Dynamic Locomotion and Whole-Body Control for Compliant Humanoids

Michael A. Hopkins

Dissertation submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy
in
Computer Engineering

Alexander Leonessa, Co-chair
A. Lynn Abbott, Co-chair
Dennis W. Hong
Craig A. Woolsey
William T. Baumann

8 December 2014
Blacksburg, Virginia

Keywords: Humanoid Robot, Dynamic Locomotion, Whole-Body Control, Impedance Control, Series Elastic Actuator, Divergent Component of Motion, Quadratic Program

Dynamic Locomotion and Whole-Body Control for Compliant Humanoids

Michael A. Hopkins

(ABSTRACT)

With the ability to navigate natural and man-made environments and utilize standard human tools, humanoid robots have the potential to transform emergency response and disaster relief applications by serving as first responders in hazardous scenarios. Such applications will require major advances in humanoid control, enabling robots to traverse difficult, cluttered terrain with both speed and stability. To advance the state of the art, this dissertation presents a complete dynamic locomotion and whole-body control framework for compliant (torque-controlled) humanoids. We develop low-level, mid-level, and high-level controllers to enable low-impedance balancing and walking on compliant and uneven terrain.

For low-level control, we present a cascaded joint impedance controller for series elastic humanoids with parallel actuation. A distributed controller architecture is implemented using a dual-axis motor controller that computes desired actuator forces and motor currents using simple models of the joint mechanisms and series elastic actuators. An inner-loop force controller is developed using feedforward and PID control with a model-based disturbance observer, enabling naturally compliant behaviors with low joint impedance.

For mid-level control, we implement an optimization-based whole-body control strategy assuming a rigid body model of the robot. Joint torque setpoints are computed using an efficient quadratic program (QP) given desired joint accelerations, spatial accelerations, and momentum rates of change. Constraints on the centroidal dynamics, contact forces, and joint limits ensure admissibility of the optimized setpoints. Using this approach, we develop compliant standing and stepping behaviors based on simple feedback controllers.

For high-level control, we present a dynamic planning and control approach for humanoid locomotion using a novel time-varying extension of the Divergent Component of Motion (DCM). By varying the natural frequency of the DCM, we are able to achieve generic vertical center of mass (CoM) trajectories during walking. Complementary reverse-time integration and model predictive control (MPC) strategies are proposed to generate dynamically feasible DCM plans over a multi-step preview window, supporting locomotion on uneven terrain.

The proposed approach is validated through experimental results obtained using THOR, a 34 degree of freedom (DOF) series elastic humanoid. Rough terrain locomotion is demonstrated in simulation, and compliant locomotion and push recovery are demonstrated in hardware. We discuss practical considerations that led to a successful implementation on the THOR hardware platform and conclude with an application of the presented control framework for humanoid firefighting onboard the ex-USS Shadwell, a decommissioned Navy ship.

This work was supported by ONR through grant N00014-11-1-0074 and by DARPA through grant N65236-12-1-1002.

Acknowledgments

I would like to thank my parents, Roger and Amelia, and my sister, Mandy, for their limitless support. I could not have succeeded without their constant encouragement throughout the years.

Thanks to my co-advisors, Dr. Alexander Leonessa and Dr. Lynn Abbott, whom I have great respect for providing me with invaluable advice and feedback on a wide variety of topics and always offering to help at the first opportunity.

Thanks to Dr. Dennis Hong for providing me the great opportunity to work in RoMeLa for many years and awarding me the trust and freedom to pursue my own research ideas. His enthusiasm and support allowed me to work on a number of rewarding projects.

Thanks to the remaining members of my graduate committee including, Dr. Craig Woolsey and Dr. William Baumann, who happily provided their valuable time and advice.

Thanks to my lab mates in TREC and RoMeLa who shared a common goal of building exciting, new robots despite the many challenges and setbacks along the way. The informal, creative culture and side projects always made the lab a fun place to work.

Thanks to Viktor Orekhov and Derek Lahr whose interest in series elastic actuators and dynamic walking inspired me to work on controls when I first joined the SAFFiR project. Navigating grad school would have been much more difficult without their dependable advice.

Thanks to Bryce Lee, Steve Ressler, Coleman Knabe, Jacob Webb, Jack Newton, Joe Holler, and the many others who contributed to the development of THOR. I had the good fortune to work with an extremely talented and creative group of engineers who never hesitated to help, no matter how many times I broke various components testing my control code.

Thanks to Robert Griffin for reviewing my paper submissions and dissertation and always providing excellent feedback.

Thanks to Johannes Englsberger for sharing great ideas and insights on humanoid locomotion and controls.

Finally, I would like to thank my extended family and friends for their positive influence and support throughout the years.

Contents

List of Figures **vii**

List of Tables **xi**

1 Introduction **1**

1.1 Motivation 1

1.2 Background 2

1.2.1 Dynamic Locomotion 2

1.2.2 Whole-Body Control 4

1.2.3 Compliant Bipeds and Quadrupeds 5

1.3 Approach 8

1.4 Contributions 9

1.5 Outline 10

1.6 Attribution 10

2 Joint-Space Control of a Series Elastic Humanoid with Parallel Actuation 11

2.1 Abstract 11

2.2 Introduction 11

2.3 Mechatronics Design and Analysis 13

2.3.1 Mechanical Design 14

2.3.2 Electrical Design 16

2.3.3 Joint Mechanics 17

2.3.4 Joint State Estimation . . . 19
2.4 Control Design . . . 20
2.4.1 SEA System Identification . . . 20
2.4.2 Actuator Force Controller . . . 23
2.4.3 Joint Impedance Controller . . . 26
2.5 Experimental Results . . . 27
2.5.1 Torque Control . . . 27
2.5.2 Impedance Control . . . 28
2.5.3 Compliant Locomotion . . . 32
2.6 Conclusion . . . 33

3 Optimization-Based Whole-Body Control of a Compliant Humanoid 34
3.1 Abstract . . . 34
3.2 Introduction . . . 34
3.3 Humanoid Dynamics . . . 36
3.3.1 Rigid Body Dynamics . . . 37
3.3.2 Centroidal Dynamics . . . 37
3.3.3 Divergent Component of Motion Dynamics . . . 37
3.4 Whole-Body Control . . . 39
3.4.1 Task-Space Formulation . . . 39
3.4.2 Model-Based Optimization (Quadratic Program) . . . 40
3.4.3 Optimization Costs . . . 41
3.4.4 Optimization Constraints . . . 41
3.4.5 Implementation . . . 42
3.5 Whole-Body Locomotion . . . 42
3.5.1 Standing . . . 43
3.5.2 Stepping . . . 45
3.6 Hardware Implementation . . . 46
3.6.1 Joint-Space Control . . . 46

3.6.2 State Estimation 48
3.7 Experimental Results 50
3.7.1 Balancing 50
3.7.2 Walking 52
3.7.3 Robustness to Unmodeled Terrain 55
3.8 Conclusion 56

4 Dynamic Locomotion on Uneven Terrain Using the Time-Varying Divergent Component of Motion 57
4.1 Abstract 57
4.2 Introduction 57
4.3 Divergent Component of Motion 59
4.4 Time-Varying DCM 60
4.5 Locomotion Planning and Control 62
4.5.1 Planning the eCMP trajectory 63
4.5.2 Planning the ω trajectory 63
4.5.3 Planning the DCM trajectory 65
4.5.4 Tracking the DCM trajectory 67
4.6 Results 68
4.7 Conclusion 72

5 Conclusion 73
5.1 Shipboard Autonomous Firefighting Robot 73
5.2 DARPA Robotics Challenge 74
5.3 Discussion 75

Bibliography 77

Appendices 85

A Ground Reference Points 86

List of Figures

1.1 Left: Humanoids in a disaster response scenario (image courtesy of DARPA). Right: THOR humanoid holding a Navy hose (image used with permission of E. Hahn). 3

1.2 Left: THOR series elastic actuator (image used with permission of J. Holler). A titanium beam serves as the spring element. Right: Location of linear actuators (red) and spring elements (blue) in THOR's lower body (image used with permission of C. Knabe). 6

1.3 Force-controlled bipeds: (A) ATLAS, hydraulic humanoid from Boston Dynamics (image used with permission of B. Lee). (B) TORO, electromechanical humanoid from DLR (image used with permission of J. Englsberger). (C) M2V2, series elastic biped from Yobotics and IHMC (image used with permission of J. Pratt) (D) SAFFiR, series elastic biped from Virginia Tech (image used with permission of B. Lee). 7

2.1 Leg kinematics of THOR, the Tactical Hazardous Operations Robot. The hip yaw-roll and ankle pitch-roll axes are actuated by linear SEAs in a parallel configuration. The hip pitch and knee pitch axes are actuated by linear SEAs in a serial configuration. Diagram used with permission of D. Lahr. Image used with permission of J. Holler. 14

2.2 (A) 2 DOF parallel hip mechanism designed by Bryce Lee, Coleman Knabe, and Viktor Orekhov (image used with permission of J. Holler). (B) Dual-axis motor controller designed by Stephen Ressler (image used with permission of S. Ressler). 15

2.3 Left: 1 DOF joint driven by a linear SEA. Right: 2 DOF joint driven by parallel SEAs. Diagrams used with permission of D. Lahr. 17

2.4 High-level block diagram of cascaded impedance controller for 2 DOF joints. 20

2.5 Forced mass-spring-damper model of a series elastic actuator with high output impedance. 21

2.6 Frequency response of the open-loop SEA plant obtained through experimental system identification. The current excitation consisted of a 2.5 A sinusoidal chirp signal with 150 Hz band-limited white noise. The estimated inverse plant model was used in the disturbance observer for the actuator force controller described in Section 2.4.2. 22

2.7 SISO force controller including PID feedback with unity feedforward and a disturbance observer (DOB) based on an empirical model of the open-loop plant, P_n. 23

2.8 Frequency response of the closed-loop SEA plant obtained through experimental system identification. The force excitation consisted of a 100 N sinusoidal chirp signal with 150 Hz band-limited white noise. The closed-loop force bandwidth is approximately 30 Hz. 25

2.9 300 N step response of the closed-loop SEA plant for various derivative gains (expressed in mA/N/s). 25

2.10 "Simple" joint impedance controller with proportional-derivative (PD) feedback and feedforward torque compensation. 26

2.11 Ankle pitch-roll torque tracking performance with and without disturbance observer. 28

2.12 THOR leg trajectory while tracking a sinusoidal chirp signal using the hip roll and yaw joints. 29

2.13 Position, velocity, and torque setpoints and estimates while tracking a sinusoidal chirp signal using the hip roll and yaw joints using low-impedance control (yaw: $k = 200$, $b = 10$, roll: $k = 400$, $b = 20$). 30

2.14 Velocity estimates obtained using the proposed incremental encoder / load cell algorithm and first order differentiation of the absolute joint encoder. . . 30

2.15 Top row: ideal and actual torque vs position for independent hip yaw and roll stiffness coefficients. Bottom row: ideal and actual torque vs velocity for independent hip yaw and roll damping coefficients. 31

2.16 THOR walking on cement. 32

2.17 Hip yaw-roll torques while walking on cement. 33

3.1 Left: Time-varying DCM dynamics. Right: Centroidal dynamics with 4 contacts. 38

3.2 Friction cone and friction pyramid approximations for foot contacts. 40

3.3 Control block diagram of locomotion subsystem. 43

3.4 Step controller state machine. 45

3.5 Hip pitch torque tracking with and without actuator disturbance observer. . 48

3.6 THOR balancing on one foot. DCM tracking is used to stabilize the CoM dynamics following a disturbance impulse applied to the back. 51

3.7 Controller response following two disturbance impulses applied to the back of the robot while balancing on the right foot. Here the x-axis is oriented to the front of the robot, and the y-axis is oriented to the left. Admissible values correspond to the outputs of the whole-body QP optimization. 51

3.8 THOR hardware platform walking on cement, gravel, and grass using identical control and estimation parameters. 52

3.9 DCM and VRP trajectories while walking on cement using the THOR hardware platform. 53

3.10 Left foot reaction forces while walking on cement using the THOR hardware platform. Admissible values correspond to the outputs of the whole-body QP optimization. 54

3.11 Center of pressure while walking on cement using the THOR hardware platform. 54

3.12 Top: Norm horizontal center of pressure error while walking on cement, gravel, and grass. Bottom: Step controller support state (0: double support, 1: single support). 55

3.13 Left: THOR standing on a balance board that rotates about a pivot. Right: Stepping onto unexpected debris and adapting to the various terrain. Images used with permission of E. Hahn. 55

4.1 Left: THOR walking on uneven terrain in simulation. Right: Time-varying Divergent Component of Motion (DCM) dynamics. 59

4.2 Top: Reference CoM and eCMP height. Bottom: LIPM natural frequency, $\alpha(t)$, and time-varying DCM natural frequency, $\omega(t)$, computed using reverse-time integration. 64

4.3 Lateral DCM and VRP trajectories for a 2-step plan. The DCM reference is computed using reverse-time integration over a 2.0 s time horizon given a terminal DCM position, $\boldsymbol{\xi}_f$. Example CoM solution trajectories are shown for initial CoM states consistent with the DCM reference. Double support phases are marked on the x-axis. 65

4.4 Lateral DCM and VRP trajectories for a 2-step plan. The DCM RK2 trajectory is computed using reverse-time integration over a 2.0 s time horizon. The DCM MPC trajectory is computed using linear MPC over a 0.5 s preview window. Double support phases are marked on the x-axis. 67

4.5 DCM, CoM, VRP, and eCMP plans given a 4-step preview window with increasing step height and turning radius. 69

4.6 Comparison of DCM and contact force trajectories for the footstep plan shown in Fig. 4.5 given constant and time-varying natural frequencies. 70

4.7 THOR walking over cinder blocks of varying height in simulation. 71

4.8 DCM and VRP trajectories while walking over cinder blocks in simulation (see Fig. 4.7). Each step is approximately 1.6 s with a double support duration of 0.32 s. 71

4.9 THOR walking with toe-off in simulation. The stride length is 0.35 m and the step duration is 1.5 s. 72

5.1 THOR suppressing a controlled fire onboard the ex-USS Shadwell in Mobile, AL. Images used with permission of Virginia Tech/Logan Wallace. 74

5.2 Left: Terrain task layout at DRC trials. Right: Debris task layout at DRC trials. Images courtesy of DARPA. 75

List of Tables

2.1 THOR Joint Specifications 15
2.2 THOR Actuator Specifications 16
2.3 Open Loop System Parameters 22

3.1 Whole-Body Controller Weights and Gains 50

Chapter 1

Introduction

1.1 Motivation

Although robotics in general has the potential to transform a wide variety of industries, emergency response scenarios are likely to be among the earliest applications for humanoid technology. With a human-inspired morphology, humanoid robots are able to utilize human tools, navigate man-made environments, and interact with humans in a natural manner. As such, humanoids have the potential to complete generic tasks in extreme environments, reducing the need to place human responders at risk. Although advances in robotic hardware may soon reach a tipping point where real-world applications become feasible, the usefulness of humanoid technology will ultimately depend on the development of control systems that allow robots to move with the speed and stability of biological systems.

In 2011, the Fukushima Daiichi nuclear power plant was devastated by multiple reactor core meltdowns following a large magnitude earthquake and tsunami. With high levels of radiation near the damaged reactors, the availability of readily fieldable humanoid robots could have significantly accelerated the short-term damage control and long-term cleanup efforts. Following these events, the DARPA Robotics Challenge (DRC) was announced to promote the development of advanced rescue robots for disaster relief scenarios. As part of the ongoing competition, robots will be required to complete a series of tasks including operating a vehicle, traversing rough terrain, using power tools, and clearing debris. The ONR Shipboard Autonomous Firefighting Robot (SAFFiR) program is a separate effort to promote the development of humanoid robots that are capable of performing damage control and maintenance tasks onboard a Navy ship.

In order to operate in the uncertain environments encountered in these scenarios, humanoids must be able to demonstrate robust manipulation and locomotion capabilities. The complex behaviors required in real-world applications often encompass multiple control objectives, e.g. maintaining balance while operating a power tool. As a result, quantitative performance

metrics can be difficult to identify. Controller design is further complicated by the dynamic properties of humanoid systems which are typically characterized by the following:

- Multibody dynamics (nonlinear system)
- High dimensionality (redundant system)
- Multiple contact modes (hybrid system)
- Actuator saturation (nonlinear system)
- Underactuation (floating-base system)

Despite the many challenges associated with humanoid locomotion, bipedal animals such as humans and birds are highly adept at traversing difficult and diverse terrain. Legged animals often employ highly dynamic gaits to achieve fast and efficient locomotion given limited range of motion and muscle strength. They also utilize active and passively compliant control strategies to maintain balance in response to external disturbances [1]. By drawing inspiration from biological systems, it may be possible to improve the design and control of humanoid robots while simultaneously furthering our knowledge of human locomotion and control.

This dissertation focuses on the advancement of dynamic locomotion and whole-body control strategies for humanoid robots with compliant actuation. The proposed methods are implemented and tested on THOR, the Tactical Hazardous Operations Robot [2], shown in Fig. 1.1. Developed as part of the DRC and SAFFiR programs, THOR is a 1.8 m tall, 60 kg electromechanical humanoid featuring 34 total degrees of freedom (DOF). Linear series elastic actuation is included in the lower body for low-impedance force control, while traditional rotary actuation is included in the upper body for high-impedance velocity control. Our goal is to develop naturally compliant control strategies that allow the THOR humanoid to operate in uncertain environments, such as Navy ships, and implement complex whole-body behaviors, such as manipulating a fire hose, in support of the DRC and SAFFiR efforts.

1.2 Background

1.2.1 Dynamic Locomotion

Model-based locomotion strategies typically generate and track dynamically feasible joint-space or task-space reference trajectories over a short time horizon in order to achieve stable walking and running gaits. Regardless of the approach, it is common to assume that inertial foothold poses, phase durations, or other gait-specific objectives are provided by a high-level planner incorporating knowledge of the surrounding terrain.

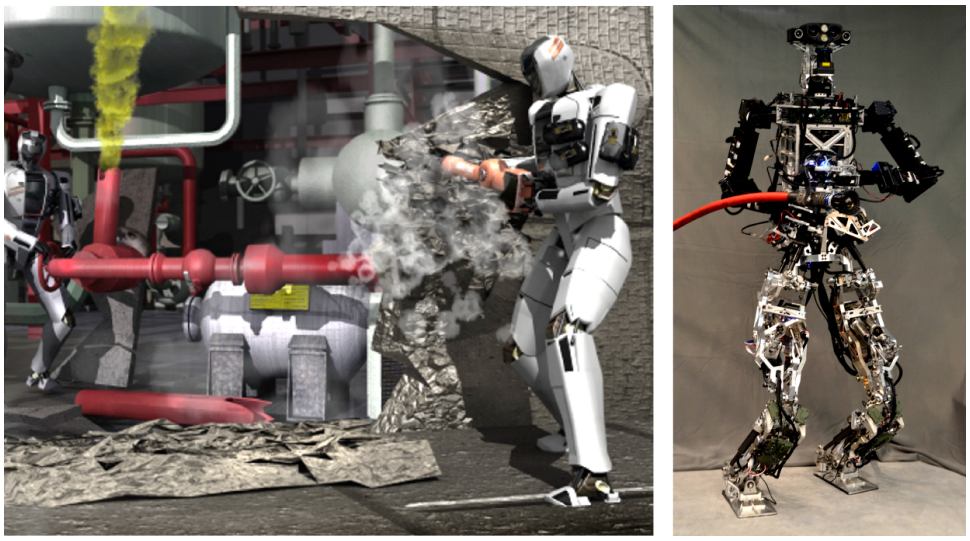

Figure 1.1: Left: Humanoids in a disaster response scenario (image courtesy of DARPA). Right: THOR humanoid holding a Navy hose (image used with permission of E. Hahn).

Direct trajectory optimization methods generate actuator or joint-space trajectories based on the nonlinear multibody dynamics. Control constraints, including range of motion and saturation limits, are typically enforced to ensure admissibility of the optimized trajectories. In [3], walking and running gaits were generated through energetic optimization of actuator trajectories using Sequential Quadratic Programming. In [4], the authors implemented simultaneous optimization of joint-space trajectories and contact modes to plan dynamic gaits for rigid body systems. The primary drawback of these methods is the complexity of the trajectory optimization problem. This is partially due to the large number of decision variables required to optimize the high-dimensional input trajectories and partially due to the nonlinearity of the dynamics. As a result, these algorithms are not yet applicable for real-time planning and control, as solution times may range from minutes to days.

Task-space locomotion strategies often compute center of mass (CoM) and end-effector trajectories based on linear models of the centroidal dynamics. Most approaches are designed to ensure admissibility of the Zero Moment Point (ZMP) in order to satisfy no-tip conditions on the support feet [5]. The linear inverted pendulum model (LIPM) assumes planar motion of the CoM and zero torque, or angular momentum rate of change, about the CoM [6]. In [6], the authors derived an analytical solution for the LIPM dynamics to compute admissible horizontal CoM trajectories for dynamic walking that satisfy the ZMP criterion. Although a majority of approaches assume zero torque about the CoM, some authors have proposed methods that produce significant angular momentum during walking [7].

The Capture Point (CP) [8] and Divergent Component of Motion (DCM) [9] are equivalent transformations of the CoM state that encode the unstable dynamics of the LIPM. By definition, the CoM can be indirectly stabilized by tracking an appropriate CP/DCM

trajectory. In [10], the authors proposed an analytical CP planner for dynamic walking, and in [11], this work was extended to enable 3D walking on uneven terrain. In [12], the authors introduced a ZMP-based CP tracking controller to compensate for time-varying vertical CoM dynamics and angular momentum rates of change during stepping, and in [13], an integrator was incorporated into the CP controller to eliminate steady-state errors.

Optimization-based locomotion strategies often compute discretized CoM or CP trajectories that minimize a cost function based on the centroidal dynamics. The cost function is typically evaluated over a one to three step preview window in order to optimize the smoothness and stability of the planned gait. ZMP preview controllers minimize the total CoM jerk and ZMP tracking error using LQR control [14]. Although LIPM assumptions are common in these approaches, some authors have proposed methods to incorporate predefined CoM height and centroidal torque trajectories [15; 16]. Alternatively, several model predictive control (MPC) techniques have been proposed to optimize the CoM trajectory and ground reaction forces online. In [17], the authors proposed a linear MPC approach that solves the ZMP preview control problem using quadratic programming. This approach is extended in [18] to incorporate linear ZMP constraints and variable sampling rates. Other approaches have investigated simultaneous CoM planning and online footstep adaptation [19].

In each approach, CoM and end-effector trajectories are typically tracked using local reference controllers coupled with an inverse kinematics or inverse dynamics solver. Inverse kinematics (IK) solvers compute feedforward joint-space velocities given desired task-space velocities. IK solutions are typically employed in position and velocity-controlled systems and do not require a rigid body dynamic model. As a result, ZMP feedback is often used to compensate for dynamic errors. Model-based inverse dynamics (ID) solvers compute feedforward joint torques given admissible joint-space or task-space accelerations. ID solutions are typically employed in torque-controlled systems for low-impedance control. Virtual model control (VMC) serves as an alternative approach that computes feedforward joint torques given desired end-effector wrenches using only a kinematic model [20]. By assuming quasistatic equilibrium conditions, VMC approaches do not compensate for the natural dynamics of the mechanical system.

1.2.2 Whole-Body Control

In recent years, optimization-based whole-body control approaches have generated considerable interest in the legged robotics community. Given a set of reduced-dimensional motion tasks, task-space whole-body control formulations compute optimal joint-space accelerations and contact forces subject to the rigid body dynamics and frictional contact constraints. Motion tasks typically include desired centroidal momentum rates of change, end-effector accelerations, and joint accelerations for multi-contact behaviors such as walking. Since joint torque setpoints are often computed from the optimized accelerations and contact forces, whole-body control can be categorized as a type of inverse dynamics.

Early whole-body control formulations utilized null-space projection methods to prioritize motion tasks and enforce dynamic constraints [21]. Within the past five years, a number of researchers have proposed convex optimization techniques to solve the inverse dynamics and whole-body control problem [22–29]. These approaches resolve redundancies in high-dimensional systems through regularization of the optimized joint setpoints. Recently, optimization-based inverse dynamics algorithms have been successfully ported to torque-controlled hardware platforms, although some implementations have required additional joint position and velocity feedback [30].

In general, whole-body control can serve as the basis for dynamic locomotion, bimanual manipulation, or any other behavior requiring multi-contact control. Task-space formulations simplify the design of locomotion controllers by allowing direct control of CoM and end-effector accelerations [31]. Centroidal momentum control has been studied extensively in the context of dynamic balancing [32–34]. Linear momentum rates of change are often defined to stabilize the CoM dynamics during locomotion, while angular momentum is often carefully regulated during human locomotion [35]. Given a finite base of support, it is not always possible to simultaneously satisfy linear and angular momentum objectives. In such cases, task weighting or prioritization can be used to resolve conflicts between opposing objectives.

1.2.3 Compliant Bipeds and Quadrupeds

A number of studies have discussed the benefits of mechanical compliance in legged animals. Enhanced metabolic efficiency, energy storage, and passive stability have been observed in a variety of species [36]. Drawing inspiration from biology, researchers have begun to incorporate passive compliance into the design of legged robots. Although several works have investigated improvements in metabolic efficiency using compliant exoskeletons and prostheses [37], most researchers in the field of legged robotics have focused on the use of series elasticity for low-impedance force control. The remainder of this section provides a review of compliant (force-controlled) bipeds and quadrupeds organized by type of actuator.

In [38], Pratt et al. proposed the series elastic actuator (SEA), which is characterized by a spring element in series with the actuator transmission. By reducing mechanical impedance, this configuration has been shown to improve the fidelity and stability of closed-loop force controllers while simultaneously increasing the shock tolerance of the actuator [38–41]. Fig. 1.2 shows an example SEA used in the design of THOR, including the location of the actuators and spring elements in the lower body. Linear forces are generated by a brushless DC motor that drives a two-stage transmission composed of a belt drive and precision ball screw. A compliant titanium beam in series with the actuator provides the desired elasticity.

Linear SEAs The MIT Leg Lab pioneered the use of series elastic actuation in walking robots such as the M2 biped [42]. The developers of M2 used linear ball screw driven SEAs with inline die springs to actuate the hip, knee, and ankle joints. Actuator forces were

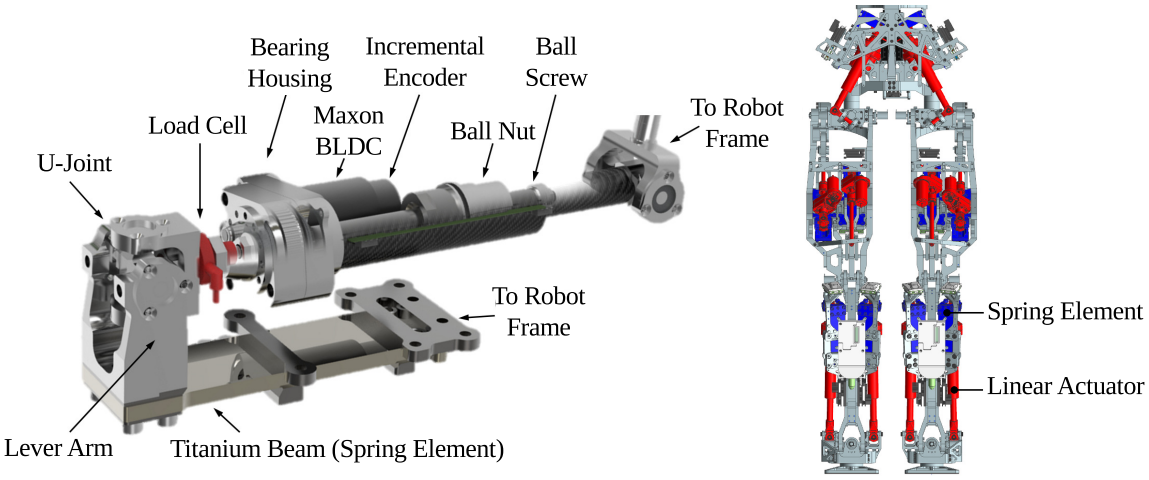

Figure 1.2: Left: THOR series elastic actuator (image used with permission of J. Holler). A titanium beam serves as the spring element. Right: Location of linear actuators (red) and spring elements (blue) in THOR's lower body (image used with permission of C. Knabe).

estimated by measuring the spring deflection via an encoder. The Yobotics IHMC M2V2 biped features similar SEAs in all 12 DOF [43]. In [44], the authors demonstrated compliant walking and push recovery on M2V2 using CP feedback and virtual model control. Several planar bipeds have demonstrated 2D walking using ball screw driven linear SEAs including MIT's Spring Flamingo [20] and UT Austin's HUME [45]. The SAFFiR biped, a predecessor of THOR, achieved 3D walking on a variety of compliant terrain using a hybrid position and force control approach with linear series elastic actuation in the ankles [46]. Like SAFFiR, the THOR humanoid features parallel actuation in both the hip and ankles [2], and like M2V2, THOR features linear SEAs in all 12 DOF of the lower body. The THOR SEA utilizes a unique spring element consisting of a titanium beam that is loaded in bending and an inline load cell for direct force measurement.

Rotary SEAs Rotary SEAs are typically characterized by an electric motor and gearbox coupled with a rotary spring. NASA's Valkyrie humanoid features rotary SEAs with harmonic drive reducers and custom torsion springs [47]. As in most designs, output torques are estimated from the measured spring deflections. The StarlETH quadruped, developed by ETH Zurich, includes highly compliant rotary SEAs that have allowed it to walk on rough terrain using virtual model control [48]. IIT's COMAN biped has demonstrated compliant balancing and walking using rotary SEAs in the hip, knee, and ankle pitch DOF [49].

Cable Driven SEAs A number of compliant bipeds have incorporated cable drives to displace the motor and transmission from the actuated joint. MIT's Spring Turkey [50]

and Spring Flamingo [20] utilize cable drives to reduce the overall leg inertia by moving the hip, knee, and ankle actuators into the torso. TU Delft's Flame and Tulip bipeds utilize cable drives with inline tension springs to implement series elasticity [51]. Both robots have demonstrated successful limit cycle walking. LIRMM's SHERPA biped features a modular 2 DOF joint design driven by linear SEAs in a parallel cable drive configuration [52].

Force-Controllable Actuators Although lacking explicit series elasticity, a number of alternative force-controllable actuation technologies have been incorporated into legged robot designs. VUB's Lucy biped uses antagonistic pneumatic actuators with pressure feedback to allow joint torques and stiffnesses to be controlled independently during walking [53]. DLR's torque-controlled humanoid, TORO, uses rotary harmonic drive actuators with a torque sensor mounted in series with the output link [54]. TORO has demonstrated compliant multi-contact balancing and dynamic walking using DCM-based planning and control. A number of force-controlled bipeds and quadrupeds feature hydraulic actuators, including Boston Dynamics' ATLAS humanoid, IIT's HyQ quadruped [55], and the SARCOS biped [25]. Closed-loop force control is implemented using hydraulic pressure sensors or inline load cells. Several institutions have implemented whole-body control and locomotion on the ATLAS humanoid as part of the DARPA Robotics Challenge [24; 29; 30]. Fig. 1.3 shows a variety of force-controlled bipeds using hydraulic, electric, and series elastic actuation.

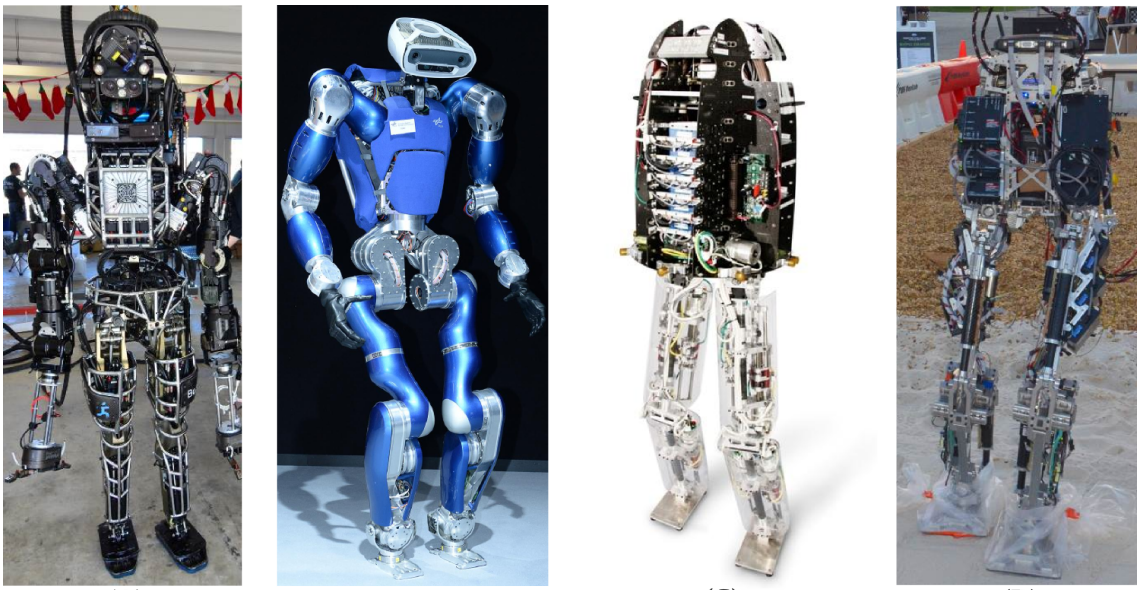

Figure 1.3: Force-controlled bipeds: (A) ATLAS, hydraulic humanoid from Boston Dynamics (image used with permission of B. Lee). (B) TORO, electromechanical humanoid from DLR (image used with permission of J. Englsberger). (C) M2V2, series elastic biped from Yobotics and IHMC (image used with permission of J. Pratt) (D) SAFFiR, series elastic biped from Virginia Tech (image used with permission of B. Lee).

1.3 Approach

This dissertation presents a complete dynamic locomotion and whole-body control framework for compliant humanoids. We discuss the low-level, mid-level, and high-level control strategies developed to enable torque-controlled humanoids to traverse compliant and uneven terrain. Experimental results obtained using the THOR hardware platform demonstrate the practicality of the proposed approach, which is readily extensible to a variety of multi-contact behaviors.

As discussed in the previous section, a number of legged robots feature rotary joints driven by linear SEAs in a parallel or serial configuration. In order to advance low-level control strategies for series elastic humanoids, we present a cascaded joint-space impedance control approach for this class of mechanisms. Distributed control is enabled through the use of a dual-axis motor controller that computes joint mechanics, actuator forces, and motor currents locally. A high-performance inner-loop force controller is developed using a combination of feedforward and PID control with a disturbance observer (DOB) based on an empirical model of the open-loop plant. The proposed approach enables low-impedance control of all 12 DOF in the THOR lower body. We include experimental results for torque control and compliant trajectory tracking on THOR.

Our mid-level control approach consists of an optimization-based whole-body controller that assumes a rigid body model of the robot. Joint torque and acceleration setpoints are computed using an efficient quadratic program (QP) formulation given desired motion tasks from a high-level locomotion controller. Constraints on the centroidal dynamics, frictional contact forces, and joint position/torque limits ensure admissibility of the optimized joint setpoints. Using this approach, we develop compliant standing and stepping behaviors that command desired task-space accelerations and momentum rates of change using a simple controller that stabilizes the centroidal dynamics. We discuss practical considerations and adaptations that led to a successful implementation on the THOR hardware platform and include experimental results for push recovery and compliant walking on various terrain including cement, gravel, and grass.

In order to advance high-level control strategies for dynamic locomotion, we present a dynamic planning and control approach using a novel time-varying extension of the Divergent Component of Motion (DCM). By varying the natural frequency of the DCM, we are able to achieve generic vertical CoM trajectories during single and double support. Given desired ZMP and vertical CoM trajectories, reverse-time integration of the discretized DCM dynamics is used to generate admissible DCM plans over a multi-step preview window. To account for discontinuities during replanning, we implement linear model predictive control (MPC) over a short time horizon. DCM tracking control is achieved using a time-varying proportional-integral controller based on the Virtual Repellent Point (VRP). The proposed approach is implemented on the THOR hardware and simulation platforms. We demonstrate rough terrain locomotion and heel-toe walking in simulation.

1.4 Contributions

The contributions of this dissertation are as follows:

1. The development of a complete control framework for dynamic locomotion, whole-body control, and joint-space control of compliant humanoids. The proposed approach is validated through balancing and walking experiments using the THOR humanoid.

2. The design and characterization of a cascaded impedance controller for rotary joints driven by linear series elastic actuators in a parallel configuration. We utilize a model-based disturbance observer based on the open-loop actuator dynamics in order to decouple disturbance estimates from the inner-loop force controllers and simplify gain selection.

3. The design and implementation of an optimization-based whole-body controller for the THOR hardware platform, including a discussion of practical considerations and lessons learned. To our knowledge, only three other adult-sized humanoids have demonstrated low-impedance whole-body control and locomotion at the time of writing.

4. The presentation of a novel time-varying extension of the Divergent Component of Motion (DCM) that simplifies dynamic locomotion planning and control on uneven terrain.

5. The development of a real-time dynamic planning algorithm for humanoid locomotion using reverse-time integration and model predictive control of the time-varying DCM. Our approach supports generic ZMP, angular momentum, and CoM height trajectories.

6. The development of a momentum control law for dynamic balancing based on the time-varying DCM dynamics.

1.5 Outline

The remainder of this dissertation is organized in manuscript format. Chapters 2, 3, and 4 are intended to serve as independent manuscripts that describe the low-level, mid-level, and high-level control strategies implemented on the THOR humanoid, respectively.

Chapter 2 presents a distributed joint-space impedance control approach for series elastic humanoids with parallel actuation.

Chapter 3 presents an optimization-based whole-body control and locomotion framework implemented on the THOR humanoid. We include experimental results using the THOR hardware.

Chapter 4 presents a dynamic planning and control framework for humanoid locomotion using the time-varying DCM. We include experimental results using a 30 DOF model of THOR in the Gazebo simulation environment.

Chapter 5 concludes with an example application of the presented control framework for shipboard firefighting using the THOR humanoid. We also include a discussion of the DARPA Robotics Challenge effort and future research directions.

Appendix A provides a brief overview of ground reference points defined in the bipedal locomotion literature.

1.6 Attribution

Chapter 2 is a journal article in preparation for submission. Stephen Ressler contributed the electronic design and embedded software framework for the dual-axis motor controller hardware. Derek Lahr assisted in the derivations of the joint mechanics and contributed valuable insight and feedback for the early drafts of the manuscript.

Chapter 3 contains material submitted to the 2015 IEEE International Conference on Robotics and Automation (ICRA) [56]. The paper includes two co-authors. Alexander Leonessa and Dennis W. Hong contributed valuable insight and feedback for the early drafts of the manuscript.

Chapter 4 contains material published in the 14th IEEE-RAS International Conference on Humanoid Robots (Humanoids) [57]. The paper includes two co-authors. Alexander Leonessa and Dennis W. Hong contributed valuable insight and feedback for the early drafts of the manuscript.

Chapter 2

Joint-Space Control of a Series Elastic Humanoid with Parallel Actuation

2.1 Abstract

As model-based whole-body control approaches become increasingly popular in the field of humanoid robotics, there is a real need for reliable torque and impedance control strategies for compliant hardware platforms. In this paper we present a joint-space control approach for THOR, a series elastic humanoid with parallel actuation. The THOR platform features 2 DOF parallel hip and ankle mechanisms driven by linear series elastic actuators. A cascaded joint impedance controller with feedforward torque compensation is implemented using a novel dual-axis motor controller. To enable distributed control, the joint statics and velocity kinematics are computed locally given the available sensor measurements, and a joint-space rigid body dynamic model is assumed for high-level control. Actuator force commands, computed from the desired joint setpoints, are tracked using a combination of feedforward and PID control with a disturbance observer (DOB) based on an empirical model of the open-loop plant. The effectiveness of the proposed approach is verified through joint tracking and locomotion experiments on the THOR humanoid.

2.2 Introduction

The field of legged robotics is currently undergoing a major transition in terms of actuation and control technology. Although researchers have demonstrated that robust dynamic locomotion can be achieved using stiff position-controlled actuation [58], these systems often require the use of outer-loop admittance controllers to account for unpredictable internal and external forces that can arise due to a lack of direct joint torque sensing. This can result in

stability issues in the presence of unmodeled dynamics and safety concerns for applications requiring human-robot interaction. In recent years a number of legged robots have featured force-controllable actuators including M2V2 [43], TORO [54], COMAN [49], StarlETH [48], HyQ [55], Hume [45], NASA's Valkyrie, and Boston Dynamics' ATLAS. This is in part due to the increased safety and adaptability associated with compliant control, and in part due to a new interest in whole-body control strategies which optimize desired joint torques in order to control task-space quantities such as centroidal momentum rate of change and end-effector accelerations [23–25; 29; 30].

In order to achieve reliable force control for legged locomotion, Pratt et al. proposed the series elastic actuator (SEA), which is characterized by a spring element in series with the actuator transmission [38]. By reducing mechanical impedance, this configuration has been demonstrated to improve force fidelity and stability while increasing shock tolerance [38–41]. Rotary SEAs and Variable Stiffness Actuators have been incorporated into a number of leg and arm designs including COMAN [49], the DLR Hand Arm System [59], and Valkyrie. Linear SEAs have been primarily incorporated into hip, knee, and ankle assemblies using levers or cable drives to generate torque about a revolute or universal joint. Examples include Spring Flamingo [20], M2V2 [43], HUME [45], and SAFFiR [60]. In many scenarios this approach can result in improved inertia distribution compared to designs with rotary actuators mounted at the joint, although similar benefits can be achieved by relocating rotary actuators using four-bar linkages or belt drives. In traditional leg designs, actuators are arranged in a serial configuration to increase actuator modularity and reduce control complexity. However, packaging and power requirements often complicate the mechanical design of serial hip, knee, and ankle joints using linear actuators.

Parallel actuation can offer favorable packaging and torque characteristics for humanoid robots. Linear actuated parallel mechanisms resemble muscle and tendon configurations often found in legged animals. One of the most common examples found in bipedal robots is a universal joint driven by a pair of linear actuators. This configuration was incorporated into the 2 DOF ankle of M2 [42] and has since been featured in a number of bipeds [43; 60; 61]. The SHERPA biped features an alternative 2 DOF joint design driven by linear SEAs in a parallel cable drive configuration [52]. Parallel mechanisms are well suited for hip and ankle joints where the desired torque vectors are typically skewed towards one axis and can be generated by the combined force outputs of each actuator [42; 52]. The arrangement of joint axes and actuators can be chosen to optimize packaging and torque requirements for each revolute joint. Unfortunately, parallel configurations typically offer reduced range of motion compared to serial designs. Additionally, the MIMO nature of parallel actuation presents a challenge to traditional distributed control system designs that utilize single-axis motor controllers to close actuator force control loops locally.

Regardless of the mechanical design, a number of control issues must be addressed in order to achieve reliable joint-space control using SEAs. By introducing a spring element, SEAs offer increased stability at the cost of reduced force control bandwidth [38]. Tracking performance can degrade in the presence of unmodeled dynamics such as nonlinear friction

forces. This problem is especially apparent when friction is high relative to output inertia. In motion control applications, low-gain position and velocity feedback can be used to help compensate for unmodeled dynamics by introducing joint-space stiffness and damping. In this approach, virtual impedance is added in parallel with feedforward torques computed using an inverse dynamics solver in order to improve tracking performance. This is a form of "simple" impedance control [62]. Several authors have presented single-axis impedance controllers leveraging inner-loop force controllers for series elastic, hydraulic, and harmonic drive actuators [63–68]. Tsoi et al. proposed an impedance control approach for a parallel ankle manipulator with linear actuation [69].

This paper presents a joint-space control approach developed for the 2 DOF parallel hip and ankle mechanisms central to the design of THOR, the Tactical Hazardous Operations Robot. As illustrated in Fig. 2.1 and 2.2, the parallel mechanism consists of a universal joint actuated by two electromechanical SEAs. A cascaded joint-space impedance controller with feedforward torque compensation is implemented on a novel dual-axis motor controller. To enable distributed control, the joint statics and velocity kinematics are computed locally using joint and motor encoder measurements, and a joint-space rigid body dynamic model is assumed for high level control. Actuator force commands, computed from the desired joint setpoints, are tracked using an inner-loop force controller that combines feedforward and PID control with a model-based disturbance observer (DOB) using inline load cell measurements. In comparison to similar force controller implementations, the use of an inverse plant based on an empirical model of the open-loop actuator dynamics simplifies tuning of the cascade controller by decoupling DOB estimates from force controller gains. The proposed approach is verified through experiments in locomotion and joint trajectory tracking using the THOR hardware platform.

2.3 Mechatronics Design and Analysis

THOR, the Tactical Hazardous Operations Robot, is a torque-controlled humanoid developed as part of the DARPA Robotics Challenge and ONR Shipboard Autonomous Firefighting Robot (SAFFiR) programs. The 1.8 m tall, 60 kg robot features a total of 34 DOF, including 6 DOF legs, 7 DOF arms, a 2 DOF waist, a 2 DOF neck, and 2 DOF hands. The lower body of THOR includes force-controllable series elastic actuators and novel dual-axis motor controllers to enable distributed joint-space impedance control. As the proposed control approach requires knowledge of the underlying mechanics, we will provide a brief overview of the THOR hip and ankle mechatronics. In particular, we will focus on the design of the 2 DOF mechanism common to each joint assembly and include a derivation of the joint-space kinematics and statics. A comprehensive review of the THOR design can be found in [2].

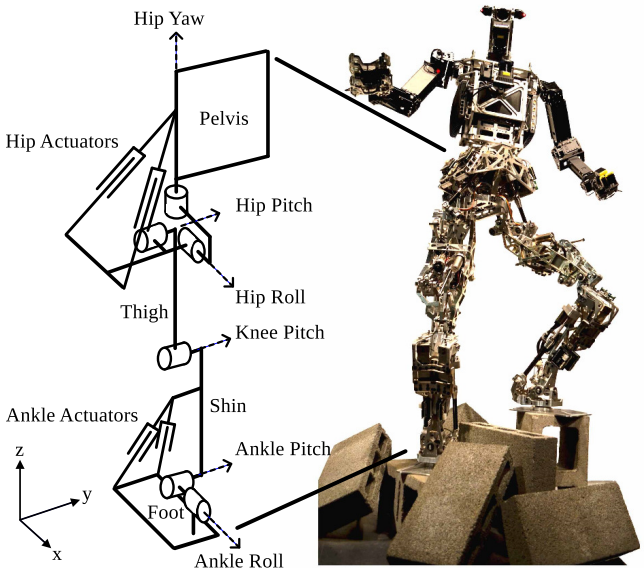

Figure 2.1: Leg kinematics of THOR, the Tactical Hazardous Operations Robot. The hip yaw-roll and ankle pitch-roll axes are actuated by linear SEAs in a parallel configuration. The hip pitch and knee pitch axes are actuated by linear SEAs in a serial configuration. Diagram used with permission of D. Lahr. Image used with permission of J. Holler.

2.3.1 Mechanical Design

As illustrated in Fig. 2.1, the hip yaw-roll and ankle pitch-roll axes of THOR are actuated by parallel SEAs. Although the link lengths, actuator configurations, and joint axes vary between the two assemblies, the hip and ankle share a common mechanical design. Each parallel mechanism incorporates a universal joint driven by a pair of linear actuators. A rendering of the hip yaw-roll assembly is shown in Fig. 2.2. The coupled actuators induce moments about the joint axes by applying forces on the lever arms attached to the lower linkage. As discussed in [2], the hip pitch and knee pitch are serially actuated using Hoken's mechanisms for extended range of motion [70]. Although we do not analyze the hip pitch and knee pitch mechanisms here, the control approach is identical to the hip yaw-roll and ankle pitch-roll with the exception of the Jacobian derivation. The performance specifications of the hip yaw-roll and ankle pitch-roll mechanisms are given in Table 2.1. Refer to [2] and [71] for further details.

Custom electromechanical linear SEAs were designed to actuate each joint in the lower body [70]. Each actuator is driven by a Maxon EC 4 Pole motor with a 48 V power stage. The rotary to linear transmission is composed of a 3:1 belt reduction and a high-precision ball screw with a 2 mm pitch. Universal joints at either end of the actuator constrain the ball nut rotation in order to generate translation, and series elasticity is implemented using a titanium beam that is loaded in bending. In order to measure the reaction force due

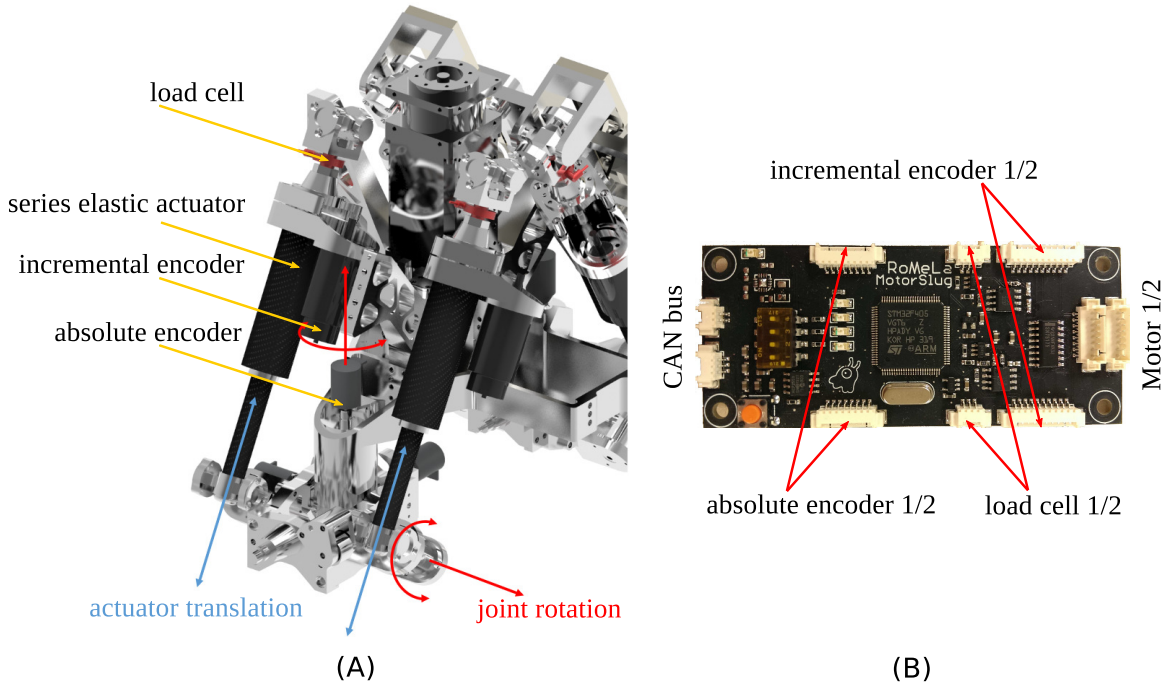

Figure 2.2: (A) 2 DOF parallel hip mechanism designed by Bryce Lee, Coleman Knabe, and Viktor Orekhov (image used with permission of J. Holler). (B) Dual-axis motor controller designed by Stephen Ressler (image used with permission of S. Ressler).

Table 2.1: THOR Joint Specifications

Parameter	Minimum Value	Maximum Value	Units
Hip Yaw Position	-20	45	deg
Hip Roll Position	-30	45	deg
Ankle Pitch Position	-55	35	deg
Ankle Roll Position	-30	30	deg
Hip Yaw Torque	$(-115, -74)$	$(74, 115)$	Nm
Hip Roll Torque	$(-175, -126)$	$(126, 175)$	Nm
Ankle Pitch Torque	$(-290, -173)$	$(173, 290)$	Nm
Ankle Roll Torque	$(-222, -62)$	$(62, 222)$	Nm

to spring deflection, a Futek LCM–200 load cell sensor is mounted in between the motor housing and the titanium beam. The spring stiffness selection was motivated by the desired force fidelity rather than energy efficiency and was informed by experimental results obtained using configurable compliance on the SAFFiR platform, a predecessor to THOR [72]. The THOR SEA can achieve linear velocities of 0.2 m/s and is capable of delivering peak and continuous forces of 2225 N and 640 N, respectively. Table 2.2 lists relevant specifications of the THOR SEA. Refer to [70] for further details.

Table 2.2: THOR Actuator Specifications

Parameter		Nominal Value	Units
Ball Screw Pitch	γ	2	mm
Motor Pulley Ratio	b	3	
Motor Torque Constant	k_τ	0.0255	Nm/A
Spring Stiffness	k_S	655	N/mm
Continuous Force		640	N
Max Force		2225	N

2.3.2 Electrical Design

As illustrated in Fig. 2.2, the proposed joint-space control approach is implemented on a custom dual-axis motor controller. Each 6 DOF leg is equipped with three controllers actuating hip yaw and hip roll, hip pitch and knee pitch, and ankle pitch and ankle roll, respectively. A comprehensive overview of the electrical design and software architecture for the THOR motor controller is given in [73]. The device implements sensor acquisition, actuator force control, and joint-space impedance control for SEAs in a parallel or serial configuration using a 32-bit ARM Cortex-M4 processor. By closing the joint control loops locally, we are able to achieve higher sample rates and reduced signal delay compared to a centralized controller subject to communication latencies. The motor controller includes two 12-bit ADC channels for actuator load cell sensing and dual connectors for absolute and quadrature incremental encoders. Pulse width modulated signals are relayed to the motor power stages to control the desired motor current in each SEA in the THOR lower body.

The motor controllers on each leg are networked on a designated CAN channel, and sensorimotor data is communicated to and from a high-level field computer using CANopen, a common fieldbus protocol used in industrial applications. Process data objects are defined to transmit joint-space position, velocity, and torque setpoints to the motor controllers at a rate of 500 Hz. The corresponding position, velocity, and torque estimates are returned to the high-level control computer using a synchronous read/write pattern. Desired joint stiffness and damping coefficients can also be updated online using additional CAN frames. Following extensive testing on the THOR hardware platform, we concluded that a 500 Hz bus rate was

sufficient for dynamic locomotion using an optimization-based whole-body controller that relies primarily on joint torque control.

2.3.3 Joint Mechanics

In this section we derive the forward and inverse mechanics for the 2 DOF hip and ankle assemblies implemented on the THOR hardware platform. A diagram of the parallel actuated mechanism is shown in Fig. 2.3.

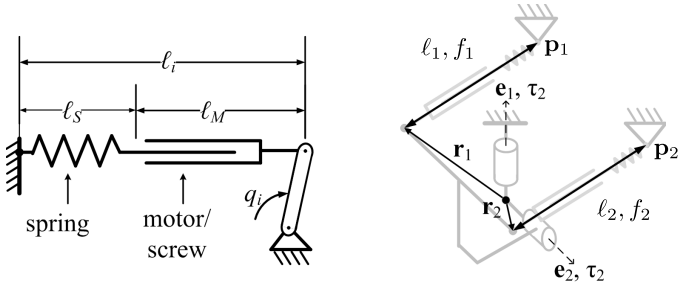

Figure 2.3: Left: 1 DOF joint driven by a linear SEA. Right: 2 DOF joint driven by parallel SEAs. Diagrams used with permission of D. Lahr.

Each actuator, $i = 1, 2$, generates a torque about each joint axis by exerting a force, f_i, between an attachment point, $\mathbf{p}_i$, fixed to the upper link and a moment arm, $\mathbf{r}_i$, that rotates with the lower link. As a result, the total moment, τ_i, along each joint axis, $\hat{\mathbf{e}}_i$, is dependent on the forces imparted by both actuators. The actuator configuration is encoded by the vector of actuator lengths,

$$\boldsymbol{\ell} = \begin{bmatrix} \ell_1 \\ \ell_2 \end{bmatrix} = \boldsymbol{\ell}_S + \boldsymbol{\ell}_M, \tag{2.1}$$

where $\boldsymbol{\ell}_S \in \mathbb{R}^2$ and $\boldsymbol{\ell}_M \in \mathbb{R}^2$ represent the spring displacement and ball screw travel for each SEA. The joint configuration is given by the vector of joint angular positions, $\mathbf{q} \in \mathbb{R}^2$.

The inverse velocity kinematics for this mechanism can be expressed as,

$$\dot{\boldsymbol{\ell}} = \mathbf{J}\dot{\mathbf{q}}, \tag{2.2}$$

where $\dot{\boldsymbol{\ell}} \in \mathbb{R}^2$ is the vector of linear actuator velocities, $\mathbf{J} = \frac{\partial \boldsymbol{\ell}}{\partial \mathbf{q}} \in \mathbb{R}^{2\times 2}$ is the mechanism Jacobian, and $\dot{\mathbf{q}} \in \mathbb{R}^2$ is the vector of joint angular velocities. Similarly, the forward statics are given by the dual relationship,

$$\boldsymbol{\tau} = \mathbf{J}^T\mathbf{f}, \tag{2.3}$$

where $\boldsymbol{\tau} \in \mathbb{R}^2$ and $\mathbf{f} \in \mathbb{R}^2$ represent the joint torque and actuator force vectors, respectively [74].

For a given joint configuration, the mechanism Jacobian can be computed as follows. First, define the joint axis, $\hat{\mathbf{e}}_i$, and actuator moment arm, $\mathbf{r}_i$, for each DOF, $i = 1, 2$, as follows,

$$\begin{aligned} \hat{\mathbf{e}}_i &= \mathbf{R}(\mathbf{q})\hat{\mathbf{e}}_{0,i} \\ \mathbf{r}_i &= \mathbf{R}(\mathbf{q})\mathbf{r}_{0,i}, \end{aligned} \tag{2.4}$$

where $\mathbf{R}(\mathbf{q}) \in \mathbb{R}^{3\times3}$ represents the lower link orientation with respect to the upper link and $\hat{\mathbf{e}}_{0,i} \in \mathbb{R}^2$ and $\mathbf{r}_{0,i} \in \mathbb{R}^2$ represent the nominal joint axes and actuator moment arms given $\mathbf{q} = \mathbf{0}$. Next, compute the direction of travel of each linear actuator,

$$\hat{\mathbf{n}}_i = \frac{\mathbf{r}_i - \mathbf{p}_i}{\|\mathbf{r}_i - \mathbf{p}_i\|}, \tag{2.5}$$

given the position of the upper actuator attachment point, $\mathbf{p}_i$. Finally, compute the joint-space torque vector by projecting the moments applied by each actuator onto the current joint axes, i.e.

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{e}}_1 & \hat{\mathbf{e}}_2 \end{bmatrix}^T \begin{bmatrix} \mathbf{r}_1 \times \hat{\mathbf{n}}_1 & \mathbf{r}_2 \times \hat{\mathbf{n}}_2 \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \end{bmatrix}. \tag{2.6}$$

Note that (2.6) is in the form of (2.3). It follows that

$$\mathbf{J} = \begin{bmatrix} \mathbf{r}_1 \times \hat{\mathbf{n}}_1 & \mathbf{r}_2 \times \hat{\mathbf{n}}_2 \end{bmatrix}^T \begin{bmatrix} \hat{\mathbf{e}}_1 & \hat{\mathbf{e}}_2 \end{bmatrix}. \tag{2.7}$$

The hip and ankle mechanisms of THOR are designed such that (2.7) is non-singular throughout the range of motion of the universal joint. Thus, the forward velocity kinematics and inverse statics can be computed using the inverse Jacobian via

$$\dot{\mathbf{q}} = \mathbf{J}^{-1}\dot{\boldsymbol{\ell}}, \tag{2.8}$$

and

$$\mathbf{f} = \mathbf{J}^{-T}\boldsymbol{\tau}. \tag{2.9}$$

The 2 DOF parallel mechanism can be modeled as a closed-loop kinematic graph. To simplify dynamic computations, we approximate the full system using a joint-space kinematic tree model and convert joint torques to actuator forces using the inverse statics solution (2.9). In the absence of external contact forces, the joint-space dynamics are given by the standard rigid body dynamics equation,

$$\boldsymbol{\tau} = \mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}), \tag{2.10}$$

where $\mathbf{H}(\mathbf{q})$ is the joint-space inertia matrix and $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is the vector of centrifugal, Coriolis and gravity torques [75]. Thus, given a rigid body model of the articulated system, we can estimate the required joint torques to accelerate each joint by solving the inverse dynamics (2.10).

2.3.4 Joint State Estimation

As shown in Fig. 2.2, each SEA is instrumented with a load cell sensor and incremental motor encoder for force and velocity estimation. The actuator forces, $\mathbf{f}$, are directly measured using the inline load cells. This approach offers high fidelity force sensing without the need to measure the spring displacements using linear potentiometers or encoders. The motor rotor positions, $\boldsymbol{\theta}$, are measured using the pre-transmission incremental encoders.

In order to compute the actuator velocities, $\dot{\boldsymbol{\ell}}$, we first define the actuator length vector, $\boldsymbol{\ell}$, in terms of the actuator force and rotor position measurements. The spring displacement vector can be computed using Hooke's law,

$$\boldsymbol{\ell}_S = -\frac{1}{k_S}\mathbf{f}, \tag{2.11}$$

and the ball screw travel vector can be computed using the ball screw pitch and motor pulley ratio defined in Table 2.2,

$$\boldsymbol{\ell}_M = \frac{\gamma}{2\pi b}\boldsymbol{\theta} + \mathbf{C}. \tag{2.12}$$

Here $\mathbf{C}$ is a constant position offset depending on the initial actuator configuration. Substituting (2.11) and (2.12) into (2.1) and differentiating with respect to time we obtain a solution for the actuator velocity vector, i.e.

$$\dot{\boldsymbol{\ell}} = -\frac{1}{k_S}\dot{\mathbf{f}} + \frac{\gamma}{2\pi b}\dot{\boldsymbol{\theta}}. \tag{2.13}$$

The rotor position and actuator force signal are differentiated using a first order derivative filter $D(s) = (s \cdot \omega_c)/(s + \omega_c)$ with a corner frequency of $\omega_c = 2\pi \cdot 50$ rad/s to attenuate high frequency noise.

The hip and ankle assemblies are also outfitted with 15-bit absolute encoders on each joint axis to enable direct measurement of the joint position vector, $\mathbf{q}$. The absolute joint encoders are unaffected by position estimation errors otherwise incurred due to unmodeled flex in the actuator structural components. Using the joint position measurements, the actuator position offset, $\mathbf{C}$, is computed during initialization by solving the inverse position kinematics. The measured joint positions are also used to compute the mechanism Jacobian (2.7) at each time step. Joint torque estimates are computed by solving the forward statics (2.3) given the actuator force measurements. Likewise, joint velocity estimates are obtained by solving the forward velocity kinematics (2.8) given the actuator velocity measurements, i.e.

$$\dot{\mathbf{q}} = \mathbf{J}^{-1}\left(-\frac{1}{k_S}\dot{\mathbf{f}} + \frac{\gamma}{2\pi b}\dot{\boldsymbol{\theta}}\right). \tag{2.14}$$

2.4 Control Design

This section presents an overview of the proposed controller architecture shown in Fig. 2.4. As in [66–68], we implement a cascaded joint impedance controller with an inner-loop force controller. Unlike many SEA implementations, the proposed force controller commands motor currents as opposed to motor velocities. We will begin by discussing system identification techniques for the linear series elastic actuators presented in Section 2.3 and will subsequently summarize the proposed force and impedance controllers.

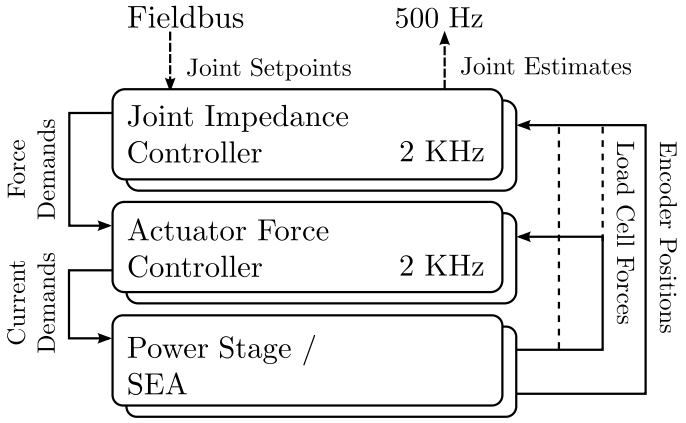

Figure 2.4: High-level block diagram of cascaded impedance controller for 2 DOF joints.

2.4.1 SEA System Identification

As in [39] the linear SEA presented in Section 2.3 is modeled as a second order forced mass-spring-damper system shown in Fig. 2.5. Here k_S is the spring stiffness, M_{EQ} is the lumped sprung mass including the reflected rotor inertia and the combined mass of the ball screw and motor housing, f_M is the linear force applied by the motor, and b_M is the transmission damping coefficient due to friction. The motor force, f_M, is linearly proportional to the motor current, i_M, and can be computed using $f_M = k_f \cdot i_M$, where the motor force coefficient is given by

$$k_f = \frac{2\pi k_\tau b}{\gamma / e}. \tag{2.15}$$

Here k_τ is the motor torque constant, b is the motor pulley ratio, γ is the ball screw pitch, and e is the transmission efficiency.

The lumped model characterizes the actuator dynamics in scenarios where the effective output impedance is high, e.g. when the end-points are connected to ground and the actuator length remains constant. During operation, the effective impedance of each end-point varies depending on the joint configuration. To account for moving output inertias, a more accurate

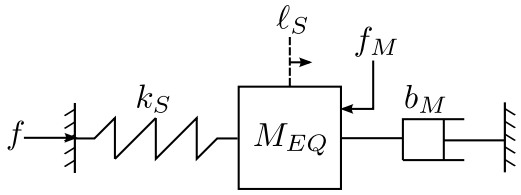

Figure 2.5: Forced mass-spring-damper model of a series elastic actuator with high output impedance.

rack and pinion model was defined in [76]. In practice, however, the high impedance model remains a useful approximation for estimation and control. The open-loop transfer function is given by

$$\frac{F}{F_M} = \frac{\frac{k_S}{M_{EQ}}}{s^2 + \frac{b_M}{M_{EQ}}s + \frac{k_S}{M_{EQ}}}, \tag{2.16}$$

where the plant output, f, represents the spring reaction force measured by the load cell sensor. As discussed in [76], the difference between the spring force and actuator output force is negligible for frequencies less than 100 Hz given the mechanical design of our actuators. This is well above the desired control bandwidth of the linear actuator. Thus, for control purposes we assume that the actuator force acting on the joint mechanism is equal to the spring reaction force.

Experimental system identification was used to estimate the unknown model parameters of the second order system (2.16). The open-loop frequency response, $\frac{F}{I_M}$, was computed by exciting the THOR SEA with a predefined current signal and measuring the force response using the inline load cell. The current excitation consisted of a 2.5 A sinusoidal chirp signal with 150 Hz band-limited white noise. During this process, the actuator end-points where locked at a fixed length to emulate infinite output impedance, and the force response was sampled at 400 Hz. The frequency response function (FRF) was computed using the H1 estimate, $H(s) = \frac{S_{yu}(s)}{S_{uu}(s)}$, where $S_{yu}(s)$ is the cross-spectrum of the response and excitation signals and $S_{uu}(s)$ is the auto-spectrum of the excitation signal.

The transfer function coefficients in (2.16) were estimated by fitting a second order model to the experimental frequency response data, and the motor force coefficient, k_f, was estimated by computing the D.C. gain. Fig. 2.6 shows the H1 estimate of the open-loop frequency response function and the resulting system model. Table 2.3 lists the theoretical and empirical model parameters obtained through system identification. For the theoretical D.C. gain calculation we assumed $e = 0.95$ and $k_\tau = 0.0255$ Nm/A based on the estimated mechanical efficiency and nominal motor specifications.

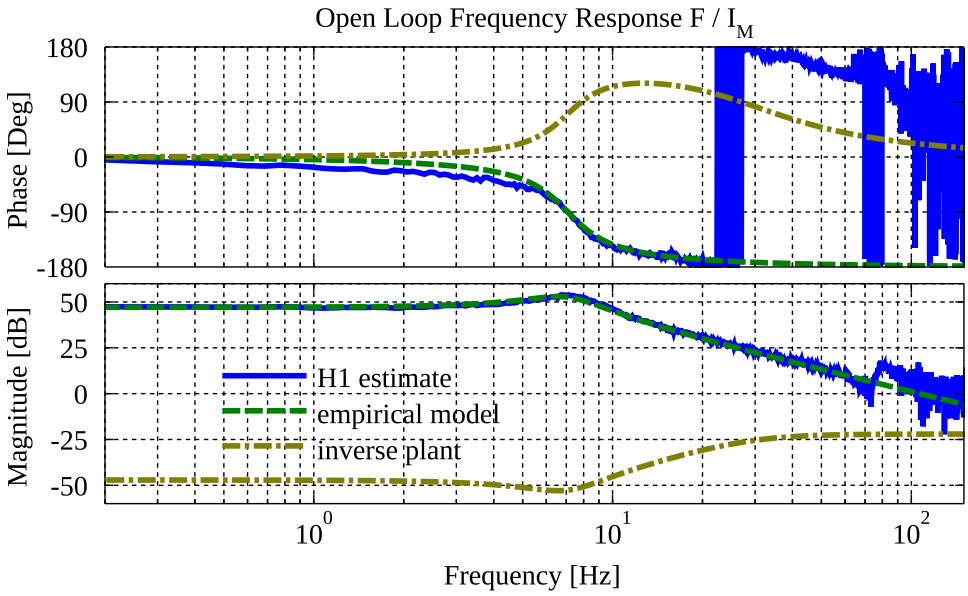

Figure 2.6: Frequency response of the open-loop SEA plant obtained through experimental system identification. The current excitation consisted of a 2.5 A sinusoidal chirp signal with 150 Hz band-limited white noise. The estimated inverse plant model was used in the disturbance observer for the actuator force controller described in Section 2.4.2.

Table 2.3: Open Loop System Parameters

Parameter		Theoretical	Empirical
Natural Frequency	$\omega_n = \sqrt{\frac{k_S}{M_{EQ}}}$	7.50 Hz	7.08 Hz
Damping Ratio	$\zeta = \frac{b_M}{2\sqrt{k_S \cdot M_{EQ}}}$	unknown	0.27
D.C. Gain	$k_f = \frac{2\pi k_\tau b}{\gamma / e}$	228.3 N/A	226.6 N/A

2.4.2 Actuator Force Controller

The actuator force controller is designed to track the desired force reference signal, $\mathbf{f}_d = [\, f_{d_1}\ f_{d_2}\,]^T$, commanded by the joint impedance controller. Each linear actuator is controlled using an independent SISO force controller that commands desired motor currents. Similar to [40] and [41], we use a combination of proportional-integral-derivative (PID) control with unity feedforward and a model-based disturbance observer (DOB). However, unlike these approaches which design the DOB around a nominal closed-loop plant, we design the DOB around the open-loop plant using the empirical SEA model described in Section 2.4.1. As discussed below, this approach simplifies tuning of control gains given our controller architecture.

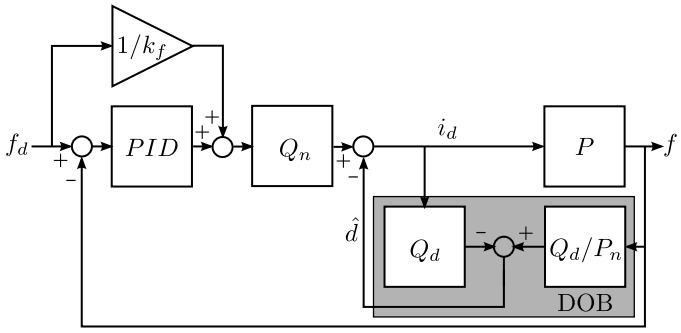

Figure 2.7: SISO force controller including PID feedback with unity feedforward and a disturbance observer (DOB) based on an empirical model of the open-loop plant, P_n.

A block diagram of the proposed force controller is shown in Fig. 2.7. Here the plant P includes the power stage and SEA. A standard PID controller is used to regulate the force error signal with the transfer function,

$$PID(s) = \left(K_p + \frac{1}{s} K_i + \frac{s \cdot \omega_c}{s + \omega_c} K_d \right), \tag{2.17}$$

where $K_p = 30$ mA/N, $K_i = 2$ mA/Ns, and $K_d = 0.25$ mA/N/s are the proportional, integral, and derivative gains selected for the experiments presented in Section 3.7. A first order low-pass filter ($\omega_c = 2\pi \cdot 50$ rad/s) is applied to the derivative term to compensate for high frequency noise in the error signal. Additionally, a unity feedforward signal is included to offset the D.C. component of the error signal. The current reference signals, $\mathbf{i}_d = [\, i_{d_1}\ i_{d_2}\,]^T$, are converted into PWM signals that drive the respective power stages of the associated linear actuators.

In order to avoid exciting high frequency modes in the mechanical design, a second order notch filter is applied to the feedforward and feedback paths. The transfer function is given by

$$Q_n(s) = \frac{s^2 + \omega_n^2}{s^2 + \frac{\omega_n}{Q} s + \omega_n^2}, \tag{2.18}$$

where ω_n represents the desired notch frequency. For the experiments presented in Section 3.7 we selected $\omega_n = 2\pi \cdot 249$ and $\omega_n = 2\pi \cdot 340$ rad/s ($Q = 5$) for the hip and ankle actuators, respectively, based on frequency response data obtained using a spectrum analyzer.

In practice, unmodeled dynamics in the SEA plant can result in poor tracking performance, especially at low reference frequencies. Disturbance observers can be used to compensate for nonlinear friction forces, time-varying output impedances, and other unmodeled disturbances. A disturbance observer estimates the linear disturbance signal signal, $\hat{d}$, by comparing the commanded reference signal with the expected reference signal given the nominal plant dynamics, P_n. This requires the use of an inverse plant filter, $1/P_n$, to compute the expected reference from the output signal. In [40] and [41] the authors design DOBs around the closed-loop dynamics given a nominal controller. In this approach, the disturbance signal is subtracted from the force reference signal, f_d. In our approach a DOB is designed around the nominal open-loop dynamics to estimate disturbances in the current reference signal, i_d. Using this design the inverse plant filter remains independent of the force controller, allowing the gains to be modified online without updating the nominal plant.

In the proposed approach, P_n is given by the open-loop SEA transfer function (2.16) obtained through experimental system identification. The raw inverse plant filter, $1/P_n$, has a relative degree of -2 and is therefore non-causal. To make the filter realizable, we apply a second order low-pass Butterworth filter,

$$Q_d(s) = \frac{\omega_c^2}{s^2 + \sqrt{2}\omega_c s + \omega_c^2}, \tag{2.19}$$

with $\omega_c = 2\pi \cdot 30$ rad/s. The modified transfer function, Q_d/P_n, attenuates high frequency noise that appears in the disturbance estimate. The frequency response of the inverse plant filter is shown in Fig. 2.6. To avoid damaging the motor, the current demand is limited to ± 10 A using a saturation function. Because the current demand is limited after the disturbance observer feedback path, the disturbance estimate can diverge whenever the actuator saturates unless appropriate safeguards are implemented. To account for this issue, we disable the disturbance observer feedback whenever the current exceeds 95% of the maximum value.

In order to assess the performance of the actuator force controller, we repeated the system identification procedure discussed in Section 2.4.1 for the closed-loop system. The force excitation signal consisted of a 100 N sinusoidal chirp signal with 150 Hz band-limited white noise. The resulting frequency response is shown in Fig. 2.8 in addition to a fourth order all-pole model. The force bandwidth of the proposed controller is approximately 30 Hz which is comparable to results found throughout the SEA control literature. The stable resonance near 80 Hz is believed to be the result of a bending mode due to ball screw tipping. The controller step response is shown in Fig. 2.9 for various derivative gains.

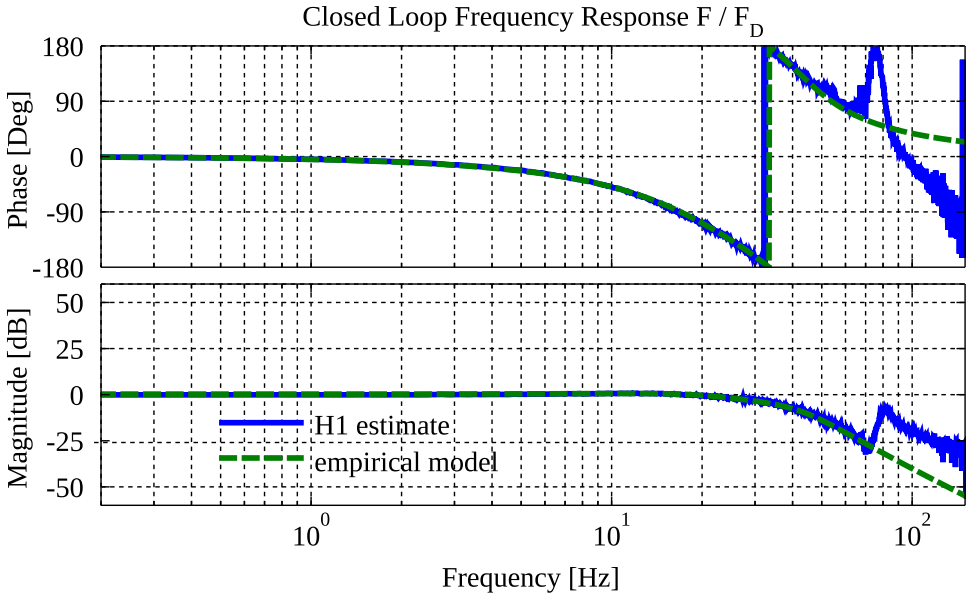

Figure 2.8: Frequency response of the closed-loop SEA plant obtained through experimental system identification. The force excitation consisted of a 100 N sinusoidal chirp signal with 150 Hz band-limited white noise. The closed-loop force bandwidth is approximately 30 Hz.

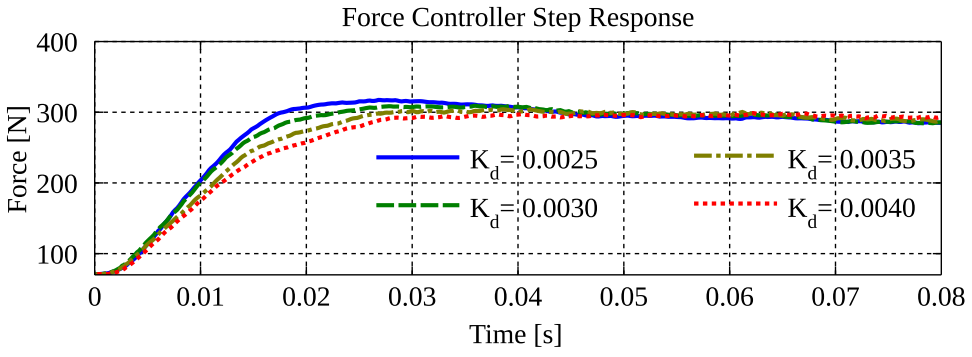

Figure 2.9: 300 N step response of the closed-loop SEA plant for various derivative gains (expressed in mA/N/s).

2.4.3 Joint Impedance Controller

Joint-space control is implemented using the "simple" impedance controller shown in Fig. 2.10. The proposed controller is designed to track joint position, velocity, and torque setpoints which are consistent with a desired acceleration profile for each joint axis. These setpoints are updated at 500 Hz by a high-level field computer that computes feedforward joint torque setpoints given desired joint-space and task-space accelerations using an optimization-based inverse dynamics solver.

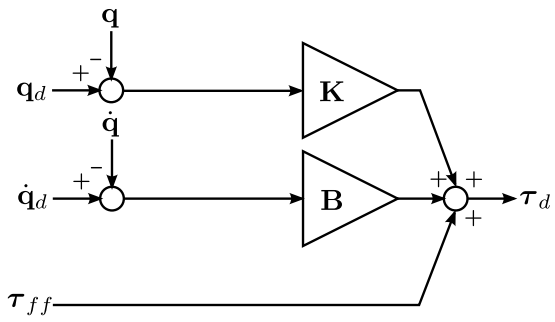

Figure 2.10: "Simple" joint impedance controller with proportional-derivative (PD) feedback and feedforward torque compensation.

In practice, modeling errors can result in poor acceleration tracking using feedforward torque control. The proposed joint impedance controller compensates for unmodeled dynamics by adding explicit position and velocity feedback to the joint torque controller. The desired output torque vector is given by

$$\boldsymbol{\tau}_d = \underbrace{\begin{bmatrix} k_1 & 0 \\ 0 & k_2 \end{bmatrix}}_{\mathbf{K}} (\mathbf{q}_d - \mathbf{q}) + \underbrace{\begin{bmatrix} b_1 & 0 \\ 0 & b_2 \end{bmatrix}}_{\mathbf{B}} (\dot{\mathbf{q}}_d - \dot{\mathbf{q}}) + \boldsymbol{\tau}_{ff}, \tag{2.20}$$

where $\mathbf{q}_d \in \mathbb{R}^2$ and $\dot{\mathbf{q}}_d \in \mathbb{R}^2$ represent the desired joint position and velocity vectors and $\boldsymbol{\tau}_{ff} \in \mathbb{R}^2$ represents the vector of feedforward joint torques. Scalars k_i and b_i, for $i = 1, 2$, represent the joint stiffness and damping coefficients, respectively. This controller is equivalent to a proportional-derivative (PD) controller with feedforward torque compensation. Because joint torque control results in a nominal joint-space inertia of zero, the closed-loop system behaves like a virtual torsional spring and damper in parallel with a joint torque source.

By varying the joint stiffness and damping coefficients, we can emulate a variety of joint control strategies. Pure torque control can be implemented with zero PD feedback. Low-gain PD feedback can help improve acceleration tracking while operating in a low stiffness mode to support compliant interaction, and high-gain PD feedback can be used to compensate for large disturbances while operating in a high stiffness mode.

Given the desired joint torques, the actuator force setpoints are computed by solving the inverse statics of the parallel mechanism,

$$\mathbf{f}_d = \mathbf{J}^{-T}\boldsymbol{\tau}_d. \tag{2.21}$$

Although quasistatic equilibrium assumptions are often violated during normal operation, this approach serves as a practical, approximate solution for distributed joint control. By solving the joint statics and kinematics locally, the task-space and joint-space dynamics are effectively decoupled. The local controller is isolated from the system dynamics (2.10), and the high-level controller is isolated from the underlying joint mechanics. This is a major benefit of the proposed mechatronics and control approach.

As discussed in [68], the stability of the joint impedance controller is dependent on the inner-loop force controller bandwidth. Intuitively, both the outer and inner-loop feedback terms contribute to the maximum overall loop gain, which is constrained by the actuator dynamics, derivative filtering, etc. [68]. As a result, it may be necessary to lower the force controller gains online in order to support high joint impedances. By incorporating a disturbance observer based on the open-loop actuator dynamics, we are able to modify the force controller gains without updating the nominal plant model used by the DOB. This simplifies tuning of the cascaded impedance controller by allowing a single empirical open-loop model to be used for disturbance rejection, independent of the controller gains.

For the experiments presented in the following section, the inner-loop force controller gains were tuned experimentally to obtain a critically damped response given the desired force bandwidth (approximately 30 Hz at 10% of the peak force amplitude). Desired outer-loop impedances were determined experimentally through locomotion and trajectory tracking experiments.

2.5 Experimental Results

The proposed joint-space controller was verified through a series of experiments conducted using the THOR hardware platform. In this section we present test results for torque and impedance tracking using the parallel hip and ankle mechanisms. We also demonstrate an application of the proposed controller for compliant locomotion using an optimization-based whole-body controller to generate joint setpoints.

2.5.1 Torque Control

Fig. 2.11 compares the joint torque tracking performance for the ankle pitch-roll mechanism with and without the use of a disturbance observer in the actuator force controller. For this experiment, the shin and foot were constrained using a test fixture to prevent joint rotation.

A 25 Nm sinusoidal chirp trajectory ranging from 0 to 1.5 Hz was commanded on the ankle pitch axis, and a 0 Nm reference was commanded on the ankle roll axis. The results show that the use of a current DOB significantly reduces torque errors resulting from unmodeled friction forces and dynamic coupling between the parallel actuators. Although the pitch torque amplitude is approximately 10% of the maximum load, the roll torque error remains less than 0.8 Nm.

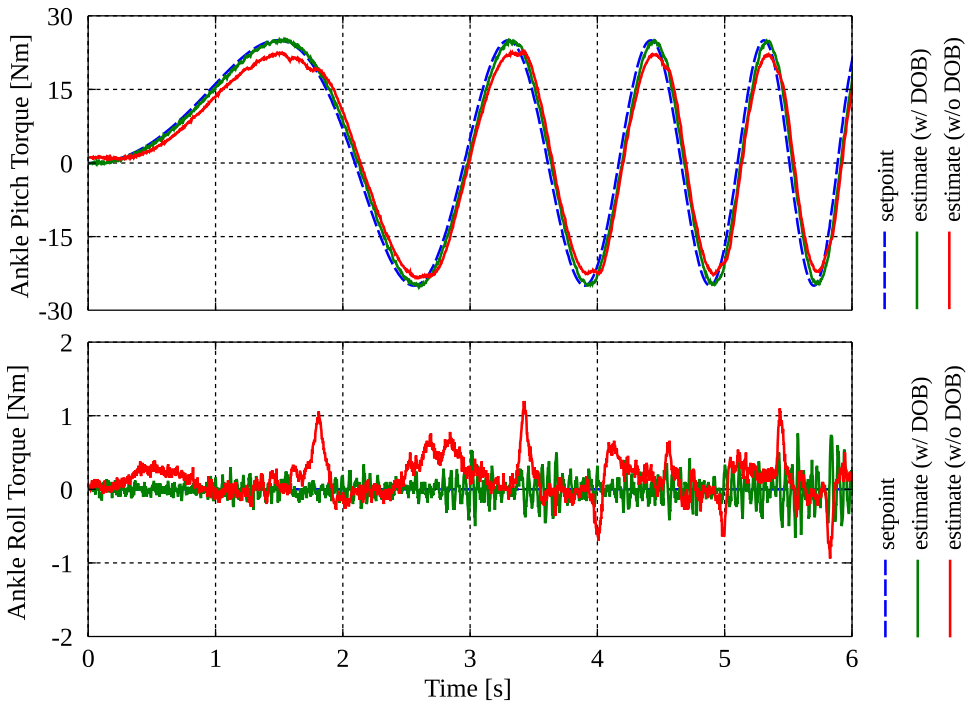

Figure 2.11: Ankle pitch-roll torque tracking performance with and without disturbance observer.

2.5.2 Impedance Control

To assess the performance of the outer-loop impedance controller, a desired 2 DOF joint reference trajectory was commanded on the hip yaw-roll mechanism. A partial time lapse of the test trajectory is show in Fig. 2.12. In order to permit unrestricted motion of the hip and ankle assemblies, the robot's torso link was mounted to a fixed base frame, allowing the legs to swing freely. The hip position setpoints were defined by the sinusoidal chirp signal, $\mathbf{q}_d(t) = 1/5 \cdot [\sin(5\pi/12 \cdot t^2) \quad \sin(5\pi/24 \cdot t^2)]^T$ rad/s, over the duration $t \in [0, 6]$ s. Desired velocity and acceleration setpoints were computed via the analytical derivatives, and feedforward torque setpoints were computed using the recursive Newton-Euler algorithm [75] given a rigid body model of the 11.1 kg leg.

Moderate stiffness and damping coefficients were chosen for each joint axis to test compliant

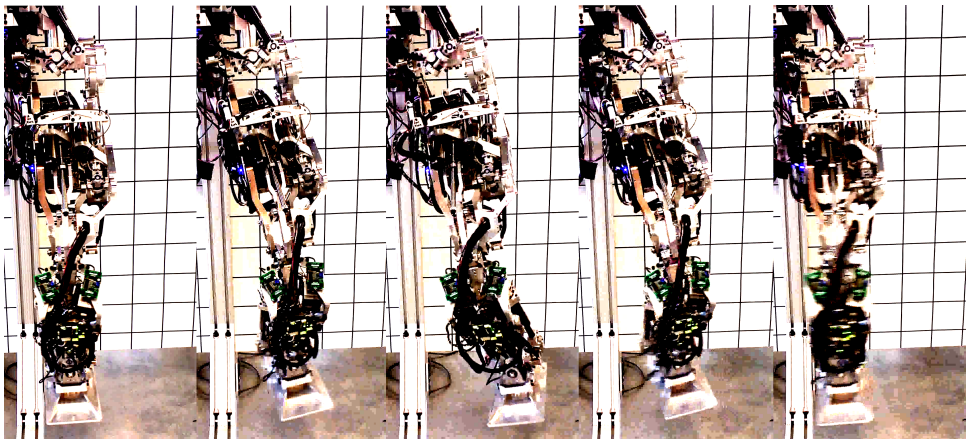

Figure 2.12: THOR leg trajectory while tracking a sinusoidal chirp signal using the hip roll and yaw joints.

trajectory tracking (yaw: $k = 200$ Nm/rad, $b = 10$ Nm/rad/s, roll: $k = 400$ Nm/rad, $b = 20$ Nm/rad/s). The roll axis was assigned a higher impedance to account for the larger relative rotational inertia. The desired and estimated position, velocity, and torque trajectories for the yaw and roll axes are shown in Fig. 2.13. Note that the computed feedforward torques closely match the measured output torques, allowing accurate trajectory tracking despite relatively low proportional-derivative gains. As a point of reference, in [64] the authors selected gains of $k = 4500$ Nm/rad and $b = 250$ Nm/rad/s to test high stiffness and damping control of a similarly proportioned robot.

Fig. 2.14 shows the hip roll velocity estimates obtained using the proposed load cell and incremental encoder algorithm (2.14) versus direct numerical differentiation of the joint position signal obtained from the absolute joint encoder. The low relative resolution of the post-transmission absolute encoder results in significant quantization error in the derivative estimate despite heavy low pass filtering ($\omega_c = 2\pi \cdot 50$ Hz). The proposed algorithm produces a more accurate joint velocity estimate by leveraging the higher resolution SEA sensor measurements via the forward velocity kinematics solution.

In a related experiment, the hip impedance controller was assigned stiffness and damping coefficients differing by one order of magnitude between each joint axis. The desired feedforward torques were set to zero in order to test the joint-space proportional-derivative controller in the presence of exogenous forces. The hip yaw stiffness was set to one tenth the value of the hip roll and the hip yaw-roll joint was manually rotated to generate interaction forces. Suitable damping coefficients where chosen for each joint axis to ensure stability of the controller. The measured output torque versus displacement for the two targeted joint stiffnesses is shown in the top row of Fig. 2.15. The apparent hysteresis in the torque curves can be partially attributed to the torque contribution of the derivative action. In order to test the accuracy of the derivative controller, the experiment was repeated with zero stiffness. The hip yaw damping was set to one tenth of the hip roll damping, and the hip yaw-roll

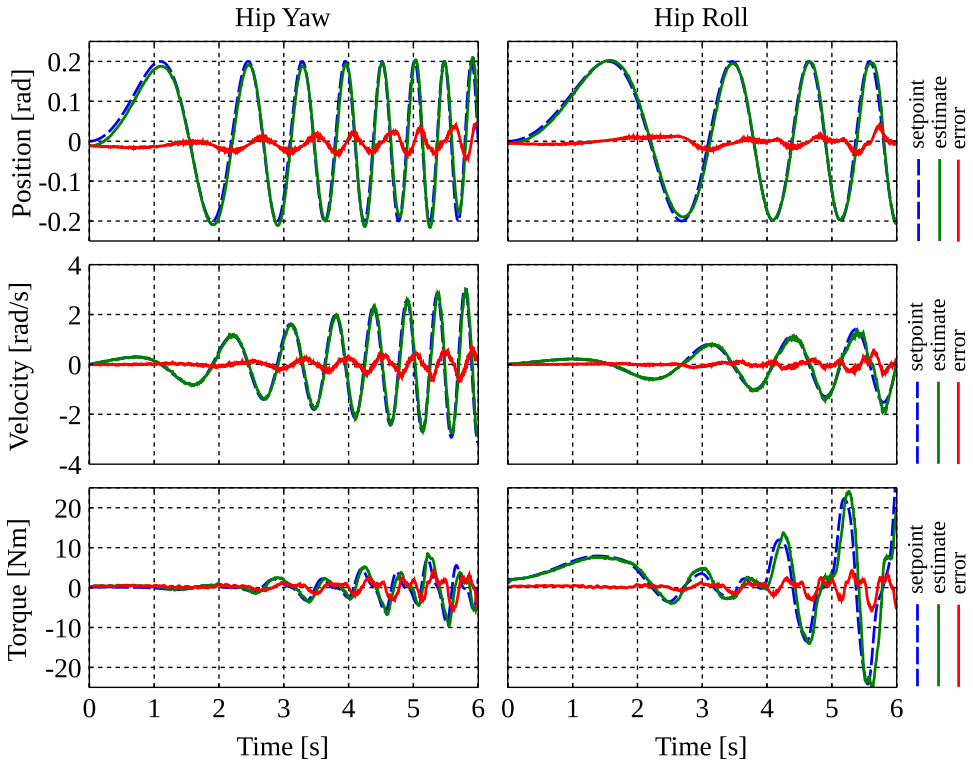

Figure 2.13: Position, velocity, and torque setpoints and estimates while tracking a sinusoidal chirp signal using the hip roll and yaw joints using low-impedance control (yaw: $k = 200$, $b = 10$, roll: $k = 400$, $b = 20$).

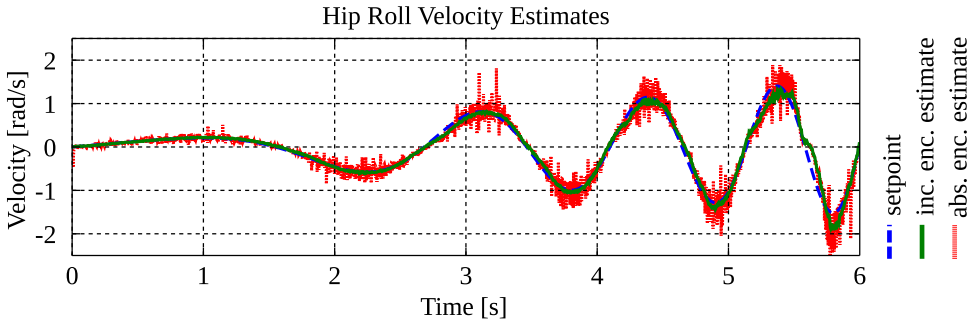

Figure 2.14: Velocity estimates obtained using the proposed incremental encoder / load cell algorithm and first order differentiation of the absolute joint encoder.

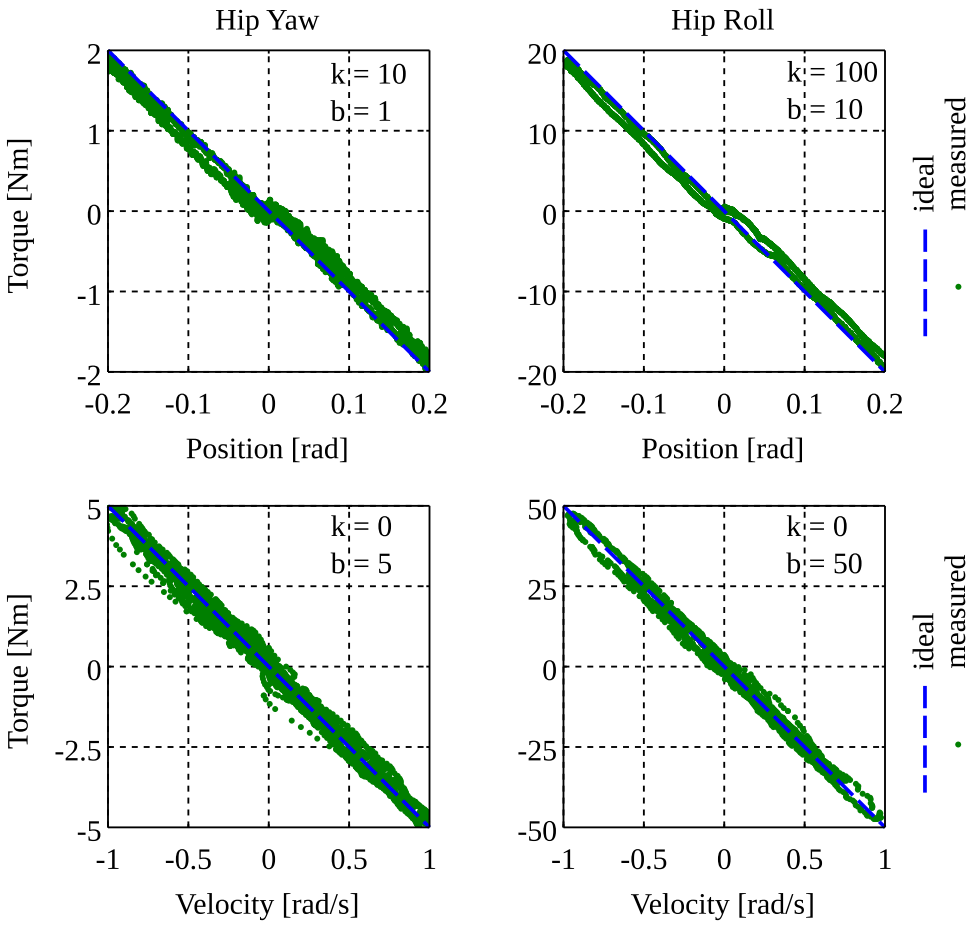

Figure 2.15: Top row: ideal and actual torque vs position for independent hip yaw and roll stiffness coefficients. Bottom row: ideal and actual torque vs velocity for independent hip yaw and roll damping coefficients.

joint was rotated as before. The measured output torque versus angular velocity for the two targeted damping coefficients is shown in the bottom row of 2.15. The results demonstrate that the proposed controller is able to achieve independent joint impedances along each axis.

2.5.3 Compliant Locomotion

Fig. 2.16 shows the THOR hardware platform walking forward using the proposed joint-space controller. The stride length is 0.175 m and the step duration is 3.5 s with a double support phase of 1.3125 s. Given desired momentum rates of change and end-effector accelerations from a high-level step controller, we compute joint torque and acceleration setpoints using an optimization-based whole-body controller similar to [24; 25]. Constraints on the centroidal dynamics, frictional contact forces, and joint position/torque limits are included to ensure admissibility of the torques and accelerations. Joint position and velocity setpoints are computed using leaky integration of the optimized accelerations to enable compliant control.

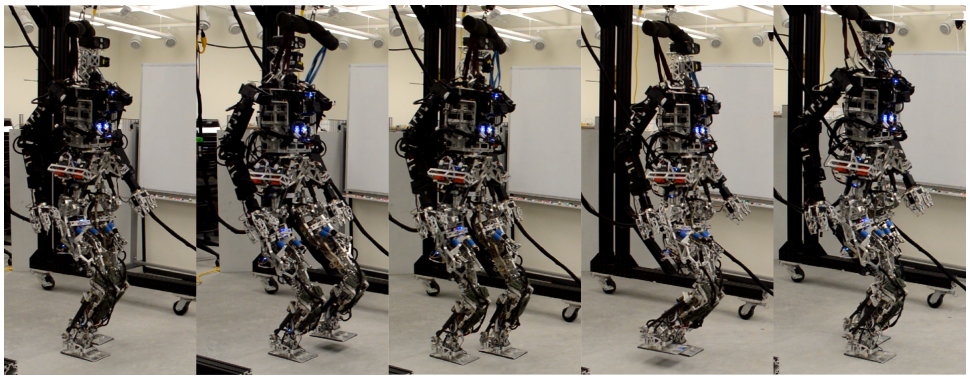

Figure 2.16: THOR walking on cement.

The proposed joint torque controller enables stable locomotion given adequate damping feedback to account for unmodeled dynamics and mechanical resonances. We found that joint position feedback is generally not necessary for our application as outer-loop feedback is typically used to introduce Cartesian impedances via joint torque control. Nevertheless, joint position feedback may be useful for tasks that require precise joint tracking such as dexterous manipulation. Fig. 2.17 shows the hip yaw and roll torques corresponding to the walking sequence in Fig. 2.16. Note that the peak hip roll torque is an order of magnitude larger than the hip yaw torque. The parallel design of the hip yaw-roll mechanism allows the control authority of both actuators to be allocated primarily to roll actuation, resulting in peak actuator forces that are significantly lower than a serial configuration incorporating the same SEAs.

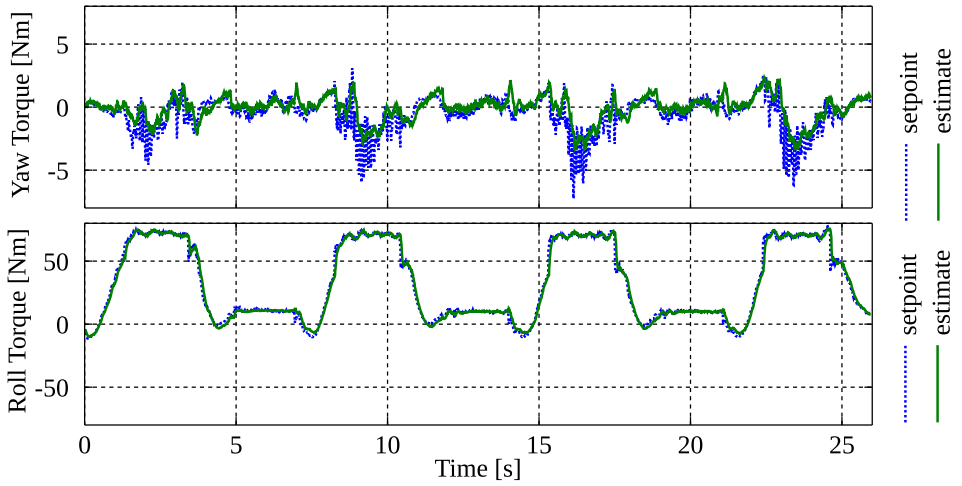

Figure 2.17: Hip yaw-roll torques while walking on cement.

2.6 Conclusion

This paper presented a cascaded joint impedance controller for a class of 2 DOF parallel mechanisms using linear series elastic actuators. The proposed controller was implemented on the parallel hip and ankle mechanisms of the THOR humanoid platform using a novel dual-axis motor controller. Our approach effectively decouples the parallel joint mechanics from the whole-body dynamics by computing feedforward torques for the multi-body system using a joint-space rigid body model and converting joint torques to actuator forces using an inverse statics solution computed by the motor controller microprocessor. The use of a model-based disturbance observer based on the open loop actuator dynamics simplifies tuning of the cascaded impedance controller by allowing the actuator force controller gains to be modified independently of the inverse plant model.

Experimental results obtained using the THOR humanoid verify the effectiveness of the proposed approach for compliant legged locomotion applications. Although we presented results for a class of 2 DOF parallel mechanisms, the proposed approach is easily extensible to other fully-actuated parallel and serial mechanisms. Future efforts will focus on improving the proposed actuator force controller using a novel linear series elastic actuator model proposed in [76] to account for moving output inertias. We are also interested in pursuing more extensive characterization of the stability of the joint impedance controller through the development of an external position-controlled end-effector designed to excite the parallel mechanism with high precision. Additional research will focus on development of similar approaches for over-actuated and under-actuated parallel rotational manipulators for biologically inspired robots.

Chapter 3

Optimization-Based Whole-Body Control of a Compliant Humanoid

3.1 Abstract

As whole-body control approaches begin to enter the mainstream of humanoid robotics research, there is a real need to address the challenges and pitfalls encountered in hardware implementations. This paper presents an optimization-based whole-body control framework enabling compliant locomotion on THOR, a 34 DOF humanoid with series elastic actuation. Given desired momentum rates of change, end-effector accelerations, and joint accelerations from a high-level locomotion controller, joint torque setpoints are computed using an efficient quadratic program (QP) formulation assuming a rigid body model of the robot. Constraints on the centroidal dynamics, frictional contact forces, and joint position/torque limits ensure admissibility of the optimized joint torques and accelerations. Dynamic balancing is implemented using a novel time-varying controller for the Divergent Component of Motion (DCM). In order to validate the effectiveness of the proposed approach, push recovery and compliant walking are demonstrated on the THOR hardware platform. We discuss practical considerations and adaptations that led to the successful implementation of optimization-based whole-body control on our hardware system.

3.2 Introduction

Compliant bipeds and quadrupeds are becoming increasingly prevalent as researchers attempt to mimic the speed and adaptability of locomotion behaviors found in nature [43; 48; 49; 54; 60]. This has created an interest in locomotion strategies that are robust to unmodeled dynamics encountered by hardware systems operating in real environments.

The problem is further complicated in the field of humanoid robotics, where multi-objective whole-body control is required to implement many common behaviors given the inherent high dimensionality and nonlinearity of the dynamics.

Early whole-body control formulations implemented multi-objective control with dynamic constraints using null-space projection methods [21]. Recently, a number of researchers have proposed convex optimization techniques to solve the inverse dynamics and whole-body control problem given multi-contact constraints [22–29]. These approaches compute admissible joint torques to minimize tracking errors for multiple motion tasks including desired momentum rates of change, end-effector accelerations, and joint accelerations. In general, these formulations can serve as the basis for dynamic locomotion, whole-body manipulation, or any behavior requiring multi-contact control.

In [23], the authors proposed an optimization-based framework to compute joint accelerations given whole-body momentum rate of change objectives. In [24], this approach was extended to simultaneously optimize contact forces using a model-based quadratic program (QP). In the proposed formulation multiple motion constraints are included to enable task-space control and a Newton-Euler constraint ensures admissibility of the ground reaction forces and joint accelerations. In [29], the authors implemented a similar approach, including a quadratic cost term to stabilize the center of mass (CoM) dynamics. In [27], accelerations, torques, and contact forces were optimized using a QP formulation that minimizes the weighted error of various motion tasks [27].

Several authors have proposed extensions to task-space control and inverse dynamics solvers to support hierarchical prioritization of motion tasks. In [22], the authors solved a sequence of linearly constrained QPs to enforce strict prioritization of multiple task-space objectives. A similar approach was implemented in [28] including joint torque and position limits. In [26], a hierarchical whole-body controller was implemented using an alternate conic optimization formulation, and in [25] the authors implemented a hierarchical inverse dynamics solver with reduced floating-base dynamic constraints.

Implementing model-based whole-body control on real hardware platforms is exceptionally challenging due to numerous issues arising from communication delays, actuator friction, mechanical bias, structural compliance, and a variety of other electromechanical phenomena. In [25], the authors demonstrated successful balancing using hierarchical inverse dynamics on a purely torque-controlled biped. Most implementations, however, have required some form of joint-space position and velocity feedback to compensate for unmodeled dynamics. As an example, a hybrid inverse kinematics and inverse dynamics controller was implemented in [30] to achieve successful locomotion on the ATLAS robot using PD feedback in conjunction with torque control.

Dynamic balancing is typically implemented through the design of feedback controllers to stabilize the robot's centroidal dynamics. This is made possible through control of the robot's centroidal momentum rate of change via direct optimization of external contact forces. Several authors implement Cartesian PD controllers to track desired CoM trajectories

using linear momentum rate of change objectives [23; 25; 26]. In [29] the authors minimized a time-varying LQR cost function for the Zero Moment Point (ZMP) dynamics directly in the quadratic program formulation.

In [8], the authors introduced the Capture Point (CP) transformation which allows the horizontal centroidal dynamics to be stabilized using a simple CMP-based control law [10; 77]. As shown in [13], this controller is equivalent to the best CoM-ZMP regulator [78], known to maximize the horizontal stability margins for CoM tracking. In [24], CP tracking was implemented using a momentum-based whole-body controller. In [11], the authors introduced the three-dimensional Divergent Component of Motion (DCM), which constitutes an extension of the CP transformation that simplifies locomotion planning and control on uneven terrain.

In this paper we present an implementation of optimization-based whole-body control and compliant locomotion on a new torque-controlled humanoid, THOR [2]. Inspired by the work of [24] and [25], we propose an efficient quadratic program formulation based on the centroidal dynamics that includes frictional contact constraints and joint position and torque limits. The centroidal dynamics are stabilized using a novel momentum controller based on the time-varying DCM dynamics, implemented for the first time on hardware. These components are assembled into a general framework for compliant locomotion on uneven terrain.

As whole-body control approaches begin to enter the mainstream of humanoid robotics research, there is a real need to address the challenges and pitfalls encountered in real hardware implementations. We discuss practical considerations and adaptations that led to the successful implementation of optimization-based whole-body control on our hardware system. Here joint-space control and state estimation serve as a primary concern. A cascaded joint-space impedance controller is proposed to achieve reliable torque tracking with low-gain velocity feedback. Joint velocity setpoints are computed using leaky integration of the optimized joint accelerations, and inner-loop damping is implemented using pre-transmission velocity estimates. A simple state estimator is implemented using kinematic Kalman filters for real-time performance. The proposed approach is validated by push recovery and compliant walking experiments using the THOR hardware platform.

3.3 Humanoid Dynamics

In this section we present a summary of humanoid dynamics and provide an overview of the time-varying Divergent Component of Motion transformation used to stabilize the centroidal dynamics during locomotion.

3.3.1 Rigid Body Dynamics

The configuration of an articulated humanoid with n actuated degrees of freedom (DOF) can be expressed by the vector, $\mathbf{q} = \begin{bmatrix} \mathbf{q}_0^T & \mathbf{q}_n^T \end{bmatrix}^T \in \mathbb{R}^{6+n}$, where $\mathbf{q}_0 \in \mathbb{R}^6$ encodes the 6 DOF translation and orientation of the floating-base frame and $\mathbf{q}_n \in \mathbb{R}^n$ represents the n DOF vector of actuated joint positions. The full rigid-body equations of motion are given by

$$\begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau} \end{bmatrix} = \mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) - \sum_c \mathbf{J}_c^T \mathbf{f}_c, \tag{3.1}$$

where $\boldsymbol{\tau} \in \mathbb{R}^n$ is the vector of actuated joint torques, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is the vector of centrifugal, Coriolis and gravity torques, $\mathbf{H}(\mathbf{q})$ is the joint-space inertia matrix, and $\mathbf{J}_c$ and $\mathbf{f}_c \in \mathbb{R}^3$ are the point Jacobians and corresponding reaction forces at each contact point, $\mathbf{r}_c \in \mathbb{R}^3$ [75]. Here we define the contact subscript, $c = 1 : N$, given N frictional contact points.

3.3.2 Centroidal Dynamics

Alternatively, the centroidal dynamics define the reduced equations of motion for the center of mass, $\mathbf{x} = [x_{com}, y_{com}, z_{com}]^T$, and linear and angular momentum of the system, $\mathbf{l} \in \mathbb{R}^3$ and $\mathbf{k} \in \mathbb{R}^3$, as illustrated in Fig. 3.1. The total momentum rate of change about the center of mass (CoM) is given by

$$\dot{\mathbf{h}} = \begin{bmatrix} \dot{\mathbf{l}} \\ \dot{\mathbf{k}} \end{bmatrix} = \sum_c \mathbf{W}_c \mathbf{f}_c + \mathbf{w}_g, \tag{3.2}$$

where $\mathbf{W}_c \in \mathbb{R}^{6\times 3}$ maps contact forces to wrenches acting about the CoM and $\mathbf{w}_g = [0, 0, -mg, 0, 0, 0]^T$ encodes the force of gravity [23; 24]. The momentum rate of change is related to the joint velocities and accelerations by the equality,

$$\dot{\mathbf{h}} = \dot{\mathbf{A}}\dot{\mathbf{q}} + \mathbf{A}\ddot{\mathbf{q}}, \tag{3.3}$$

where $\mathbf{A} \in \mathbb{R}^{6\times 6+n}$ represents the Centroidal Momentum Matrix (CMM). Given the current configuration, $\mathbf{A}$ can be computed using the efficient recursive algorithm defined in [34].

3.3.3 Divergent Component of Motion Dynamics

The three-dimensional DCM, $\boldsymbol{\xi} = \mathbf{x} + \frac{1}{\omega_0}\dot{\mathbf{x}}$, is a linear transformation of the CoM state that divides the linear centroidal dynamics into stable and unstable first-order components where $\omega_0 = \sqrt{\frac{g}{\Delta z}}$ represents the natural frequency of the linear inverted pendulum [11; 79]. For a constant CoM height, Δz, the horizontal projection of the DCM is equivalent to the two-dimensional Capture Point (CP) [8], defined as the point at which the Centroidal Moment Pivot (CMP) [35] must be placed at any time to allow the CoM to come to a complete rest. In [57], we proposed a time-varying extension of the three-dimensional DCM to simplify

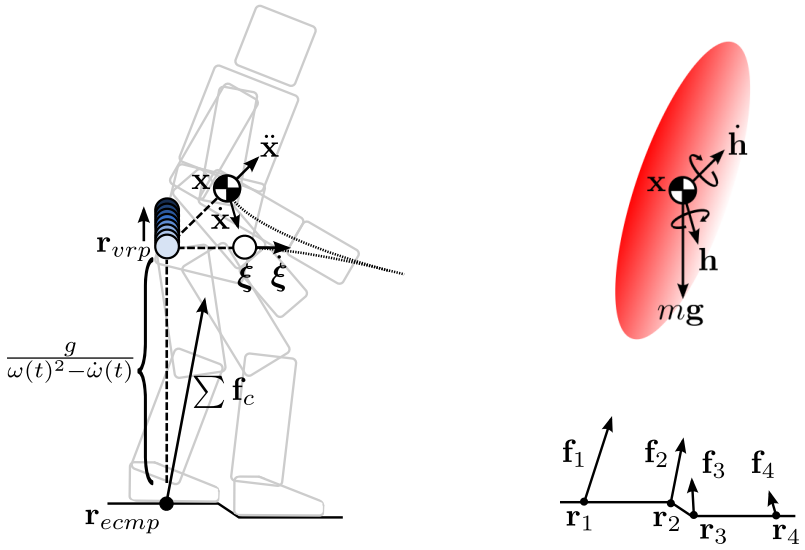

Figure 3.1: Left: Time-varying DCM dynamics. Right: Centroidal dynamics with 4 contacts.

dynamic planning and control of the linear centroidal dynamics for humanoid locomotion. Here we briefly summarize those results.

The time-varying Divergent Component of Motion is defined as

$$\boldsymbol{\xi} = \mathbf{x} + \frac{1}{\omega(t)}\dot{\mathbf{x}}, \tag{3.4}$$

where $\omega(t) > 0$ is the time-varying natural frequency of the DCM. Solving (3.4) for $\dot{\mathbf{x}}$ yields the first-order CoM dynamics,

$$\dot{\mathbf{x}} = \omega(\boldsymbol{\xi} - \mathbf{x}). \tag{3.5}$$

Here we have introduced the relaxed notation, $\omega := \omega(t)$, omitting the natural frequency's explicit dependence on time. The CoM dynamics are asymptotically stable with respect to an equilibrium point at $\boldsymbol{\xi}$. Thus, the linear centroidal dynamics can be indirectly stabilized by regulating the DCM.

As discussed in [57], we can also derive first-order equations of motion for the time-varying DCM in terms of the Virtual Repellent Point (VRP),

$$\dot{\boldsymbol{\xi}} = \left(\omega - \frac{\dot{\omega}}{\omega}\right)(\boldsymbol{\xi} - \mathbf{r}_{vrp})\,. \tag{3.6}$$

First introduced in [11], the Virtual Repellent Point, $\mathbf{r}_{vrp}$, represents the unstable equilibrium point of the DCM dynamics and can be viewed as a three-dimensional analog to the CMP. The time-varying extension of the VRP is defined in terms of the linear momentum rate of change and CoM position,

$$\mathbf{r}_{vrp} = \mathbf{x} - \frac{\dot{\mathbf{l}}}{m\,(\omega^2 - \dot{\omega})}, \tag{3.7}$$

where m is the total mass of the robot.

Using (3.2) we can define the linear momentum rate of change as $\dot{\mathbf{l}} = \sum \mathbf{f}_c - m\mathbf{g}$ where $\mathbf{g} = [\,0\;0\;g\,]^T$ is the gravity vector. Thus, we can alternatively express the VRP as

$$\mathbf{r}_{vrp} = \mathbf{x} - \frac{\sum \mathbf{f}_c - m\mathbf{g}}{m\left(\omega^2 - \dot{\omega}\right)} = \mathbf{r}_{ecmp} + \frac{\mathbf{g}}{\omega^2 - \dot{\omega}}, \tag{3.8}$$

where

$$\mathbf{r}_{ecmp} = \mathbf{x} - \frac{\sum \mathbf{f}_c}{m\left(\omega^2 - \dot{\omega}\right)} \tag{3.9}$$

is the time-varying extension of the enhanced Centroidal Moment Pivot (eCMP) [11]. The eCMP encodes the direction and magnitude of all contact forces acting on the robot and lies on the line passing through the CoM, parallel to the net contact force, $\sum \mathbf{f}_c$. When the eCMP intersects the ground plane it is equivalent to the canonical CMP [35]. In comparison, the VRP encodes the direction and magnitude of all external forces acting on the robot, including gravitational forces.

As illustrated in Fig. 3.1, the horizontal position of the VRP is equivalent to the eCMP, while the vertical position varies depending on the natural frequency trajectory. If the eCMP lies in the base of support, it is possible to avoid generating a horizontal moment about the CoM by placing the center of pressure (CoP) at the eCMP. Thus, through appropriate planning of the eCMP trajectory, the linear centroidal dynamics can be stabilized using a control law based on the VRP without generating significant angular momentum during locomotion.

3.4 Whole-Body Control

In this section we provide an overview of the proposed optimization-based whole-body controller implementation. First, we describe the task-space formulation used to represent motion tasks and frictional contact constraints. Then we discuss the specific quadratic program formulation used to optimize joint accelerations and contact forces given multiple task-space objectives.

3.4.1 Task-Space Formulation

Arbitrary motion tasks such as the acceleration of a Cartesian frame or contact point can be expressed as

$$\dot{\mathbf{v}}_t = \dot{\mathbf{J}}_t\dot{\mathbf{q}} + \mathbf{J}_t\ddot{\mathbf{q}}, \tag{3.10}$$

where $\mathbf{J}_t$ is the associated task-space Jacobian matrix. In this work we model contact points using Coulomb friction and linear acceleration constraints, i.e. $\dot{\mathbf{v}}_c = 0$. Tipping and slipping

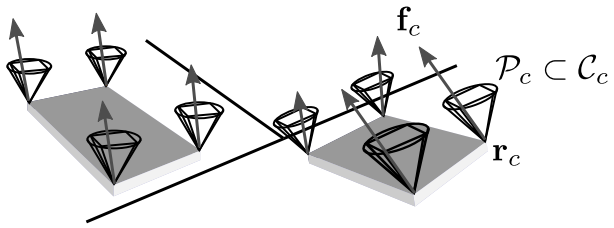

Figure 3.2: Friction cone and friction pyramid approximations for foot contacts.

of supporting surfaces is avoided by ensuring that reaction forces at each contact point remain inside the corresponding friction cone, i.e. $\mathbf{f}_c \in \mathcal{C}_c$ [80], as illustrated in Fig. 3.2.

As in [23; 24; 29; 81], we employ a polyhedral approximation of the friction cone, $\mathcal{P}_c \subset \mathcal{C}_c$, at each contact point, $c = 1 : N$, allowing contact forces to be computed using a linear generating function,

$$\mathbf{f}_c = \boldsymbol{\beta}_c \boldsymbol{\rho}_c. \tag{3.11}$$

The columns of $\boldsymbol{\beta}_c \in \mathbb{R}^{3\times 4}$ are unilateral bases that span the volume of admissible forces in inertial coordinates, and $\boldsymbol{\rho}_c \in \mathbb{R}^4$ is a vector of non-negative coefficients determined via optimization.

3.4.2 Model-Based Optimization (Quadratic Program)

Given multiple motion tasks, $\dot{\mathbf{v}}_t$, the goal of whole-body control is to compute joint torques, $\boldsymbol{\tau}$, that minimize the tracking error for each task while satisfying dynamic constraints including available control authority, range of motion, and limits on the frictional contact forces. Motion tasks may include desired momentum rates of change, spatial accelerations of coordinate frames attached to individual links, or joint-space accelerations computed using an arbitrary high-level controller. Inspired by the work of [24] and [25], the proposed whole-body controller optimizes desired joint accelerations, $\ddot{\mathbf{q}}$, and generalized contact forces, $\boldsymbol{\rho} = \begin{bmatrix} \boldsymbol{\rho}_1^T & \dots & \boldsymbol{\rho}_N^T \end{bmatrix}^T$, using a linearly constrained quadratic program (QP) in the form,

$$\min_{\ddot{\mathbf{q}},\boldsymbol{\rho}} \left\| \mathbf{C}_b \left(\mathbf{b} - \dot{\mathbf{J}}\dot{\mathbf{q}} - \mathbf{J}\ddot{\mathbf{q}} \right) \right\|^2 + \lambda_{\ddot{\mathbf{q}}} \|\ddot{\mathbf{q}}\|^2 + \lambda_{\boldsymbol{\rho}} \|\boldsymbol{\rho}\|^2 \tag{3.12}$$

subject to

$$\dot{\mathbf{A}}\dot{\mathbf{q}} + \mathbf{A}\ddot{\mathbf{q}} = \sum_c \mathbf{W}_c \mathbf{f}_c + \mathbf{w}_g \tag{3.13}$$

$$\underline{\mathbf{q}} \leq \mathbf{q} + T\dot{\mathbf{q}} + \frac{1}{2}T^2\ddot{\mathbf{q}} \leq \bar{\mathbf{q}} \tag{3.14}$$

$$\underline{\boldsymbol{\tau}} \leq \boldsymbol{\tau} \leq \bar{\boldsymbol{\tau}} \tag{3.15}$$

$$\mathbf{0} \leq \boldsymbol{\rho}, \tag{3.16}$$

where $\mathbf{b}$ is the vector of desired motion tasks and $\mathbf{J}$ is the corresponding matrix of stacked Jacobians, i.e.

$$\mathbf{b} = \begin{bmatrix} \dot{\mathbf{h}}_d \\ \ddot{\mathbf{q}}_d \\ \dot{\mathbf{v}}_{d_1} \\ \vdots \end{bmatrix}, \qquad \mathbf{J} = \begin{bmatrix} \mathbf{A} \\ \boldsymbol{I} \\ \mathbf{J}_1 \\ \vdots \end{bmatrix}. \tag{3.17}$$

The QP cost function and constraint equations are described in the subsections below.

3.4.3 Optimization Costs

The cost function (3.12) is designed to minimize the weighted quadratic error of the task-space objectives, $\mathbf{b}$, given a semi-positive definite weighting matrix, $\mathbf{Q}_b = \mathbf{C}_b^T \mathbf{C}_b$. Joint acceleration and contact force regularization terms are also included to ensure that the QP is strictly convex given $\lambda_{\ddot{\mathbf{q}}},\ \lambda_{\boldsymbol{\rho}} > 0$. The weighting matrix allows soft prioritization of motion tasks. High weights are assigned to contact accelerations to ensure contact points remain stationary relative to support surfaces, while low weights are often assigned to angular momentum rate of change objectives to improve dynamic stability. Although this approach does not allow a strict hierarchy of task prioritizations as in [25; 26], we have found that highly-weighted costs can be more forgiving than hard constraints in scenarios where one or more tasks are ill-conditioned.

3.4.4 Optimization Constraints

Constraints (3.13)-(3.16) are linear in the decision variables and ensure admissibility of the optimized joint accelerations and contact forces. Equation (3.13) follows from (3.2) and (3.3) and enforces the centroidal dynamic constraints [24]. The final inequality (3.16) enforces Coulomb friction constraints at each contact point. As proposed in [28], joint range of motion constraints are implemented using (3.14) where $\underline{\mathbf{q}}$ and $\bar{\mathbf{q}}$ represent the joint position limits and T represents a time constant determining the maximum rate of convergence to either limit. For the experiments presented in Section 3.7 we use $T \approx 0.15$ s.

A number of QP formulations include joint torques, $\boldsymbol{\tau}$, as additional decision variables to permit explicit torque limits in the optimization [26–28]. In these approaches, the full rigid body equations of motion (3.1) are included as linear equality constraints. As discussed in [25], it is possible to eliminate these additional decision variables and constraints noting that the torque vector is a linear function of the joint acceleration and contact force vectors. From (3.1) and (3.11), we have

$$\begin{bmatrix} \mathbf{0} \\ \boldsymbol{\tau} \end{bmatrix} = \begin{bmatrix} \mathbf{H}(\mathbf{q}) & \sum_c \mathbf{J}_c^T \boldsymbol{\beta}_c \mathbf{S}_c \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\rho} \end{bmatrix} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}), \tag{3.18}$$

where $\mathbf{S}_c$ is a selection matrix that projects $\boldsymbol{\rho}$ to $\boldsymbol{\rho}_c$. Joint torque limits are implemented by substituting the lower n equations of (3.18) into (3.15) where $\underline{\boldsymbol{\tau}}$ and $\bar{\boldsymbol{\tau}}$ represent lower and upper torque limits. Admissibility of the joint torques, joint accelerations, and contact forces is ensured by the centroidal dynamics constraint (3.13). This is a departure from [25] where the floating-base dynamics are constrained by the upper 6 equations of (3.1); however, this is an equivalent formulation that also constrains the net momentum rate of change.

3.4.5 Implementation

At each time step, admissible joint accelerations, $\ddot{\mathbf{q}}_a$, and contact forces, $\boldsymbol{\rho}_a$, are computed using the proposed model-based optimization (3.12). Corresponding joint torque setpoints, $\boldsymbol{\tau}_a$, are computed from (3.18). The QP is solved using an Eigen implementation of QuadProg++ based on the active set method described in [82]. Given 8 active contact points and 30 actuated DOF, the optimization executes at 800 Hz on a quad-core i7 processor with OpenMP support. This enables real-time whole-body control for tasks such as standing manipulation and dynamic walking. The task-space Jacobians, Centroidal Momentum Matrix, and joint-space inertia matrix required by the optimization are computed using a rigid body model obtained from a detailed CAD design. The vector of centrifugal, Coriolis and gravity torques is computed using the Recursive Newton Euler algorithm [75].

3.5 Whole-Body Locomotion

This section provides an overview of the proposed locomotion framework using whole-body control. For reference, a high-level block diagram is shown in Fig. 3.3. Dynamic behaviors such as standing and stepping are implemented using state machines that respond to external events such as toe-off and heel-strike. Following each event, state-specific parameters such as desired end-effector waypoints and step durations are passed to planning modules to generate joint-space and task-space trajectories for dynamic locomotion.

At each time step, desired motion tasks are computed using DCM, angular momentum, end-effector, and joint-space tracking controllers. Optimal joint torques are computed using the whole-body controller framework presented in Section 3.4, and compliant joint-space control is implemented to track the resulting joint trajectories. Finally, a Kalman filter-based state estimator is implemented to compute the necessary states for whole-body control. The following subsections present a detailed overview of the compliant standing and stepping behaviors implemented using the proposed framework. Note that the presented approach can be applied to a large variety of tasks including climbing, fall recovery, and whole-body manipulation.

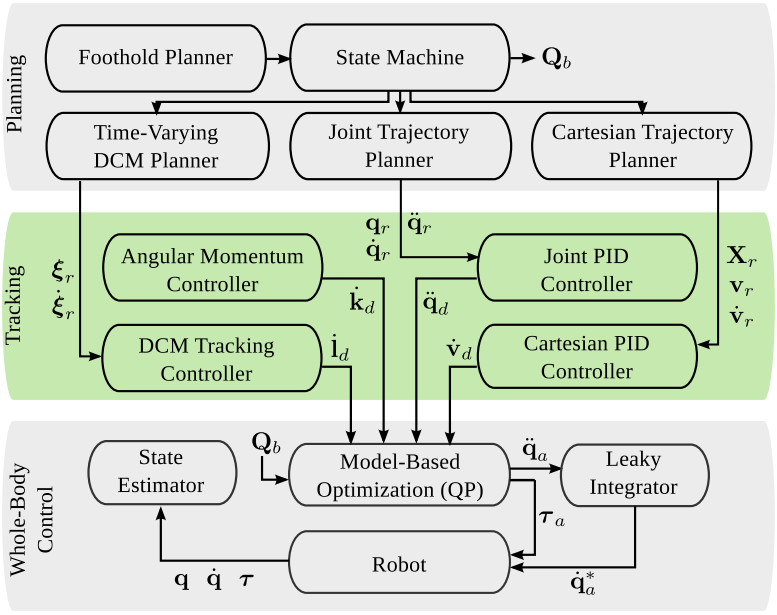

Figure 3.3: Control block diagram of locomotion subsystem.

3.5.1 Standing

The stand controller enables the robot to maintain balance in double or single support while tracking upper body joint trajectories for generic manipulation tasks.

Momentum Control

Dynamic balancing is implemented by tracking a DCM reference trajectory, $\boldsymbol{\xi}_r(t)$, that nominally lies above the center of the support polygon. As discussed in Section 3.3.3, the time-varying DCM dynamics can be stabilized by defining an appropriate control law for the VRP. In [11], the authors proposed a proportional tracking controller for the time-invariant DCM dynamics assuming $\omega(t) = \omega_0$. Integral action was introduced into the DCM controller in [13] to compensate for steady state errors due to model uncertainties.

We define the following control law based on the time-varying DCM dynamics (3.6),

$$\mathbf{r}_{vrp} = \boldsymbol{\xi} - \frac{1}{\omega - \frac{\dot{\omega}}{\omega}} \left(\dot{\boldsymbol{\xi}}_r + \mathbf{k}_\xi \left(\boldsymbol{\xi}_r - \boldsymbol{\xi} \right) + \mathbf{k}_\Xi \int \left(\boldsymbol{\xi}_r - \boldsymbol{\xi} \right) dt \right), \tag{3.19}$$

where $\boldsymbol{\xi}_r$ and $\dot{\boldsymbol{\xi}}_r$ represent the reference DCM position and velocity. The first term cancels the DCM drift dynamics, and the second term implements a proportional-integral controller with unity feedforward. The non-negative feedback gains $\mathbf{k}_\xi$ and $\mathbf{k}_\Xi$ determine the bandwidth and steady-state characteristics of the DCM controller. Here we assume that the natural

frequency of the DCM satisfies $\omega - \dot{\omega}/\omega > 0$. As discussed in [57], the natural frequency trajectory is derived from a desired CoM height trajectory which can be designed to satisfy this condition given $z_{com} - z_{ecmp} > 0$ and $\ddot{z}_{com} > -g$. The latter constraint limits the proposed approach to locomotion behaviors without a flight phase.

Given the commanded VRP setpoint from the DCM tracking controller (3.19), the desired linear momentum rate of change is derived from the VRP definition (3.7) such that

$$\dot{\mathbf{h}}_d = \begin{bmatrix} \dot{\mathbf{l}}_d \\ \dot{\mathbf{k}}_d \end{bmatrix} = \begin{bmatrix} m\left(\omega^2 - \dot{\omega}\right)\left(\mathbf{x} - \mathbf{r}_{vrp}\right) \\ -b_{\mathbf{k}}\mathbf{k} \end{bmatrix}. \tag{3.20}$$

Here the angular momentum rate of change is derived from a simple damping controller. The damping coefficient, $b_{\mathbf{k}} \geq 0$, regulates the total angular velocity in the system.

Upper Body Control

Joint-space acceleration objectives are used to track upper-body joint trajectories. Joint tracking is achieved using a standard PD controller in the form,

$$\ddot{\mathbf{q}}_d = \ddot{\mathbf{q}}_r + k_{\mathbf{q}}\left(\mathbf{q}_r - \mathbf{q}\right) + b_{\mathbf{q}}\left(\dot{\mathbf{q}}_r - \dot{\mathbf{q}}\right), \tag{3.21}$$

where $\mathbf{q}_r$, $\dot{\mathbf{q}}_r$ and $\ddot{\mathbf{q}}_r$ are the reference joint position, velocity, and acceleration vectors. The PD gains, $k_{\mathbf{q}} \geq 0$ and $b_{\mathbf{q}} \geq 0$, can be adjusted online depending on the current task. The reference trajectories may be specified by an external planner to allow standing manipulation, in which case high optimization weights are assigned to support accurate tracking. Low optimization weights may also be assigned to allow the arms to assist in balancing by generating angular momentum and or shifting the center of mass.

Lower Body Control

Pelvis and foot trajectories are generated using piecewise 5^{th} order minimum jerk polynomials. The desired state of each Cartesian frame is expressed by a transform $\mathbf{X}_r = \begin{bmatrix} \mathbf{R}_r & \mathbf{r}_r \\ \mathbf{0} & 1 \end{bmatrix}$, twist $\mathbf{v}_r = \begin{bmatrix} \dot{\mathbf{r}}_r & \boldsymbol{\omega}_r \end{bmatrix}^T$, and spatial acceleration $\dot{\mathbf{v}}_r = \begin{bmatrix} \ddot{\mathbf{r}}_r & \dot{\boldsymbol{\omega}}_r \end{bmatrix}^T$. The 6 DOF trajectories are tracked using a Cartesian PD controller in the form,

$$\dot{\mathbf{v}}_d = \begin{bmatrix} \ddot{\mathbf{r}}_d \\ \dot{\boldsymbol{\omega}}_d \end{bmatrix} = \begin{bmatrix} \ddot{\mathbf{r}}_r + \mathbf{K}_{\mathbf{r}}\left(\mathbf{r}_r - \mathbf{r}\right) + \mathbf{B}_{\mathbf{r}}\left(\dot{\mathbf{r}}_r - \dot{\mathbf{r}}\right) \\ \dot{\boldsymbol{\omega}}_r + \mathbf{K}_{\mathbf{R}}\left(\theta\hat{\mathbf{e}}\right) + \mathbf{B}_{\mathbf{R}}\left(\boldsymbol{\omega}_r - \boldsymbol{\omega}\right) \end{bmatrix}, \tag{3.22}$$

where $\theta\hat{\mathbf{e}}$ is the axis-angle representation of the rotational error. Here $\mathbf{K}_{\mathbf{r}}$, $\mathbf{K}_{\mathbf{R}}$, $\mathbf{B}_{\mathbf{r}}$, and $\mathbf{B}_{\mathbf{R}}$ are diagonal stiffness and damping matrices. Note that the pelvis linear accelerations are uncontrolled to avoid over-constraining the optimization.

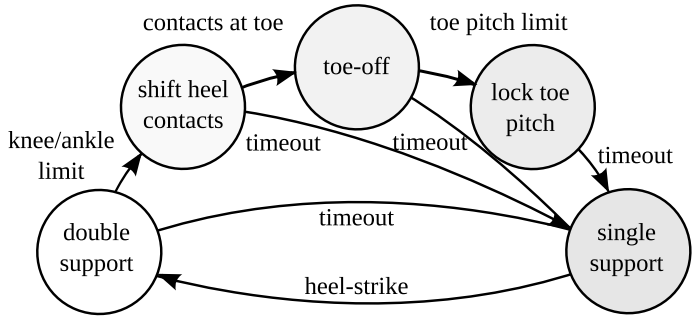

Figure 3.4: Step controller state machine.

3.5.2 Stepping

The step controller implements single-step and multi-step behaviors given desired foothold poses and step durations from a high-level footstep planner.

Trajectory Planning

The DCM reference trajectory is updated at the onset of each step using the time-varying DCM planner described in [57]. First, explicit CoP and vertical CoM trajectories are computed from the desired foothold poses. Next, reverse-time integration is used to plan a dynamically feasible DCM trajectory over a multi-step time horizon satisfying the final boundary conditions. Finally, a model predictive controller is used to compute an admissible reference trajectory over a short preview window satisfying the initial boundary conditions.

State Machine

The step behavior employs the same subset of controllers as the stand behavior. However, an additional finite state machine is implemented to handle transitions between various contact phases as illustrated in Fig. 3.4. During the double support phase, both feet remain in contact with the support surface. Eight active contact constraints are added to the whole-body controller optimization corresponding to the four corners of each foot. Following the preplanned double support duration, the state machine transitions to the single support phase, breaking contact with the swing foot and configuring appropriate optimization weights for swing foot tracking. Heel-strike events are detected using a force-torque sensor in the sole of the swing foot. When the swing foot is sufficiently loaded following ground contact, the state machine transitions back to double support.

In order to increase the maximum stride length, we adopt a reactive toe-off strategy to compensate for range of motion limits. If the measured knee or ankle pitch of the swing leg reaches a soft position limit during double support, the two contact points on the heel of

the swing foot are shifted to the toe in order to appropriately bound the center of pressure. Next the heel contacts are disabled to enable toe-off by allowing the foot to pivot about the two toe contacts. If the toe pitch reaches a critical limit, the swing foot tracking controller is configured to prevent the foot from rotating further during the double support phase.

3.6 Hardware Implementation

The proposed whole-body locomotion framework was implemented on THOR, a 34 DOF compliant humanoid designed to compete in the DARPA Robotics Challenge [2]. The THOR lower body is equipped with force-controllable linear series elastic actuators [70], while the upper body is equipped with stiff rotary actuators. Compliant actuation in the lower body enables direct regulation of internal and external forces, allowing the robot to naturally adapt to uncertain terrain features. In comparison, stiff actuation in the upper body offers precise trajectory tracking for manipulation tasks. The locomotion framework shown in Fig. 3.3 executes at 150 Hz on a quad-core i7 computer mounted in the robot's chest.

Due to the reliance on a high DOF dynamic model, optimization-based whole-body control approaches can be sensitive to inaccuracies in the inertial estimates of the rigid bodies. To reduce model errors, we maintain a detailed CAD model that includes material properties reflecting the empirical weights of physical sub-assemblies. Additional estimation and control errors can arise from a variety of sources including sensor noise, communication delays, friction forces, and structural compliance. Because we assume a rigid body model with a torque source at each joint, reliable joint-space control and estimation is critical to the success of the whole-body controller when implemented on a physical system. This section discusses the low-level controllers and estimators implemented on the THOR hardware platform.

3.6.1 Joint-Space Control

We begin by describing the joint-space control approach used to track the admissible torque and acceleration setpoints obtained from the whole-body controller QP (3.12). The upper body joint trajectories are tracked using a high-gain velocity controller, while the lower body trajectories are tracked using a cascaded joint impedance controller with an inner-loop force controller. The impedance controller implements high-fidelity joint torque control with low-gain velocity feedback. As demonstrated in [25], it is possible to implement whole-body inverse dynamics using a purely torque-controlled hardware interface. However, inner-loop velocity feedback can increase the overall stability of the system by compensating for integrated acceleration errors and introducing active damping at a high sample rate. In our platform, linear series elastic actuators are used to generate torques about the lower body joints via linear to rotary and parallel linkages [83; 84]. The torque setpoints, $\boldsymbol{\tau}_a^*$, and integrated velocity setpoints, $\dot{\mathbf{q}}_a^*$, are relayed to embedded joint controllers at each time step.

Torque and Velocity Setpoints

The joint torque setpoints, $\boldsymbol{\tau}_a^*$, are obtained from the optimized joint torques, $\boldsymbol{\tau}_a$, by applying a second order notch filter to suppress a resonance near the high force bandwidth of the actuators. Without filtering, the leg joints tend to oscillate at approximately 20 Hz when the robot is in single support and the pitch actuators approach their peak output force. The filter transfer function is given by

$$H(s) = \frac{s^2 + \omega_n^2}{s^2 + \frac{\omega_n}{Q}s + \omega_n^2}, \tag{3.23}$$

where the desired notch frequency is set to $\omega_n = 2\pi \cdot 20$ rad/s, and the Q-factor is set to 10 for the experiments presented in Section 3.7.

We employ a similar approach to [85] to compute the joint velocity setpoints, $\dot{\mathbf{q}}_a^*$, using leaky integration of the optimized joint accelerations, $\ddot{\mathbf{q}}_a$. The leaky integrator dynamics are expressed by the first-order ODE,

$$\ddot{\mathbf{q}}_a^* = \alpha\left(\dot{\mathbf{q}} - \dot{\mathbf{q}}_a^*\right) + \ddot{\mathbf{q}}_a, \tag{3.24}$$

where $\dot{\mathbf{q}}_a^* = \int \ddot{\mathbf{q}}_a^*\, dt$. The leaking rate, $\alpha \geq 0$, determines the rate at which the integral drifts towards the estimated velocity. In the course of testing, we chose $\alpha = 75$ for the lower body joints and $\alpha = 0$ for the upper body joints to account for the different actuator impedances.

Force Control

The inner-loop actuator force controller incorporates feedforward and PID control with a model-based disturbance observer (DOB) based on measurements from an inline load cell sensor. The DOB estimates unmodeled disturbances such as nonlinear friction forces and attempts to cancel their effect through negative feedback. Given the measured output force, the disturbance signal is estimated by comparing the commanded motor current with the expected motor current obtained from an inverse model of the open-loop actuator plant. Using the combined approach, we are able to achieve a control bandwidth of approximately 30 Hz at 10% of the peak force amplitude with excellent low-frequency force control. This is despite observing approximately 200 N of stiction in the transmission.

Joint torques are controlled by tracking equivalent actuator forces obtained from an inverse statics solution for the linear to rotary mechanism. Fig. 3.5 shows the torque tracking performance for the hip pitch joint with and without the use of a DOB. With the leg free to swing, a torque reference trajectory was commanded to emulate the swing phase during stepping. The results show that the DOB feedback significantly reduces torque tracking errors due to stiction and unmodeled dynamics. We found that the corresponding improvement in torque control was critical to the success of our whole-body locomotion implementation.

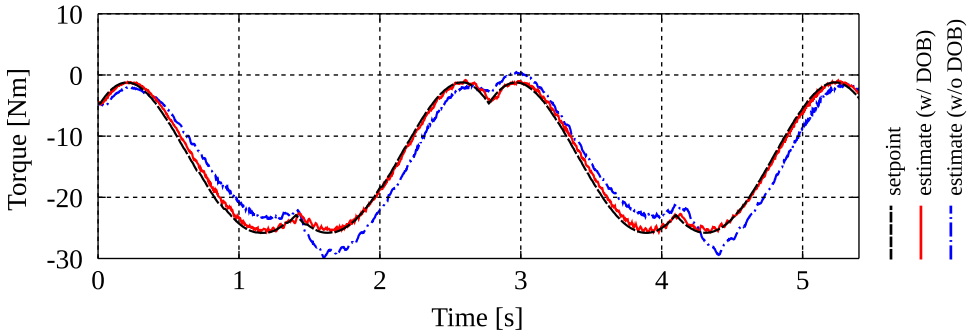

Figure 3.5: Hip pitch torque tracking with and without actuator disturbance observer.

Impedance Control

The authors experimented with two variations of a "simple" joint impedance controller. The first controller introduces outer-loop damping based on joint velocity estimates from post-transmission absolute encoders at the joints. The desired actuator forces are given by

$$\mathbf{f}_a = \mathbf{J}_{\boldsymbol{\ell}}^{-T} \left(\boldsymbol{\tau}_a^* + b_{\mathbf{q}} \left(\gamma \dot{\mathbf{q}}_a^* - \dot{\mathbf{q}} \right) \right), \tag{3.25}$$

where $\mathbf{J}_{\boldsymbol{\ell}}$ is the mechanism Jacobian that maps angular joint velocities, $\dot{\mathbf{q}}$, to linear actuator velocities, $\dot{\boldsymbol{\ell}}$. The joint damping coefficient $b_{\mathbf{q}} \geq 0$ determines the velocity feedback gain, and the scalar $\gamma \in [0, 1]$ biases the joint velocity setpoint towards zero in order to improve the stability of the derivative action. If $b_{\mathbf{q}} = 0$, (3.25) reduces to a simple torque controller. If $\gamma = 0$, the velocity feedback term introduces viscous damping.

The second controller introduces inner-loop damping based on actuator velocity estimates from pre-transmission incremental encoders at the motors, i.e.

$$\mathbf{f}_a = \mathbf{J}_{\boldsymbol{\ell}}^{-T} \boldsymbol{\tau}_a^* + b_{\boldsymbol{\ell}} \left(\gamma \dot{\boldsymbol{\ell}}_a^* - \dot{\boldsymbol{\ell}} \right). \tag{3.26}$$

In this scenario, joint-space damping is indirectly achieved by tracking desired linear actuator velocities computed via forward kinematics, $\dot{\boldsymbol{\ell}}_a^* = \mathbf{J}_{\boldsymbol{\ell}} \dot{\mathbf{q}}_a^*$. Although the effective joint damping varies dependent on the configuration, this approach significantly improves the stability of the derivative action, presumably due to the collocation of the motor and velocity sensor. In the experiments presented in the following section, we use $b_{\boldsymbol{\ell}} = 10000$ N/m/s and $\gamma = 0.625$.

3.6.2 State Estimation

In order to improve the whole-body controller performance, we implemented a minimal state estimator to suppress sensor noise resulting from mechanical vibration, electrical interference, and digital quantization. The estimated states include the current robot configuration, $\mathbf{q}$, and generalized velocities, $\dot{\mathbf{q}}$. The n DOF joint position and velocity estimates are obtained

from the post-transmission absolute encoders mounted at each joint. The authors found that the use of absolute encoder derivatives for velocity estimation results in stable whole-body control as long as active joint damping is implemented using the proposed inner-loop velocity feedback (3.26).

In our system, the floating-base frame is defined at the robot's pelvis. The pelvis link is equipped with an Attitude and Heading Reference System (AHRS) to measure the angular velocity and orientation relative to the inertial frame. As in [86], we compute linear position and velocity estimates for the floating-base frame using a combination of leg kinematics and inertial measurements. Assuming no-slip conditions on the active foot contacts, the estimated pelvis frame is given by

$$ {}^{0}\mathbf{X}_{p} = \frac{1}{N} \sum_{c=1}^{N} {}^{0}\mathbf{X}_{c} \, {}^{c}\mathbf{X}_{p}, \tag{3.27} $$

where ${}^{c}\mathbf{X}_{p}$ is the pelvis transform with respect to each contact point (obtained using forward kinematics), and ${}^{0}\mathbf{X}_{c}$ is the contact point transform relative to the inertial frame. The translational component of ${}^{0}\mathbf{X}_{c}$ is constant during each support phase and is only updated during heel-strike. The rotational component is defined such that the orientation of the pelvis transform, ${}^{0}\mathbf{X}_{p}$, is equal to the measured AHRS orientation.

The signal-to-noise ratio of the position measurements is much larger than the differentiated velocity measurements. As a result, we do not filter the position states in our system. To compensate for noise in the velocity estimates, we implement a kinematic Kalman filter using a process model based on the admissible joint-space and task-space acceleration inputs from the whole-body controller. At every time-step, the filtered velocity estimates are computed using SISO prediction and correction equations,

$$ \begin{aligned} v_{k}^{-} &= A v_{k-1}^{+} + B u_{k}, \\ v_{k}^{+} &= v_{k}^{-} + K \left(y_{k} - \left(C v_{k}^{-} + D u_{k} \right) \right), \end{aligned} \tag{3.28} $$

given $A = 1$, $B = \Delta T$, $C = 1$, $D = 0.5\Delta T$. Here y_k represents the velocity measurement, u_k represents the corresponding acceleration input obtained via the whole-body controller QP, and v_k^- and v_k^+ represent the *a priori* and *a posteriori* velocity estimates, respectively. In this formulation the Kalman gain is given by

$$ K = PC^{T}(CPC^{T} + R)^{-1} = P(P + R)^{-1}, \tag{3.29} $$

where R is the measurement covariance and P is the steady-state *a priori* error covariance.

The process and measurement covariances were selected experimentally. Aside from filtering the floating-base and joint-space velocity estimates, we also filter task-space velocity and centroidal momentum estimates using the same Kalman filter formulation. The omission of position states greatly simplifies the design of the estimator and enables an efficient real-time implementation.

3.7 Experimental Results

This section presents experimental results obtained using the THOR hardware platform. A comprehensive list of whole-body controller weights and feedback gains is provided in Table 3.1. Note that the linear momentum objective is weighted an order of magnitude higher than the angular momentum objective. This is meant to improve dynamic stability by permitting large restoring forces when it is not possible to simultaneously satisfy both linear and angular objectives. Although unlisted, we also implemented integral feedback for the linear and angular swing foot acceleration objectives and pelvis angular acceleration objectives with integral gains of diag (100, 100, 100), diag (100, 100, 100), and diag (50, 50, 30), respectively. The DCM proportional gain was set to $k_{\boldsymbol{\xi}} = 2.5$ in the horizontal axes and $k_{\boldsymbol{\xi}} = 7.5$ in the vertical axis. We found that a higher stiffness in the vertical axis was necessary to achieve comparable performance due to the larger reaction forces. The DCM integral gain was set to $k_{\boldsymbol{\Xi}} = 0.75$ in order to compensate for steady-state errors.

Table 3.1: Whole-Body Controller Weights and Gains

Motion Task	Units	Weight	Stiffness	Damping
$\dot{\mathbf{l}}$	N	10	-	-
$\dot{\mathbf{k}}$	Nm	1	0	0
$\ddot{\mathbf{q}}_{\text{arms}}$	rad/s^2	6	15	10
$\ddot{\mathbf{q}}_{\text{waist}}$	rad/s^2	100	40	20
$\ddot{\mathbf{r}}_{\text{contact points}}$	m/s^2	10000	0, 0, 0	0, 0, 0
$\ddot{\mathbf{r}}_{\text{swing foot}}$	m/s^2	1000	150, 150, 200	20, 20, 20
$\dot{\boldsymbol{\omega}}_{\text{swing foot}}$	rad/s^2	1000	100, 100, 100	10, 10, 20
$\dot{\boldsymbol{\omega}}_{\text{pelvis}}$	rad/s^2	30	70, 70, 30	30, 30, 15

3.7.1 Balancing

Fig. 3.6 shows the robot balancing on one foot following a disturbance force applied to the back. The robot is able to recover from the unexpected disturbance by pitching its hip and shoulder joints in order generate the necessary reaction forces to stabilize the DCM. This is made feasible by compliant joint control in the lower body. Low damping coefficients allow the leg joints to immediately accelerate in response to the impulse without shifting the CoP to the edge of the foot.

The controller setpoints and estimates for two consecutive pushes are shown in Fig. 3.7. The x-axis of the inertial frame is oriented to the front of the robot and the y-axis is oriented to the left. The region between the virtual toe and heel contacts is marked for reference. The whole-body controller formulation ensures that the optimized CoP setpoint does not exceed this safety region, thereby preventing the support foot from tipping by satisfying the

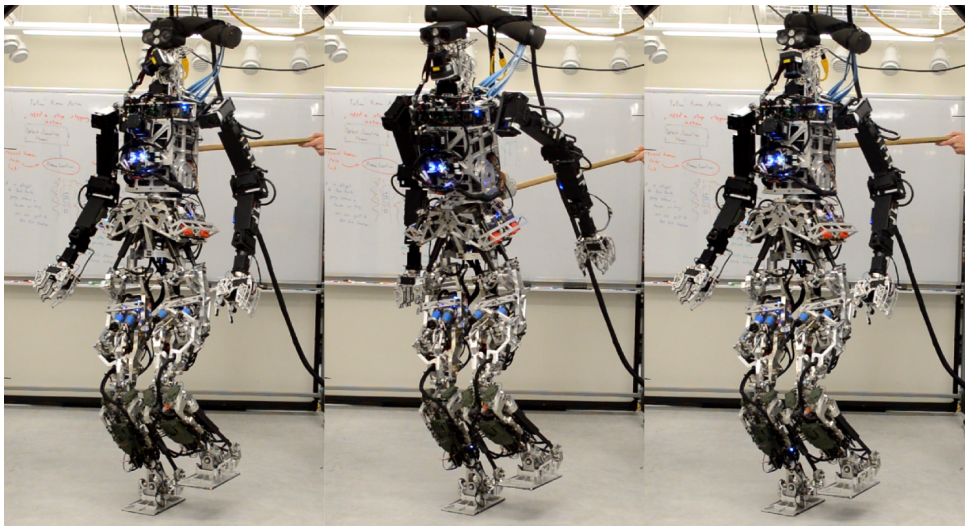

Figure 3.6: THOR balancing on one foot. DCM tracking is used to stabilize the CoM dynamics following a disturbance impulse applied to the back.

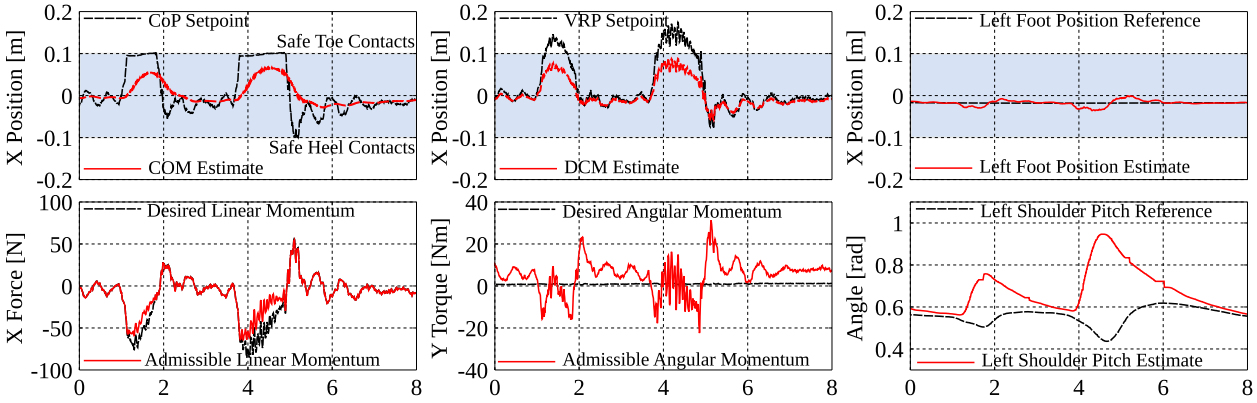

Figure 3.7: Controller response following two disturbance impulses applied to the back of the robot while balancing on the right foot. Here the x-axis is oriented to the front of the robot, and the y-axis is oriented to the left. Admissible values correspond to the outputs of the whole-body QP optimization.

ZMP criterion. The VRP setpoint, however, is allowed to leave the base of support in order to stabilize the DCM dynamics. Consequently, a net angular momentum rate of change is induced along the y-axis.

Note that the left shoulder pitch joint diverges significantly from the desired reference trajectory in order to generate the necessary spin angular momentum. Contrarily, the high weights of the swing foot objectives allow accurate reference tracking throughout the disturbance. Once the linear dynamics are stabilized, the pelvis and arm objectives dominate, allowing the robot to return to its original pose.

3.7.2 Walking

Fig. 3.8 shows the robot walking on various surfaces including cement, gravel, and soft grass. Each step is characterized by a double support duration of 1.3125 s, single support duration of 2.1875 s, stride length of 0.175 m, and swing foot apex of 0.09 m. The optimization weights, controller gains, and estimation parameters are identical in each test (refer to Table 3.1). Note that, although the whole-body QP formulation assumes a rigid body contact model, the step controller is able to maintain stability on compliant and unstable terrain.

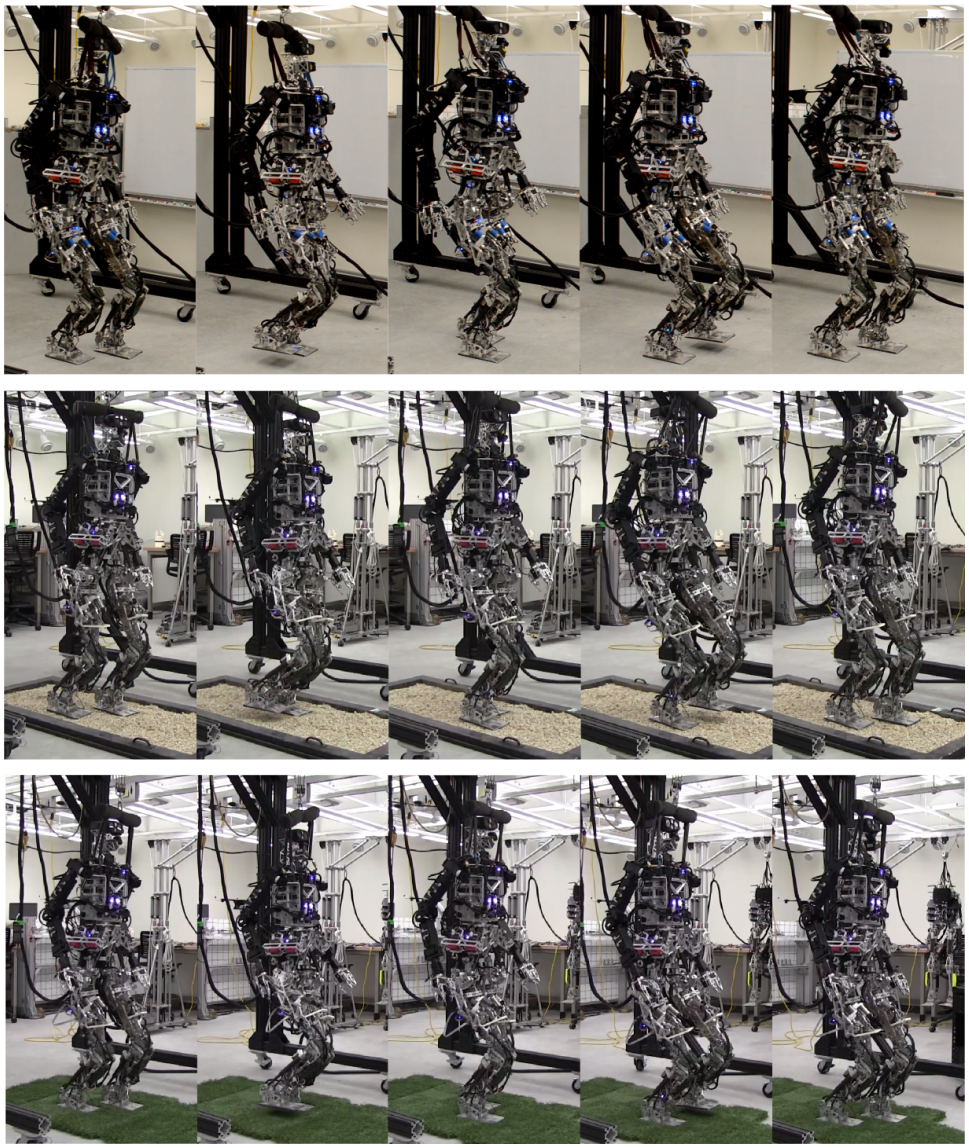

Figure 3.8: THOR hardware platform walking on cement, gravel, and grass using identical control and estimation parameters.

Fig. 3.9 shows the desired and estimated DCM and VRP trajectories while walking on the cement surface depicted in Fig. 3.8. As the results demonstrate, the robot is able to track the DCM reference trajectory using the VRP-based momentum controller defined in (3.19) with only minor deviations from the nominal VRP trajectory. The largest errors occur during heel-strike, when the DCM and VRP plans have just been updated, and the swing foot impact forces can be difficult to predict.

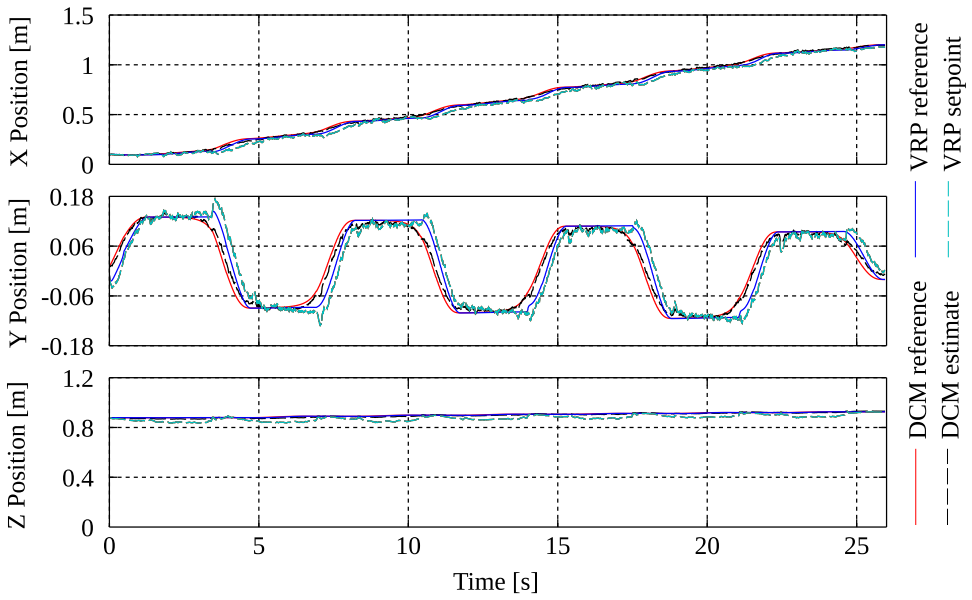

Figure 3.9: DCM and VRP trajectories while walking on cement using the THOR hardware platform.

As discussed in Section 3.6, compliant actuation in the lower body enables accurate control of the ground reaction forces during locomotion. Fig. 3.10 compares the optimized and estimated reaction forces acting on the left foot while walking on cement. Here the foot wrenches were estimated using a six-axis force-torque sensor mounted in between the ankle and the sole. In this scenario, the locomotion framework is able to achieve the desired foot wrench using a purely model-based control approach, i.e. without introducing a feedback path based on the force-torque data. Although significant transient errors appear in the optimized tangential forces, the DCM tracking controller appears robust to the corresponding variations in linear momentum rate of change.

Fig. 3.11 shows the resulting horizontal CoP trajectory (estimated from the normal forces and tangential torques acting on each foot). The accuracy of the CoP trajectory is partially determined by errors in the corresponding reaction forces. For this experiment, the horizontal CoP reference was defined to be equivalent to the VRP/eCMP reference shown in Fig. 3.9. As demonstrated in the balancing experiment, the optimized CoP is typically collocated with the optimized VRP/eCMP in order to minimize horizontal angular momentum rates of change during locomotion. Thus, CoP excursions in the xy plane are also correlated with

restoring forces generated by the momentum controller, which shift the eCMP setpoint away from the reference trajectory.

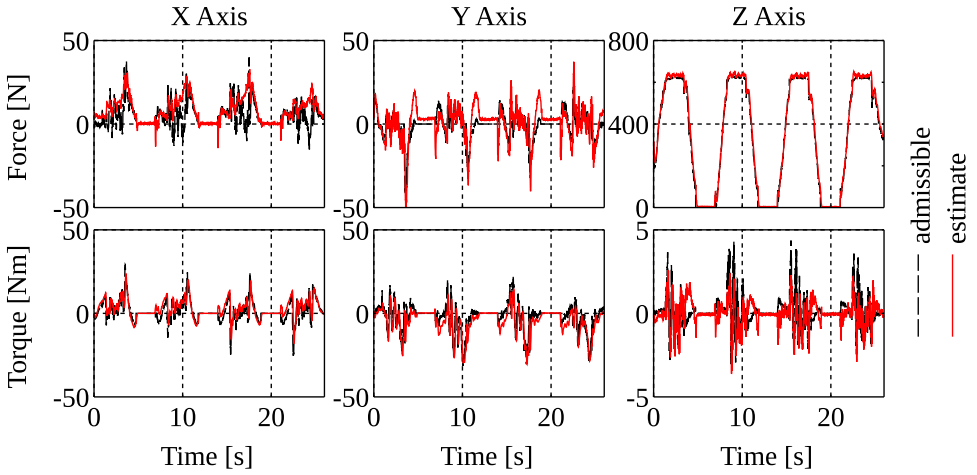

Figure 3.10: Left foot reaction forces while walking on cement using the THOR hardware platform. Admissible values correspond to the outputs of the whole-body QP optimization.

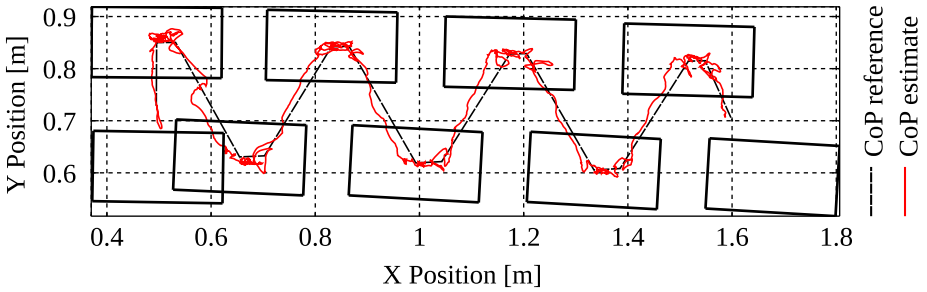

Figure 3.11: Center of pressure while walking on cement using the THOR hardware platform.

Fig. 3.12 shows the norm horizontal CoP error for the walking experiments depicted in Fig. 3.8. Larger CoP errors appear in the gravel and grass data due to the effects of unmodeled surface compliance. The gravel is free to shift underfoot, while the grass deflects significantly under load. In the grass experiment, CoP errors exceeding 7.5 cm occur during the transition from single to double support. Deviations from the nominal CoP trajectory can limit control authority of the linear and angular momentum rates of change given a finite base of support. Although the stability margins decrease on compliant surfaces, the robot is able to successfully traverse the varying terrain using the same control gains and optimization weights employed on stiff surfaces.

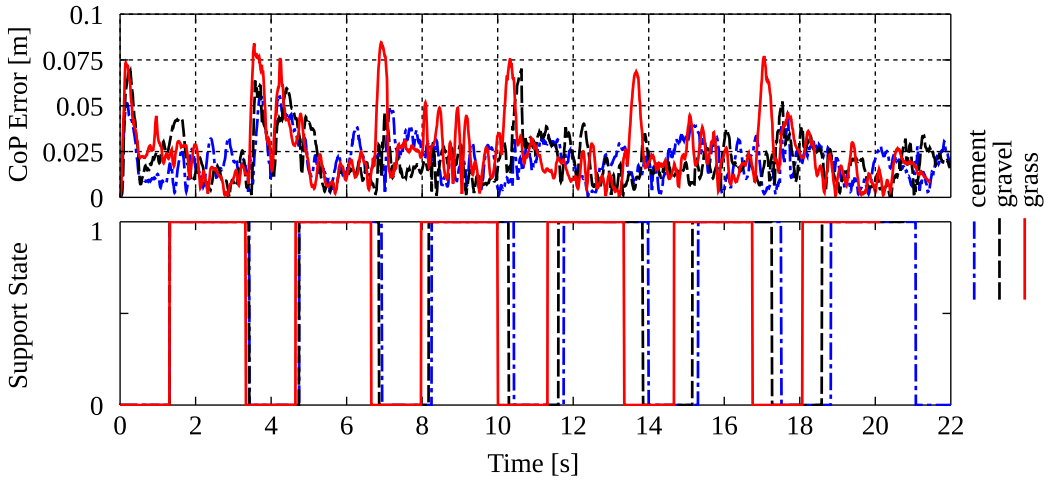

Figure 3.12: Top: Norm horizontal center of pressure error while walking on cement, gravel, and grass. Bottom: Step controller support state (0: double support, 1: single support).

3.7.3 Robustness to Unmodeled Terrain

The use of compliant joint control in the lower body results in behaviors that are inherently robust to uncertain and unstable terrain. Fig. 3.13 shows the robot standing and stepping onto various terrain, initially modeled as a flat surface. In the leftmost image, the robot stands on a balance board that rotates about a pivot, and in the images to the right, the robot steps onto unexpected debris. In each scenario, the controller is able to safely adapt to uncertain terrain through the use of low-impedance task-space feedback coupled with accurate joint torque control.

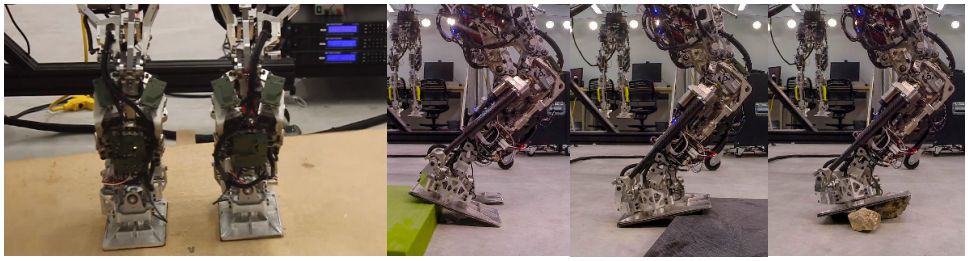

Figure 3.13: Left: THOR standing on a balance board that rotates about a pivot. Right: Stepping onto unexpected debris and adapting to the various terrain. Images used with permission of E. Hahn.

3.8 Conclusion

The proposed control framework has enabled the THOR platform to navigate shipboard test environments as part of the Navy Shipboard Autonomous Fire Fighting (SAFFiR) program. The use of compliant actuation in the lower body supports locomotion behaviors that are robust to unmodeled terrain features resulting from surface compliance or perception errors. Furthermore, the incorporation of low-gain joint velocity feedback enables accurate tracking of the task-space objectives using a cascaded joint-space impedance controller.

Future research efforts will focus on improvements in model-based state estimation for compliant humanoids. We are also investigating improved joint-space torque control approaches using a novel model for linear series elastic actuators. As whole-body control approaches become increasingly robust, new opportunities are arising for optimization-based locomotion planners that incorporate novel task-space and joint-space cost terms and constraints. We are currently interested in methods to handle uncertainty and failure during locomotion, including the application of adaptive control methods for dynamic step recovery.

Chapter 4

Dynamic Locomotion on Uneven Terrain Using the Time-Varying Divergent Component of Motion

4.1 Abstract

This paper presents a framework for dynamic humanoid locomotion on uneven terrain using a novel time-varying extension of the Divergent Component of Motion (DCM). By varying the natural frequency of the DCM, we are able to achieve generic CoM height trajectories during stepping. The proposed planning algorithm computes admissible DCM reference trajectories given desired ZMP plans for single and double support. This is accomplished using reverse-time integration of the discretized DCM dynamics over a finite time horizon. To account for discontinuities during replanning, linear model predictive control (MPC) is implemented over a short preview window. DCM tracking control is achieved using a time-varying proportional-integral controller based on the Virtual Repellent Point (VRP). The effectiveness of the combined approach is verified in simulation using a 30 DOF model of THOR, a compliant torque-controlled humanoid.

4.2 Introduction

After decades of research into legged robots, humanoid locomotion controllers are still unable to match the grace and robustness of biological systems. The problem is complicated by the underactuated and nonlinear nature of the associated multibody dynamics. The dynamic planning problem typically requires the generation of centroidal reference trajectories that can be realized using admissible contact forces given mode-dependent contact constraints.

To achieve real-time performance, many authors introduce simplifying assumptions which limit locomotion to flat terrain or impose certain restrictions on the vertical center of mass (CoM) or angular momentum trajectories.

A common approach in the walking literature is to generate reference CoM trajectories from admissible Zero-Moment Point (ZMP) trajectories in order to satisfy no-tipping conditions on the support feet [5]. In [6], a CoM pattern generator was proposed based on the analytic solution to the linear inverted pendulum (LIPM) dynamics. In [14], the authors introduced a preview controller to minimize ZMP tracking errors by optimizing the CoM jerk. In [17], a similar model predictive control (MPC) approach was proposed for ZMP tracking using a linearly constrained quadratic program. Although these approaches are based on the LIPM dynamics, some authors have incorporated desired CoM height and angular momentum trajectories using time-varying LQR solutions [15; 29].

The Capture Point (CP) [8] and Divergent Component of Motion (DCM) [9] are equivalent transformations of the CoM state that encode the unstable dynamics of the linear inverted pendulum. By definition, the CoM can be indirectly stabilized by tracking an appropriate CP trajectory. In [10], the authors proposed a recursive algorithm to compute CP trajectories for dynamic walking given desired ZMP waypoints, and in [11] this work was extended to three-dimensional DCM trajectories to enable locomotion on uneven terrain. In [13], integral action was incorporated into the CP controller to eliminate steady-state errors, and in [12], the authors introduced a ZMP-based CP tracking controller to compensate for time-varying vertical CoM dynamics and horizontal angular momentum rates of change during stepping. Because these methods assume a constant natural frequency of the LIPM for trajectory planning, the ZMP can deviate from the desired reference when the vertical CoM dynamics do not match the time-invariant LIPM dynamics.

This paper introduces a dynamic planning and control approach for humanoid robots using a novel time-varying extension of the DCM. By modifying the definition of the time-varying natural frequency, we derive simple first-order dynamics for the DCM in terms of the Virtual Repellent Point (VRP) introduced in [11]. Using this framework, we present a method to plan DCM reference trajectories for dynamic walking on uneven terrain given generic vertical CoM and ZMP trajectories. This is accomplished using reverse-time integration of the discretized DCM dynamics over a finite time horizon. To account for discontinuities during replanning, linear model predictive control (MPC) is implemented over a short preview window. The effectiveness of the combined approach is verified through experiments using a simulated 30 DOF model of the THOR humanoid shown in Fig. 4.1. Although we focus on trajectory generation for the three-dimensional DCM, the proposed algorithms can also be applied to two-dimensional CP planning and control.

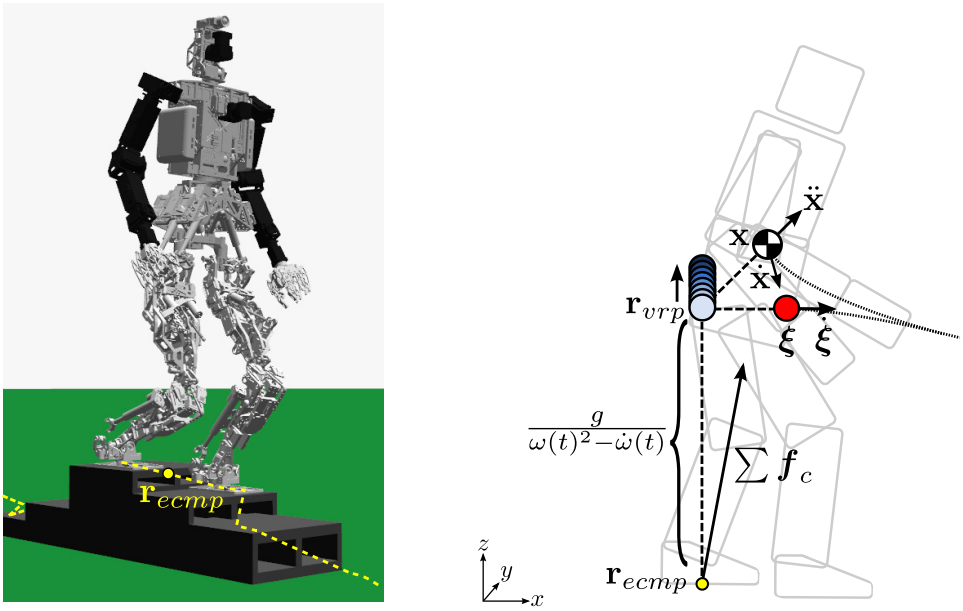

Figure 4.1: Left: THOR walking on uneven terrain in simulation. Right: Time-varying Divergent Component of Motion (DCM) dynamics.

4.3 Divergent Component of Motion

The Divergent Component of Motion (DCM) is a linear transformation of the center of mass (CoM) state used to separate the second-order CoM dynamics into stable and unstable first-order components [9]. In [11], the authors introduced a three-dimensional extension of the DCM defined by

$$\boldsymbol{\xi} = \mathbf{x} + \frac{1}{\omega_0}\dot{\mathbf{x}}, \tag{4.1}$$

where $\mathbf{x} = [\, x_{com},\ y_{com},\ z_{com}\,]^T$ is the CoM position, $\dot{\mathbf{x}}$ is the CoM velocity, and $\omega_0 = \sqrt{\frac{g}{\Delta z_{com}}}$ is the natural frequency of the linear inverted pendulum [6] given a nominal CoM height, Δz_{com}.

The unstable first-order DCM dynamics are given by

$$\dot{\boldsymbol{\xi}} = \omega_0 \left(\boldsymbol{\xi} - \mathbf{r}_{vrp}\right), \tag{4.2}$$

where $\mathbf{r}_{vrp}$ represents the Virtual Repellent Point (VRP) [11]. Given $\omega_0 > 0$, this system has a single pole in the right-half plane. The DCM is repelled away from the VRP at a rate proportional to its distance. This unstable equilibrium point is defined as

$$\mathbf{r}_{vrp} = \mathbf{r}_{ecmp} + [\, 0,\, 0,\, \Delta z_{com}\,]^T, \tag{4.3}$$

where

$$\mathbf{r}_{ecmp} = \mathbf{x} - \frac{\sum \mathbf{f}_c}{m\omega_0^2} \tag{4.4}$$

represents the enhanced Centroidal Moment Pivot (eCMP) [11]. The eCMP encodes the direction and magnitude of all contact forces, $\sum \mathbf{f}_c \in \mathbb{R}^3$, acting on the CoM given the total mass, m, and CoM position, $\mathbf{x}$ [11]. When the eCMP intersects the ground plane it coincides with the canonical CMP ground reference point [35]. For planning purposes the eCMP and CMP are often defined to lie within the base of support as shown in Fig. 4.1 in order to ensure admissibility of contact forces during stepping [77].

From (4.1) the stable first-order CoM dynamics are given by

$$\dot{\mathbf{x}} = \omega_0(\boldsymbol{\xi} - \mathbf{x}). \tag{4.5}$$

By definition, the CoM converges to the DCM with a time constant of $\frac{1}{\omega_0}$. Since the zero dynamics of the DCM transformation are asymptotically stable, the centroidal dynamics can be indirectly stabilized by tracking a desired DCM reference trajectory. This property has been exploited by a number of researchers to develop dynamic walking strategies which do not require direct planning of the CoM trajectory [9; 11–13].

4.4 Time-Varying DCM

In this section we adapt the Divergent Component of Motion dynamics to account for a time-varying natural frequency, i.e.

$$\boldsymbol{\xi} = \mathbf{x} + \frac{1}{\omega(t)}\dot{\mathbf{x}}. \tag{4.6}$$

As discussed in Section 4.5, this relaxation of the original DCM definition enables improved planning of the vertical CoM trajectory during locomotion. When convenient we will use the notation $\omega := \omega(t)$, omitting the natural frequency's explicit dependence on time.

From (4.6) the time-varying CoM dynamics are given by

$$\dot{\mathbf{x}} = \omega(\boldsymbol{\xi} - \mathbf{x}). \tag{4.7}$$

Assuming $\omega(t) > \rho$ for some small positive value ρ, this system can be shown to be asymptotically stable with respect to the DCM equilibrium, noting that for the Lyapunov function $V(\mathbf{x}, t) = \frac{1}{2}\mathbf{x}^T\mathbf{x}$ we have $\dot{V}(\mathbf{x}, t) = -\omega(t)\mathbf{x}^T\mathbf{x} < -\rho\mathbf{x}^T\mathbf{x} < 0$. Differentiating the time-varying DCM definition (4.6) and substituting the CoM dynamics (4.7) gives

$$\begin{aligned}
\dot{\boldsymbol{\xi}} &= \dot{\mathbf{x}} - \frac{\dot{\omega}}{\omega^2}\dot{\mathbf{x}} + \frac{1}{\omega}\ddot{\mathbf{x}} \\
&= \left(1 - \frac{\dot{\omega}}{\omega^2}\right)\dot{\mathbf{x}} + \frac{1}{\omega}\ddot{\mathbf{x}} \\
&= \left(\omega - \frac{\dot{\omega}}{\omega}\right)(\boldsymbol{\xi} - \mathbf{x}) + \frac{1}{\omega}\ddot{\mathbf{x}} \\
&= \left(\omega - \frac{\dot{\omega}}{\omega}\right)\left(\boldsymbol{\xi} - \left(\mathbf{x} - \frac{\ddot{\mathbf{x}}}{\omega^2 - \dot{\omega}}\right)\right).
\end{aligned} \tag{4.8}$$

By redefining the VRP in terms of the CoM position and acceleration, i.e.

$$\mathbf{r}_{vrp} = \mathbf{x} - \frac{\ddot{\mathbf{x}}}{\omega^2 - \dot{\omega}}, \tag{4.9}$$

the time-varying DCM dynamics can be expressed as

$$\dot{\boldsymbol{\xi}} = \left(\omega - \frac{\dot{\omega}}{\omega}\right)(\boldsymbol{\xi} - \mathbf{r}_{vrp}). \tag{4.10}$$

This system is inherently unstable with respect to the VRP given $\omega - \dot{\omega}/\omega > 0$. Similar to the time-invariant case, the CoM dynamics can be indirectly stabilized by tracking a desired DCM reference trajectory using an appropriate control law based on the VRP.

Applying Newton's second law, the CoM acceleration can be expressed as $\ddot{\mathbf{x}} = \frac{1}{m}\sum \mathbf{f}_c - \mathbf{g}$ where $\mathbf{g} = [\,0, 0, g\,]^T$ is the gravity vector. Substituting this expression into (4.9) provides the VRP definition in terms of the eCMP,

$$\mathbf{r}_{vrp} = \mathbf{x} - \frac{\sum \mathbf{f}_c - m\mathbf{g}}{m\,(\omega^2 - \dot{\omega})} = \mathbf{r}_{ecmp} + \frac{\mathbf{g}}{\omega^2 - \dot{\omega}}, \tag{4.11}$$

where the eCMP is redefined as

$$\mathbf{r}_{ecmp} = \mathbf{x} - \frac{\sum \mathbf{f}_c}{m\,(\omega^2 - \dot{\omega})}. \tag{4.12}$$

As illustrated in Fig. 4.1, the vertical VRP-eCMP offset varies over time depending on the natural frequency and its first derivative while the horizontal offset remains zero.

Note that for a constant natural frequency, $\omega(t) = \omega_0$ and $\dot{\omega}(t) = 0$, the time-varying CoM-DCM dynamics and VRP-eCMP definitions are equivalent to the time-invariant equations introduced by Englsberger et al. [11]. Examining the z-axis constraints defined in (4.9) and (4.11), $\omega(t)$ can be defined in terms of the vertical CoM and eCMP dynamics using the nonlinear differential equation,

$$\omega^2 - \dot{\omega} = \frac{\ddot{z}_{com}}{z_{com} - z_{vrp}} = \frac{\ddot{z}_{com} + g}{z_{com} - z_{ecmp}}. \tag{4.13}$$

Thus, the natural frequency of the time-varying DCM is related to the natural frequency of the LIPM [78] by the expression, $\sqrt{\omega^2 - \dot{\omega}} = \sqrt{\frac{\ddot{z}_{com}+g}{\Delta z_{com}}}$ where $\Delta z_{com} = z_{com} - z_{ecmp}$. Note that this is a significant departure from previous CP and DCM publications which define $\omega = \sqrt{\frac{\ddot{z}_{com}+g}{\Delta z_{com}}}$ [12; 13].

If $\omega = 0$ or $\omega^2 - \dot{\omega} = 0$, the DCM dynamics (4.10) become uncontrollable. In the first case, the time-varying DCM is undefined, and in the second case the commanded contact forces, $\sum \mathbf{f}_c$, become zero. As a result, it is necessary to enforce the constraints $\omega > \rho$ and $\omega^2 - \dot{\omega} > \beta$ for some small positive values ρ and β. This restricts the application of the time-varying DCM to dynamic behaviors such as walking which do not include a flight state.

4.5 Locomotion Planning and Control

In this section we present a novel approach to planning and control for humanoid walking using the time-varying DCM. We will assume that a high level footstep planner provides information regarding the desired inertial foothold poses and step durations over a finite time horizon. Given these constraints, the dynamic planning problem addresses the generation of DCM reference trajectories that can be tracked using admissible contact forces during walking. In the proposed approach, trajectories are generated at the onset of each step given a desired terminal DCM state. The complementary dynamic control problem addresses how to track desired DCM reference trajectories using state feedback.

In [11] and [79], the authors propose a method to plan 3D DCM trajectories by computing analytic solutions of the reverse-time DCM dynamics given eCMP reference points which lie in the base of support. In order to achieve smooth eCMP and contact force trajectories during double support, the computed DCM reference trajectories are post-processed using polynomial interpolation of critical waypoints. This approach has proven very versatile, enabling fast trajectory planning for walking over uneven terrain using heel-strike and toe-off [79]. In order to compute closed-form solutions, however, these methods introduce certain restrictions on the generated eCMP trajectories and assume a constant natural frequency of the DCM.

In the proposed approach, we compute discrete reference trajectories for the time-varying DCM using a combination of reverse-time integration and model predictive control. By computing solutions numerically rather than analytically, it is possible to achieve generic eCMP trajectories during double and single support. Additionally, by planning appropriate trajectories for the natural frequency of the time-varying DCM, it is possible to achieve arbitrary kinematically feasible CoM height trajectories while stepping. The approach is summarized by the following steps described in this section.

1. Plan the eCMP trajectory given a desired footstep plan.
2. Plan the ω trajectory given a desired vertical CoM trajectory.
3. Plan the DCM trajectory using reverse-time integration and model predictive control given a desired terminal DCM state.
4. Track the DCM reference trajectory using a proportional-integral controller based on the time-varying VRP.

4.5.1 Planning the eCMP trajectory

Given a desired footstep plan, we begin by computing a ZMP[1] trajectory that intersects the base of support during each contact phase. This ensures that the derived eCMP and DCM reference trajectories will satisfy the ZMP criterion [5] given the commanded contact forces. The proposed approach supports arbitrary ZMP trajectories, enabling flat-footed or heel-toe walking. In the experiments presented in Section 4.6, the ZMP trajectory is defined by piecewise 3$^{\text{rd}}$-order polynomials. During double support, the ZMP moves to the center of the new support foot, and during single support, the ZMP moves to the toe-off position.

It is usually desirable to avoid generating significant angular momentum during walking. As a result, many dynamic planning algorithms assume collocation of the ZMP and CMP, resulting in zero angular momentum rate of change, or torque about the CoM. In this case, it is sufficient to define the eCMP to be equal to the ZMP over the planned preview window in order to satisfy the ZMP criterion. Alternatively, if the expected CoM height trajectory is known in advance, we can shift the eCMP away from the ZMP in order to achieve nonzero horizontal angular momentum rates of change, i.e.

$$\mathbf{r}_{ecmp} = \mathbf{r}_{zmp} + \frac{1}{m\,(g + \ddot{z}_{com})} \begin{bmatrix} \tau_y & -\tau_x & 0 \end{bmatrix}^T. \tag{4.14}$$

Here τ_x and τ_y represent the desired horizontal moments about the CoM and $\ddot{z}_{com}$ represents the desired vertical acceleration of the CoM.

4.5.2 Planning the ω trajectory

The time-varying DCM dynamics derived in Section 4.4 provide a simple framework for planning dynamically feasible DCM trajectories given a non-constant natural frequency. Excluding joint torque and range of motion limitations, any continuously differentiable ω trajectory can be realized subject to the constraints $\omega > \rho$ and $\omega^2 - \dot{\omega} > \beta$ for some small positive values ρ and β. In practice, however, it is often useful to compute an explicit plan for the CoM height when stepping over uneven terrain. In this case, the desired ω trajectory can be derived from a nominal vertical CoM trajectory, $z_{com}(t)$, with uniformly continuous velocity.

Solving (4.13) for $\dot{\omega}$ yields the nonlinear ω dynamics,

$$\dot{\omega} = \omega^2 - \alpha^2(t), \qquad \omega(t_0) = \omega_0, \tag{4.15}$$

where $\alpha(t)$ is the natural frequency of the LIPM defined by $\sqrt{\frac{\ddot{z}_{com}+g}{z_{com}-z_{ecmp}}}$. Thus, the reverse-time dynamics are given by

$$\dot{\omega}_r = \alpha^2(t_r) - \omega^2, \qquad \omega(t_f) = \omega_f, \tag{4.16}$$

[1]Note that the three-dimensional ZMP is not well-defined in the walking literature. Here we assume equivalence of the ZMP and center of pressure.

where $\dot{\omega}_r = -\dot{\omega}$ is the reverse-time derivative of the natural frequency, ω_f is the value of ω at the final time t_f, and $t_r = t_f - (t - t_0)$ is the reverse time variable.

Given $\omega_f > 0$, the reverse-time solution, $\omega(t_r)$, converges to $\alpha(t_r)$ as $t_r \to -\infty$ and $\dot{\alpha}(t_r) \to 0$. Furthermore, if $\omega_f > \rho$ and $\alpha(t_r) > \rho \ \forall t_r < t_f$ for some small positive value ρ, then $\omega(t_r) > \rho \ \forall t_r < t_f$. To see this note that if $\alpha^2(t_r) > \omega^2(t_r) > \rho^2$ then $\dot{\omega}_r > 0$. Referring back to the forward-time dynamics, these conditions also imply $\omega^2 - \dot{\omega} = \alpha^2(t) > \rho^2 > \beta$ for some small positive value β. Thus, given a CoM height trajectory satisfying $\alpha(t) > \rho$, i.e. $z_{com} - z_{ecmp} > 0$ and $\ddot{z}_{com} > -g$, an admissible ω trajectory can be computed from the natural frequency of the LIPM using reverse-time integration.

Given a desired terminal state, ω_f, we discretize the reverse-time dynamics and integrate backwards in time using a second-order Runge-Kutta method. When tracking the derived time-varying DCM trajectory, the CoM height closely follows the nominal vertical CoM trajectory assuming no large perturbations occur. Fig. 4.2 shows the planned eCMP, CoM height and natural frequency trajectories corresponding to a footstep plan with increasingly taller steps. The nominal vertical CoM trajectory is defined by piecewise 5^{th}-order polynomials, and the final time-varying natural frequency is initialized to $\omega_f = \alpha(t_f)$.

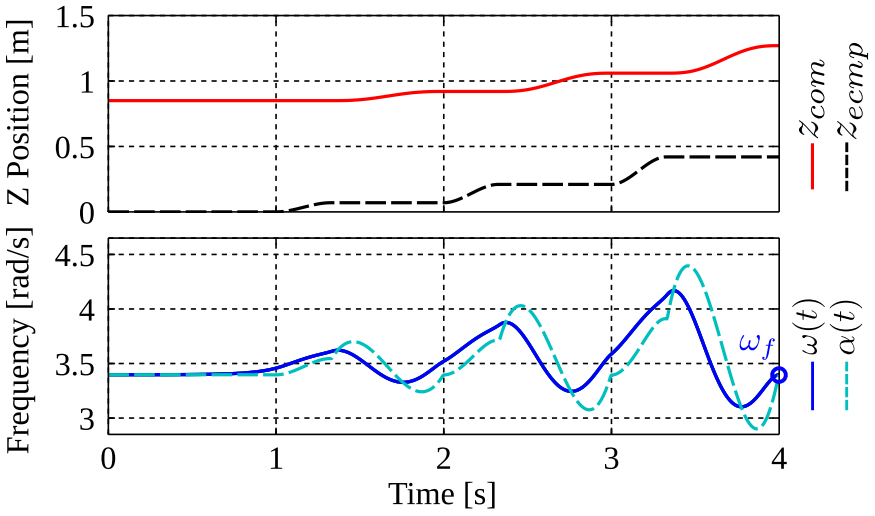

Figure 4.2: Top: Reference CoM and eCMP height. Bottom: LIPM natural frequency, $\alpha(t)$, and time-varying DCM natural frequency, $\omega(t)$, computed using reverse-time integration.

It should be noted that reverse-time integration can result in discontinuities between consecutive plans, i.e. $\omega(t_0) \neq \omega_0$, due to the lack of initial boundary conditions. In practice, however, this discontinuity is sufficiently small as to not cause noticeable issues when replanning at the onset of each step. This is partially due to the ω dynamics, which ensure $\omega^2(t_0) - \dot{\omega}(t_0) = \omega_0^2 - \dot{\omega}_0$ given continuous eCMP and CoM height reference trajectories.

4.5.3 Planning the DCM trajectory

In order to generate the final DCM reference trajectory, we propose a two-part approach consisting of reverse-time integration for long-term planning and model predictive control for short-term planning. Given the desired eCMP and ω trajectories, a corresponding VRP trajectory can be computed from (4.11). As discussed in Section 4.4, the time-varying DCM dynamics are unstable with respect to the VRP, causing the DCM to diverge with time. However, in reverse-time the VRP acts like an attractor,

$$\dot{\boldsymbol{\xi}}_r = -\left(\omega - \frac{\dot{\omega}}{\omega}\right)\left(\boldsymbol{\xi} - \mathbf{r}_{vrp}(t_r)\right), \qquad \boldsymbol{\xi}(t_f) = \boldsymbol{\xi}_f, \tag{4.17}$$

where $\dot{\boldsymbol{\xi}}_r = -\dot{\boldsymbol{\xi}}$ is the reverse-time derivative of the DCM.

Using a similar approach to the previous section, we apply reverse-time integration in order to compute a stable DCM trajectory given a desired terminal state $\boldsymbol{\xi}(t_f) = \boldsymbol{\xi}_f$. For the experiments in Section 4.6, the terminal DCM position is chosen to lie above the center of the support polygon during the final double support phase in the preview window. Fig. 4.3 shows an example DCM reference trajectory computed using reverse-time integration over a 2.0 s time horizon for a 2-step plan in the y-axis.

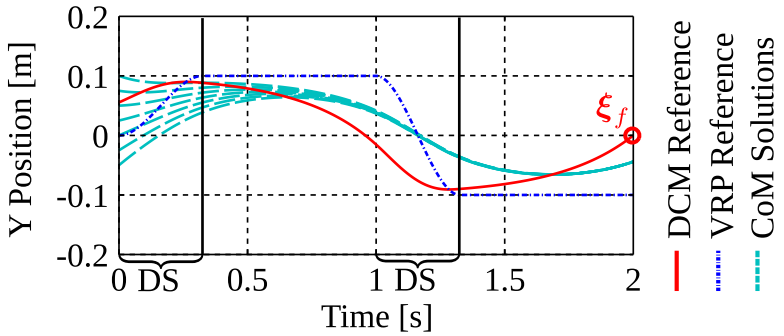

Figure 4.3: Lateral DCM and VRP trajectories for a 2-step plan. The DCM reference is computed using reverse-time integration over a 2.0 s time horizon given a terminal DCM position, $\boldsymbol{\xi}_f$. Example CoM solution trajectories are shown for initial CoM states consistent with the DCM reference. Double support phases are marked on the x-axis.

Reverse-time integration of the DCM dynamics results in a discontinuity in the reference trajectory occurring at the beginning of each planning window. Although larger preview windows can be used to minimize the discontinuity, in practice, the initial DCM offset can be significant (> 0.01 m) in the event of step timing and/or foothold positioning errors. This can result in significant ZMP errors during transition periods. In order to enforce initial boundary conditions on the DCM state, we implement discrete-time model predictive control (MPC) over a short preview window occurring at the beginning of each time horizon. The model predictive controller drives the DCM state to the reference trajectory computed via reverse-time integration in a smooth manner.

In order to compute the MPC solution, we first define the discrete-time DCM dynamics along a single axis,

$$\begin{aligned} \begin{bmatrix} \xi_{k+1} \\ \dot{\xi}_{k+1} \end{bmatrix} &= \underbrace{\begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}}_{\mathbf{A}} \begin{bmatrix} \xi_k \\ \dot{\xi}_k \end{bmatrix} + \underbrace{\begin{bmatrix} \Delta t^2/2 \\ \Delta t \end{bmatrix}}_{\mathbf{B}} u_k \\ y_k &= \underbrace{\begin{bmatrix} 1 & -\omega_k/(\omega_k^2 - \dot{\omega}_k) \end{bmatrix}}_{\mathbf{C}_k} \begin{bmatrix} \xi_k \\ \dot{\xi}_k \end{bmatrix}. \end{aligned} \tag{4.18}$$

Here Δt represents the sample period, ξ_k and $\dot{\xi}_k$ represent the DCM position and velocity, u_k represents the DCM acceleration, and y_k represents the VRP at time step k. Given an N-step preview window, the discrete-time DCM trajectory, $\mathbf{v}_\xi \in \mathbb{R}^{2N}$, can be expressed in terms of the initial state, $\boldsymbol{\xi}_0 \in \mathbb{R}^2$, and input trajectory $\mathbf{v}_u \in \mathbb{R}^N$ using the recursive relationship,

$$\mathbf{v}_\xi = \underbrace{\begin{bmatrix} \mathbf{A} \\ \mathbf{A}^2 \\ \vdots \\ \mathbf{A}^N \end{bmatrix}}_{\boldsymbol{\Phi}_0^\xi} \boldsymbol{\xi}_0 + \underbrace{\begin{bmatrix} \mathbf{B} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{AB} & \mathbf{B} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{N-1}\mathbf{B} & \mathbf{A}^{N-2}\mathbf{B} & \dots & \mathbf{B} \end{bmatrix}}_{\boldsymbol{\Phi}_u^\xi} \mathbf{v}_u.$$

Similarly, the VRP output trajectory, $\mathbf{v}_y \in \mathbb{R}^N$, is given by

$$\mathbf{v}_y = \underbrace{\begin{bmatrix} \mathbf{C}_1\mathbf{A} \\ \mathbf{C}_2\mathbf{A}^2 \\ \vdots \\ \mathbf{C}_N\mathbf{A}^N \end{bmatrix}}_{\boldsymbol{\Phi}_0^y} \boldsymbol{\xi}_0 + \underbrace{\begin{bmatrix} \mathbf{C}_1\mathbf{B} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{C}_2\mathbf{AB} & \mathbf{C}_2\mathbf{B} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_N\mathbf{A}^{N-1}\mathbf{B} & \mathbf{C}_N\mathbf{A}^{N-2}\mathbf{B} & \dots & \mathbf{C}_N\mathbf{B} \end{bmatrix}}_{\boldsymbol{\Phi}_u^y} \mathbf{v}_u.$$

By expressing the state and output histories in terms of the input trajectory and initial state, the MPC problem can be formulated as a quadratic program (QP) with the elements of $\mathbf{v}_u$ as the decision variables. This approach is inspired by the work of Wieber and Krause et al. who proposed similar MPC approaches for the CoM and CP dynamics [17; 87].

The MPC optimization is formulated as

$$\min_{\mathbf{v}_u} \frac{1}{2}\Delta\mathbf{v}_y^T\mathbf{Q}\Delta\mathbf{v}_y + \frac{1}{2}\mathbf{v}_u^T\mathbf{R}\mathbf{v}_u + \frac{1}{2}\Delta\mathbf{v}_\xi^T\mathbf{F}\Delta\mathbf{v}_\xi, \tag{4.19}$$

where $\Delta\mathbf{v}_y = \mathbf{v}_y - \mathbf{v}_{y,s}$ and $\Delta\mathbf{v}_\xi = \mathbf{v}_\xi - \mathbf{v}_{\xi,s}$ represent the output and state error vectors given the VRP and DCM solution trajectories, $\mathbf{v}_{y,s}$ and $\mathbf{v}_{\xi,s}$, obtained via reverse-time integration. Here $\mathbf{Q}$ and $\mathbf{F}$ are positive-semidefinite cost matrices weighting the VRP and DCM error terms. The third cost matrix, $\mathbf{R} > 0$, penalizes the DCM acceleration, thereby smoothing the optimized trajectories.

The unconstrained QP (4.19) is a variant of the linear-quadratic regulator (LQR) and can be solved using the least squares formulation, $\mathbf{b}_u = \mathbf{A}_u \mathbf{v}_u$ given

$$\mathbf{b}_u = -\mathbf{Q}\boldsymbol{\Phi}_u^{yT}(\boldsymbol{\Phi}_0^y \boldsymbol{\xi}_0 - \mathbf{v}_{y,s}) - \mathbf{F}\boldsymbol{\Phi}_u^{\xi\,T}(\boldsymbol{\Phi}_0^\xi \boldsymbol{\xi}_0 - \mathbf{v}_{\xi,s})$$

$$\mathbf{A}_u = \boldsymbol{\Phi}_u^{yT}\mathbf{Q}\boldsymbol{\Phi}_u^y + \boldsymbol{\Phi}_u^{\xi\,T}\mathbf{F}\boldsymbol{\Phi}_u^\xi + \mathbf{R}.$$

Noting that $\mathbf{A}_u$ is a positive-definite symmetric matrix, the solution can be computed efficiently using Cholesky decomposition. In order to compute the three-dimensional MPC solution over the desired preview window, we solve separate QPs for the x, y and z trajectories, noting that the linear DCM dynamics are decoupled in each axis.

Fig. 4.4 shows the optimized lateral DCM and VRP trajectories given the 2-step plan used in Fig. 4.3. As before, reverse-time integration is used to compute reference trajectories over a 2 s horizon. The MPC solution is computed over the first 0.5 s with $N = 100$, $\mathbf{Q} = \boldsymbol{I}$, $\mathbf{R} = 10^{-4}\boldsymbol{I}$, and $\mathbf{F} = 10^6\mathbf{S}_N$ where $\mathbf{S}_N$ is a selection matrix designed to weight only the final value of $\Delta\mathbf{v}_\xi$. In this example, the CoM velocity is initialized to 0 m/s to simulate a transition from standing to stepping. As a result, the initial CoM state and DCM position are inconsistent with the reverse-time DCM plan. The MPC optimization drives the DCM state to the reverse-time solution within the desired 0.5 s window.

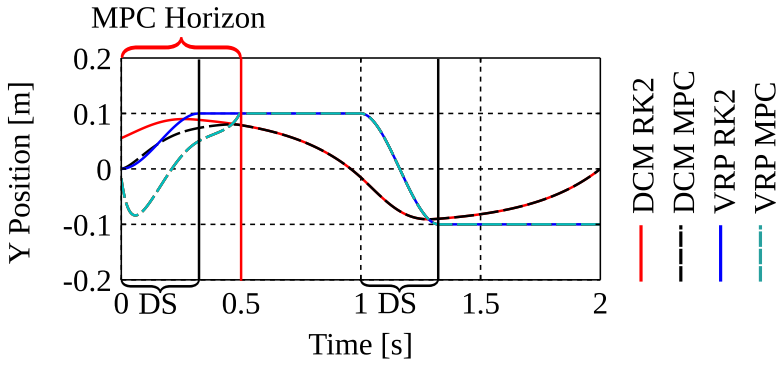

Figure 4.4: Lateral DCM and VRP trajectories for a 2-step plan. The DCM RK2 trajectory is computed using reverse-time integration over a 2.0 s time horizon. The DCM MPC trajectory is computed using linear MPC over a 0.5 s preview window. Double support phases are marked on the x-axis.

4.5.4 Tracking the DCM trajectory

The time-varying DCM trajectory is updated at the beginning of each double support phase and tracked using the following control law,

$$\mathbf{r}_{vrp} = \boldsymbol{\xi} - \frac{1}{\omega - \frac{\dot{\omega}}{\omega}}\left(\dot{\boldsymbol{\xi}}_r + \mathbf{k}_\xi\left(\boldsymbol{\xi}_r - \boldsymbol{\xi}\right) + \mathbf{k}_\Xi \int \left(\boldsymbol{\xi}_r - \boldsymbol{\xi}\right) dt\right),$$

where $\boldsymbol{\xi}_r$ and $\dot{\boldsymbol{\xi}}_r$ are the reference DCM position and velocity and k_ξ and k_Ξ are non-negative feedback gains. The first term cancels the nominal DCM dynamics, and the second term implements a proportional-integral controller with unity feedforward. Given the commanded VRP setpoint and CoM estimate, we compute the desired linear momentum rate of change using $\dot{\boldsymbol{l}}_d = m\left(\omega^2 - \dot{\omega}\right)\left(\mathbf{x} - \mathbf{r}_{vrp}\right)$ based on the VRP definition (4.9). For the experiments presented in Section 4.6, the momentum rate of change objective is tracked using a model-based whole-body controller similar to [24; 25].

We note that it is also possible to implement traditional model predictive control by replanning the DCM reference trajectory at each time step. In this case, the proposed tracking controller is not required, as the desired linear momentum rate of change is computed directly from the reference VRP. Although more computationally intensive, this approach may offer increased robustness to large disturbances and is intended as a future research direction.

4.6 Results

Fig. 4.5 shows an example reference trajectory generated using the proposed approach. The desired footstep plan consists of 4 forward steps of increasing height (0.07, 0.14, 0.21 and 0.28 m) with a step duration of 1 s, double support duration of 0.33 s, and single support duration of 0.67 s. The nominal vertical CoM trajectory is constant during each double support phase and increases to a height of 0.85 m above the support foot during each single support phase. The eCMP, VRP, and DCM trajectories are discretized with a sample period of $\Delta t = 0.005$ s.

Fig. 4.6 compares the corresponding DCM and contact force trajectories given a constant and time-varying natural frequency (shown in Fig. 4.2). Note that the vertical DCM trajectories vary significantly due to the natural frequency's effect on the VRP height. Although affected to a lesser degree, the time-varying and time-invariant horizontal DCM trajectories begin to diverge as $\left\|\left(\omega - \frac{\dot{\omega}}{\omega}\right) - \omega_0\right\|$ becomes large. This trend also appears in the commanded contact forces. While the specific normal forces associated with a constant natural frequency may be sub-optimal for stepping over varied terrain due to range of motion or torque limitations, the time-varying DCM dynamics provide a principled approach to planning DCM trajectories with generic normal force profiles. This could enable the design of DCM-based walking trajectories that more closely emulate biological models such as SLIP [88].

The proposed planning and control approach was tested using a 30 DOF model of THOR, the Tactical Hazardous Operations Robot [2], simulated in Gazebo [89]. THOR is a torque-controlled humanoid developed to compete in the DARPA Robotics Challenge. Fig. 4.7 shows the simulated THOR platform walking over 0.075 and 0.15 m tall cinder blocks, and Fig. 4.8 shows the resulting DCM and VRP trajectories. The step duration is 1.6 s and the double support duration is 0.32 s. For this experiment, the DCM plan was computed

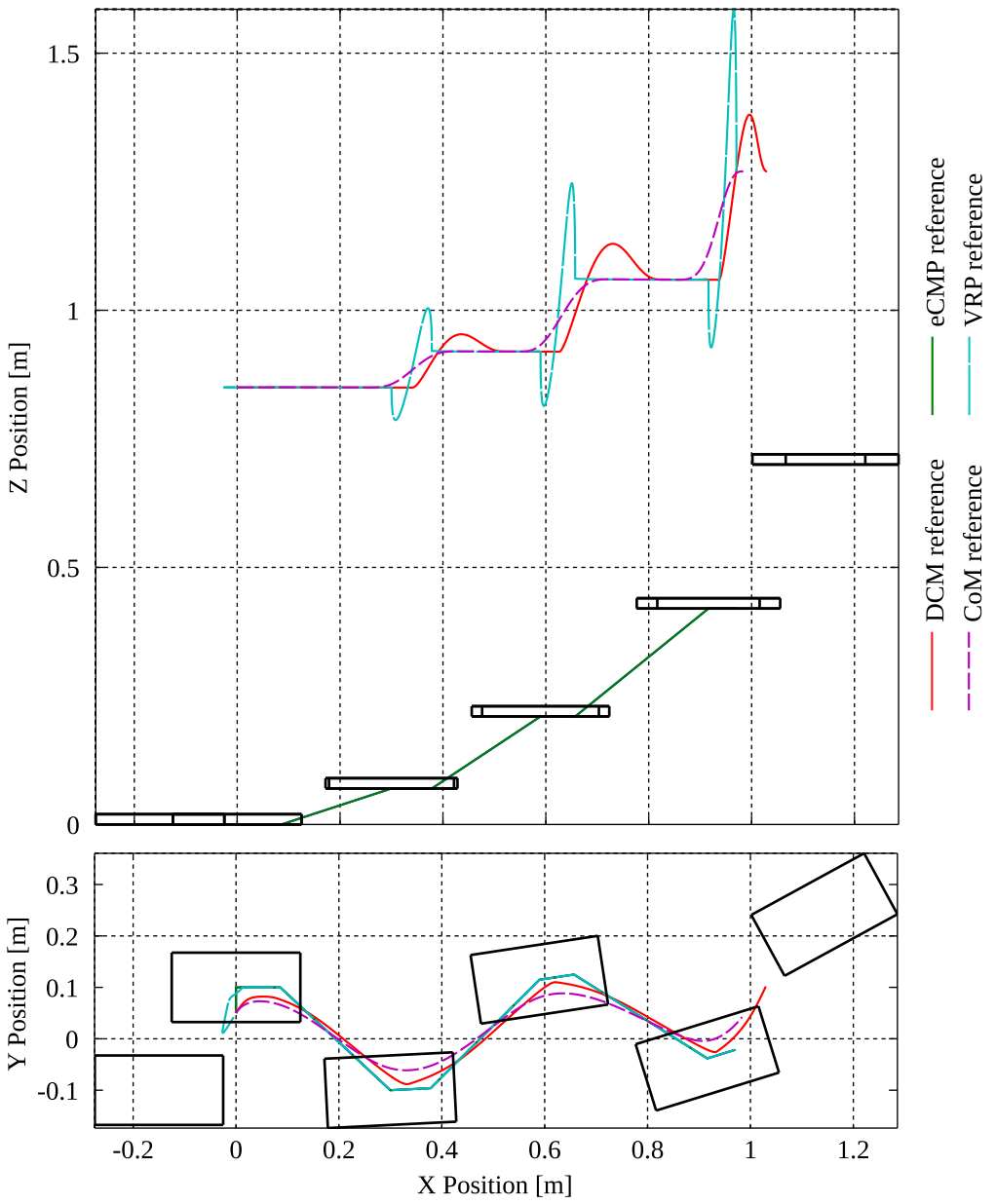

Figure 4.5: DCM, CoM, VRP, and eCMP plans given a 4-step preview window with increasing step height and turning radius.

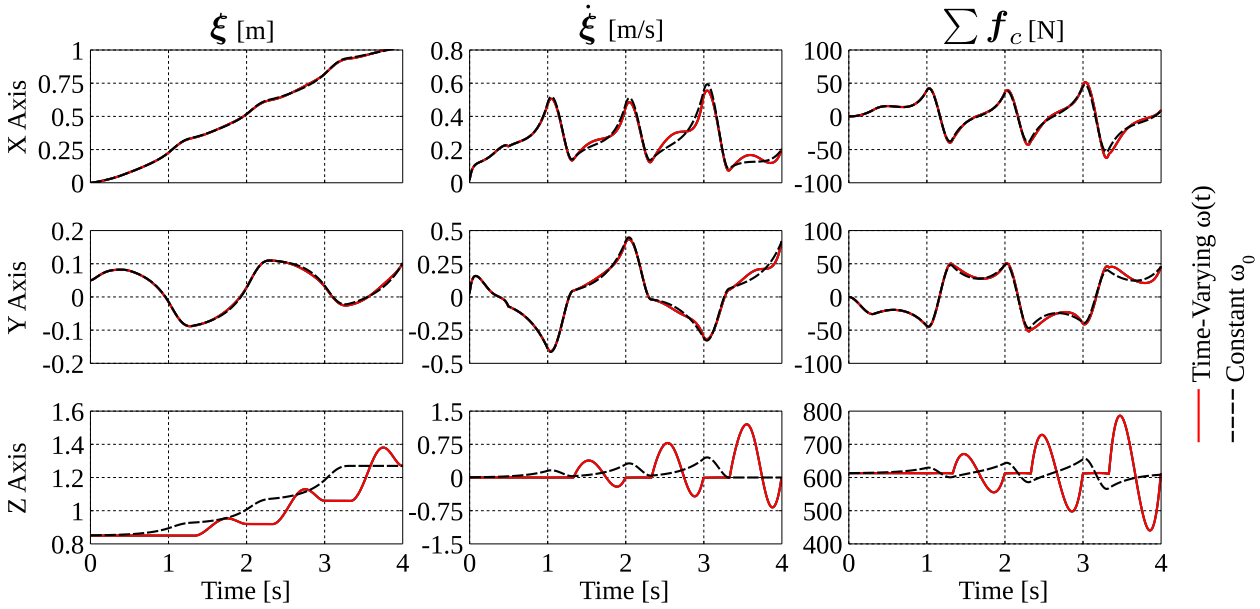

Figure 4.6: Comparison of DCM and contact force trajectories for the footstep plan shown in Fig. 4.5 given constant and time-varying natural frequencies.

at the beginning of each double support phase given a 3-step preview window. As seen in Fig. 4.8, the robot is able to track the time-varying DCM reference trajectory with minimal deviations to the nominal VRP reference using the DCM feedback controller presented in Section 4.5.4. Here the controller gains were set to $k_\xi = 2.0$ and $k_\Xi = 0.3$.

As discussed in section 4.5.1, the proposed time-varying DCM planner supports generic ZMP reference trajectories to enable heel-toe walking on flat and uneven terrain. Fig. 4.9 shows the THOR humanoid walking with a 0.35 m stride in simulation. As illustrated in this sequence, the step controller uses a toe-off phase to compensate for range of motion limits encountered in the knee and ankle joints during walking. This approach can be used to increase the maximum step size or to raise the CoM height in order to achieve more efficient walking.

The step controller updates at 200 Hz real-time on a 3.7 GHz i7 processor. The current unoptimized dynamic planning implementation takes approximately 0.002 s to compute a DCM reference trajectory over a 6.0 s time horizon with a 0.5 s MPC preview window and a 0.005 s sample period. The walking behavior is implemented using a simple state machine to transition between double and single support phases. Admissible joint torque setpoints are computed using a QP-based whole-body controller given the desired end-effector accelerations, momentum rates of change, and frictional contact constraints. In order to mitigate the effects of sensor noise, Kalman filtering is applied to the joint-space velocity measurements, and the DCM position is computed from the estimated joint states using a 36 DOF floating-base model of the robot.

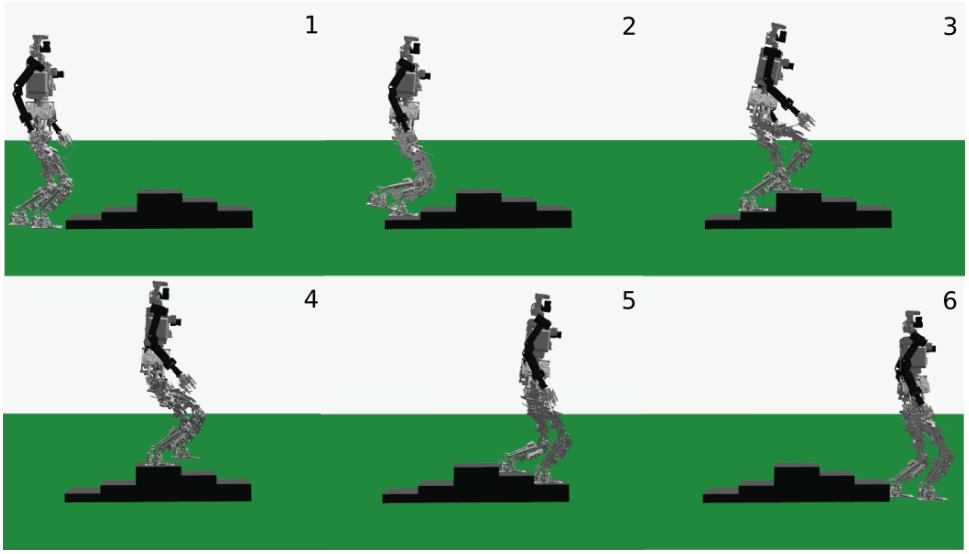

Figure 4.7: THOR walking over cinder blocks of varying height in simulation.

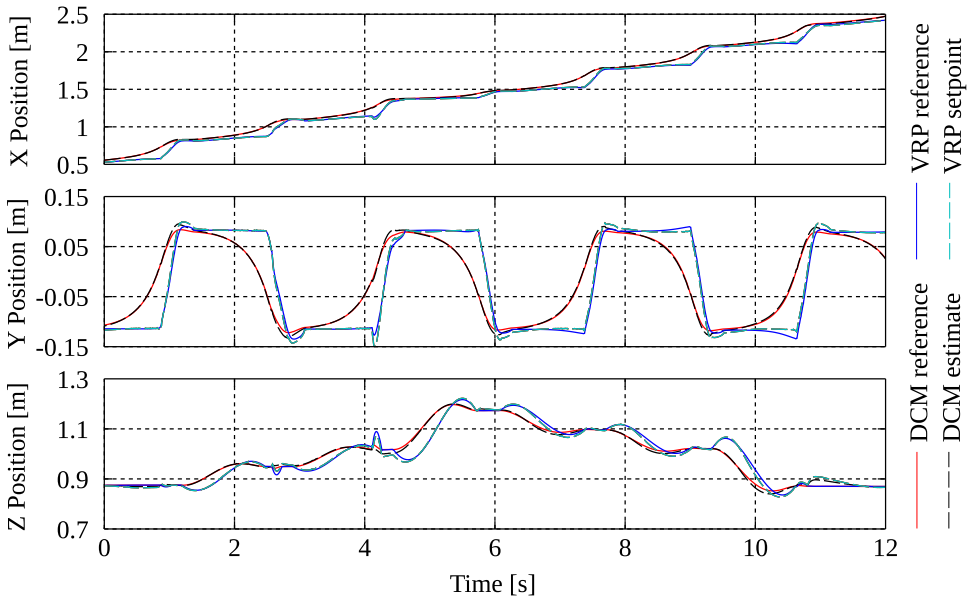

Figure 4.8: DCM and VRP trajectories while walking over cinder blocks in simulation (see Fig. 4.7). Each step is approximately 1.6 s with a double support duration of 0.32 s.

Figure 4.9: THOR walking with toe-off in simulation. The stride length is 0.35 m and the step duration is 1.5 s.

4.7 Conclusion

We presented a novel time-varying extension of the DCM and showed that this framework could be used to plan and track DCM reference trajectories with desired ZMP and CoM height characteristics. The time-varying natural frequency of the DCM effects the ground reaction forces commanded during stepping and may allow the design of more robust locomotion behaviors on uneven terrain. Although we presented a three-dimensional DCM tracking controller to stabilize the planned trajectories, the proposed approach could also be used for horizontal CP tracking in conjunction with a high-gain CoM height controller. By incorporating reverse-time integration for long-term planning and model predictive control for short-term planning, we are able to achieve real-time performance over a relatively large preview window.

Future research will focus on improvements to the time-varying DCM tracking controller to account for phase delay in the VRP feedback and unmodeled dynamics. We are also investigating full MPC and time-varying LQR approaches for continuous replanning at each time step to simplify adaptive behaviors such as dynamic footstep placement.

Chapter 5

Conclusion

The goal of this dissertation was to advance the state of the art in locomotion planning and control for compliant humanoids. Our work was motivated by the wealth of potential emergency response and disaster relief applications for humanoid robots, each requiring major advances in modern control systems. We presented a complete control framework for dynamic locomotion and whole-body control that was validated using the THOR humanoid, a prototype rescue robot developed as part of the ONR Shipboard Autonomous Firefighting Robot (SAFFiR) and DARPA Robotics Challenge (DRC) projects. This chapter describes the application of the presented approach to control problems encompassed by these two projects, including a discussion of future research efforts.

5.1 Shipboard Autonomous Firefighting Robot

The goal of the ONR SAFFiR project is to develop an autonomous humanoid robot to assist in routine maintenance and damage control tasks onboard a US Navy ship. In November 2014, the THOR humanoid successfully extinguished a controlled fire onboard the ex-USS Shadwell using the control framework presented in this dissertation. As part of a major demonstration, the robot was required to traverse a ship corridor, grasp a standard Navy handline hose, approach a compartment containing a fire, aim the nozzle, and suppress the fire using live water. Fig. 5.1 shows THOR completing the fire suppression task. The robot is wearing a water resistant suit to protect the internal electronics from airborne particulates and spray emitted by the nozzle.

The deck was coated in a relatively low-friction epoxy that varied in slope by approximately 5 degrees due to warping from multiple fire tests conducted on the ship. Additionally, the fire nozzle exerted a significant wrench on the grasping hand due to the weight and pressure of the water-filled hose. Despite these challenges, the robot was able to walk to the compartment door while grasping the nozzle and maintain balance while extinguishing

the fire. A high-level footstep planner provided desired foothold poses and step durations in order to properly position the robot. The use of a task-space control formulation greatly reduced the complexity of the whole-body behaviors used to carry and aim the hose. We repurposed the stand and step behaviors defined in Chapter 3 and assigned large whole-body controller weights to the upper body acceleration objectives to provide accurate trajectory tracking for manipulating the hose.

Figure 5.1: THOR suppressing a controlled fire onboard the ex-USS Shadwell in Mobile, AL. Images used with permission of Virginia Tech/Logan Wallace.

5.2 DARPA Robotics Challenge

The DARPA Robotics Challenge (DRC) is an ongoing competition promoting the development of fieldable robots for disaster relief applications. The competition will require robots to complete a series of challenging locomotion and manipulation tasks given a degraded communications link to an operator control station. The DRC finals, to be held in June 2015, will likely rank teams based on the ability to complete a number of sequential tasks in an allocated time window, simulating a disaster relief scenario. The final tasks may include driving and exiting a vehicle, passing through a door, passing through a debris-filled corridor, using a power tool to cut through a wall, turning a valve, traversing a terrain course, inserting

a plug into a receptacle, and climbing a set of stairs, among other tasks. Fig. 5.2 shows example layouts for the terrain and debris tasks featured at the DRC trials event held in December 2013.

As discussed in Chapter 4, the proposed control framework is designed to support dynamic locomotion on uneven terrain. Unfortunately, we are currently limited by the available knee torque of the THOR humanoid, which prevents the robot from stepping up or down at significant heights. In preparation for the DRC finals, the THOR design will undergo major modifications to support the necessary joint torque requirements for locomotion on terrain similar to Fig. 5.2. The new humanoid, ESCHER, will feature approximately double the available knee torque, allowing it to ascend steps at a height of at least 10 inches. Porting the presented whole-body control and locomotion framework to the ESCHER hardware platform should not pose a major challenge, as many of the proposed algorithms are generic to torque-controlled humanoids and ESCHER shares a similar mechatronic design to THOR.

Figure 5.2: Left: Terrain task layout at DRC trials. Right: Debris task layout at DRC trials. Images courtesy of DARPA.

5.3 Discussion

This dissertation presented a hierarchical control system enabling the THOR hardware platform to balance in the presence of significant external forces and to traverse various terrain including cement, gravel, and grass. Our approach consists of three primary components. First, a low-level joint-space impedance controller was developed for rotary joints driven by linear SEAs in a parallel or serial configuration (an increasingly common design found in the compliant humanoids). Second, an optimization-based inverse dynamics and whole-body control approach was developed to support compliant locomotion behaviors on the THOR hardware platform. Third, a novel dynamic planning and control approach

was proposed using a time-varying extension of the Divergent Component of Motion. The combined approach represents a principled approach to dynamic locomotion and whole-body control for compliant humanoids.

Future research efforts will focus on improvements in model-based state estimation and control to enable fast, stable locomotion over unstructured terrain. Since the presented locomotion planning approach enables generic ZMP and vertical CoM trajectories during walking, a natural extension of this work relates to planning reference trajectories that optimize performance metrics such as gait efficiency and stability. Although not discussed in this dissertation, the proposed approach can be readily extended to multi-contact whole-body locomotion behaviors such as climbing a ladder or entering a vehicle. Since foot placement and step timing adjustments play a critical role in the stability of legged systems, we are also interested in the application of adaptive control approaches for dynamic step recovery.

One of the primary lessons learned during the course of our research is that, in addition to an accurate dynamic model, reliable low-level control and state estimation are critical to the success of inverse dynamics-based approaches. Most task-space control formulations assume a rigid body model for articulated humanoids where the available actuators are represented as pure joint torque sources. It may be possible to significantly improve the whole-body controller performance by incorporating a model of the actuator dynamics into the multibody dynamic model used by the inverse dynamics optimization. We are also interested in utilizing more accurate models for linear series elastic actuators in order to improve the performance of the actuator force controller implemented on THOR and ESCHER.

Bibliography

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

[2] B. Lee, "Design of a humanoid robot for disaster response," Master's thesis, Virginia Polytechnic Institute and State University, April 2014.

[3] M. Srinivasan and A. Ruina, "Computer optimization of a minimal biped model discovers walking and running," *Nature*, vol. 439, no. 7072, pp. 72–75, 2006.

[4] M. Posa and R. Tedrake, "A direct method for trajectory optimization of rigid bodies through contact," *Int. J. Rob. Res.*, vol. 33, no. 1, pp. 69–81, 2014.

[5] P. Sardain and G. Bessonnet, "Forces acting on a biped robot. Center of Pressure-Zero Moment Point," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, vol. 34, no. 5, pp. 630–637, Sept 2004.

[6] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3D linear inverted pendulum mode: a simple modeling for a biped walking pattern generation," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, vol. 1, 2001, pp. 239–246.

[7] K. Harada, K. Miura, M. Morisawa, K. Kaneko, S. Nakaoka, F. Kanehiro, T. Tsuji, and S. Kajita, "Toward human-like walking pattern generator," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Oct 2009, pp. 1071–1077.

[8] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture Point: A Step toward Humanoid Push Recovery," in *Humanoid Robots (Humanoids), 6th IEEE-RAS International Conference on*, Dec 2006, pp. 200–207.

[9] T. Takenaka, T. Matsumoto, and T. Yoshiike, "Real time motion generation and control for biped robot -1st report: Walking gait pattern generation," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Oct 2009, pp. 1084–1091.

[10] J. Englsberger, C. Ott, M. Roa, A. Albu-Schaffer, and G. Hirzinger, "Bipedal walking control based on Capture Point dynamics," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Sept 2011, pp. 4420–4427.

[11] J. Englsberger, C. Ott, and A. Albu-Schaffer, "Three-dimensional bipedal walking control using Divergent Component of Motion," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Nov 2013, pp. 2600–2607.

[12] J. Englsberger and C. Ott, "Integration of vertical COM motion and angular momentum in an extended Capture Point tracking controller for bipedal walking," in *Humanoid Robots (Humanoids), 12th IEEE-RAS International Conference on*, Nov 2012, pp. 183–189.

[13] M. Morisawa, S. Kajita, F. Kanehiro, K. Kaneko, K. Miura, and K. Yokoi, "Balance control based on Capture Point error compensation for biped walking on uneven terrain," in *Humanoid Robots (Humanoids), 12th IEEE-RAS International Conference on*, Nov 2012, pp. 734–740.

[14] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," in *Robotics and Automation (ICRA), IEEE International Conference on*, vol. 2, 2003, pp. 1620–1626.

[15] J. Park and Y. Youm, "General ZMP preview control for bipedal walking," in *Robotics and Automation, 2007 IEEE International Conference on*, April 2007, pp. 2682–2687.

[16] J. Mayr, H. Gattringer, and H. Bremer, "Online walking gait generation with predefined variable height of the center of mass," in *Intelligent Robotics and Applications*, ser. Lecture Notes in Computer Science, S. Jeschke, H. Liu, and D. Schilberg, Eds. Springer Berlin Heidelberg, 2011, vol. 7102, pp. 569–578.

[17] P.-B. Wieber, "Trajectory free linear model predictive control for stable walking in the presence of strong perturbations," in *Humanoid Robots (Humanoids), 6th IEEE-RAS International Conference on*, Genova, Italy, Dec 2006, pp. 137–142.

[18] D. Dimitrov, P. B. Wieber, H. Ferreau, and M. Diehl, "On the implementation of model predictive control for on-line walking pattern generation," in *Robotics and Automation (ICRA), IEEE International Conference on*, May 2008, pp. 2685–2690.

[19] H. Diedam, D. Dimitrov, P. B. Wieber, K. Mombaur, and M. Diehl, "Online walking gait generation with adaptive foot positioning through linear model predictive control," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Sept 2008, pp. 1121–1126.

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

[21] L. Sentis and O. Khatib, "A whole-body control framework for humanoids operating in human environments," in *Robotics and Automation (ICRA), IEEE International Conference on*, May 2006, pp. 2641–2648.

[22] M. de Lasa and A. Hertzmann, "Prioritized optimization for task-space control," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Oct 2009, pp. 5755–5762.

[23] S.-H. Lee and A. Goswami, "Ground reaction force control at each foot: A momentum-based humanoid balance controller for non-level and non-stationary ground," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Oct 2010, pp. 3157–3162.

[24] T. Koolen, J. Smith, G. Thomas, S. Bertrand, J. Carff, N. Mertins, D. Stephen, P. Abeles, J. Englsberger, S. McCrory, J. van Egmond, M. Griffioen, M. Floyd, S. Kobus, N. Manor, S. Alsheikh, D. Duran, L. Bunch, E. Morphis, L. Colasanto, K.-L. Ho Hoang, B. Layton, P. Neuhaus, M. Johnson, and J. Pratt, "Summary of team IHMC's virtual robotics challenge entry," in *Humanoid Robots (Humanoids), 13th IEEE-RAS International Conference on*, Oct 2013.

[25] A. Herzog, L. Righetti, F. Grimminger, P. Pastor, and S. Schaal, "Experiments with a hierarchical inverse dynamics controller on a torque-controlled humanoid," *arXiv preprint arXiv:1305.2042*, 2013.

[26] P. M. Wensing and D. E. Orin, "Generation of dynamic humanoid behaviors through task-space control with conic optimization," in *Robotics and Automation (ICRA), IEEE International Conference on*, May 2013, pp. 3103–3109.

[27] S. Feng, X. Xinjilefu, W. Huang, and C. G. Atkeson, "3D walking based on online optimization," in *Humanoid Robots (Humanoids), 13th IEEE-RAS International Conference on*, Oct 2013.

[28] L. Saab, O. Ramos, F. Keith, N. Mansard, P. Soueres, and J. Fourquet, "Dynamic whole-body motion generation under rigid contacts and other unilateral constraints," *Robotics, IEEE Transactions on*, vol. 29, no. 2, pp. 346–362, April 2013.

[29] S. Kuindersma, F. Permenter, and R. Tedrake, "An efficiently solvable quadratic program for stabilizing dynamic locomotion," in *Robotics and Automation (ICRA), IEEE International Conference on*, May 2014.

[30] S. Feng, E. Whitman, X. Xinjilefu, and C. G. Atkeson, "Optimization based full body control for the Atlas robot," in *Humanoid Robots (Humanoids), 14th IEEE-RAS International Conference on*, Nov 2014.

[31] M. de Lasa, I. Mordatch, and A. Hertzmann, "Feature-based locomotion controllers," *ACM Transactions on Graphics*, vol. 29, no. 4, 2010.

[32] D. Orin and A. Goswami, "Centroidal momentum matrix of a humanoid robot: Structure and properties," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Sept 2008, pp. 653–659.

[33] A. Macchietto, V. Zordan, and C. R. Shelton, "Momentum control for balance," *ACM Transactions on Graphics*, vol. 28, no. 3, 2009.

[34] D. E. Orin, A. Goswami, and S.-H. Lee, "Centroidal dynamics of a humanoid robot," *Autonomous Robots*, pp. 1–16, 2013.

[35] M. B. Popovic and H. Herr, "Ground reference points in legged locomotion: Definitions, biological trajectories and control implications," *Int. J. Robot. Res*, vol. 24, no. 12, p. 1013–1032, 2005.

[36] T. J. Roberts and E. Azizi, "Flexible mechanisms: the diverse roles of biological springs in vertebrate movement," *The Journal of experimental biology*, vol. 214, no. 3, pp. 353–361, 2011.

[37] S. K. Au, J. Weber, and H. Herr, "Powered ankle–foot prosthesis improves walking metabolic economy," *Robotics, IEEE Transactions on*, vol. 25, no. 1, pp. 51–66, 2009.

[38] G. Pratt and M. Williamson, "Series elastic actuators," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on. 'Human Robot Interaction and Cooperative Robots'*, vol. 1, Aug 1995, pp. 399–406.

[39] D. Robinson, J. Pratt, D. Paluska, and G. Pratt, "Series elastic actuator development for a biomimetic walking robot," in *Advanced Intelligent Mechatronics, IEEE/ASME International Conference on*, 1999, pp. 561–568.

[40] K. Kong, J. Bae, and M. Tomizuka, "Control of rotary series elastic actuator for ideal force-mode actuation in human-robot interaction applications," vol. 14, no. 1, pp. 105–118, 2009.

[41] N. Paine, S. Oh, and L. Sentis, "Design and control considerations for high-performance series elastic actuators," *Mechatronics, IEEE/ASME Transactions on*, vol. 19, no. 3, pp. 1080–1091, June 2014.

[42] D. J. Paluska, "Design of a humanoid biped for walking research," Master's thesis, Massachusetts Institute of Technology, Sept 2000.

[43] J. Pratt and B. Krupp, "Design of a bipedal walking robot," in *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, vol. 6962, Apr. 2008.

[44] 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, 2012.

[45] M. Slovich, N. Paine, K. Kemper, A. Metger, A. Edinger, J. Weber, and L. Sentis, "Building HUME: A bipedal robot for human-centered hyper-agility," in *Dynamic Walking Meeting*, 2012.

[46] D. F. Lahr, "Design and control of a bipedal robot," Ph.D. dissertation, Virginia Polytechnic Institute and State University, March 2014.

[47] N. Paine, "High-performance series elastic actuation," Ph.D. dissertation, University of Texas at Austin, 2014.

[48] M. Hutter, C. Gehring, M. Bloesch, M. Hoepflinger, C. D. Remy, and R. Siegwart, "StarlETH: A compliant quadrupedal robot for fast, efficient, and versatile locomotion," in *15th International Conference on Climbing and Walking Robot (CLAWAR)*, 2012.

[49] N. Tsagarakis, S. Morfey, G. Cerda, L. Zhibin, and D. Caldwell, "COMpliant huMANoid COMAN: Optimal joint stiffness tuning for modal frequency control," in *Robotics and Automation (ICRA), IEEE International Conference on*, May 2013, pp. 673–678.

[50] J. Pratt, P. Dilworth, and G. Pratt, "Virtual model control of a bipedal walking robot," in *Robotics and Automation (ICRA), IEEE International Conference on*, Apr 1997, pp. 193–198.

[51] 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 (IROS), IEEE/RSJ International Conference on*, Sept 2008, pp. 2486–2491.

[52] I. Olaru, S. Krut, and F. Pierrot, "Novel mechanical design of biped robot SHERPA using 2 DOF cable differential modular joints," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, 2009, pp. 4463–4468.

[53] B. Vanderborght, *Dynamic Stabilisation of the Biped Lucy Powered by Actuators with Controllable Stiffness*. Springer, 2010, vol. 63.

[54] J. Englsberger, A. Werner, C. Ott, B. Henze, M. A. Roa, G. Garofalo, R. Burger, A. Beyer, O. Eiberger, K. Schmid, and A. Albu-Schäffer, "Overview of the torque-controlled humanoid robot TORO," in *Humanoid Robots (Humanoids), 14th IEEE-RAS International Conference on*, Nov 2014.

[55] C. Semini, N. G. Tsagarakis, E. Guglielmino, M. Focchi, F. Cannella, and D. G. Caldwell, "Design of HyQ - a hydraulically and electrically actuated quadruped robot," *Journal of Systems and Control Engineering*, vol. 225, no. 6, pp. 831–849, 2011.

[56] M. A. Hopkins, D. W. Hong, and A. Leonessa, "Compliant locomotion using whole-body control and Divergent Component of Motion tracking," in *Robotics and Automation (ICRA), IEEE International Conference on*, May 2015. (Under Review).

[57] M. A. Hopkins, D. W. Hong, and A. Leonessa, "Humanoid locomotion on uneven terrain using the time-varying Divergent Component of Motion," in *Humanoid Robots (Humanoids), 14th IEEE-RAS International Conference on*, Nov 2014.

[58] Y. Ito, S. Nozawa, J. Urata, T. Nakaoka, K. Kobayashi, Y. Nakanishi, K. Okada, and M. Inaba, "Development and verification of life-size humanoid with high-output actuation system," in *Robotics and Automation (ICRA), IEEE International Conference on*, May 2014, pp. 3433–3438.

[59] M. Grebenstein, A. Albu-Schaffer, T. Bahls, M. Chalon, O. Eiberger, W. Friedl, R. Gruber, S. Haddadin, U. Hagn, R. Haslinger, H. Hoppner, S. Jorg, M. Nickl, A. Nothhelfer, F. Petit, J. Reill, N. Seitz, T. Wimbock, S. Wolf, T. Wusthoff, and G. Hirzinger, "The DLR hand arm system," in *Robotics and Automation (ICRA), IEEE International Conference on*, May 2011, pp. 3175–3182.

[60] D. Lahr, V. Orekhov, B. Lee, and D. Hong, "Development of a parallely actuated humanoid, SAFFiR," in *ASME International Design Engineering Technical Conference*, 2013.

[61] M. Gienger, K. Loffler, and F. Pfeiffer, "Towards the design of a biped jogging robot," in *Robotics and Automation (ICRA), IEEE International Conference on*, vol. 4, 2001, pp. 4140–4145.

[62] N. Hogan, "Impedance control: An approach to manipulation," in *American Control Conference*, 1984, pp. 304–313.

[63] G. Pratt, P. Willisson, C. Bolton, and A. Hofman, "Late motor processing in low-impedance robots: impedance control of series elastic actuators," in *American Control Conference*, vol. 4, June 2004, pp. 3245–3251.

[64] D. Bentivegna, C. Atkeson, and J.-Y. Kim, "Compliant control of a hydraulic humanoid joint," in *Humanoid Robots (Humanoids), 7th IEEE-RAS International Conference on*, Nov 2007, pp. 483–489.

[65] H. Vallery, J. Veneman, E. van Asseldonk, R. Ekkelenkamp, M. Buss, and H. van Der Kooij, "Compliant actuation of rehabilitation robots," *Robotics Automation Magazine, IEEE*, vol. 15, no. 3, pp. 60–69, Sept 2008.

[66] M. Hutter, C. D. Remy, M. A. Hoepflinger, and R. Siegwart, "High compliant series elastic actuation for the robotic leg ScarlETH," in *Proceedings of the International Conference on Climbing and Walking Robots (CLAWAR)*, 2011.

[67] N. L. Tagliamonte and D. Accoto, "Passivity constraints for the impedance control of series elastic actuators," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 228, no. 3, pp. 138–153, 2014.

[68] M. Focchi, G. A. Medrano-Cerda, T. Boaventura, M. Frigerio, C. Semini, J. Buchli, and D. G. Caldwell, "Robot impedance control and passivity analysis with inner torque and velocity feedback loops," *arXiv preprint arXiv:1406.4047*, 2014.

[69] Y. H. Tsoi, S. Xie, and G. Mallinson, "Joint force control of parallel robot for ankle rehabilitation," in *ICCA IEEE International Conference on Control and Automation*, 2009, pp. 1856–1861.

[70] C. Knabe, B. Lee, V. Orekhov, and D. Hong, "Design of a compact, lightweight, electromechanical linear series elastic actuator," in *ASME International Design Engineering Technical Conference*, 2014.

[71] B. Lee, C. Knabe, V. Orekhov, and D. Hong, "Design of a human-like range of motion hip joint for humanoid robots," in *ASME International Design Engineering Technical Conference*, 2014.

[72] V. Orekhov, D. Lahr, B. Lee, and D. Hong, "Configurable compliance for series elastic actuators," in *ASME International Design Engineering Technical Conference*, 2013.

[73] S. A. Ressler, "Design and implementation of a dual axis motor controller for parallel and serial series elastic actuators," Master's thesis, Virginia Polytechnic Institute and State University, April 2014.

[74] K. J. Waldron and H. K. H., "Series-parallel dualities in actively coordinated mechanisms," *The International Journal of Robotics Research*, vol. 10, no. 5, pp. 473–480, 1991.

[75] R. Featherstone, *Rigid Body Dynamics Algorithms*. Springer Berlin, 2008, vol. 49.

[76] V. L. Orekhov, C. S. Knabe, M. A. Hopkins, and D. W. Hong, "An unlumped model for linear series elastic actuators with ball screw drives," in *Robotics and Automation (ICRA), IEEE International Conference on*, 2015. (Under Review).

[77] 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, Aug 2012.

[78] T. Sugihara, "Standing stabilizability and stepping maneuver in planar bipedalism based on the best COM-ZMP regulator," in *Robotics and Automation (ICRA), IEEE International Conference on*, May 2009, pp. 1966–1971.

[79] J. Englsberger, T. Koolen, S. Betrand, J. Pratt, C. Ott, and A. Albu-Schaffer, "Trajectory generation for continuous leg forces during double support and heel-to-toe shift based on divergent component of motion," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, Sept 2014.

[80] Y. Abe, M. da Silva, and J. Popovic, "Multiobjective control with frictional contacts," in *Symposium on Computer Animation*, M. Gleicher and D. Thalmann, Eds. Eurographics Association, 2007, pp. 249–258.

[81] C. Ott, M. Roa, and G. Hirzinger, "Posture and balance control for biped robots based on contact force optimization," in *Humanoid Robots (Humanoids), 11th IEEE-RAS International Conference on*, Oct 2011, pp. 26–33.

[82] D. Goldfarb and A. Idnani, "A numerically stable dual method for solving strictly convex quadratic programs," *Mathematical Programming*, vol. 27, no. 1, pp. 1–33, 1983.

[83] C. Knabe, B. Lee, and D. Hong, "An inverted straight line mechanism for augmenting joint range of motion in a humanoid robot," in *ASME International Design Engineering Technical Conference*, 2014.

[84] B. Lee, C. Knabe, V. Orekhov, and D. Hong, "Design of a human-like range of motion hip joint for humanoid robots," in *ASME International Design Engineering Technical Conferences*, 2014.

[85] S. Bertrand, T. Wu, and J. Pratt, "Momentum-based control framework and capturibility-based walking control - application to the humanoid robot Atlas." Presented at the Institute for Human and Machine Cognition (IHMC), Pensacola, Florida, March 14, 2014.

[86] M. F. Fallon, M. Antone, N. Roy, and S. Teller, "Drift-free humanoid state estimation fusing kinematic, inertial and LIDAR sensing," in *Humanoid Robots (Humanoids), 14th IEEE-RAS International Conference on*, Nov 2014.

[87] M. Krause, J. Englsberger, P.-B. Wieber, and C. Ott, "Stabilization of the capture point dynamics for bipedal walking based on model predictive control," in *IFAC Symposium on Robot Control (SYROCO)*, vol. 10, no. 1, 2012, pp. 165–171.

[88] H. Geyer, A. Seyfarth, and R. Blickhan, "Compliant leg behaviour explains basic dynamics of walking and running," in *Proc. Biol Sci.*, Nov 2006.

[89] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *Intelligent Robots and Systems (IROS), IEEE/RSJ International Conference on*, vol. 3, Sept 2004, pp. 2149–2154.

Appendices

Appendix A

Ground Reference Points

Ground reference points such as the center of pressure (CoP), Zero Moment Point (ZMP) [5], and Centroidal Moment Pivot (CMP) [35] are commonly employed throughout the bipedal literature to analyze the centroidal dynamics and derive control laws that satisfy weak stability criteria. The center of pressure (CoP) is defined as the average point of pressure (or normal force) over the support surface, and therefore cannot exist outside the convex hull of the contact points, often referred to as the base of support. When two feet are in contact with the ground, the net CoP is defined as the pressure-weighted average of the individual foot CoPs, assuming no other points of contact exist between the robot and environment [5].

The Zero Moment Point (ZMP) is commonly used to derive CoM trajectories that satisfy no-tip constraints on the support feet. The ZMP is defined as the point on the ground where the component of reaction torque tangential to the surface normal is equal to zero. This is expressed by the equality,

$$\dot{\mathbf{k}} = (\mathbf{r}_{zmp} - \mathbf{x}) \times \sum \mathbf{f}_c + \sum \boldsymbol{\tau}_n, \tag{A.1}$$

where $\mathbf{x} = [\, x_{com}, y_{com}, z_{com} \,]^T$ is the CoM position, $\dot{\mathbf{k}}$ is the net angular momentum rate of change, $\sum \mathbf{f}_c$ is the net ground reaction force, and $\sum \boldsymbol{\tau}_n$ is the total normal torque applied by each foot. As discussed in [5], the ZMP is equivalent to the net CoP on flat terrain. Unfortunately, the three-dimensional ZMP is not well-defined in the walking literature, although some authors have suggested varying definitions [5; 35]. If the ZMP lies strictly inside the base of support, ground reaction forces can be directed without tipping the foot about an edge. This property has made ZMP planning and control a popular approach for dynamic walking.

The Centroidal Moment Pivot (CMP) is commonly used in conjunction with the ZMP to control the spin angular momentum during balancing. The CMP is defined as the point on the ground that lies on the line parallel to the total ground reaction force, passing through

the the CoM [35]. This is expressed by the constraint,

$$\mathbf{0} = (\mathbf{r}_{cmp} - \mathbf{x}) \times \sum \mathbf{f}_c, \tag{A.2}$$

Collocation of the ZMP and CMP ensures that the moment about the CoM resulting from the total ground reaction force is equal to zero. In practice, many ZMP-based approaches assume ZMP/CMP equivalence in order to avoid generating spin angular momentum during locomotion. Although the ZMP must remain inside the support polygon, the CMP can leave the base of support in order to stabilize the linear CoM dynamics by inducing a nonzero angular momentum rate of change.

For a constant CoM height and flat ground plane, the horizontal CoM dynamics are given by

$$\mathbf{P}\ddot{\mathbf{x}} = \omega_0^2 \left(\mathbf{P}\mathbf{x} - \mathbf{P}\mathbf{r}_{cmp}\right). \tag{A.3}$$

These dynamics are equivalent to the linear inverted pendulum model (LIPM) [6]. Here $\mathbf{P} \in \mathbb{R}^{2\times 2}$ is a matrix operator that projects the CoM and CMP onto the xy-plane, and $\omega_0 = \sqrt{\frac{g}{z_0}}$ is the natural frequency of the linear inverted pendulum given a nominal CoM height of z_0 [77].