Distributed Feedback Control Algorithms for Cooperative Locomotion: From Bipedal to Quadrupedal Robots

Vinaykarthik R. Kamidi

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
Mechanical Engineering

Kaveh Akbari Hamed, Chair
Alexander Leonessa
Corina Sandu
Ryan Williams

March 11, 2022
Blacksburg, Virginia

Keywords: Legged Locomotion, Distributed Nonlinear Control, Distributed Quadratic Programming, Real-Time Control, Hybrid Systems
Copyright 2022, Vinaykarthik R. Kamidi

Distributed Feedback Control Algorithms for Cooperative Locomotion: From Bipedal to Quadrupedal Robots

Vinaykarthik R. Kamidi

(ABSTRACT)

This thesis synthesizes general and scalable distributed nonlinear control algorithms with application to legged robots. It explores both naturally decentralized problems in legged locomotion, such as the collaborative control of human-lower extremity prosthesis and the decomposition of high-dimensional controllers of a naturally centralized problem into a network of low-dimensional controllers while preserving equivalent performance. In doing so, strong nonlinear interaction forces arise, which this thesis considers and sufficiently addresses. It generalizes to both symmetric and asymmetric combinations of subsystems. Specifically, this thesis results in two distinct distributed control algorithms based on the decomposition approach.

Towards synthesizing the first algorithm, this thesis presents a formal foundation based on decomposition, Hybrid Zero Dynamics (HZD), and scalable optimization to develop distributed controllers for hybrid models of collaborative human-robot locomotion. This approach considers a centralized controller and then decomposes the dynamics and parameterizes the feedback laws to synthesize local controllers. The Jacobian matrix of the Poincaré map with local controllers is studied and compared with the centralized ones. An optimization problem is then set up to tune the parameters of the local controllers for asymptotic stability. It is shown that the proposed approach can significantly reduce the number of controller parameters to be optimized for the synthesis of distributed controllers, deeming the method computationally tractable. To evaluate the analytical results, we consider a human amputee with the point of separation just above the knee and assume the average physical parameters of a human male. For the lower-extremity prosthesis, we consider the PRleg, a powered knee-ankle prosthetic leg, and together, they form a 19 Degrees of Freedom (DoF) model. A multi-domain hybrid locomotion model is then employed to rigorously assess the performance of the aforestated control algorithm via numerical simulations. Various simulations involving the application of unknown external forces and altering the physical parameters of the human model unbeknownst to the local controllers still result in stable amputee locomotion, demonstrating the inherent robustness of the proposed control algorithm.

In the later part of this thesis, we are interested in developing distributed algorithms for the real-time control of legged robots. Inspired by the increasing popularity of Quadratic programming (QP)-based nonlinear controllers in the legged locomotion community due to their ability to encode control objectives subject to physical constraints, this thesis exploits the idea of distributed QPs. In particular, this thesis presents a formal foundation

to systematically decompose QP-based centralized nonlinear controllers into a network of lower-dimensional local QPs. The proposed approach formulates a feedback structure between the local QPs and leverages a one-step communication delay protocol. The properties of local QPs are analyzed, wherein it is established that their steady-state solutions on periodic orbits (representing gaits) coincide with that of the centralized QP. The asymptotic convergence of local QPs' solutions to the steady-state solution is studied via Floquet theory. Subsequently, to evaluate the effectiveness of the analytical results, we consider an 18 DoF quadrupedal robot, A1, as a representative example. The network of distributed QPs mentioned earlier is condensed to two local QPs by considering a front-hind decomposition scheme. The robustness of the distributed QP-based controller is then established through rigorous numerical simulations that involve exerting unmodelled external forces and introducing unknown ground height variations. It is further shown that the proposed distributed QPs have reduced sensitivity to noise propagation when compared with the centralized QP.

Finally, to demonstrate that the resultant distributed QP-based nonlinear control algorithm translates equivalently well to hardware, an extensive set of blind locomotion experiments on the A1 robot are undertaken. Similar to numerical simulations, unknown external forces in the form of aggressive pulls and pushes were applied, and terrain uncertainties were introduced with the help of arbitrarily displaced wooden blocks and compliant surfaces. Additionally, outdoor experiments involving a wide range of terrains such as gravel, mulch, and grass at various speeds up to 1.0 (m/s) reiterate the robust locomotion observed in numerical simulations. These experiments also show that the computation time is significantly dropped when the distributed QPs are considered over the centralized QP.

Distributed Feedback Control Algorithms for Cooperative Locomotion: From Bipedal to Quadrupedal Robots

Vinaykarthik R. Kamidi

(GENERAL AUDIENCE ABSTRACT)

Inspiration from animals and human beings has long driven the research of legged locomotion and the subsequent design of the robotic counterparts: bipedal and quadrupedal robots. Legged robots have also been extended to assist human amputees with the help of powered prostheses and aiding people with paraplegia through the development of exoskeleton suits. However, in an effort to capture the same robustness and agility demonstrated by nature, our design abstractions have become increasingly complicated. As a result, the ensuing control algorithms that drive and stabilize the robot are equivalently complicated and subjected to the curse of dimensionality. This complication is undesirable as failing to compute and prescribe a control action quickly destabilizes and renders the robot uncontrollable.

This thesis addresses this issue by seeking nature for inspiration through a different perspective. Specifically, through some earlier biological studies on cats, it was observed that some form of locality is implemented in the control of animals. This thesis extends this observation to the control of legged robots by advocating an unconventional solution. It proposes that a high-dimensional, single-legged agent be viewed as a virtual composition of multiple, low-dimensional subsystems. While this outlook is not new and forms precedent to the vast literature of distributed control, the focus has always been on large-scale systems such as power networks or urban traffic networks that preserve sparsity, mathematically speaking. On the contrary, legged robots are underactuated systems with strong interaction forces acting amongst each subsystem and dense mathematical structures. This thesis considers this problem in great detail and proposes developments that provide theoretical stability guarantees for the distributed control of interconnected legged robots. As a result, two distinctly different distributed control algorithms are formulated.

We consider a naturally decentralized structure appearing in the form of a human-lower extremity prosthesis to synthesize distributed controllers using the first control algorithm. Subsequently, the resultant local controllers are rigorously validated through extensive full-order simulations. In order to validate the second algorithm, this thesis considers the problem of quadrupedal locomotion as a representative example. It assumes for the purposes of control synthesis that the quadruped is comprised of two subsystems separated at the geometric center, resulting in a front and hind subsystem. In addition to rigorous validation via numerical simulations, in the latter part of this thesis, to demonstrate that distributed controllers preserve practicality, rigorous and extensive experiments are undertaken in indoor and outdoor settings on a readily available quadrupedal robot A1.

Dedication

To my parents, Aruna R. Kamidi and Laxma R. Kamidi.

Acknowledgments

To Dr. Kaveh Akbari Hamed: I am deeply appreciative for your mentorship and the opportunities you have provided me throughout my doctoral studies. You have made this the most rewarding journey and better than I could ever imagine. I am still amazed at the depth and breadth of knowledge you possess and the ease with which you convey the most complex concepts. I am grateful to have seen the world of math and controls through your perspective and will most certainly miss our discussions on various topics of control theory. Besides being a mentor, you have extended your friendship to me. I will cherish our spur-of-the-moment chats on the most random topics. I now realize that these chats had always come when I was struggling with roadblocks that are now a major part of this thesis. Those little distractions allowed me to reset and refocus. In closing, I hope to emulate your scientific approach and grow to become more like you.

I also acknowledge my thesis committee: Dr. Alexander Leonessa, Dr. Corina Sandu, and Dr. Ryan Williams, for their efforts in improving this thesis and the valuable advice they provided during my preliminary examination.

I have been fortunate enough to collaborate and develop ideas with some of the brilliant members of the legged locomotion community. I particularly enjoyed my discussions with Dr. Aaron D. Ames and Dr. Robert D. Gregg. I want to express my sincerest gratitude to them for the opportunity to learn and for the valuable insights that helped shape parts of this thesis.

I have had the most fantastic time at the Hybrid Dynamic Systems and Robot Locomotion Lab (HDSRL). I appreciate the time I spent with the other members (in no particular order): Jesseop Kim, Abhishek Pandala, Randy Fawcett, Leila Amanzadeh, and Basit Muhammad Imran. They were always available to discuss and provide genuine feedback that helped polish my ideas.

I would also like to thank my best friends: Ashiya Gillani, Shruthi Yellu (who is also my aunt) and Swetcha Yellu (my goofy little cousin), for being my support system during the most challenging times. They have kept me grounded, focused, and cheered me on. They have lent me their ear and acted as my venting mechanism. They have helped create an incredible ecosystem that made life outside the lab equivalently enjoyable.

Thank you to my parents, Aruna R. Kamidi and Laxma R. Kamidi, for their unconditional love and support. This journey could not have been possible without you. Thank you for everything, for the freedom to pursue my dreams and for encouraging me every step of the way.

Contents

List of Figures xii

List of Tables xvii

List of Videos xviii

1 Introduction 1

1.1 Motivation for Study . . . 2
1.2 Current Limitations . . . 3
1.3 Current Scope of Decentralized Architecture . . . 4
1.4 Research Goals, Objectives, and Contributions . . . 6
1.5 Significance and Importance . . . 9
1.6 State-of-the-Art in Legged Locomotion: Modeling and Control . . . 9
1.6.1 Dynamic Modeling . . . 9
1.6.2 Trajectory Optimization . . . 14
1.6.3 Existing Control Approaches . . . 17
1.7 Relevant Publications & Attribution . . . 20
1.8 Contributed Works . . . 21
1.9 Outline . . . 22

I Distributed Controllers: A Scalable Approach based on Hybrid Zero Dynamics 24

2 Preliminaries: HZD-Based Legged Locomotion Control 25

2.1 Introduction . . . 25
2.2 Hybrid Systems Formulation . . . 25

2.3 Continuous-Time Dynamics . . . 26
2.3.1 Contact Models . . . 27
2.3.2 Constrained Continuous-Time Dynamics . . . 28
2.4 Discrete-Time Dynamics . . . 29
2.5 Open-Loop Hybrid Model . . . 30
2.6 Centralized Feedback Controllers . . . 31
2.6.1 Family of Parameterized Virtual Constraint Controllers . . . 31
2.7 Summary . . . 32

3 Synthesis of Distributed Controllers **33**
3.1 Introduction . . . 33
3.1.1 Contributions of this Chapter . . . 33
3.2 Dynamic Decomposition . . . 34
3.3 Control Decomposition . . . 35
3.3.1 Output Modification . . . 35
3.4 Exponential Stability Analysis . . . 37
3.4.1 Poincaré Sections Analysis for Closed-loop Systems . . . 37
3.5 Distributed Control Synthesis . . . 41
3.6 Summary . . . 41

4 Applications to Cooperative Human-Robot Locomotion **43**
4.1 Introduction . . . 44
4.1.1 Contributions of this Chapter . . . 44
4.1.2 Related Work and Motivation . . . 44
4.2 Physical Model . . . 45
4.3 Gait Optimization . . . 47
4.3.1 Problem Formulation . . . 48
4.3.2 Constraints . . . 48
4.3.3 Cost Function . . . 49

4.3.4 NLP via FROST . . . 50
4.4 System Decomposition . . . 50
4.5 Numerical Simulations . . . 52
4.5.1 Asymptotically Stable Walking: Decomposition from a Stable Orbit . 53
4.5.2 Robustness Analysis . . . 54
4.5.3 Centralized vs Distributed Controllers . . . 57
4.6 Summary and Discussion . . . 58

II Distributed Quadratic Programming-based Controllers: A Scalable Approach based on I-O Linearization 60

5 Network of Distributed Quadratic Programming-Based Controllers 61
5.1 Introduction . . . 61
5.1.1 Motivation . . . 61
5.1.2 Contributions of this Chapter . . . 62
5.1.3 Related work . . . 62
5.2 Problem Formulation . . . 62
5.3 Network of Distributed QPs . . . 64
5.3.1 Synthesis of Local QPs . . . 65
5.4 Asymptotic Convergence Analysis . . . 66
5.5 Summary . . . 67

6 Application to Quadrupedal Locomotion 68
6.1 Introduction . . . 69
6.1.1 Related Work . . . 69
6.1.2 Contributions of this Chapter . . . 69
6.2 Robot Model Description . . . 70
6.3 QP-Based I-O Linearizing Controller . . . 71
6.3.1 Centralized QP . . . 71

6.3.2 Distributed QPs for Quadrupedal Locomotion . . . 73
6.4 Numerical Simulations . . . 74
6.4.1 Nominal Trotting and Robustness Analysis . . . 76
6.4.2 Differentiating Distributed from Centralized Controllers . . . 76
6.5 Hardware Realization: Real-Time Control on A1 . . . 77
6.5.1 Indoor Experiments . . . 78
6.5.2 Outdoor Experiments . . . 79
6.6 Summary and Discussion . . . 81

7 Conclusions and Future Work **85**
7.1 Summary of Contributions . . . 85
7.1.1 Decomposition-Based Distributed Controllers . . . 85
7.1.2 Decomposition-Based Distributed QP Controllers . . . 86
7.1.3 Experimental Validation on a Quadrupedal Platform . . . 86
7.2 Possible Directions for Future Work . . . 87
7.3 Final Thoughts . . . 88

Bibliography **89**

List of Figures

1.1 Some representative examples of existing state-of-the-art legged robots: (a) A1 (Unitree Robotics) [8], (b) Anymal-C (ANYbotics) [1], (c) Vision60 (Ghost Robotics) [6], (d) HYQ [182], (e) Spot (Boston Dynamics) [3], (f) MIT mini cheetah [7], (g) Cassie and Digit (Agility Robotics) [2], (h) Atlas (Boston Dynamics) [3], (i) Powered Knee/Ankle Prosthesis (Bionic Engineering Lab) [154], (j) Ampro3 [220], (k) PRleg [71], and (l) Open-Source Prosthesis Leg [36]. 1

1.2 General, high-level, illustration of different feedback control architectures available for controlling a *decentralized* structure. (a) centralized control architecture, (b) decentralized control architecture, and (c) distributed control architecture. The figure adopts customary notation used in the controls society, where x denotes the states and u denotes the controls. 3

1.3 Overview of the research aims R1-R3 addressed by this thesis. This main objective of this thesis is to develop general, scalable, and distributed control algorithms and the subsequent extension to the various applications of legged locomotion: from bipeds to quadrupeds. 8

1.4 Illustrations of three different approaches toward modeling legged robots with the help of an advanced quadrupedal robot, Vision 60: (a) the entire robot abstracted as a LIP model, (b) centroidal model: the entire robot abstracted as an ellipsoid, and (c) full-order methodology that accounts for all the rigid links in the dynamic model. 10

1.5 Illustrations of two predominant rigid-body abstractions: (a) fixed-base serial, tree structure (b) floating-base serial, tree structure. 13

2.1 Illustration of a multi-domain hybrid model emulating a quadrupedal ambling gait. 26

2.2 Typical foot contacts observed in legged locomotion: (a) point contact, (b) line contact, and, (c) planar contact. 28

3.1 Illustration of the proposed decomposition framework, applicable to any legged locomotion model: (a) front-hind decomposition of quadrupedal robot (top and isometric view), (b) left-right decomposition of quadrupedal robot (top and isometric view), (c) left-right decomposition of humanoid robot (isometric view), and (d) decomposition of human-prosthesis model (isometric view). 34

4.1 Illustration of the cooperative locomotion problem considered: (a) human-robot model visualized as a single agent, (b) human-robot model decomposed into two sub systems, (c) closer look at the lower-extremity prosthesis abstracted as a stick-figure model. (d) CAD model of the robotic prosthesis considered in this work and, (e) assembled life-size robotic prosthesis, PRleg [71]. 43

4.2 (a) Illustration of the individual links in the human-prosthesis model, (b) schematic presenting the joint notation followed for the human subsystem, (c) illustration of the joint notation adopted for the prosthetic subsystem, and, (d) description of the foot profile assumed and the corresponding illustration of link lengths and step constraints used for TO. 45

4.3 Multi-contact hybrid model of cooperative human-prosthesis locomotion: a 4-domain approximation of human-walking that captures the heel-rise phase. We denote $\{v_1, v_3\}$ as push-off phases and $\{v_2, v_4\}$ as flat-footed phases. 46

4.4 illustration of the synthesized distributed controller for the cooperative-human locomotion problem. (a) Controller block for the human subsystem, Σ_h, that assumes the parameters for the prosthetic subsystem, Σ_r, is on the orbit and calculates the control action, u_h. (b) Controller block for the prosthetic subsystem, Σ_r that assumes the parameters for the human subsystem, Σ_h, is on the orbit and calculates the control action, u_r. (c) Distributed control actions that are calculated concurrently and applied to the cumulative model. 51

4.5 Illustration of an elaborate sequence of simulations undertaken to evaluate the performance of the proposed distributed controllers: the black node, (a) depicts the simulation as a result of decomposition from a stable orbit. The blue nodes, (b)-(e) denote the robustness analysis. The red node, (g) indicates the comparative analysis between centralized and distributed controllers when a white Gaussian noise is induced into the prosthesis joints. 52

4.6 Snapshots captured at the 100^{th} step, illustrating stable walking as a result of the distributed controller synthesized from a stable orbit, $\mathcal{O}_{\text{stable}}$. (a) and (c) capture double stance phase, (b) and (d) capture heel-rise phase. 53

4.7 Limit cycles and outputs illustrating an asymptotically stable walking gait at 0.2 (m/s). The results are accumulated for cooperative human-robot walking over 100 steps. 53

4.8 (a)-(d) Snapshots captured at 8^{th} step illustrating the destabilizing sequence triggered by an intentionally designed, poorly tuned, centralized controller. (e)-(h) Snapshots captured at 100^{th} step illustrating stable walking as a result of the distributed controller that is synthesized from an unstable orbit, $\mathcal{O}_{\text{unstable}}$, extracted from the simulation in (a)-(d). 54

4.9 Limit cycles depicting convergence to an orbit as a result of distributed controllers synthesized from an *unstable* centralized orbit. The results are accumulated for cooperative human-robot walking over 100 steps. 55

4.10 (a)-(c) Limit cycles depicting convergence to an orbit albeit an external disturbance force of 125 (N) applied along the x-axis. (d) Torques in the prosthesis subsystem. (e)-(g) Limit cycles depicting convergence to an orbit albeit an external disturbance force applied of 200 (N) along the y-axis. (h) Torques in the prosthesis subsystem. 56

4.11 (a)-(c) Limit cycles depicting convergence to an orbit when the torso mass is varied to 32 (kg). (d) Torques in the prosthesis subsystem. (e)-(g) Limit cycles depicting convergence to an orbit albeit varying the torso mass to 45 (kg). (h) Torques in the prosthesis subsystem. 57

4.12 Simulation results demonstrating the performance of centralized, (a)-(b) and distributed controllers, (c)-(d) when the velocity components ($\dot{q}_r$) of Σ_r are induced with white Gaussian noise. (a),(c) capture the torques of Σ_h while (b),(d) capture Σ_r torques. Simulation was executed for 40 steps but results are truncated to 1 step to clearly highlight differences. 58

5.1 Illustration of the proposed network of distributed QPs based on the decomposition approach. 63

6.1 A composite of three separate images from the same experimental run illustrating the evolution of the distributed QP-based control algorithm on a quadrupedal platform, A1 [8]. Here, the robot is blindly trotting forward (from right to left) in the presence of arbitrarily dispersed wooden blocks at the speed of 0.5 (m/s). 68

6.2 (a) Joint description of a 18-DoF quadrupedal robot. (b) Illustration of the two-domain trot gait considered in this realization. Here, the filled circle indicates contact with ground. 70

6.3 An illustration of the outputs and the different structures of QP-based controllers considered: (a) nominal centralized, (b) augmented centralized, and (c) distributed controller adopted from Chapter 5. 72

6.4 Illustration of an extensive set of simulations and experiments undertaken to evaluate the performance of the proposed distributed QP-based controllers. The blue nodes indicate simulation case studies: (a) nominal forward walking, (b)-(d) robustness analysis, and (e) performance of the the proposed distributed controller in the presence of noisy velocities. The red nodes signify experimental evaluation: (f) a series of indoor experiments and (g) a series of outdoor experiments. 74

6.5 Phase plots, (a)-(e) of the body roll, body pitch, body yaw, hip pitch, and knee pitch corresponding to the *simulations* with nominal trot at 0.8 (m/s), payload at 0.3 (m/s), planks at 0.3 (m/s) and external disturbances (see Figure 6.8 (a)-(c)) applied during in-place trotting. 76

6.6 Hip roll torque (from simulation) in the distributed control scheme against the centralized control scheme. In this case study, the noise was injected into the velocity components of the front subsystem (front right leg), and plotted here are the torques of the right hind leg. The propagation of noise into the hind subsystem in the centralized control scheme is visibly higher. Here, the gray overlay indicates ground contact. 77

6.7 Snapshots illustrating a series of blind locomotion experiments (indoor) utilizing the proposed distributed QP-based controllers on the quadrupedal platform, A1. (a) nominal forward trot, (b) nominal backward trot, (c) backward trot in a field of arbitrarily displaced wooden blocks, (d) lateral trot on arbitrarily displace wooden blocks, (e) forward locomotion in a field of arbitrarily dispersed wooden blocks, (f) trotting on a compliant surface (gym mat) scattered with wooden blocks, (g) forward locomotion with a payload of 4.54 (kg) (36% uncertainty), (h) aggressive tethered pulling while trotting in-place. 78

6.8 (a)-(c) Simulation case-study evaluating the performance of the distributed controller in presence of external disturbance provided by a 3 (kg) object flying at the robot in the positive y-axis. (d)-(f) Experiments demonstrating recovery from an aggressive push during in-place trotting. 79

6.9 Phase plots, (a)-(e) of the body roll, body pitch, body yaw, hip pitch, and knee pitch corresponding to the *experiments* with payloads (see Figure 6.7 (c)), tethered pulling (see Figure 6.7 (d)) and external pushes (see Figure 6.8 (d)-(f)). An overlay of the nominal trot at 0.5 (m/s) is also provided. 80

6.10 Comparison of the hip roll torque of the front right leg against the corresponding steady-state torque during nominal trot experiment, demonstrating close convergence. 80

6.11 Snapshots illustrating various blind locomotion experiments in natural environments (outdoor) utilizing the proposed distributed QP-based controllers on the quadrupedal platform, A1. (a) Gravel with an abrupt change in terrain height at the speed of 0.4 (m/s), (b) mulch at the speed of 0.35 (m/s), (c) concrete at the speed of 0.35 (m/s), (d) grass at the speed of 1.0 (m/s). 80

6.12 (a) A big picture view to clearly illustrate the inclination of the terrain A1 is set to traverse. Inset: (b)-(e) close-up snapshots (view from left to right) illustrating stable downhill locomotion on the inclined landscape while blindly trotting backward at the speed of 0.35 (m/s). 81

6.13 Virtual constraints (outputs) for all the outdoor experimental runs on A1. The *top row* indicates the outputs of the front subsystem and the *bottom row* indicates the outputs in the hind subsystem. Here, ϕ, θ and ψ indicate roll, pitch and yaw, respectively. 82

6.14 Phase plots of the body roll, body pitch and body yaw corresponding to, (a)-(c) gravel, (d)-(e) mulch, (j)-(k) 0.1 (m/s), and (m)-(o) inclined slope. The y-axis denotes the angular velocities. 83

6.15 Joint torques and GRFs at the leg ends from the outdoor experimental runs: (a)-(e) loosely displaced gravel, (f)-(j) mulch, (k)-(o) grass at the speed of 1.0 (m/s), and (p)-(t) downhill locomotion. 84

List of Tables

1.1 A Non-Extensive List of Relevant Legged Locomotion Control Literature . . 19

2.1 Contact Constraints Associated with Typical Foot Contacts in Legged Locomotion 27

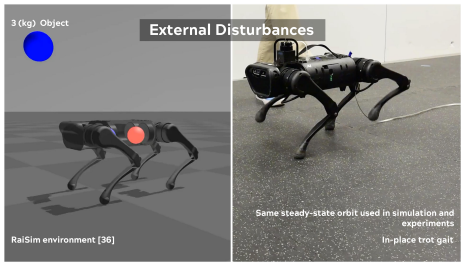
4.1 Physical Parameters Utilized for the Cooperative Locomotion Model 47

4.2 TO Parameters for the Cooperative Human-Robot Locomotion Problem . . 50

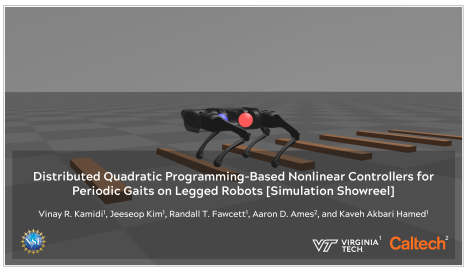
List of Videos

This section enumerates a list of videos that were published during the development of this thesis.

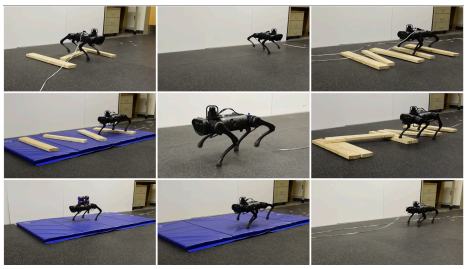
Distributed Controllers for Human-Robot Locomotion [Supplementary Video]
https://youtu.be/MiGhjggJhpI



Distributed Quadratic Programming-Based Nonlinear Controllers for Periodic Gaits on Legged Robots
https://youtu.be/VoYYCZrJOps



Simulation Showreel: Distributed QPs on A1
https://youtu.be/k-wg1svFVxo



Experiment Showreel: Distributed QPs on A1
https://youtu.be/ED4ZfMZJPE8

List of Abbreviations

ASLIP Assymetric Spring Loaded Inverted Pendulum

CBF Control Barrier Functions

CLF Control Lyapunov Functions

CoM Center of Mass

CoP Center of Pressure

DMPC Distributed Model Predictive Control

DoF Degrees of Freedom

DP Dynamic Programming

EoM Equations of Motion

GRFs Ground Reaction Forces

HZD Hybrid-Zero Dynamics

I-O Input-Output

IMU Inertial Measurement Unit

IP Interior Point Method

ISS Input-to-State-Stability

KKT Karush-Kuhn-Tucker

LIP Linear Inverted Pendulum

LLS Lateral Leg Spring

LP Linear Program

MPC Model Predictive Control

NLP Nonlinear Programming Problem

OCP Optimal Control Problem

ODE Ordinary Differential Equation

QP Quadratic Program

SLIP Spring Loaded Inverted Pendulum

TO Trajectory Optimization

VCs Virtual Constraints

WBC Whole Body Control

Chapter 1

Introduction

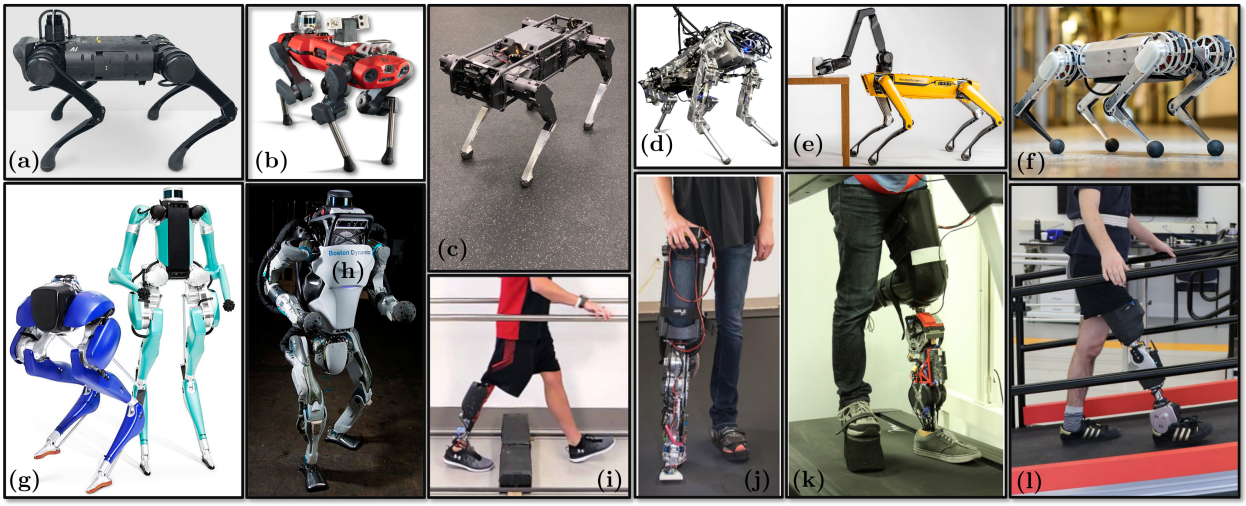


Figure 1.1: Some representative examples of existing state-of-the-art legged robots: (a) A1 (Unitree Robotics) [8], (b) Anymal-C (ANYbotics) [1], (c) Vision60 (Ghost Robotics) [6], (d) HYQ [182], (e) Spot (Boston Dynamics) [3], (f) MIT mini cheetah [7], (g) Cassie and Digit (Agility Robotics) [2], (h) Atlas (Boston Dynamics) [3], (i) Powered Knee/Ankle Prosthesis (Bionic Engineering Lab) [154], (j) Ampro3 [220], (k) PRleg [71], and (l) Open-Source Prosthesis Leg [36].

Legged robots are uniquely placed in that they do not require any new infrastructure unlike wheeled robots, and are fit to operate in preexisting structures designed around human needs, thus becoming ideal candidates to assist alongside humans in hazardous environments. They can additionally restore functionality to lower-extremity amputees through powered prostheses or paraplegics through exoskeletons. However, this has been a longstanding goal and has proven to be a difficult endeavor. This is evident from the many falls observed in the DARPA Robotics Challenge (DRC) [5] highlighting the *critical limitations* in feedback control theory of high-dimensional legged robots. This observation has been ingrained in the research community for quite a while now and has led to some impressive demonstrations [52, 53, 161] aftermath DRC.

However, to address the many failures observed in DRC, control algorithms have become increasingly complex and nonlinear, which are not necessarily tractable by existing numerical methods. This complexity then raises the question of *scalability*— Legged robots being

inherently unstable systems, complex controllers can not take forever to prescribe the following control action. Failure to prescribe the following control action in a timely manner destabilizes and renders the system uncontrollable. This puts us in a rather paradoxical situation, wherein we want to capture the intricate details of legged locomotion but avoid the computational complexity it brings along. We are then left asking if it is possible to account for all the nuances of legged robots but still end up with a tractable and efficient structure that existing numerical methods can address?

1.1 Motivation for Study

Much of the physical attributes in legged robotics are bio-inspired—almost all modern legged robots bear close resemblances to their biological counterparts (see Figure 1.1) and a class of modeling techniques is informed by biology (biomechanics, see Section 1.6.1). Similarly, plenty of studies has been undertaken to observe the *structure* in the control architectures employed by animals/humans (neurophysiology). This section brings to focus one such specific study that paralyzed cats by severing the spinal cord, and as a consequence, any direct connection of the hind legs with the brain was nonexistent [126]. This naturally formed two subsystems, the first subsystem comprised of the front legs whose connection with the brain was still intact, and the second subsystem was composed of the hind legs. This is what we will refer to as a "decentralized" structure (interchangeably refereed to as a subsystem throughout). In this study, the cat was then placed on a treadmill wherein, it was able to execute a wide variety of gaits, indicating that its ability to locomote was unimpaired. *This demonstrates that a great deal of locality in control is employed by the biological counterparts.* This observation has existed for quite a while now but has *not* been widely translated to the feedback control of legged robots.

The necessity for general, scalable, and efficient control algorithms (theoretical standpoint): On the other hand, with rapid technological advancements in design, legged robot hardware is becoming significantly complex (see Figure 1.1 again). This is reflected by the increasing nonlinearites enabled by the high Degrees of Freedom (DoFs) in current state-of-the-art designs. Existing nonlinear control approaches consider the entire robot as a single system (this is what we will refer to as a "centralized" structure) and makes use of centralized feedback control algorithms, in which multiple links and joints of the robot communicate their sensor measurements to a central controller (see Figure 1.2 (a)). This controller interprets the data from all the joints and relays decisions back to all the actuators. However, the approach is restrictive in that, they *fail to scale* with the dimensionality of the robots. For instance, consider feedback linearization, a widely accepted nonlinear control method that requires the inversion of a state-dependent "decoupling" matrix to compute the control action at every time sample. The computational complexity of this operation scales as $O(n^3)$ [199], where n is the total number of DoFs. Furthermore, this inversion operation of the full decoupling matrix propagates and amplifies local modeling errors/disturbances

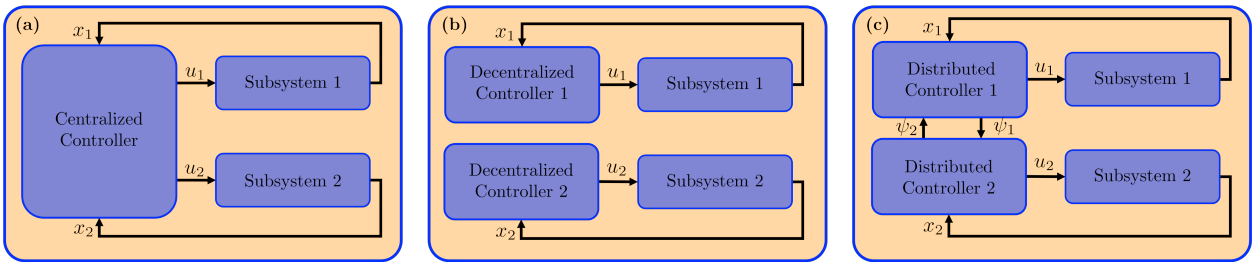


Figure 1.2: General, high-level, illustration of different feedback control architectures available for controlling a *decentralized* structure. (a) centralized control architecture, (b) decentralized control architecture, and (c) distributed control architecture. The figure adopts customary notation used in the controls society, where x denotes the states and u denotes the controls.

across all the joints in the closed-loop dynamics. It is then fair to summarise that this curse of dimensionality is a *theoretical roadblock* impeding the deployment of traditional centralized nonlinear controllers. *This motivates the need for scalable, efficient, and local nonlinear feedback control theory.* Interestingly enough, adopting the decentralized structure, observed in the biological counterparts, allows the formulation of simplified and local control problems. This is due to the fact that decentralized control problems define lower-dimensional subsystems (e.g., each leg of the robot) that use only sensory feedback local to each subsystem. Each subsystem usually has a dedicated processing unit and a local controller to make its own decisions based on the local measurements (see Figure 1.2 (b)). However, the absence of communication between controllers in decentralized strategies may limit the achievable performance. An alternative strategy could be the use of distributed control schemes that have provision for minimal communication amongst the local controllers (see Figure 1.2 (c)).

1.2 Current Limitations

The bedrock of existing approaches that address the control of legged robots, by default, consider the *centrality* structure. In particular, nonlinear controllers developed to stabilize periodic orbits of bipedal robots (with both point foot and flat foot) and quadrupedal robots (with point foot) follow a single layer central structure [14, 16, 20, 25, 28, 30, 47, 55, 62, 63, 91, 95, 101, 109, 110, 124, 130, 149, 171, 174, 175, 177, 192, 194, 195, 211, 212, 213]. Equivalently, controllers developed with reduced-order models of locomotion also make use of multi-level centralized architectures wherein the first layer is reserved for generating trajectories of the Center of Mass (COM) and the corresponding kinematic trajectories for the joints in the robot. The second layer is then comprised of the whole body, inverse kinematic controllers [83, 99, 216] to set the robot in motion. Recently, Quadratic Programming (QP)-based nonlinear controllers have gained increasing popularity in the legged locomotion community due to their ability to encode control objectives subject to physical constraints. For example, they have been used in the context of hierarchical control wherein the problem was casted

as Model Predictive Control (MPC) problem for path planning of reduced-order locomotion models (see, e.g., [42, 56, 68]) or used in the whole-body motion control of full-order models (see, e.g., [76, 81, 127]). While these efforts are impressive in their own right, they do *not* address the theoretical roadblocks highlighted in Section 1.1 and further pose a problem to a specific class of legged robots that are aimed at restoring functionality to amputees and paraplegics.

The necessity for stabilizing distributed nonlinear control algorithms for legged locomotion (practical standpoint): Existing centralized controllers *cannot* be easily transferred to the problem of human-robot cooperative legged locomotion. This restriction is fundamental: powered prosthesis and the human user are, by definition, decentralized subsystems whose interaction with each other is enabled by an external coupling (the same is the case for exoskeletons and human users). Previous work that addresses the problem of cooperative, human-powered prosthesis locomotion has made use of centralized nonlinear architectures [35, 220] neglecting the interaction forces at play between the subsystems or Linear Time-Varying (LTV) controllers that in addition to neglecting the interaction forces, also fail to capture the nonlinearities in a gait cycle. Moreover, existing approaches for decentralized controllers are tailored to the stabilization of equilibrium points for Ordinary Differential Equations (ODEs) but not periodic orbits of hybrid dynamical systems [38, 60, 61, 86, 103, 188, 209]. In addition to high dimensionality, legged locomotion is hybrid in nature and in the interest of capturing the nuances in locomotion of its biological counterparts, we introduce underactuated phases into the hybrid model. *This practical roadblock motivates the need for systematic development of general, stabilizing distributed nonlinear control algorithms that can be realizable on various legged robots.* A short survey highlighting the impediments that prevent the extension of existing decentralized/distributed control methodologies to the problem of legged locomotion provides further motivation and is presented next.

1.3 Current Scope of Decentralized Architecture

The idea of utilizing decentralized/distributed controllers is not new, but through this brief discussion, we intend to show that the benefits of this control architecture has only been considered for large-scale systems (for e.g., traffic, water, power networks, etc.) and has been widely extended only to *dynamically decoupled systems* in robotics. Further, this discussion will also highlight that the vast body of existing work only addresses structures that are presented with sparsity. For the sake of completeness, since decentralized and distributed control have a close relationship, we will include both in the scope of this short survey.

In [205] stability of the closed-loop system was ensured by distributed controllers enabled by multiple communications among subsystems and the use of system-wide control objective functions. Similarly, in [135, 137], decentralized control architecture with continuous and asynchronous feedback for nonlinear systems was introduced. This was then extended

to a Distributed Model Predictive Control (DMPC) for network control systems utilizing Lyapunov-based MPC in [136]. However, they all make a common assumption that each subsystem has access to the dynamical model of the entire system. This limitation was then addressed by allowing access to *only* the local dynamics for each subsystem with the introduction of stability-constrained DMPC [49] and game theory based DMPC [143] but for linear systems. These methods show promise but cannot be directly translated to our problem and would require significant reformulation to consider nonlinearities and interaction forces. Furthermore, it would necessitate the use of reduced-order models, considering the fact that MPC based algorithms *do not* mesh well with high-dimensional full-order models.

In recent years, there have been significant theoretical and numerical advances in the development of efficient methods that consider a *distributed* treatment of QPs. Examples include, but are not limited to, active-set methods [112, 142], Lagrangian decomposition or dual decomposition [133], and distributed multiple shooting methods [181]. These methods, however, depend upon the sparse nature of the equality constraints[125]. They translate very well to the problem of network systems or DMPC [69] as the subsystems are weakly coupled, hence preserving sparsity. On the contrary, legged robots are underactuated, high-dimensional, and inherently unstable hybrid dynamical systems. By nature, any morphological consideration of subsystems will result in these subsystems being strongly coupled.

On the other hand, decentralized/distributed control is not an alien concept in the field of robotics; it has been widely accepted and is an active research thrust. Particularly, in [163], a switched systems-based approach was introduced to concurrently generate goals and trajectories in a decentralized manner for a multi-agent network of ground robots. Similarly, in [33] decentralized feedback controllers were synthesized to navigate a known environment with obstacles for a team of heterogeneous agents. Further, extensive work aimed towards multi-agent coordination and cooperative formation control utilizing decentralized/distributed architectures has been undertaken in [21, 23, 34, 57, 121, 138, 139, 203, 215, 217]. However, these approaches only address weakly coupled systems, and hence its applicability can not be extended to our use case. In addition, legged robots are inherently unstable and hybrid, which further increases the complexity in the design of distributed controllers.

Moreover, state-of-the-art decentralized/distributed techniques that address weakly coupled systems are limited to teams of ground vehicles [22], aerial vehicles [153, 196], and heterogeneous robots [134] *but not* the non-intuitive, decentralized/distributed control of a *single* legged robot with strongly coupled subsystems. However, in the context of legged locomotion, decentralized feedback controllers for powered prosthetic legs have been introduced in [92, 96, 146, 189, 200], but have mostly utilized LTV control methods that are tedious in that they require different control parameters to be tuned at different time periods to account for the nonlinear dynamics in a gait cycle. This results in clinicians spending a significant amount of time and effort in tuning control parameters (further necessitates significant training time to familiarize clinicians with control parameters) [189] for each individual patient. This is further elaborated in good detail in Chapter. 4.

With the help of Section 1.1 and Section 1.2, we have highlighted that *key roadblocks* exist theoretically: lack of *scalable, efficient,* control algorithms and practically: lack of nonlinear control algorithms for *naturally* decentralized systems, due to the consideration of centralized architecture in high-dimensional legged robots. In addition to these motivating reasons for the development of distributed controllers, we have also highlighted that the biological counterparts take advantage of the decentralized structure. Further, in Section 1.2, we have discussed that the state-of-the-art in legged locomotion control is fixated on the notion of centralization by presuming the entire problem as a composition of "one" structure. These centralized methodologies are not compliant with the distributed information sharing and decision-making structure we are interested in, thus indicating that these key roadblocks still constitute an open problem. Moreover, in this section, we have highlighted that the existing scientific literature on decentralized/distributed control is primarily focused on large-scale systems and multi-agent, weakly coupled, Unmanned Ground Vehicles (UGV) and Unmanned Aerial Vehicles (UAV) systems (both outside the scope of legged locomotion). Collectively, this indicates the existence of a *fundamental gap in knowledge* of distributed control algorithms for motion control of an inherently unstable, underactuated, and complex hybrid dynamical, single-agent legged robot. The next section identifies the specific areas of development that will be addressed by this thesis.

1.4 Research Goals, Objectives, and Contributions

The ***overarching goal*** of this thesis is to establish a formal foundation based on decomposition, hybrid systems theory, nonlinear systems theory, and optimization to develop a class of general, scalable, and efficient distributed feedback control algorithms for the cooperative control of legged robots. This thesis considers both visibly decentralized structures and "single" legged robots that by nature possess a centralized structure but are envisioned as a composition of "multiple" distributed structures for control synthesis. Since this thesis considers generality, subsystems can appear to be either symmetric or asymmetric. Additionally, as noted earlier in Section 1.3, this unconventional approach imposes the need for non-sparse formulations and the consideration of strong coupling amongst subsystems. One of this thesis's primary goals is to address these implementational challenges explicitly. In order to do so, the thesis proposes the following ***specific goals***:

(**1**) Creation of general and scalable distributed nonlinear feedback control algorithms for full-order hybrid dynamical models of legged robots based on decomposition and geometric control.

(**2**) Establishing concrete and provable methods to simplify the search for optimal and stabilizing local controllers.

(**3**) Creation of general and scalable network of distributed QP-based control algorithms by decomposing a centralized QP for the real-time control of legged robots.

(**4**) Establishing analytical tools for tuning distributed QPs' parameters such that the solution converges to that of the centralized QP.

(**5**) Transferring the theoretical contributions and the resultant distributed control algorithms into practice through extensive experiments on a physical quadrupedal robot, A1 (see Figure 1.1 (a)).

These goals are structured into ***specific objectives*** with ***key contributions*** through the help of the following research aims (see Figure 1.3):

(R1) Research Aim 1: Scalable Nonlinear Distributed Control Algorithms based on Decomposition and Hybrid Zero Dynamics: The objective of this research aim is to present a systematic and scalable approach, based on decomposition, Hybrid Zero Dynamics (HZD), and optimization, to develop nonlinear distributed controllers that exponentially stabilize collaborative human-robot locomotion with *asymmetric* models. Specifically, (1) We propose a decomposition approach for the synthesis of distributed controllers. The integration of decomposition and optimization allows for a significant reduction of the parametric space as we only need to modify and optimize the already-existing centralized HZD controller rather than searching for local controllers from scratch. This leads to a small-scale optimization problem. (2) We study the properties of the Jacobian of the Poincaré map of the centralized and distributed controllers and show that there exists an upper bound on the norm of the difference of Jacobian matrices that significantly simplifies the search for optimal local control parameters. (3) We numerically validate the analytical foundation by designing distributed controllers for multi-domain and collaborative human-robot locomotion with a powered knee-ankle prosthetic leg and nontrivial feet, having a total of 19 DoFs (see **Task R1** in Figure 1.3). This research aim comprises specific goals (1) and (2).

(R2) Research Aim 2: Scalable Distributed Quadratic Programming Controllers based on Decomposition: The objective of this research aim is to present a formal foundation to systematically decompose QP-based and centralized nonlinear controllers into a network of lower-dimensional and local QPs for periodic behaviors in hybrid systems, with application to robotic locomotion. Here, we aim to develop a network of distributed QPs with an inherent feedback structure that preserves the steady-state solution of the centralized QP on periodic orbits. Through this research aim, we aim to theoretically establish a set of sufficient conditions under which the solutions of the local QPs asymptotically converge to the steady-state solution of the centralized QP (see **Task R2** in Figure 1.3). We aim to numerically investigate the effectiveness of the synthesized controller by performing rigorous numerical simulations in an existing physics-based simulator. This research aim comprises specific goals (3) and (4).

(R3) Research Aim 3: Reduction to Practice of the Proposed Distributed Control Algorithms on a Quadrupedal Platform: The objective of this research aim is to apply the proposed synthesis approach (**Task R2**) utilizing local QPs to experimentally realize robust locomotion on the 18-DOF, advanced quadrupedal robot, A1. In particular, this

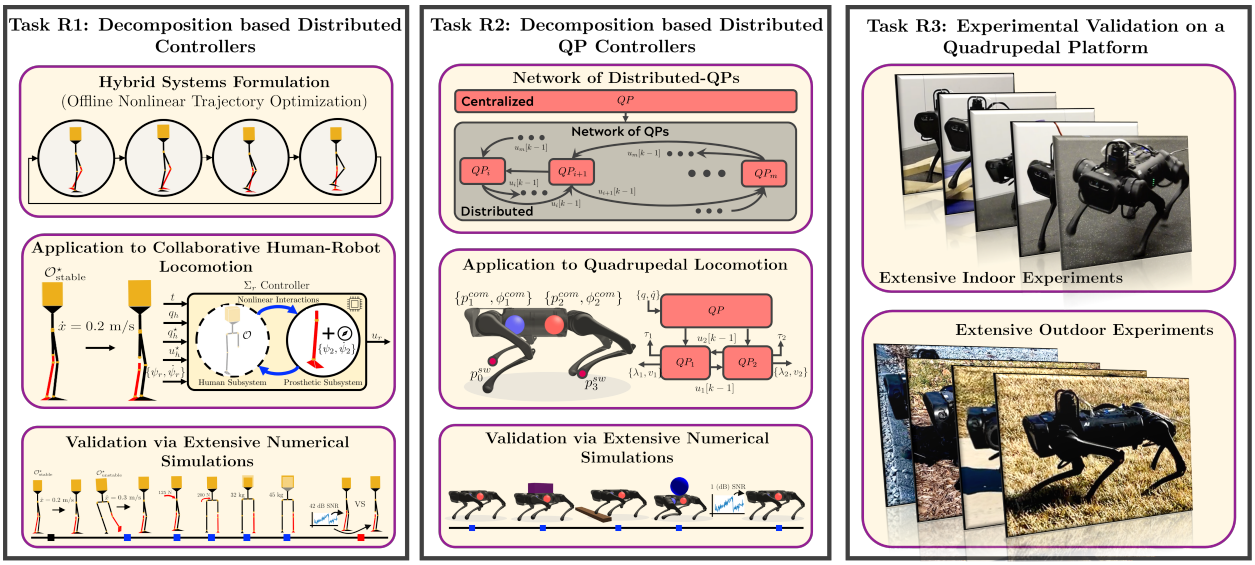


Figure 1.3: Overview of the research aims R1-R3 addressed by this thesis. This main objective of this thesis is to develop general, scalable, and distributed control algorithms and the subsequent extension to the various applications of legged locomotion: from bipeds to quadrupeds.

research aim seeks to verify the theoretical contributions via an extensive set of experimental studies (see **Task R3** in Figure 1.3) for quadrupedal locomotion at varying speeds and in the presence of different uncertainties in both indoor and outdoor settings. This research aim comprises of the specific goal (5).

Structure of the thesis: Two *different* distributed controllers arise from systematically addressing the research objectives (**R1-R2**). Then, in order to preserve accessibility, this thesis is structured into two self-contained parts. Part I will be composed of Chapter 2, Chapter 3, and Chapter 4 that will address the research objective **R1**. Part II comprises of Chapter 5 and Chapter 6 and will discuss in detail the proposed methodology to address the research objectives **R2** and **R3**.

With the help of part I and part II, this thesis makes the following ***specific achievements***:

(**1**) For **R1**, it is shown that the proposed approach can effectively synthesize local controllers for the comprehensive model of human-robot locomotion. In particular, we reduce the number of local controller parameters to be tuned and optimized to almost 5.2% of that in [12].

(**2**) For **R2**, it is shown that the proposed local QPs have less sensitivity to noise propagation compared to the original centralized QP. It is further shown that the total solve time of the local QPs is reduced by 3.82 times in comparison to the centralized QPs.

(**3**) For **R3**, an extensive array of experiments both in the indoor and outdoor settings demonstrate stable locomotion for varying speeds and in the presence of unmodelled

uncertainties using the distributed QP controllers developed in **R2**. Additionally, to the best of the author's knowledge, this thesis reports the *first* successful hardware implementation of a model-based *distributed* controller on a quadrupedal platform.

1.5 Significance and Importance

The *theoretical significance* of the corresponding distributed control algorithms developed in this thesis is multifold and enumerated in the following: (1) they enable a high degree of cooperation amongst the subsystems by accounting for the *strong nonlinear interactions* at play in complex and high dimensional legged robots; (2) they provably stabilize *full-order*, decentralized hybrid models as opposed to reduced-order abstractions of legged robots; (3) they consider *underactuation* and *non* flat-footed walking, catering to the broader facets of legged locomotion; and (4) they form a powerful *general* tool to construct parameterized, nonlinear distributed feedback controllers for problems beyond the scope of legged locomotion.

Furthermore, these distributed control algorithms are *practically significant*, in that: (1) they decompose the high-dimensional problem into smaller sub-problems that are liberated from the *curse of dimensionality* thus disabling any computational bottlenecks; (2) they have better fault-tolerance and are presented with inherent robustness that is facilitated by the division among the subsystems—disturbances originated in one subsystem *cannot* propagate to the other subsystem.

1.6 State-of-the-Art in Legged Locomotion: Modeling and Control

In this section, a brief survey of the state-of-the-art approaches in dynamic modeling, Trajectory Optimization (TO), and feedback control of legged locomotion is provided. The majority of the methods described in this section are independent of legged locomotion and can be applied to a wide range of dynamic problems. However, we will refrain from generalization and restrict the discussion to the realm of legged locomotion. Additionally, efforts have been made to frame this discussion in a platform-agnostic manner (i.e., applicable to bipeds, quadrupeds, etc.) to the best extent.

1.6.1 Dynamic Modeling

Legged robots fall under the class of systems identified as "floating-base" systems whose underlying dynamics can be encapsulated by either reduced-order models or full-order models.

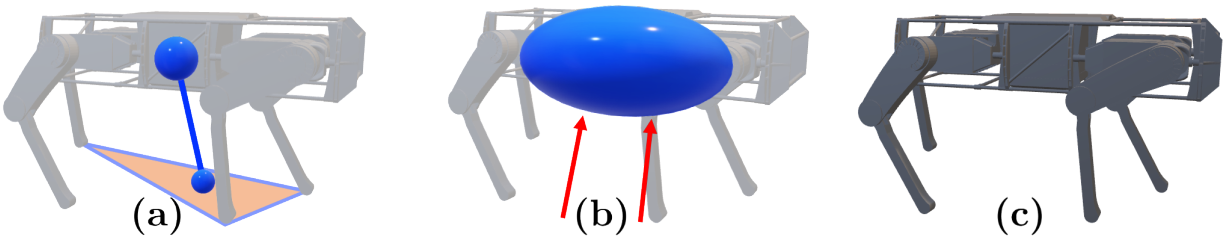


Figure 1.4: Illustrations of three different approaches toward modeling legged robots with the help of an advanced quadrupedal robot, Vision 60: (a) the entire robot abstracted as a LIP model, (b) centroidal model: the entire robot abstracted as an ellipsoid, and (c) full-order methodology that accounts for all the rigid links in the dynamic model.

The choice of one over the other is pivoted around multiple factors and is not straightforward. In what follows below, we aim to highlight the advantages and disadvantages of each method to paint a complete picture and better aid the user's choice.

Reduced-order Models

This set of models is primarily preferred because of its ability to encapsulate a complex model in a simple structure. Besides the fact that these models result in simple control strategies, they are widely accepted because they bear resemblances to models employed by animals and humans, as evident from biomechanical studies [40, 43, 64, 80, 117]. This observation was exploited by the majority of approaches in DRC through the implementation of template-based models. In what follows below, we will discuss two such models in detail.

A) Linear Inverted Pendulum (LIP)

The most elementary model in the reduced-order family is the Linear Inverted Pendulum (LIP) model that abstracts the entire robot as a point mass that is balanced on a massless inverted pendulum [97, 115]. This assumption implies that the robot's joint velocities produce negligible momentum and is rational in one of two cases: (1) when the robot's limb mass is negligible compared to the entire mass of the robot, e.g., MIT Cheetah 3 [42], or, Cassie [4], (2) under the assumption that the desired velocity of the robot is considerably slow. Further, LIP stems from the Inverted Pendulum dynamics, which is originally nonlinear in nature. To linearize these dynamics, the vertical component of the Center of Mass (CoM) is constrained to a horizontal plane. To better explain this, let us assume that the Cartesian coordinates of the CoM in the inertial frame are given by $r \in \mathbb{R}^3$ where $r \coloneqq \text{col}(r_x, r_y, r_z)$ and "col" is the column operator. Here r_x encodes the position along the x-axis (r_y and r_z follow accordingly). Then according to the constraint, $\ddot{r}_z \equiv 0$. With all the assumptions and

constraints now in place, the LIP model can be finally given by

$$\begin{bmatrix} \ddot{r}_x \\ \ddot{r}_y \end{bmatrix} = \frac{g}{r_z} \begin{bmatrix} r_x - u_x^{\text{cop}} \\ r_y - u_y^{\text{cop}} \end{bmatrix}, \tag{1.1}$$

where, g is the gravitational constant and $u^{\text{cop}} \in \mathbb{R}^2$ indexes the Cartesian coordinates of the Center of Pressure (CoP). The LIP framework has sustained the test of time and is considered an essential tool till date as it can be efficiently solved while respecting the real-time constraints and is evident from its use by different approaches [20, 72, 98, 170]. However, the assumptions made in formulating the LIP model can become restrictive when complex *dynamic* motions are considered, that may entail flight-phases, body re-orientation, or overcoming irregular obstacles necessitating vertical jumps (i.e., $\ddot{r}_z \equiv 0$ has to be violated). Another necessary and important constraint imposed by the LIP models is that the CoP (an accumulation of all the vertical components exerted by each individual foot) must lie in the convex hull that is constructed by the feet in contact with the ground. Condensing the 3-dimensional forces at the foot into the CoP results in loss of information corresponding to vertical forces exerted at each foot. This becomes restrictive in that it does not allow us to impose friction cone constraints on each individual foot, thus acting as a roadblock towards modeling many important characteristics exhibited by legged robots.

To navigate these limitations, few extensions have been proposed to further the LIP model. One such model is the Spring Loaded Inverted Pendulum (SLIP) model that abstracts the inverted pendulum model as a point mass balancing on a massless spring. The SLIP model allows the consideration of flight phases and can be used to model both walking and running [84]. Furthermore, it has shown to approximate the CoM trajectories and Ground Reaction Forces (GRFs) of a wide range of animals/humans [44] strengthening its appeal to similarly approximate robot dynamics. Other examples of reduced-order dynamics for legged locomotion include the Assymetric Spring Loaded Inverted Pendulum (ASLIP) [167] and the Lateral Leg Spring (LLS) model [168].

B) Centroidal Dynamics

To model a N link, rigid, floating-based body connected by m actuated joints (accounts for m DoFs), we first attach a "fictitious" link (see Figure 1.5 (b)) that tethers the base of the robot to the "world" frame or the "inertial" frame. This contributes an additional 6 DoFs that enables us to precisely locate the rigid-body's linear position, $p \in \mathbb{R}^3$ and the angular orientation, $\phi \in \text{SO}(3)$, where SO(3) is the special orthogonal group, with respect to the global frame. It is to be noted that one can either use the Euler angles, $\phi \in \mathbb{R}^3$ or the quaternions to obtain the global orientation. In this discussion and through out this thesis, we will choose the former due to it's ease of availability although it is prone to singularities in some instances, e.g., improper initialization. The total DoFs of such a floating-base model can then be given by, $n := m+6$. To make further progress in this discussion and throughout this thesis, we will adopt the standard notation used in the robotics community. With that in place, $q := \text{col}(p, \phi, q_1, ..., q_m)$ and $q \in \mathbb{R}^n$ denotes the generalized coordinates that will

capture the evolution of all DoFs and their corresponding velocities are promptly encoded in $\dot{q} \in \mathbb{R}^n$. Together, q and $\dot{q}$ combine to form the state vector $x \coloneqq \text{col}(q, \dot{q})$, where $x \in \mathbb{R}^{2n}$.

With the standard notation in place, we can now proceed to understand the centroidal dynamic modeling framework. The Centroidal dynamic modeling method acts as an intermediate tool between the reduced-order models and the full-order models. This method exploits the fundamental property of the CoM that is defined as a virtual point where all the mass of the robot is assumed to be lumped without changing the behavior of the entire system. Consequently, one can assume that the GRFs are directly acting on CoM as an applied force rather than acting on all bodies as a distributed vector field. To compute the centroidal dynamics consider

$$h_i = \begin{bmatrix} l_i \\ k_i \end{bmatrix} = I_i \, v_i, \tag{1.2}$$

where $l_i \in \mathbb{R}^3$ is the linear momentum, $k_i \in \mathbb{R}^3$ is the angular momentum, and I_i is the spatial inertial for link i. Equation (1.2) provides the spatial momentum of each link with respect to the centroidal frame. The centroidal momentum of the entire robot with respect to the CoM frame, h_G is given as the sum of its body spatial momenta

$$h_G = \sum_{i=1}^{n} {}^{i}X_G^{\top} \, h_i = \underbrace{\sum_{i=1}^{n} {}^{i}X_G^{\top} \, A_i}_{A_G(q)} \, \dot{q} = A_G \, \dot{q}, \tag{1.3}$$

where $A_G \in \mathbb{R}^{6\times n}$ is the centroidal momentum matrix as defined in [162] that directly maps the joint space velocities, $\dot{q}$, into the robots CoM spatial momentum, $h_G \in \mathbb{R}^6$. Further, here $A_i \coloneqq M_i \, J_i \in \mathbb{R}^{6\times n}$ is the individual link's momentum matrix expressed in the link's frame. ${}^{i}X_G \in \mathbb{R}^{6\times 6}$ is the projection matrix from the centroidal to link coordinates and finally J_i is the Jacobian matrix that maps the joint space velocities into task space velocities. Finally, the centroidal dynamics can then be obtained by differentiating Equation 1.3 as follows:

$$\dot{h}_G = A_G \, \ddot{q} + \dot{A}_G \, \dot{q}. \tag{1.4}$$

Full-order Models

Full-order dynamic models, on the contrary, shy away from simple abstractions and model the robot in its entirety, i.e., it considers all the joints and links of the robot, in order to provide a *reasonably* complete description and in an attempt to capture a *reasonably* accurate representation of the robot. This approach results in nonlinearities that arise from the inertia matrices, end effector Jacobian matrices, Coriolis and gravity terms with respect to the joint configurations, etc. While motor damping and motor viscous friction introduce parametric uncertainties, the approach considers this negligible and proceeds with a *close to accurate approximation* of the model. To proceed further, we make use of the Lagrangian, a

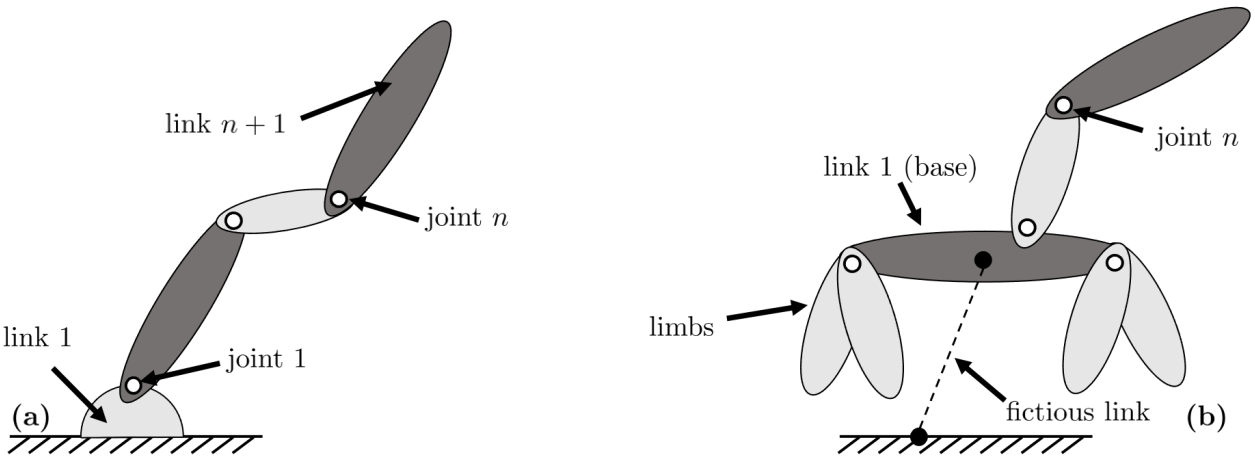


Figure 1.5: Illustrations of two predominant rigid-body abstractions: (a) fixed-base serial, tree structure (b) floating-base serial, tree structure.

real-valued function that is defined as follows:

$$\mathcal{L}(q, \dot{q}) \coloneqq K(q, \dot{q}) - U(q), \tag{1.5}$$

where K denotes the total kinetic energy of the system and U denotes the total potential energy of the system. The Lagrange's equation is then given by

$$\frac{\mathrm{d}}{\mathrm{dt}} \frac{\partial \mathcal{L}}{\partial \dot{q}} - \frac{\partial \mathcal{L}}{\partial q} = \Gamma, \tag{1.6}$$

where Γ is the vector of the generalized forces λ and torques u. Further, the kinetic energy is quadratic, i.e., $K(q, \dot{q}) = \frac{1}{2}\, \dot{q}^\top\, D(q)\, \dot{q}$, where $D(q) \in \mathbb{R}^{n \times n}$ is the positive-definite, mass-inertia matrix. Equation (1.6) can then be expressed in the following second-order form

$$D(q)\, \ddot{q} + C(q, \dot{q})\, \dot{q} + G(q) = B\, u + J^\top(q)\, \lambda, \tag{1.7}$$

where $C(q, \dot{q})\, \dot{q} + G(q) \in \mathbb{R}^n$ represents the Coriolis, centrifugal, and gravitational terms, $B \in \mathbb{R}^{n \times m}$ denotes the input distribution matrix, $J(q)$ represents the contact Jacobian matrix and, $\Gamma \coloneqq B\, u + J^\top\, \lambda$ denotes the sum of the generalized forces and torques acting on the rigid body (see [213, Appendix B] for the complete derivation). From this complete representation, one specific observation of interest that aligns with the thrust of this thesis can be made. In particular, $D(q)$, $C(q, \dot{q})\, \dot{q}$ and $G(q)$ scale up with the increase in number of DoFs. This leads to the computational bottleneck that discourages the use of real-time Optimal Control Problems (OCPs) that simultaneously looks for feasible trajectories, q and the corresponding suitable control action, u. Furthermore, irrespective of the modeling choice, for floating-base systems, as discussed earlier, the first six components of q are comprised of unactuated DoFs. For cases, where the robot is in "flight-phase" or the consideration of "heel-rise" and "toe-lift" phases in bipedal robots with non-trivial feet, the system is rendered underactuated. We

will see in Subsection 1.6.3 that there are very few approaches that addresses underactuation in legged locomotion. In this thesis, we will exploit the full-order dynamic model and its properties to design distributed controllers for the general class of legged robots, thus directly addressing the computational roadblock. Moreover, the resultant distributed controllers also address the underactuated phases in legged locomotion. The application considered in Chapter 4 has a naturally occurring underactuated phase that helps validate this claim.

Hybrid Systems Theory

In Subsection 1.6.1, we have demonstrated multiple methods to capture the continuous-time dynamics. However, legged locomotion is an interplay between continuous and discrete-time domains. It is, therefore, crucial to simultaneously capture both the continuous and discrete-time dynamics to closely model the behavior of legged robots within a stride. Towards that end, we note that hybrid systems theory exhibits characteristics of both continuous and discrete-time dynamical systems [37, 87, 103, 218] and has become a very common language to model different processes including power systems [108], and mechanical systems subjected to impacts [14, 15, 16, 25, 28, 30, 47, 48, 54, 55, 58, 62, 63, 90, 91, 95, 101, 111, 124, 145, 157, 167, 169, 171, 176, 177, 179, 194, 195, 197, 202, 213]. Hybrid models of locomotion involve ODEs describing the continuous-time phases (see Equation (1.7)) and discrete maps that capture the impact (establishing contact with the environment) or take-off (breaking contact with the environment) events [100, 111]. For the time being, we assume that we have the discrete maps in hand. This assumption is realized in Section 2.4 of Chapter 2 where there is scope to provide sufficient context that will significantly help in understanding the formulation of discrete-time dynamics.

1.6.2 Trajectory Optimization

With the dynamics in hand, we can now start synthesizing controllers either via formulating an optimal control problem (OCP) or by first generating a feasible reference trajectory and then commanding the controller to track the generated trajectory. This choice is primarily dictated by the type of dynamics one considers. In particular, reduced-order models can generate tractable OCPs that can be solved in real-time, albeit sacrificing the dynamic richness. On the other, high-dimensionality in full-order models makes it difficult to formulate a real-time control problem and necessitates the generation of full state reference trajectories. In this section, our emphasis will be on full-state TO which entails solving a NLP given that the goal for Part I constitutes the use of full-order, hybrid dynamical models that result in nonlinear constraints and/or nonlinear objective functions.

The general TO problem then deals with finding the open-loop control $u(t)$ over a finite time interval $t \in [t_0, t_f]$ such that for a given initial state x_0, the state trajectory reaches the

desired final state x_f. More formally, the TO problem can be formulated as follows:

$$\begin{aligned}
\min_{u} \quad & \mathcal{J} = l(x(t_f)) + \int_{t_0}^{t_f} g(x(t), u(t))\,\mathrm{dt} && \text{(Cost Function)} \\
\text{s.t.} \quad & \dot{x} = f(x(t), u(t)) && \text{(Dynamics)} \\
& x(t_0) = x_0 && \text{(Initial Condition)} \\
& c_{\min} \leq c(x(t), u(t)) \leq c_{\max} && \text{(Path Constraints)} \\
& b_{\min} \leq b(x(t_f)) \leq b_{\max} && \text{(Boundary Constraints).}
\end{aligned} \tag{1.8}$$

Note that Equation (1.8) is a continuous-time TO problem and has to be converted into a form that is digestible (discretization) by existing mathematical programs. This process is labeled as the "transcription" process after which, the problem can be interfaced with any commercially available solvers, e.g., IPOPT [208], SNOPT [85], or FMINCON[150]. In cases where the constraints are linear and the objective function is either linear or quadratic, transcription methods result in either a Linear Program (LP) or a QP based on the nature of the objective function. However, as pointed out earlier our constraints and/or the objective function are nonlinear in nature, therefore for our problem, transcription methods will lead to a NLP. Typically LPs, QPs and NLPs can all be structured into a similar form as follows:

$$\begin{aligned}
\min_{z} \quad & f(z) && \text{(Cost Function)} \\
\text{s.t.} \quad & c(z) = 0 && \text{(Equality Constraints)} \\
& g(z) \leq 0 && \text{(Inequality Constraints)} \\
& z_{\text{lb}} \leq z \leq z_{\text{ub}} && \text{(Bounds).}
\end{aligned} \tag{1.9}$$

An initial condition (i.e., guess) z_0 is still needed, and a proper initialization could potentially reduce the optimization time. However, there is no concrete method to guide the choice of the initial condition, and it is usually a heuristic. This transition from Equation (1.8) to Equation (1.9) can be facilitated by multiple ways but for the general class of problems in robotics, the preferred method is direct collocation method. We shall make the case for this in the context of our specific problem in the following discussion.

Direct vs Indirect Methods

The broad class of transcription methods can be classified as a direct or an indirect method. Direct methods first discretize the problem and then optimize for a solution. On the contrary, indirect methods first find optimality conditions and then discretizes these conditions to feed to a solver. The advantage of indirect methods is that they can provide accurate solutions in comparison to direct methods by virtue of the necessary and sufficient optimality conditions. However, at the same time, these conditions are to be constructed by the user and usually require closed-form analytical expressions that may or may not be available [39].

In our case, closed-form expressions are complex and involve time-consuming computations. Furthermore, indirect methods have shorter regions of convergence in comparison, which necessitates better initialization [39]. These factors are discouraging, and thus we do not pursue indirect methods for finding feasible reference trajectories.

Shooting vs Simultaneous (Collocation) Methods

Direct methods are classified as either shooting methods or simultaneous methods (alternatively known as collocation methods). Shooting methods and collocation methods both solve the TO problem by transforming it into a NLP. Given $x(0)$ and $u(t)$, shooting methods approximate the trajectory using forward simulations. As a result, the decision variables are few, resulting in a small and dense optimization problem. The key difficulty with this method is that there is no provision for including path constraints as the intermediate state variables $x(t)$ are not present in the decision variables [39]. This is a major drawback in the context of legged locomotion, and moreover, we are specifically interested in $x(t)$. On the other hand, collocation methods account for the state variables as decision variables and allow for path constraints to be included. This makes a strong case for our use of collocation methods and *not* shooting methods. Now that the reasoning behind the choice of direct collocation methods is clear, we note that the collocation methods are further classified into trapezoidal, Hermite-Simpson, and orthogonal methods. At the same time, we also note that commercial and/or open-source packages are available, which will readily transcribe a continuous-time problem with path constraints and boundary constraints into a discrete problem while also accounting for the non-smooth dynamics that arise in a hybrid dynamical system. One such package that uses the direct collocation method is Fast Robotic Optimization and Simulation Toolkit (FROST) [106] and is specifically written for hybrid dynamic legged locomotion problems in that it allows the user to provide their multi-domain gait graphs. However, this is by no means a free lunch, a well-posed TO problem is still necessary for FROST to converge to a solution. We show this in detail in Chapter. 4.

Hermite-Simpson Collocation Method. We will briefly go over the specific details of transcription to have a general understanding of the process with the help of the Hermite-Simpson method. We begin by discretizing time over $N_g \geq 1$ intervals, where N_g is the user specified value known as *grid points* (or alternatively *knot points*). The state and the control trajectories are then discretized by representing the continuous-time trajectories, $x(t)$ and $u(t)$ by their values at each grid point as follows:

$$\begin{aligned} t &\rightarrow [t^0, t^1, \cdots, t^{N_g}], \\ x(t) &\rightarrow [x^0, x^1, \cdots, x^{N_g}], \\ u(t) &\rightarrow [u^0, u^1, \cdots, u^{N_g}]. \end{aligned} \tag{1.10}$$

Our decision variables are then x^k and u^k. As a general rule of thumb, the denser the grid points, the smoother the corresponding solution is. However, it is recommended to start

with a coarse grid to obtain an approximate solution and proceed to gradually increase the number of grid points and optimize further for a smoother solution. The dynamic equations are then imposed as constraints on the collocation points, t_c^k, defined as the midpoints of each discretized interval, $t_c^k := \frac{t^k + t^{k+1}}{2}$. The central idea being that the change in state between two collocation points is equal to the integral of system dynamics. This integral is then approximated by the Simpson quadrature and applied at each collocation point. The state and its corresponding derivative at the collocation points are given as

$$\begin{aligned} x(t_c^k) &= \frac{1}{2}\left(x(t^k) + x(t^{k+1})\right) + \frac{h}{8}\left(\dot{x}(t^k) - \dot{x}(t^{k+1})\right), \\ \dot{x}(t_c^k) &= \frac{-3}{2h}\left(x(t^k) - x(t^{k+1})\right) - \frac{1}{4}\left(\dot{x}(t^k) + \dot{x}(t^{k+1})\right), \qquad \text{(Dynamics)}, \end{aligned} \tag{1.11}$$

where $h := t^{k+1} - t^k$. Likewise the cost function is approximated by the Simpson-quadrature as follows:

$$\mathcal{J}(k) = l(x(t_f)) + \sum_{k=0}^{N-1} \frac{h}{6}\left(g^k + 4g^{k+\frac{1}{2}} + g^{k+1}\right), \quad \text{(Cost Function)}. \tag{1.12}$$

The path constraints and bounds are similarly discretized as in Equation (1.10) and pieced together with Equation (1.11) and Equation (1.12), thus attaining the structure of Equation (1.9). After this NLP is solved, we are left with the state and control trajectories at each collocation point. The continuous trajectory for controls $u(t)$ is then constructed by using Simpson quadrature to approximate each segment between the knot points. Similarly to construct the continuous state trajectory, $x(t)$, cubic Hermite spline is used.

FROST on the other hand modifies this general Hermite-Simpson collocation formulation to allow for the expression of system dynamics in an implicit form at each node facilitating the inclusion of further optimization parameters. In addition to, $t^k, x^k := (q^k, \dot{q}^k)$ and u^k (decision variables in the general formulation), FROST further introduces $\ddot{q}^k, \lambda^k, \alpha^k$ and η^k, where λ encodes the GRFs, α are the parameters of Virtual Constraints (VCs) and η are the holonomic constraints (see [107] for further details of this modification). The introduction of α in the decision variables in FROST, now allows the calculation of closed-loop control, $u(t, x)$. We can either choose to utilize the resultant α's or synthesize our own to track the states x. This synthesis of VCs in the context of centralized control is discussed in Chapter. 2.

1.6.3 Existing Control Approaches

From the first principles point of view, we are now at the final juncture. We discussed the dynamics and considered, how the choice of the model directs our next steps: undertake an *offline* TO in order to obtain a reference trajectory or formulate an *online* OCP for trajectory

generation in real-time. The previous section addressed the TO problem, after which we are left to synthesize a controller to track the resultant trajectories (final juncture). This problem will be addressed in this section along with the OCPs for online trajectory generation. There are a wide range of control methodologies that tackle the problem of trajectory tracking and similarly diverse methods that address OCPs. However, in the interest of brevity, we will narrow the scope of this discussion to control approaches/methods that are of relevance to this thesis.

Model Predictive Control (MPC)

MPC is an age-old technique whose application has been restricted to primarily large-scale and slow dynamic processes such as chemical plants, power plants, etc [45]. However, with the recent advancements in computation speed and the development of efficient algorithms [210], MPC-based approaches have also found applications in the control of safety-critical and dynamic legged robots. Specifically, the ability of the MPC approaches to account for the future actions while searching for the current optimal control action is very attractive and one of the reasons for the method to gain traction over the recent times. As highlighted earlier, MPC forms the first layer of the hierarchical control schemes and iteratively plans for the CoM motion and CoP given by either Equation (1.1) for the LIP model or simply the CoM position and orientation given by Equation (1.4) for the centroidal model. The problem is further subjected to constraints such that it respects the CoP conditions and accounts for the feasibility of GRFs. It usually takes the form of QP that is in a similar structure as in Equation (1.9). The appeal of MPC is further enhanced by the use of LIP or centroidal models in that the resulting simple structures can be transferred *online* to undertake real-time motion planning of bipeds or quadrupeds alike. It is then left to the robustness of the low-level controllers to account for the deficiencies of the reduced-order models. Alternative attempts have been made to extend the MPC approach to full-order models in [73] but without any control preview. Nonetheless, it has been successfully employed in [72, 98, 116, 170] for bipedal LIP models. However, the extension of LIP-based MPCs to quadrupedal robots is not straightforward as it lacks the provision to consider the feasibility of individual GRFs and rather approximates the cumulative GRF. This problem has been addressed in [66, 67, 207] by solving for optimal GRFs at the contacting leg ends for every time sample. This, however, increases the number of constraints and questions the computation feasibility to solve the OCP at every time sample (for e.g., 200 HZ). To address this problem then, event-based MPCs has been introduced in [20, 132] that rather than solving for an optimal solution at every time sample, computes for a solution at the beginning of every continuous-time domain triggered by a discrete event (i.e., the hybrid dynamics formulation). There have also been other alternative flavours of MPC, such as nonlinear MPC [159], policy regularized MPC [41] and QP-based WBC [74, 128].

Relevant Control Approaches for Trajectory Tracking

Table 1.1: A Non-Extensive List of Relevant Legged Locomotion Control Literature

Control Type	Reference(s)
Passivity Control	[58], [59], [79], [82], [88], [89], [123], [129], [151], [152], [193], [194], [195], [201]
Transverse Linearization	[78], [145], [185], [186], [187]
Hybrid Reduction	[26], [27], [28], [31], [90], [91], [93], [94]

In the context of trajectory tracking, different methods tailored to address the specific aspects of legged locomotion have been developed. For instance, passivity control approaches exploit passive dynamics that revolve around the idea that with proper design, minimal corrective control is required for executing walking motions. However, for practical applications modern robots have to demonstrate capabilities far beyond walking and exhibit robustness beyond minimal weight uncertainties. In addition, successful control approaches should also account for the hybrid nature encountered in all practical legged locomotion models, as highlighted in Subsection 1.6.1. State-of-the-art control design methods that address the hybrid nature of legged locomotion models are developed based on hybrid reduction [29, 31, 32, 90], controlled symmetries [194], geometric reduction [26, 29, 31, 32, 191], transverse linearization [145, 187], and HZD [30, 157, 212, 213]. Amongst these approaches, controlled symmetries, hybrid reduction and geometric reduction all assume full actuation in the model and make use of energy shaping. However, as seen in Subsection 1.6.1, when rich hybrid models are defined to capture the nuances of legged locomotion, underactuated phases often arise. This gap is bridged by transverse linearization and HZD-based approaches that explicitly address underactuation. However, transverse linearization techniques have not been translated extensively to hardware implementations. The HZD approach, on the other hand, in addition to considering underactuation is robust to the modeling uncertainties brought about by the full-order nonlinear models and has been successfully employed both in simulation and experimentation for high-dimensional legged robots. These features render the HZD approach attractive and form the focus of our immediate discussion. We remark that the HZD approach can be employed to either track the trajectories generated by offline NLP or to track the CoM trajectories generated by the real-time MPCs [20]. For the convenience of the reader, a non-exhaustive list of references that makes use of control approaches other than the HZD formulation is given in Table 1.1.

Hybrid Zero Dynamics (HZD)

The HZD approach prescribes a set of kinematic constraints that are alternatively called VCs to coordinate the motion of the robot's links in a continuous-time domain. Typi-

cally, these VCs are enforced by the action of a feedback controller (e.g., Input-Ouput(I-O) linearization [114]). The central idea behind HZD revolves around the effort to capture the stability properties of the periodic gaits of the high-dimensional system in a low-dimensional abstraction. This is facilitated when the VCs are satisfied during both the continuous- and discrete-time dynamics of the hybrid models of locomotion. In this case, the zero dynamics manifolds corresponding to the VCs become invariant under the flow of the hybrid system, which is referred to as the *hybrid invariance*. The challenge is to find VCs and the corresponding parameters that respect the hybrid invariance condition. Despite this difficulty, the HZD approach has gained success in controlling high-dimensional monopods (both 2D and 3D) [15, 167, 176], bipedal robots (both 2D and 3D) [14, 17, 24, 25, 30, 54, 55, 101, 104, 107, 148, 166, 174, 178, 197, 198, 214], prosthesis [92, 96, 219], exoskeletons [102, 105] and recently quadrupeds [18, 20, 50, 140], and multi-agent cooperative quadrupedal locomotion [19]. The extension of HZD to decentralized control has been seen in [11] for exoskeletons, however it does *not* consider the interaction forces in its formulation. Furthermore, decentralized HZD for transfemoral prosthesis with trivial point feet was considered in [12, 13]. Here, the decentralized controllers were synthesized by setting up a large-scale optimization problem based on an iterative sequence of bilinear matrix inequalities (BMIs). The BMI algorithm develops local controllers from scratch by optimizing for a significant amount of control parameters. This may result in a computational bottleneck for multi-domain locomotion models with nontrivial feet and high DoFs. The approach developed in this thesis aims to *significantly reduce* the number of optimizable control parameters (to be precise, 94.8% reduction is observed [118]).

1.7 Relevant Publications & Attribution

Chapters 3 and 4 comprise the material published in the IEEE Control System Letters (L-CSS) [118] which was also presented at the 2021 American Control Conference (ACC).

[**J2**] **V. R. Kamidi**, J. C. Horn, R. D. Gregg, and K. Akbari Hamed, "Distributed Controllers for Human-Robot Locomotion: A Scalable Approach Based on Decomposition and Hybrid Zero Dynamics," *IEEE Control Systems Letters*, 5(6):1976–1981, Dec. 2021.

[**C1**] A. Pandala*, **V. R. Kamidi***, and K. Akbari Hamed, "Decentralized Control Schemes for Stable Quadrupedal Locomotion: A Decomposition Approach from Centralized Controllers," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, pp. 3975-3981, Oct. 2020. Note: ***Equal Authors**

Chapters 5 and 6 comprise the material under review in IEEE Control System Letters (L-

CSS) and for possible presentation at the 2022 IEEE International Conference on Decision and Control (CDC).

[**J1**] **V. R. Kamidi**, J. Kim, R. T. Fawcett, A. D. Ames, and K. Akbari Hamed, "Distributed Quadratic Programming-Based Nonlinear Controllers for Periodic Gaits on Legged Robots,"*IEEE Control Systems Letters*, Under Review, Feb. 2022.

1.8 Contributed Works

The following manuscripts were written and published in peer-reviewed journals or presented at peer-reviewed conferences as part of this research. Note that some of them form the foundation of this work while other contributions are independent of this research thrust.

Journal Articles:

[**J1**] **V. R. Kamidi**, J. Kim, R. T. Fawcett, A. D. Ames, and K. Akbari Hamed, "Distributed Quadratic Programming-Based Nonlinear Controllers for Periodic Gaits on Legged Robots,"*IEEE Control Systems Letters*, Under Review, Feb. 2022.

[**J2**] **V. R. Kamidi**, J. C. Horn, R. D. Gregg, and K. Akbari Hamed, "Distributed Controllers for Human-Robot Locomotion: A Scalable Approach Based on Decomposition and Hybrid Zero Dynamics," *IEEE Control Systems Letters*, 5(6):1976–1981, Dec. 2021.

[**J3**] K. Akbari Hamed, **V. R. Kamidi**, W. Ma, A. Leonessa, and A. D. Ames, "Hierarchical and Safe Motion Control for Cooperative Locomotion of Robotic Guide Dogs and Humans: A Hybrid Systems Approach," *IEEE Robotics and Automation Letters*, vol. 5, no. 1, pp. 56-63, Jan. 2020.

Conference Proceedings:

[**C1**] **V. R. Kamidi**, J. Kim, R. T. Fawcett, A. D. Ames, and K. Akbari Hamed, "Distributed Quadratic Programming-Based Nonlinear Controllers for Periodic Gaits on Legged Robots," *IEEE International Conference on Decision and Control (CDC)*, Under Review, Feb. 2022

[**C2**] **V. R. Kamidi**, J. C. Horn, R. D. Gregg and K. A. Hamed, "Distributed Controllers for Human-Robot Locomotion: A Scalable Approach Based on Decomposition and Hybrid Zero Dynamics," *American Control Conference (ACC)*, pp. 2049-2054, May. 2021.

[**C3**] A. Pandala*, **V. R. Kamidi***, and K. Akbari Hamed, "Decentralized Control Schemes for Stable Quadrupedal Locomotion: A Decomposition Approach from Centralized Controllers," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, pp. 3975-3981, Oct. 2020. (Note: ***Equal Authors**)

[**C4**] J. B. Martin, **V. R. Kamidi**, A. Pandala, R. Fawcett, and K. Akbari Hamed, "Exponentially Stabilizing and Time-Varying Virtual Constraint Controllers for Dynamic Quadrupedal Bounding," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Las Vegas, NV, pp. 3914-3921, Oct. 2020.

[**C5**] K. Akbari Hamed, **V. R. Kamidi**, A. Pandala, W. Ma, and A. D. Ames, "Distributed Feedback Controllers for Stable Cooperative Locomotion of Quadrupedal Robots: A Virtual Constraint Approach," *American Control Conference (ACC)*, Denver, CO, pp. 5314-5321, July 2020.

1.9 Outline

This document is compiled in a manuscript format. A brief outline of the successive chapters is presented below.

Chapter 2 establishes and discusses the preliminaries, the foundational framework utilized to develop the distributed control framework. A complete description of the full-order dynamic model is given, followed by the formulation of the centralized, virtual constraint controllers.

Chapter 3 formulates the distributed control framework for legged locomotion in a platform-agnostic manner. First, it demonstrates the decomposition of the general dynamic equation for a single system into two subsystems. Next, this chapter establishes the distributed control structure that captures the strong nonlinear interaction forces and demonstrates the need for minimal communication between the subsystems. It then carefully studies the exponential stability behavior of periodic orbits (gaits) using the Poincaré return map of the distributed controllers and compares it with the centralized ones. The chapter then sets up an optimization-based approach to synthesize exponentially stabilizing distributed controllers.

Chapter 4 validates the theoretical contributions established in Chapter 3. Towards that end, this chapter considers the collaborative locomotion problem of human-lower extremity prosthesis that naturally appears as a decentralized system. It performs extensive numerical simulations to demonstrate inherent robustness in the proposed distributed control approach. The chapter concludes with a comparison study between the centralized and the distributed control frameworks.

Chapter 5 formulates and presents a formal foundation to systematically decompose QP-

based centralized nonlinear controllers into a network of lower-dimensional distributed QPs, with application to legged locomotion. The proposed approach formulates a feedback structure between the local QPs and leverages a one-step communication delay protocol. This chapter then analyzes the properties of the local QPs wherein it is established that their steady-state solutions on periodic orbits (representing gaits) coincide with that of the centralized QP. The asymptotic convergence of local QPs' solutions to the steady-state solution is then studied via Floquet theory.

Chapter 6 validates the theoretical contributions established in Chapter 5. Towards that end, it considers the problem of quadrupedal locomotion. It envisions that the centralized system is composed of two subsystems, particularly the composition of the front and hind subsystems. This chapter evaluates the effectiveness of the analytical results through rigorous numerical simulations and various experiments on a quadrupedal robot, A1, with the result being robust locomotion on different terrains and in the presence of external disturbances.

Chapter 7 concludes this thesis with a concise summary of the contributions addressed by this thesis and a brief discussion of the future directions that this work can motivate.

Part I

Distributed Controllers: A Scalable Approach based on Hybrid Zero Dynamics

Chapter 2

Preliminaries: HZD-Based Legged Locomotion Control

2.1 Introduction

In this chapter, we establish the mathematical foundations upon which the contributions in Part I of this preliminary document is built on. In particular, we propose a decomposition based distributed controller that assumes the existence of an orbit (stable or unstable) synthesized by centralized HZD-based controllers. It is therefore crucial to understand the construction of centralized controllers. Accordingly, we will begin with providing a more formal description of the hybrid systems formulation (briefly described in Subsection 1.6.1) for the general class of legged locomotion problems in Section 2.2. We will then briefly revisit the continuous-time dynamics (first introduced in Subsection 1.6.1) to address the inclusion of holonomic constraints in Section 2.3. In Section 2.4, we will discuss the second component of a hybrid system, the discrete dynamics that facilitate the transition from one continuous-time domain to other. We will then establish the open-loop hybrid model in its complete form in Section 2.5. The centralized HZD-based controller for this open-loop model will then be established in Section 2.6. Finally, this chapter concludes with a brief discussion in Section 2.7.

2.2 Hybrid Systems Formulation

Definition 2.1. (Directed Cycle). A directed cycle is a directed graph, $\Lambda \coloneqq (\mathcal{V}, \mathcal{E})$, that is composed of the set of vertices $\mathcal{V} \coloneqq \{v_1, \cdots, v_k\}$ and edges $\mathcal{E} \coloneqq \{e_1, \cdots, e_k\}$, where $e_j \coloneqq (v_{j-1} \to v_j)$ for all $j = 2, \cdots, k$ and $e_1 \coloneqq (v_k \to v_1)$.

Without generalizing to a specific platform, the open loop multi-domain hybrid model of locomotion can be given by the following tuple

$$\mathscr{HL}^{\text{ol}} = (\Lambda, \mathcal{X}, \mathcal{U}, \mathcal{D}, \mathcal{S}, \Delta, \mathcal{FG}). \tag{2.1}$$

Here, the vertices represent the continuous-time domain dynamics, referred to as domains or phases. The edges then represent the discrete-time transitions between two continuous-

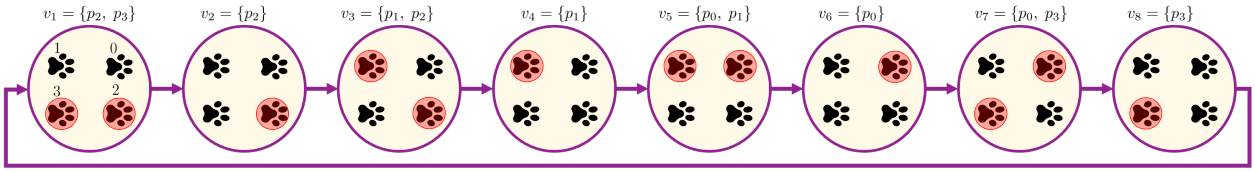


Figure 2.1: Illustration of a multi-domain hybrid model emulating a quadrupedal ambling gait.

time domains that arises from the change in the number of holonomic constraints with the environment (i.e., a new contact point is added to the set of existing contact points or breaks). The assumption is further made that the discrete-time transitions are instantaneous. The set of state manifolds for the directed cycle, Λ, is denoted by $\mathcal{X} \coloneqq \{\mathcal{X}_v\}_{v\in\mathcal{V}}$, in which $\mathcal{X}_v \subseteq \mathbb{R}^{2n}$ is the state manifold for the vertex $v \in \mathcal{V}$. The set of admissible controls is denoted by $\mathcal{U} \coloneqq \{\mathcal{U}_v\}_{v\in\mathcal{V}}$, where $\mathcal{U}_v \subseteq \mathbb{R}^m$ is the set of admissible control inputs for the domain $v \in \mathcal{V}$. Here, n and m retain the same definition from Section 1.6 and correspondingly $2n$ signifies the total number of states. With this notation in hand, $\mathcal{D} \coloneqq \{\mathcal{D}_v\}_{v\in\mathcal{V}}$ denotes the set of domain of admissibility, where $\mathcal{D}_v \subseteq \mathcal{X}_v \times \mathcal{U}_v$ is a smooth submanifold of $\mathbb{R}^n \times \mathbb{R}^m$. The set of guards for the hybrid system in Equation (2.1), given by $\mathcal{S} \coloneqq \{\mathcal{S}_e\}_{e\in\mathcal{E}}$, is the set of switching surfaces on which discrete-time transitions occur when the state and control trajectories, $(x(t), u(t))$, intersect the guard $\mathcal{S}_e \subset \mathcal{D}_v$. $\Delta \coloneqq \{\Delta_e\}_{e\in\mathcal{E}}$ is the set of reset laws (i.e., re-initialization rules) to denote the discrete-time transitions. Finally, $\mathcal{FG} \coloneqq \{\mathcal{FG}_v\}_{v\in\mathcal{V}}$ is a set of continuous-time control systems defined on $\mathcal{D}_v$, given by the input-affine state equation $\dot{x} = f_v(x) + g_v(x)\,u$, with f_v and g_v being smooth on $\mathcal{X}_v$.

Example 2.2. (Ambling gait): Consider an 8 domain ambling gait as in Figure 2.1. The legs of the robot are enumerated by $i \in \{0, 1, 2, 3\}$ with the sequence depicted in domain v_1 of the Figure 2.1. Further, as discussed, let the directed cycle of this gait be $\Lambda = (\mathcal{V}, \mathcal{E})$. In this particular example, $\mathcal{V} = \{p_{2,3}, p_2, p_{2,1}, p_1, p_{1,0}, p_0, p_{0,3}, p_3\}$ and the edge set $\mathcal{E} = \{p_{2,3} \to p_2, p_2 \to p_{2,1}, p_{2,1} \to p_1, p_1 \to p_{1,0}, p_{1,0} \to p_0, p_0 \to p_{0,3}, p_{0,3} \to p_3\}$, where p_i denotes the foot in contact with the ground in that particular domain. Moreover, $p_{i,j}$, $i \neq j \in \{0, 1, 2, 3\}$ represents the domains in which feet i and j are simultaneously in contact with the ground.

2.3 Continuous-Time Dynamics

We will construct and understand the remaining components of Equation (2.1) in this and the following sections. We will first begin by extending the floating-base dynamics in Equation (1.7) to account for the holonomic constraints arising from the contacts established with the environment in each domain. The resultant will be noted as the *constrained* continuous-time dynamics. However, before we proceed it is necessary to understand the types of contacts and the constraints they introduce.

Table 2.1: Contact Constraints Associated with Typical Foot Contacts in Legged Locomotion

Point Contact	Line Contact	Planar Contact
$\lambda^z > 0$	$\lambda^z > 0$	$\lambda^z > 0$
$\lvert\lambda^x\rvert < \frac{\mu}{\sqrt{2}}\lambda^z$	$\lvert\lambda^x\rvert < \frac{\mu}{\sqrt{2}}\lambda^z$	$\lvert\lambda^x\rvert < \frac{\mu}{\sqrt{2}}\lambda^z$
$\lvert\lambda^y\rvert < \frac{\mu}{\sqrt{2}}\lambda^z$	$\lvert\lambda^y\rvert < \frac{\mu}{\sqrt{2}}\lambda^z$	$\lvert\lambda^y\rvert < \frac{\mu}{\sqrt{2}}\lambda^z$
-	$-l_w\lambda^z < \lambda^{mx} < l_w\lambda^z$	$-l_w\lambda^z < \lambda^{mx} < l_w\lambda^z$
-	-	$-l_h\lambda^z < \lambda^{my} < l_t\lambda^z$

2.3.1 Contact Models

Towards that objective, in this subsection a brief review of the contacts and the corresponding constraints is provided. Further, given that this thesis explores both the bipedal and quadrupedal robots, we frame this discussion to address the typical contacts observed in legged robotics. In order to construct the holonomic constraints arising from the contacts with the external environment, consider a reference frame attached to foot at the contact point as shown in Figure 2.2. The ground contact wrench with respect to this frame is then given by $\lambda \coloneqq \text{col}(\lambda^x, \lambda^y, \lambda^z, \lambda^{mx}, \lambda^{my}, \lambda^{mz})$, where the first three components are the GRFs and the last three components are the ground reaction torques (i.e., moments). This provides us at most 6 holonomic constraints. For a point contact enabled by the point feet in almost all modern quadrupedal robots (see Figure 1.1), 3 holonomic constraints are associated with this contact type as noted in Table 2.1. One constraint comes from the condition imposed on the foot to avoid lifting from the ground

$$\lambda^z > 0. \tag{2.2}$$

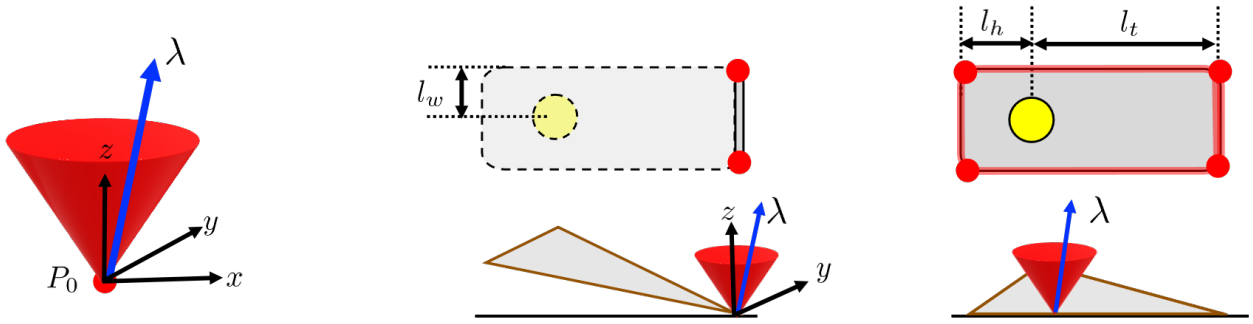
Next, to avoid linear slipping, we enforce the following constraint

$$\sqrt{(\lambda^x)^2 + (\lambda^y)^2} < \mu\lambda^z, \tag{2.3}$$

where μ is the friction coefficient. In order to use linear constraints, traditionally the friction cone in Equation (2.3) is replaced by the friction pyramid that gives us $\lvert\lambda^x\rvert < \frac{\mu}{\sqrt{2}}\lambda^z$ and $\lvert\lambda^y\rvert < \frac{\mu}{\sqrt{2}}\lambda^z$. Another contact type noticed in legged robots is the line contact that typically models the rotation of the foot about the toe or heel. As seen in Figure 2.2 (b), the rotation along the y-axis of the foot frame is free resulting in a contact wrench with five non-zero components. Their is no moment exerted about the y-axis. In addition to the constraints in Equation (2.2) and (2.3), the constraint on the moment produced by the ground is limited in the x-axis as follows

$$-l_w\lambda^z < \lambda^{mx} < l_w\lambda^z, \tag{2.4}$$

where l_w is the width of the foot as shown in Figure 2.2(b). Finally, we have planar contacts established by the flat foot contact as seen in Figure 2.2 (c). This constraints results in a



Point Contact **(a)** Line Contact **(b)** Planar Contact **(c)**

Figure 2.2: Typical foot contacts observed in legged locomotion: (a) point contact, (b) line contact, and, (c) planar contact.

contact wrench with six non-zero components and additionally a constraint on the moment produced by the y-axis is imposed as

$$-l_h\lambda^z < \lambda^{my} < l_t\lambda^z, \tag{2.5}$$

in which l_h and l_t are the distance from the foot frame to the heel and toe, respectively. As noted in [100] an additional constraint on λ^z can also be given but in addition to the size of the foot, friction coefficient it requires an unknown force distribution along the sole of the foot. As a result, this constraint is neglected when modeling straight displacement.

2.3.2 Constrained Continuous-Time Dynamics

Consider again, $q \in \mathcal{Q} \subset \mathbb{R}^n$, the generalized coordinates that capture the evolution of all DoFs and the corresponding velocities encoded in $\dot{q} \in \mathbb{R}^n$. Together q and $\dot{q}$ combine to form the state vector, $x \coloneqq \mathrm{col}(q, \dot{q}) \in \mathcal{X} \subset \mathbb{R}^{2n}$ and $\mathcal{X} \coloneqq \mathrm{T}\mathcal{Q} \coloneqq \mathcal{Q} \times \mathbb{R}^n$ represents the state manifold. The evolution of the robot in each continuous-time domain, $v \in \mathcal{V}$, can then be described by the following algebraic and second order ODE that arises from the Euler-Lagrange equations and principle of virtual work

$$\begin{aligned} D(q)\,\ddot{q} + C(q,\dot{q})\,\dot{q} + G(q) &= B\,u + J_v^{\top}(q)\lambda, \\ J_v(q)\,\ddot{q} + \frac{\partial}{\partial q}\Big(J_v(q)\dot{q}\Big)\,\dot{q} &= 0, \end{aligned} \tag{2.6}$$

where $D(q) \in \mathbb{R}^{n\times n}$ denotes the positive definite mass inertia matrix, $C(q,\dot{q})\,\dot{q} \in \mathbb{R}^n$ represents the Coriolis and centrifugal terms, and $G(q) \in \mathbb{R}^n$ contains the gravitational terms. The input distribution matrix and Lagrange multipliers are given by $B \in \mathbb{R}^{n\times m}$ with the property $\mathrm{rank}(B) = m$ and $\lambda \in \mathbb{R}^{n_v^h}$, where n_v^h defines the number of holonomic constraints

arising from the type of foot contact (see Section 2.3.1). Further, $J_v(q)\,\dot{q} = 0$, where $J_v(q) \coloneqq \frac{\partial \eta_v}{\partial q} \in \mathbb{R}^{n_v^h \times n}$ is the velocity constraint of the holonomic physical constraint that is expressed as $\eta_v(q) = 0$, in which $\eta_v \coloneqq \{\eta_c\}_{c \in \mathcal{C}_v} \in \mathbb{R}^{n_v^h}$. Here $\mathcal{C}_v$ is the index set of all holonomic constraints on $\mathcal{D}_v$. Then, by eliminating the Lagrange multipliers from Equation (2.6), we can obtain the constrained dynamics in domain v as follows:

$$D(q)\,\ddot{q} + F_v(q, \dot{q}) = T_v(q)\,u, \tag{2.7}$$

where

$$\begin{aligned} F_v(q, \dot{q}) &\coloneqq \mathrm{proj}_v H + J_v^\top (J_v\, D^{-1}\, J_v^\top)^{-1} \frac{\partial}{\partial q}(J_v \dot{q})\,\dot{q}, \\ H(q, \dot{q}) &\coloneqq C(q, \dot{q})\,\dot{q} + G(q), \\ T_v(q) &\coloneqq \mathrm{proj}_v B, \\ \mathrm{proj}_v(q) &\coloneqq I - J_v^\top (J_v\, D^{-1}\, J_v^\top)^{-1} J_v\, D^{-1}. \end{aligned} \tag{2.8}$$

Finally, the evolution of the entire system can be given as $\dot{x} = \mathrm{col}(\dot{q}, \ddot{q})$ and is expanded into

$$\dot{x} = \begin{bmatrix} \dot{q} \\ D^{-1}(q)(-F_v(q, \dot{q}) + T_v(q)\,u) \end{bmatrix}. \tag{2.9}$$

The rewritten equations of motion in Equation (2.9) can be further defined as $\dot{x} \coloneqq f_v(x) + g_v(x)\,u$ for which the state manifold is given by

$$\mathcal{X}_v \coloneqq \{x = \mathrm{col}(q, \dot{q}) \in \mathrm{T}\mathcal{Q} \mid \eta_v(q) = 0, J_v(q)\,\dot{q} = 0\}. \tag{2.10}$$

According to the construction procedure then, $\mathcal{X}_v$ is forward-invariant under the flow of the continuous-time dynamics.

2.4 Discrete-Time Dynamics

A discrete-time transition occurs every time a robot makes or breaks the contact with environment. In the event of a lift-off, i.e., when an existing contact is broken, the discrete-time dynamics are simply taken as the identity map, i.e., $x^+ = \Delta_e(x^-) \coloneqq x^-$ to preserve the continuity of position and velocity. On the contrary, in the event of a new contact, the state of the mechanical system undergoes an abrupt change in the velocity according to the instantaneous impact model between two rigid bodies [114]. To be precise, the conservation of the generalized momentum during the infinitesimal period of the impact is given in

$$\begin{aligned} D(q)\,\dot{q}^+ - D(q)\,\dot{q}^- &= J_{v+1}^\top(q)\,\delta\lambda, \\ J_{v+1}(q)\,\dot{q}^+ &= 0, \end{aligned} \tag{2.11}$$

where $\dot{q}^-$ and $\dot{q}^+$ represent the generalized velocities right before and after the impact, respectively. Further, $\delta\lambda$, represents the intensity of impulsive GRFs and/or moments during the impact and the contact Jacobian matrix, J_{v+1} corresponds to the next domain, i.e., the domain right after impact. Solving Equation (2.11), one can obtain the relationship between pre-impact and post-impact velocity as follows

$$\dot{q}^+ = (I - D^{-1}J_{v+1}^\top (J_{v+1}D^{-1}J_{v+1}^\top)^{-1}J_{v+1})\,\dot{q}^-. \tag{2.12}$$

Now using the continuity of position (i.e., $q^+ = q^-$) and Equation (2.12), we can express the discrete-time mapping as $x^+ = \Delta_e(x^-)$. Here, Δ_e is a nonlinear map that is assembled as follows:

$$\Delta_e(x^-) := \begin{bmatrix} q^- \\ (I - D^{-1}J_{v+1}^\top (J_{v+1}D^{-1}J_{v+1}^\top)^{-1}J_{v+1})\,\dot{q}^- \end{bmatrix}. \tag{2.13}$$

2.5 Open-Loop Hybrid Model

The stability of periodic orbits of multi-domain hybrid models as in Figure 2.1 can be shown to be equivalent to the one of an associated single-domain model (see [213, Theorem 4.3]). In particular, with the help of [213, Theorem 4.3], we introduce $\tilde{\Delta}$ that encodes the discrete maps of all the domains and the flows of all the continuous-time domains, except the one in consideration, domain v. We can then piece together and express the open-loop and single-domain hybrid model of locomotion as follows:

$$\Sigma^{\text{ol}} : \begin{cases} \dot{x} = f_v(x) + g_v(x)\,u, & x \in \mathcal{X}_v \\ x^+ = \tilde{\Delta}(x^-), & x^- \in \mathcal{X}_v \cap \mathcal{S}_e, \end{cases} \tag{2.14}$$

where x^- and x^+ denote the state of the system right before and right after the discrete-time transition, respectively, $\tilde{\Delta} : \mathcal{X}_v \to \mathcal{X}_v$ is a smooth (i.e., $\mathcal{C}^\infty$) reset map, and f_v and g_v are smooth functions. We note that the transition from multi-domain to single-domain models is done only to simplify the analysis and that in reality we make use of the multi-domain models to synthesise and validate our controllers (see Chapter 4).

Assumption 2.3. (Periodic Orbit and Phasing Variable): There exists a *periodic solution* (i.e., gait) to the hybrid model in Equation (2.14) for some nominal control inputs and some fundamental period $T^\star > 0$. The corresponding *periodic orbit* is represented by $\mathcal{O} \subset \mathcal{X}_v$. To define the desired evolution of the state and control inputs on $\mathcal{O}$, we make use of a time-based *phasing variable* as

$$\tau := \frac{t - t^+}{T^\star}, \tag{2.15}$$

where t^+ represents the time sample right after the discrete-time event. The evolution of the state and control inputs on $\mathcal{O}$ are further denoted by $x^\star(\tau) := \text{col}(q^\star(\tau), \dot{q}^\star(\tau))$ and $u^\star(\tau)$ for $0 \leq \tau < 1$.

2.6 Centralized Feedback Controllers

The objective of this section is to present a family of parameterized virtual constraint controllers that asymptotically stabilize the dynamic gaits for the multi-domain hybrid models of locomotion (see Figure 2.1 for an example). Virtual constraints are defined as a set of kinematic constraints that can be imposed by the action of feedback control laws to coordinate the motion of links within a stride [54, 92, 96, 144, 147, 156, 167, 174, 178]. In this work, we are only interested in relative degree 2, holonomic constraints for position tracking. Additionally, one can use relative degree 1, nonholonmic constraints or a combination of holonomic and nonholonomic constraints as in [18, 104].

2.6.1 Family of Parameterized Virtual Constraint Controllers

We will now present a centralized, HZD-based controller to exponentially stabilize the desired orbit $\mathcal{O}$ for the hybrid model of locomotion. For this purpose, we make use of the following time-varying holonomic outputs (i.e., virtual constraints) to be imposed

$$y_v(\tau, q) \coloneqq h_v \coloneqq C_v \left(q - q^\star(\tau)\right), \tag{2.16}$$

for some full-rank output matrix C_v with the property $m_v \coloneqq \dim(y_v) \leq \dim(u)$ and $q^\star(\tau)$ represents the desired evolution of the configuration variables on the orbit, $\mathcal{O}$ in terms of the phasing variable, τ. Now, consider Equation (2.16) and differentiate it along the continuous-time dynamics to obtain

$$\dot{y}_v \coloneqq \dot{h}_v \coloneqq \frac{\partial h_v}{\partial q}\dot{q} + \frac{\partial h_v}{\partial \tau}\dot{\tau}. \tag{2.17}$$

Further, differentiating Equation (2.17), we get

$$\ddot{y}_v \coloneqq \ddot{h}_v \coloneqq \frac{\partial h_v}{\partial q}\ddot{q} + \underbrace{\frac{\partial}{\partial q}\left(\frac{\partial h_v}{\partial q}\dot{q}\right)\dot{q}}_{0} + \underbrace{\frac{\partial}{\partial \tau}\left(\frac{\partial h_v}{\partial q}\dot{q}\right)\dot{\tau}}_{0} + \underbrace{\frac{\partial}{\partial q}\left(\frac{\partial h_v}{\partial \tau}\dot{\tau}\right)\dot{q}}_{0} + \frac{\partial^2 h_v}{\partial \tau^2}\dot{\tau}^2. \tag{2.18}$$

On substituting Equation (2.7) in Equation (2.18), the output dynamics become

$$\ddot{y}_v = \underbrace{-\frac{\partial h_v}{\partial q}D^{-1}F_v}_{\mathrm{L}^2_{f_v} y_v} + \underbrace{\frac{\partial h_v}{\partial q}D^{-1}T_v}_{\mathrm{L}_{g_v}\mathrm{L}_{f_v} y_v}\, u + \frac{\partial^2 h_v}{\partial \tau^2}\dot{\tau}^2, \tag{2.19}$$

which on simplification translates into

$$\ddot{y}_v = \mathrm{L}_{g_v}\mathrm{L}_{f_v} y_v\, u + \mathrm{L}^2_{f_v} y_v + \frac{\partial^2 y_v}{\partial \tau^2}(\dot{\tau}^2). \tag{2.20}$$

Here L_{f_v} and L_{g_v} are the Lie derivatives with respect to the vector field f_v and g_v. Further $\mathrm{L}_{g_v}\mathrm{L}_{f_v}y_v$ is the decoupling matrix. Equations (2.17)-(2.19) illustrates the standard I-O linearization [114]. Furthermore, we are interested in the following desired output dynamics $\ddot{y}_v + k_d\,\dot{y}_v + k_p\,y_v = 0$ for some positive gains k_p and k_d. One can then, employ a nonlinear feedback law as follows

$$u = \Gamma_v(\tau, x) := -\left(\mathrm{L}_{g_v}\mathrm{L}_{f_v}y_v\right)^{\dagger}\left(\mathrm{L}_{f_v}^2 y_v + \frac{\partial^2 y_v}{\partial \tau^2}(\dot{\tau})^2 + k_d\,\dot{y}_v + k_p\,y_v\right), \tag{2.21}$$

that results in $\lim_{t\to\infty} y(t) = 0$, where $(\mathrm{L}_{g_v}\mathrm{L}_{f_v}y_v)^{\dagger}$ denotes the Moore–Penrose inverse of the decoupling matrix $\mathrm{L}_{g_v}\mathrm{L}_{f_v}y_v \in \mathbb{R}^{m_v\times m}$. Additionally, the feedback control law in Equation (2.21) renders the zero dynamics manifold

$$\mathcal{Z}_v := \{x \in \mathcal{X}_v \mid y_v(\tau, q) = \mathrm{L}_{f_v}y_v + \frac{\partial y_v}{\partial \tau}\,\dot{\tau} = 0\}, \tag{2.22}$$

attractive and forward-invariant under the flow of the closed-loop system, $\dot{x} = f_v^{\mathrm{cl}}(\tau, x)$, where $f_v^{\mathrm{cl}}(\tau, x) := f_v(x) + g_v(x)\,\Gamma_v(\tau, x)$. The construction procedure then dictates that the $\dim(\mathcal{Z}_v) = 2(n - n_v^h - m_v)$. Additionally, two other constraints that ensure the zero-dynamics manifold is non-trivial and does not result in singularities are given as $\dim(\mathcal{Z}_v) > 1$ and $\dim(y_v) \leq \mathrm{rank}(T_v)$, where, the definition of T_v is given in Equation (2.7). Consequently, these constraints decide the number of outputs, m_v in each domain.

2.7 Summary

In this chapter, we presented in detail the method to synthesize centralized controllers utilizing the HZD framework. Without resorting to a specific legged platform, we began with demonstrating the procedure to obtained the constrained dynamic representation in a continuous-time domain. Next, we capture the transitions between continuous-time domains in the discrete dynamic equations. Together, they form an open-loop hybrid system. Before we proceeded to the development of nonlinear controller, we took a short detour to understand the typical types of contacts and the constraints they introduce into the domain in Subsection 2.3.1. We then introduced the concept of virtual constraints and the subsequent synthesis of parameterized virtual constraint controllers in Section 2.6. This chapter laid the foundations for the contributions in Part I of this thesis. In the next chapter, we will demonstrate the synthesis of distributed controllers that are decomposed from the centralized controllers synthesized in this chapter.

Chapter 3

Synthesis of Distributed Controllers

This chapter *formulates* scalable distributed control algorithms based on principles established in Chapter 2. It primarily documents the main theoretical contribution corresponding to Part I of this thesis. Additionally, we note that the theory presented in this chapter is platform agnostic and is valid for any legged robot: bipedal, quadrupedal, prosthesis, or exoskeletons (see Figure 3.1). After providing a brief summary of contributions in Subsection 3.1.1, Section 3.2 will demonstrate the dynamic decomposition that results in two distinctive subsystems. It is then be followed by the control decomposition that will provide local control laws for each individual subsystem in Section 3.3. In Section 3.4, we will present the exponential stability analysis that enables us to simplify the search for the local control parameters for the local control law in Section 3.3. Finally, Section 3.5 builds on the results of Section 3.4 and formulates a small-scale optimization problem that is transcribed into a NLP to find optimal local control parameters. This chapter then concludes with a brief summary in Section 3.6.

3.1 Introduction

3.1.1 Contributions of this Chapter

The specific outcomes of this chapter are summarized in the following contributions: (1) Development of a decomposition based approach for the synthesis of distributed controllers. We show that the integration of decomposition and optimization allows for a significant reduction in the parametric space as one only needs to modify and optimize the already-existing centralized HZD controllers rather than searching for local controllers from scratch. This approach thus leads to a *small scale optimization* problem. (2) Further, we study the properties of the Jacobian of the Poincaré map of both centralized and distributed controllers and show that there exists an upper bound on the norm of the difference of the Jacobian matrices. This allows for a *significant simplification* in the search of optimal local control parameters.

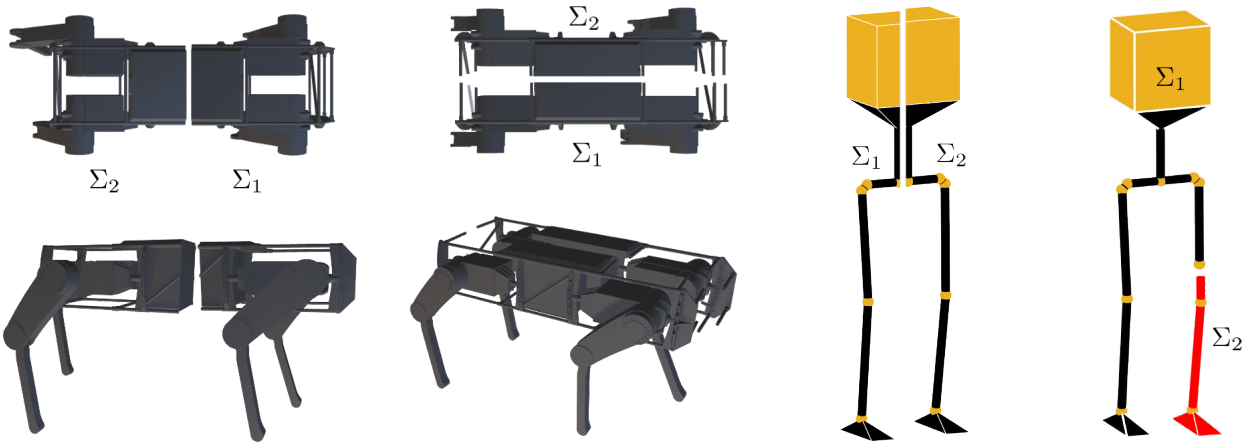


Figure 3.1: Illustration of the proposed decomposition framework, applicable to any legged locomotion model: (a) front-hind decomposition of quadrupedal robot (top and isometric view), (b) left-right decomposition of quadrupedal robot (top and isometric view), (c) left-right decomposition of humanoid robot (isometric view), and (d) decomposition of human-prosthesis model (isometric view).

3.2 Dynamic Decomposition

This section and the succeeding section develops distributed feedback controllers, based on the decomposition approach, to exponentially stabilize the periodic orbit $\mathcal{O}$. We consider the synthesis problem of local controllers for two subsystems of the hybrid model of locomotion given in Equation (2.14). Without loss of generality, we assume that the generalized coordinate vector and control inputs can be decomposed as $q = \mathrm{col}(q_1, q_2)$ and $u = \mathrm{col}(u_1, u_2)$, where $q_i \in \mathcal{Q}_i$ and $u_i \in \mathcal{U}_i$ denote the local configuration variables and local control inputs for the subsystem Σ_i for $i \in \{1, 2\}$. The local state variables are further denoted by $x_i := \mathrm{col}(q_i, \dot{q}_i) \in \mathcal{X}_i$.

We begin by first decomposing the equations of motion in Equation (2.7) as follows:

$$\left[\begin{array}{c|c} D_{ii} & D_{ij} \\ \hline D_{ji} & D_{jj} \end{array}\right] \left[\begin{array}{c} \ddot{q}_i \\ \hline \ddot{q}_j \end{array}\right] + \left[\begin{array}{c|c} C_{ii} & C_{ij} \\ \hline C_{ji} & C_{jj} \end{array}\right] \left[\begin{array}{c} \dot{q}_i \\ \hline \dot{q}_j \end{array}\right] + \left[\begin{array}{c} G_i \\ \hline G_j \end{array}\right] = \left[\begin{array}{c|c} B_{ii} & B_{ij} \\ \hline B_{ji} & B_{jj} \end{array}\right] \left[\begin{array}{c} u_i \\ \hline u_j \end{array}\right]. \tag{3.1}$$

To develop local controllers, we make the following assumption.

Assumption 3.1. (State Approximation for Subsystems): We suppose that the subsystem Σ_i for every $i \in \{1, 2\}$ can approximate the state variables for the other subsystem, i.e., Σ_j, $j \neq i \in \{1, 2\}$, by their desired values on the periodic orbit $\mathcal{O}$, that is, $x_j \approx x_j^\star(\tau)$.

Using Equation (3.1), we can write the equation of motion for the first subsystem as follows

$$D_{ii}\,\ddot{q}_i + D_{ij}\,\ddot{q}_j + H_i = B_{ii}\,u_i + B_{ij}\,u_j, \tag{3.2}$$

for all $i \neq j \in \{1,2\}$, where $H_i := C_{ii}\dot{q}_i + C_{ij}\dot{q}_j + G_i$. From Equation (3.2), one can solve for $\ddot{q}_i$, i.e.,

$$\begin{aligned}\left(D_{ii} - D_{ij}\,D_{jj}^{-1}\,D_{ji}\right)\ddot{q}_i + H_i - D_{ij}\,D_{jj}^{-1}\,H_j = \\ \left(B_{ii} - D_{ij}\,D_{jj}^{-1}B_{ji}\right)u_i \;+ \left(B_{ij} - D_{ij}\,D_{jj}^{-1}\,B_{jj}\right)u_j.\end{aligned} \tag{3.3}$$

Using Assumption 3.1, the nonlinear dynamics for subsystem i can be approximated as follows:

$$\bar{D}_{ii}\,(q_i,t)\,\ddot{q}_i + \bar{H}_i\,(q_i,\dot{q}_i,t) = \bar{B}_{ii}\,(q_i,t)\,u_i, \tag{3.4}$$

in which

$$\begin{aligned}\bar{D}_{ii} &:= D_{ii} - D_{ij}\,D_{jj}^{-1}\,D_{ji}, \\ \bar{H}_i &:= H_i - D_{ij}\,D_{jj}^{-1}\,H_j - \left(B_{ij} - D_{ij}\,D_{jj}^{-1}\,B_{jj}\right)u_j, \\ \bar{B}_{ii} &:= B_{ii} - D_{ij}\,D_{jj}^{-1}B_{ji},\end{aligned} \tag{3.5}$$

that are evaluated at $q_j = q_j^\star(\tau)$, $\dot{q}_j = \frac{\partial q_j^\star}{\partial \tau}(\tau)\,\dot{\tau}$, and $u_j = u_j^\star(\tau)$. Furthermore, the subsystem i can approximate the nonlinear dynamics of the other subsystem, i.e., the subsystem j, as

$$\bar{D}_{jj}\,(q_i,t)\,\ddot{q}_j + \bar{H}_j\,(q_i,\dot{q}_i,t) = \bar{B}_{ji}\,(q_i,t)\,u_i, \tag{3.6}$$

where

$$\begin{aligned}\bar{D}_{jj} &:= D_{jj} - D_{ji}\,D_{ii}^{-1}\,D_{ij}, \\ \bar{H}_j = H_j - D_{ji}\,D_{ii}^{-1}\,H_i &- \left(B_{jj} - D_{ji}\,D_{ii}^{-1}\,B_{ij}\right)u_j, \\ \bar{B}_{ji} &:= B_{ji} - D_{ji}\,D_{ii}^{-1}\,B_{ii}\end{aligned} \tag{3.7}$$

evaluated at $q_j = q_j^\star(\tau)$, $\dot{q}_j = \frac{\partial q_j^\star}{\partial \tau}(\tau)\,\dot{\tau}$, and $u_j = u_j^\star(\tau)$.

3.3 Control Decomposition

3.3.1 Output Modification

The virtual constraints in Equation (2.16) can be decomposed as $y = \mathrm{col}(y_1, y_2)$ for some local outputs y_1 and y_2 with the property $\dim(y_i) \leq \dim(u_i)$, $i \in \{1,2\}$. In particular, we

assume that

$$y_i = C_{ii}\left(q_i - q_i^\star(\tau)\right) + C_{ij}\left(q_j - q_j^\star(\tau)\right), \tag{3.8}$$

for all $i \neq j \in \{1, 2\}$. We remark that for the simplicity of presentation, we drop the subscript "$_v$" from y_v in Equation (2.16). Now, using Assumption 3.1, the local output y_i can be reduced to $y_i = C_{ii}\left(q_i - q_i^\star(\tau)\right)$. Our group's previous work [12] has shown that such local virtual constraints may *not* stabilize the orbit $\mathcal{O}$ for the hybrid model of locomotion. To tackle this challenge, we make the following assumption.

Assumption 3.2. (Existence of a set of Measurable Global Variables): We suppose that there are some smooth functions $\psi_i(x) = \psi_i(x_1, x_2)$, referred to as *measurable global variables*, that 1) depend on the global states x_1 and x_2, and 2) are measurable for the subsystem Σ_i along the trajectories of the hybrid model via some sensors.

We next consider some measurable holonomic global variables for the subsystem i represented by $\psi_i(q)$. We further suppose that both ψ_i and $\dot{\psi}_i$ are measurable for the subsystem Σ_i. Our motivation for this assumption comes from the fact that robotic platforms can be equipped with some inertial measurements units (IMUs) attached at the appropriate location to measure the absolute position and velocity variables that depend on the global states $(q, \dot{q})$. We then modify the local output for the subsystem Σ_i as follows:

$$y_i = C_{ii}(q_i - q_i^\star(\tau)) + E_i(\Psi(q) - \Psi_i^\star(\tau)), \tag{3.9}$$

where $\psi_i^\star(\tau)$ denotes the desired evolution of the measurable global variables ψ_i on $\mathcal{O}$ in terms of the phasing variable τ. The local output in Equation (3.9) is further parameterized by an output matrix E_i to be determined in Section 3.4. Differentiating the output in Equation (3.9) then results in

$$\begin{aligned}\ddot{y}_i = C_{ii}&\left(\ddot{q}_i - \frac{\partial^2 q_i^\star}{\partial \tau^2}(\dot{\tau})^2\right)\\ &+ E_i\left(\frac{\partial \psi_i}{\partial q}\ddot{q} + \frac{\partial}{\partial q}\left(\frac{\partial \psi_i}{\partial q}\dot{q}\right)\dot{q} - \frac{\partial^2 \psi_i^\star}{\partial \tau^2}(\dot{\tau})^2\right),\end{aligned} \tag{3.10}$$

in which the acceleration terms, i.e., $\ddot{q}_i$ and $\ddot{q} = \mathrm{col}(\ddot{q}_1, \ddot{q}_2)$, can be approximated using the nonlinear dynamics in Equation (3.4) and Equation (3.6). Furthermore, the nonlinear terms $\frac{\partial \psi_i}{\partial q}$ and $\frac{\partial}{\partial q}(\frac{\partial \psi_i}{\partial q}\dot{q})\dot{q}$ can be computed utilizing Assumption 3.1. This would result in an input-affine output dynamics as follows:

$$\ddot{y}_i = A_i(\tau, x_i)\, u_i + b_i(\tau, x_i) = -k_d\, \dot{y}_i - k_p\, y_i. \tag{3.11}$$

From the output dynamics in Equation (3.11), one can solve for the minimum 2-norm local

control solution, that is,

$$u_i = \Gamma_i(\tau, x_i, \psi_i, \dot{\psi}_i) := -A_i^{\dagger} \left(b_i + k_d \, \dot{y}_i + k_p \, y_i\right). \tag{3.12}$$

3.4 Exponential Stability Analysis

This section will investigate the exponential stabilization problem of the periodic orbit $\mathcal{O}$ under the proposed distributed feedback controllers. We will study the Jacobian matrix of the Poincaré map with distributed controllers and then set up an optimization-based approach to tune the parameters of the local controllers given in Equation (3.12).

3.4.1 Poincaré Sections Analysis for Closed-loop Systems

The evolution of the closed-loop mechanical system with centralized and distributed feedback controllers can be expressed by the following hybrid models

$$\Sigma^{\text{cl}} : \begin{cases} \begin{bmatrix} \dot{x} \\ \dot{\tau} \end{bmatrix} = \begin{bmatrix} f^{\text{cl}}(\tau, x) \\ \frac{1}{T^\star} \end{bmatrix}, & x \in \mathcal{X} \\ \begin{bmatrix} x^+ \\ \tau^+ \end{bmatrix} = \begin{bmatrix} \Delta(x^-) \\ 0 \end{bmatrix}, & x^- \in \mathcal{X} \cap \mathcal{S} \end{cases} \tag{3.13}$$

and

$$\hat{\Sigma}^{\text{cl}} : \begin{cases} \begin{bmatrix} \dot{x} \\ \dot{\tau} \end{bmatrix} = \begin{bmatrix} \hat{f}^{\text{cl}}(\tau, x) \\ \frac{1}{T^\star} \end{bmatrix}, & x \in \mathcal{X} \\ \begin{bmatrix} x^+ \\ \tau^+ \end{bmatrix} = \begin{bmatrix} \Delta(x^-) \\ 0 \end{bmatrix}, & x^- \in \mathcal{X} \cap \mathcal{S} \end{cases} \tag{3.14}$$

where

$$\hat{\Gamma}(\tau, x) \coloneqq \text{col}(\Gamma_1(\tau, x_1, \psi_1, \dot{\psi}_1), \Gamma_2(\tau, x_2, \psi_2, \dot{\psi}_2)), \tag{3.15}$$

represents the distributed controllers, and

$$\begin{aligned} f^{\text{cl}}(\tau, x) &\coloneqq f(x) + g(x)\,\Gamma(\tau, x), \\ \hat{f}^{\text{cl}}(\tau, x) &\coloneqq f(x) + g(x)\,\hat{\Gamma}(\tau, x), \end{aligned} \tag{3.16}$$

denotes the closed-loop vector fields with centralized and distributed feedback laws, respectively. For future purposes, we define the augmented states as $x_a \coloneqq \text{col}(x, \tau) \in \mathcal{X}_a \coloneqq \mathcal{X} \times \mathbb{R}_+$,

augmented vector fields as

$$f_a^{\mathrm{cl}}(x_a) \coloneqq \mathrm{col}(f^{\mathrm{cl}}(\tau, x), \frac{1}{T^\star}), \tag{3.17}$$
$$\hat{f}_a^{\mathrm{cl}}(x_a) \coloneqq \mathrm{col}(\hat{f}^{\mathrm{cl}}(\tau, x), \frac{1}{T^\star}),$$

augmented reset law as $\Delta^{\mathrm{cl}}(x_a) \coloneqq \mathrm{col}(\Delta(x), 0)$, and $\mathcal{S}_a \coloneqq \mathcal{S} \times \mathbb{R}_+$. We then denote the Poincaré maps for the closed-loop hybrid models in Equation (3.13) and Equation (3.14) by $P_a : \mathcal{S}_a \to \mathcal{S}_a$ and $\hat{P}_a : \mathcal{S}_a \to \mathcal{S}_a$, respectively.

Lemma 3.3. (Fixed Points): *Under Assumptions 2.3-3.2, $x_{af}^\star \coloneqq col(x_f^\star, 1)$ is a fixed point for the Poincaré maps, i.e., $P_a(x_{af}^\star) = \hat{P}_a(x_{af}^\star) = x_{af}^\star$, where $x_f^\star \coloneqq x^\star(1) = \overline{\mathcal{O}} \cap \mathcal{S}$.*

The proof is immediate according to the construction procedure. From Lemma 3.3, $\mathcal{O}_a \coloneqq \{(x, \tau) \,|\, x = x^\star(\tau), 0 \leq \tau < 1\}$ is a periodic orbit for the closed-loop hybrid systems in Equation (3.13) and Equation (3.14). Furthermore, let us denote the time evolution of the augmented state trajectory along $\overline{\mathcal{O}}_a$ (i.e., the closure of $\mathcal{O}_a$) by $x_a^\star(t)$ for $0 \leq t \leq T^\star$. We then define the Jacobian matrices of the augmented vector fields along $x_a^\star(t)$ by $A(t)$ and $\hat{A}(t)$, that is,

$$A(t) \coloneqq \frac{\partial f_a^{\mathrm{cl}}}{\partial x_a}(x_a^\star(t)), \tag{3.18}$$
$$\hat{A}(t) \coloneqq \frac{\partial \hat{f}_a^{\mathrm{cl}}}{\partial x_a}(x_a^\star(t)),$$

for $t \in [0, T^\star]$. We now make the following assumption that enables us to define a bounded relationship between the Jacobian matrices of the augmented vector fields.

Assumption 3.4. There are $L > 0$ and $\mu > 0$ such that for all $t \in [0, T^\star]$, $\|A(t)\|_2 \leq L$ and $\|A(t) - \hat{A}(t)\|_2 \leq \mu$.

Before we can study the variational equations, we present the following theorem that bypasses the issue of existence and uniqueness and focuses on the closeness of the solutions.

Theorem 3.5. [122, Theorem 3.4]: *Let $f(t, x)$ be piecewise continuous in t and Lipschitz in x on $[t_0, t_1] \times W$ with a Lipschitz constant L, where $W \subset \mathbb{R}^n$ is an open connected set. Let $y(t)$ and $z(t)$ be solutions of*

$$\dot{y} = f(t, y), \quad y(t_0) = y_0,$$

and

$$\dot{z} = f(t, z) + g(t, z), \quad z(t_0) = z_0,$$

such that $y(t)$, $z(t) \in W$ *for all* $t \in [t_0, t_1]$. *Suppose that*

$$\|g(t,x)\| \leq \mu, \quad \forall (t,x) \in [t_0,t_1] \times W,$$

for some $\mu > 0$. *Then,*

$$\|y(t) - z(t)\| \leq \|y_0 - z_0\| \exp[L(t-t_0)] + \frac{\mu}{L}\{\exp[L(t-t_0)] - 1\}, \tag{3.19}$$

$\forall t \in [t_0, t_1]$.

We can now present the following lemma that provides a bounded relationship between the trajectory sensitivity matrices of the centralized and distributed closed-loop ODEs. It makes use of the results presented in Theorem 3.5.

Lemma 3.6. (Variational Equations): *Under Assumptions 2.3-3.4, consider the variational equations* $\dot{\Phi}(t) = A(t)\,\Phi(t)$ *and* $\dot{\hat{\Phi}}(t) = \hat{A}(t)\,\hat{\Phi}(t)$ *for* $t \in [0, T^\star]$ *with* $\Phi(0) = \hat{\Phi}(0) = I$. *Let us define* $\varphi(t) \coloneqq \mathrm{vec}(\Phi(t))$ *and* $\hat{\varphi}(t) \coloneqq \mathrm{vec}(\hat{\Phi}(t))$, *where the "*vec*" denotes the vectorization operator. Then, there is* $\delta > 0$ *such that for all* $t \in [0, T^\star]$,

$$\|\varphi(t) - \hat{\varphi}(t)\|_2 \leq \frac{\mu\,\delta}{L}\left(\exp(L\,t) - 1\right). \tag{3.20}$$

Proof. From properties of the vectorization operator,

$$\dot{\varphi} = \mathrm{vec}(A(t)\,\Phi) = (I \otimes A(t))\,\mathrm{vec}(\Phi) = (I \otimes A(t))\,\varphi, \tag{3.21}$$

and similarly, $\dot{\hat{\varphi}} = (I \otimes \hat{A}(t))\,\hat{\varphi}$, where "$\otimes$" denotes the Kronecker product. Furthermore, we can show that

$$\begin{aligned} \|I \otimes A(t)\|_2 &= \|A(t)\|_2 \leq L, \\ \|I \otimes A(t) - I \otimes \hat{A}(t)\|_2 &= \|A(t) - \hat{A}(t)\|_2 \leq \mu, \end{aligned} \tag{3.22}$$

for all $t \in [0, T^\star]$. These properties together with $\varphi(0) = \hat{\varphi}(0) = \mathrm{vec}(I)$, boundedness of the solutions, i.e., $\exists\ \delta > 0$ such that $\|\varphi(t)\|_2 \leq \delta$ and $\|\hat{\varphi}(t)\|_2 \leq \delta$ for $t \in [0, T^\star]$, and applying Theorem 3.5 to the ODEs complete the proof. □

Note that the augmented Poincaré maps, P_a and $\hat{P}_a$ are an $n+1$-dimensional system. Our group's previous results allowed the reduction of the exponential stability problem of the corresponding fixed points into an n-dimensional discrete-time system. This transformation is enabled by the following theorem.

Theorem 3.7. [149, Theorem 1]: *Suppose there exists a time-varying feedback law as in Equation (2.21), where in the system evolves per the time-based phasing variable as in Assumption 2.3 and resets on the switching manifolds as* $\tau^+ = 0$. *Then the following statements hold:*

1) *The Jacobian linearization of the Poincaré map for the augmented system takes the following structure*

$$\frac{\partial P_a}{\partial x_a}(x_{af}^\star) = \begin{bmatrix} \frac{\partial P_x}{\partial x}(x_f^\star, 1) & 0 \\ \frac{\partial P_\tau}{\partial x}(x_f^\star, 1) & 0 \end{bmatrix}, \tag{3.23}$$

here, $P_a = \mathrm{col}(P_x, P_\tau)$ *is a decomposition corresponding to* (x, τ)*, i.e.,* $P_x \in \mathbb{R}^n$ *and* $P_\tau \in \mathbb{R}$*.*

2) *The periodic orbit* $\mathcal{O}$ *is orbitally exponentially stable for the close-loop hybrid system, if and only if,*

$$\left| eig \left\{ \frac{\partial P_x}{\partial x}(x_f^\star) \right\} \right| < 1. \tag{3.24}$$

We are now in a good position to present one of the *main results* of this chapter through the following theorem.

Theorem 3.8. (Upper Bound on the Norm of the Difference of the Jacobian Matrices): *Under Assumptions 2.3-3.4, there is a constant* $\beta > 0$ *such that*

$$\left\| \frac{\partial P_a}{\partial x_a}\left(x_{af}^\star\right) - \frac{\partial \hat{P}_a}{\partial x_a}\left(x_{af}^\star\right) \right\|_2 \leq \frac{\mu\,\beta}{L}\left(\exp(L\,T^\star) - 1\right). \tag{3.25}$$

Proof. From [17, Theorem 2] and Theorem 3.7, the Jacobian linearization of the Poincaré return maps can be given by

$$\begin{aligned} \frac{\partial P_a}{\partial x_a}(x_{af}^\star) &= \Pi\,\Phi(T^\star)\,\frac{\partial \Delta_a}{\partial x_a}(x_{af}^\star), \\ \frac{\partial \hat{P}_a}{\partial x_a}(x_{af}^\star) &= \Pi\,\hat{\Phi}(T^\star)\,\frac{\partial \Delta_a}{\partial x_a}(x_{af}^\star), \end{aligned} \tag{3.26}$$

where Π is the saltation matrix, defined by

$$\Pi := I - \frac{1}{\frac{\partial s_a}{\partial x_a}(x_{af}^\star)\, f_a^{\mathrm{cl}}(x_{af}^\star)}\, f_a^{\mathrm{cl}}(x_{af}^\star)\,\frac{\partial s_a}{\partial x_a}(x_{af}^\star). \tag{3.27}$$

Here, we assume that the augmented switching manifold $\mathcal{S}_a$ can be expressed as the zero-level set of the switching function $s_a(x_a)$. From Lemma 3.3, $f_a^{\mathrm{cl}}(x_{af}^\star) = \hat{f}_a^{\mathrm{cl}}(x_{af}^\star)$, and hence, Π is indeed the saltation matrix for both Jacobian matrices. From the properties of the vectorization operator, there is a scalar $\gamma > 0$ such that

$$\|\Phi(T^\star) - \hat{\Phi}(T^\star)\| \leq \gamma \|\varphi(T^\star) - \hat{\varphi}(T^\star)\|, \tag{3.28}$$

which in combination with Lemma 3.6 and norm properties results in

$$\left\| \frac{\partial P_a}{\partial x_a}(x^\star_{af}) - \frac{\partial \hat{P}_a}{\partial x_a}(x^\star_{af}) \right\| \leq \|\Pi\| \left\| \frac{\partial \Delta_a}{\partial x_a}(x^\star_{af}) \right\| \frac{\mu\,\delta\,\gamma}{L}(\exp(L\,T^\star) - 1). \tag{3.29}$$

Finally, defining $\beta \coloneqq \delta\,\gamma\,\|\Pi\|\|\frac{\partial \Delta_a}{\partial x_a}(x^\star_{af})\|$ completes the proof. □

3.5 Distributed Control Synthesis

In order to synthesize distributed controllers, we look for some output matrices E_i, $i \in \{1, 2\}$ in Equation (3.9) to exponentially stabilize $\mathcal{O}$ via proposed local controllers. For this purpose, let us denote the columns of E_i by ξ_i, that is, $\xi_i \coloneqq \text{vec}(E_i)$. The local controllers Γ_i in (3.12) can be then represented by *parameterized feedback laws* $\Gamma_i(\tau, x_i, \psi_i, \dot{\psi}_i, \xi_i)$. We further define the control parameters as $\xi \coloneqq \text{col}(\xi_1, \xi_2)$. This allows us to parameterize the Jacobian matrix of the Poincaré map by ξ which can be denoted by $\frac{\partial \hat{P}_a}{\partial x_a}(x^\star_{af}, \xi)$. The exponential stabilization problem then consists of finding the controller parameters ξ such that the eigenvalues of $\frac{\partial \hat{P}_a}{\partial x_a}(x^\star_{af}, \xi)$ strictly lie inside the unit circle. We then set up a NLP to look for stabilizing parameters ξ as follows:

$$\begin{aligned} \min_{\xi} \quad & \frac{1}{2}\|\xi\|_2^2 \\ \text{s.t.} \quad & \left| \text{eig}\left(\frac{\partial \hat{P}_a}{\partial x_a}\,(x^\star_{af}, \xi) \right) \right| < 1. \end{aligned} \tag{3.30}$$

Here, we search for the minimum 2-norm controller parameters to make modified outputs in Equation (3.9) close to the decomposed ones (3.8) while imposing the exponential stability constraint. Our previous work in [12] synthesized distributed controllers by looking for both output matrices C_{ii} and E_i in Equation (3.9) which may have a significant number of decision variables and hence, it may result in a large-scale and computationally intensive NLP. The proposed approach, however, significantly reduces the number of parameters to be optimized for the synthesis of distributed controllers. More specifically, we do *not* search for C_{ii} and only focus on E_i.

3.6 Summary

This chapter accomplishes two things: 1) Demonstration of the synthesis of decomposition based distributed controllers, and 2) Mathematically establishing methods that simplify the search for stabilizing, optimal local control parameters. We presented the dynamic decomposition and the subsequent control decomposition that results in two comparatively simpler

local control problems. We then showed that there exists an upper bound on the norm of the difference of Jacobian Matrices of centralized and distributed controllers thus significantly simplifying the search for optimal local control parameters. With this knowledge, we then demonstrated the synthesis of distributed controllers by setting up a small scale NLP that searches for the local control parameters. In the next chapter, we will validate the proposed theory in this chapter with the help of an asymmetric system that by nature consists of a decentralized structure: the combination of human and a lower extremity prosthesis.

Chapter 4

Applications to Cooperative Human-Robot Locomotion

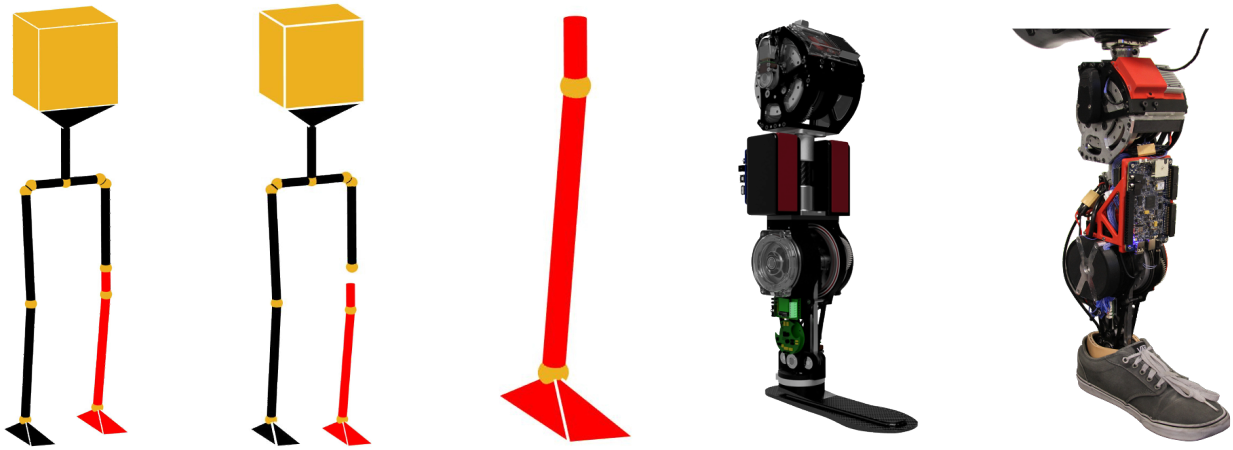


Figure 4.1: Illustration of the cooperative locomotion problem considered: (a) human-robot model visualized as a single agent, (b) human-robot model decomposed into two sub systems, (c) closer look at the lower-extremity prosthesis abstracted as a stick-figure model. (d) CAD model of the robotic prosthesis considered in this work and, (e) assembled life-size robotic prosthesis, PRleg [71].

This chapter seeks to *validate* the theoretical contributions described in Chapter 3 through the help of a case-study on human-robot cooperative locomotion model (see Figure 4.1). After summarizing the contributions of this chapter in Subsection 4.1.1, a brief overview (supplementary to the literature review in Chapter 1) of the existing approaches that addresses powered prosthetic locomotion is presented in Subsection 4.1.2. Building on this, we highlight the *fundamental gap* that necessitates the use of the proposed distributed controller framework, further in this section. In Section 4.2 a complete model description is provided, Section 4.3 will then formulate a NLP-based TO problem for cooperative walking as a single system. Section 4.4 will systematically decompose the system into two asymmetric subsystems and the results of rigorous numerical simulations undertaken are presented in Section 4.5. Further, for convenience, a comprehensive video[1] accompanying the paper[2]

[1]Video: https://youtu.be/MiGhjggJhpI
[2]Paper: https://doi.org//10.1109/LCSYS.2020.3045669

which compiles animations of all the numerical simulations is made available. Finally, this chapter concludes with a brief discussion in Section 4.6.

4.1 Introduction

4.1.1 Contributions of this Chapter

The specific outcomes of this chapter are summarized with the following contributions: (1) We numerically validate the analytical framework established in Chapter 3 by designing distributed controllers for multi-domain and *collaborative* human-robot locomotion with a powered knee-ankle prosthetic leg and nontrivial feet, having a total of 19 DOF (see Figure 4.1). We show that the proposed approach can effectively synthesize local controllers for this comprehensive model of locomotion. In particular, with this approach we were able to reduce the number of local control parameters to be tuned and optimized to almost 5.2% of that in [12]. (2) We lay support to the claim made earlier about the decentralized architecture "providing inherent robustness" by a comparative numerical analysis between the centralized and distributed controllers utilizing this *asymmetric* platform. In particular, we inject a disturbance into prosthesis velocity measurements and demonstrate that the distributed controller restricts this disturbance to the prosthetic subsystem. However, the centralized controller enables the propagation of the disturbance into the human subsystem as well. Furthermore, the amplitude of the propagated noise in centralized controller is significantly higher than that of the distributed controller.

4.1.2 Related Work and Motivation

Without the need to exploit inherent symmetries (e.g., Quadrupeds: envisioned as a set of bipeds or Bipeds: envisioned as a set of monopods), we choose to demonstrate the performance of the distributed controllers on a naturally disjoint system that arises by the extrinsic coupling of powered prosthetic legs with lower-extremity amputees. The notion of decentralization/distributed control is not new to the problem of powered prosthesis. In [200], impedance, [70] torque, [10] kinematic, based reference trajectory tracking, decentralized feedback controllers were implemented. However, these linear control method necessitates different control parameters be tuned at different time periods to account for the nonlinear dynamics in locomotion. This results in a "finite state machine" [204], requiring clinicians to spend significant time and effort tuning each controller to a patient [190] and risks instability when perturbations invoke the wrong controller at the wrong time [92, 96, 146].

The limitations of sequential control method could possibly be addressed by the nonlinear controllers used in dynamic walking robots [14, 16, 25, 30, 55, 101, 131, 144, 145, 147, 174, 177, 183, 184, 212, 213]. However, their centralized architectures require instrumenting the

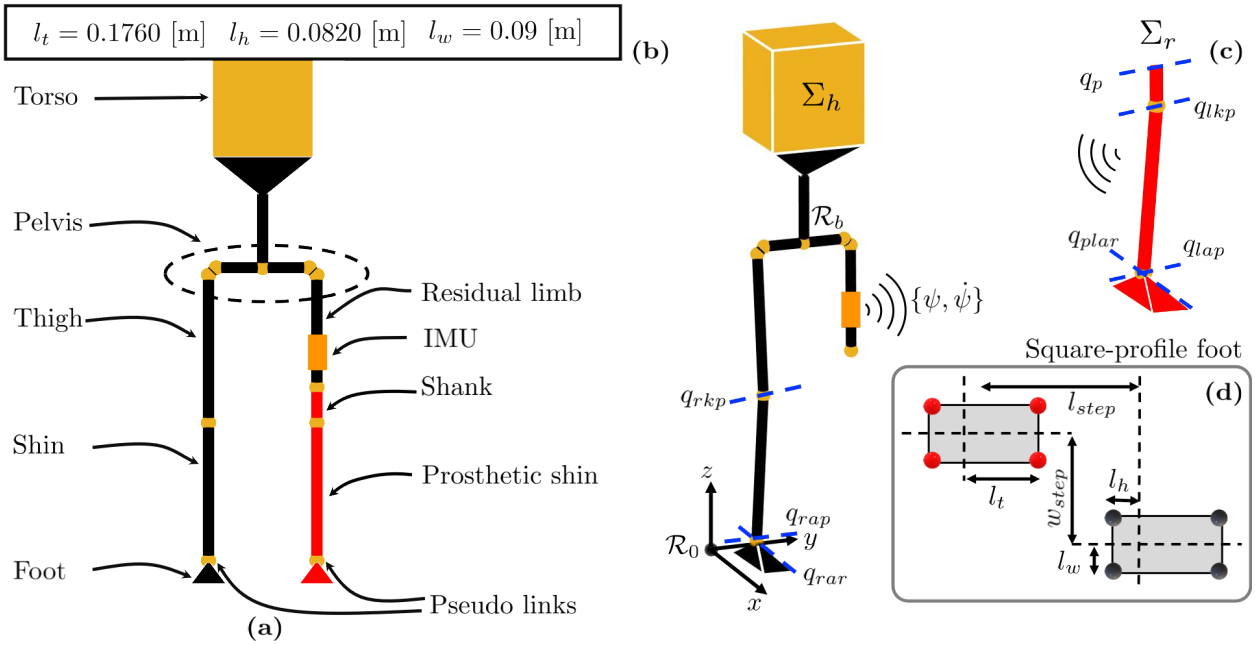


Figure 4.2: (a) Illustration of the individual links in the human-prosthesis model, (b) schematic presenting the joint notation followed for the human subsystem, (c) illustration of the joint notation adopted for the prosthetic subsystem, and, (d) description of the foot profile assumed and the corresponding illustration of link lengths and step constraints used for TO.

human body with a profusion of sensors to coordinate the interconnected subsystems. This poses the problem of practicality as it is clinically infeasible for users of prosthetic legs to wear sensors at all their intact joints, besides causing discomfort for human users. Furthermore, there is no provision to consider the interaction forces between the two interconnected systems under the centralized architecture. To compensate for interactions, [96] made use of high-gain controllers, [146, 172] made use of expensive force sensors to measure and cancel the interaction forces in linear methods. These *key roadblocks* to cooperative human-machine walking can be addressed by the systematically designed distributed nonlinear feedback controllers in Chapter 3. We show in the following sections, that the use of this method does not necessitate any expensive sensors except a light-weight Inertial Measurement Unit (IMU) to avail the prosthetic subsystem with the orientation and the corresponding change of the orientation of the human subsystem in the global frame.

4.2 Physical Model

We consider a 3-D, lower-extremity amputee with the residual limb being the left leg (see Figure 4.2 (a)). While arms play a critical role in balancing humans/humanoid robots when executing complex dynamic maneuvers but resort to an inactive role otherwise, and since

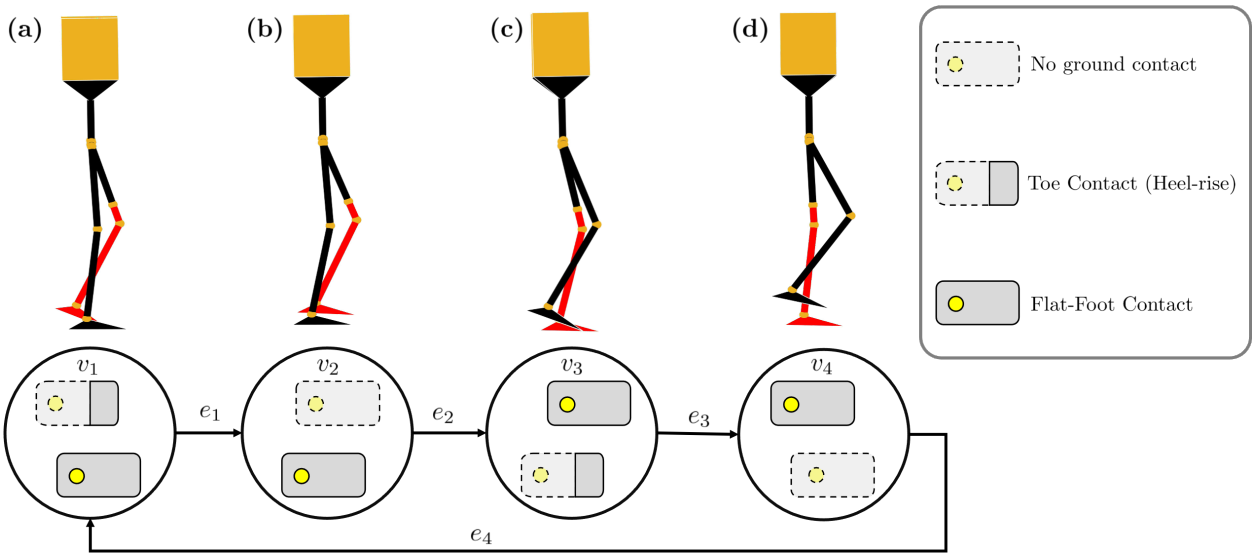


Figure 4.3: Multi-contact hybrid model of cooperative human-prosthesis locomotion: a 4-domain approximation of human-walking that captures the heel-rise phase. We denote $\{v_1, v_3\}$ as push-off phases and $\{v_2, v_4\}$ as flat-footed phases.

our interests lie in *robust cooperative walking*, which does not entail any extreme maneuvers, in our abstraction arms are not modeled. For the human model, we consider a torso rigidly attached to the pelvis (see Figure 4.2 (b)). The pelvis encompasses a 3 DoF hip joint at each leg that accounts to the hip yaw, hip pitch and hip roll which are encoded by $\{q_{rhy}, q_{rhp}, q_{rhr}\}$ and $\{q_{lhy}, q_{lhp}, q_{lhr}\}$ for the right and left hip joints, respectively. The right branch is further extended by the thigh link which connects the hip yaw joint and the 1 DoF knee joint, denoted by q_{rkp}. The shin link bridges the gap between knee and the 2 DoF ankle joint: ankle pitch, q_{rap}, and ankle roll, q_{rar}. The right branch finally terminates in the foot link and with the intention to preserve the anthropomorphism in our abstraction, this link is modeled as a square-profile foot (see Figure 4.2 (d)) as opposed to the simplification into a point foot (refer to Chapter 2). Alternatively, the left branch of the human model terminates in the residual limb (thigh).

Correspondingly, we consider a 2 DoF powered prosthesis robot developed in [71]. A passive joint, q_p is attached to the prosthetic shank to model the coupling between the human and the robot (see Figure 4.2). On the other end, the shank terminates at the prosthesis knee joint, q_{lkp} which is then connected to the ankle pitch joint, q_{lap} via a prosthetic shin. Further, a passive ankle roll joint, q_{plar} is modeled to account for the passive inversion/eversion introduced by the split toe design of the physical prosthetic foot [9]. We assume a rigid spring with stiffness, $k = 75$ and damping, $c = 0.5$, will model the effect of the split toe design. To complete the description of the floating-base model, we attach a frame $\mathcal{R}_b$ to the geometric center of the pelvis. This introduces 6 DoFs: $\{x, y, z, r, p, y\}$ that will enable us to precisely locate the human-robot model with respect to the global frame, $\mathcal{R}_0$.

Here, x, y, z encode the absolute position and r, p, y correspond to the absolute roll, pitch and yaw angles, respectively. We are now in a position to form our minimal coordinate representation, $q(t) \in \mathbb{R}^{19}$ that accurately describes the state of the model at any point in time. Correspondingly, the state vector is chosen as $x(t) \coloneqq \text{col}(q, \dot{q}) \in \mathbb{R}^{38}$. Note that the actuated joints of the model are given in $q_{\text{act}}(t) \in \mathbb{R}^{11}$ and it shapes the control vector, $u(t) \in \mathbb{R}^{11}$. The vectors $q(t)$ and $q_{\text{act}}(t)$ are expanded below for convenience.

$$q(t) \coloneqq \text{col}\big(x, y, z, r, p, y, q_{rar}, q_{rap}, q_{rkp}, q_{rhy}, q_{rhr}, q_{rhp}, q_{lhp}, q_{lhr}, q_{lhy}, q_p, q_{lkp}, q_{lap}, q_{lar}\big),$$
$$q_{\text{act}}(t) \coloneqq \text{col}\big(q_{rar}, q_{rap}, q_{rkp}, q_{rhy}, q_{rhr}, q_{rhp}, q_{lhp}, q_{lhr}, q_{lhy}, q_{lkp}, q_{lap}\big).$$

Table 4.1: Physical Parameters Utilized for the Cooperative Locomotion Model

	Human Parameters			Prosthesis Parameters	
Segment	Mass [kg]	Length [m]	Segment	Mass [kg]	Length [m]
Torso	40.82	0.6358	Shank	0.9371	0.0959
Hip	1.5140	0.12	Shin	3.1712	0.3733
Thigh	9.0460	0.42	Talus[3]	0.125	0
Thigh (residual limb)	8.209	0.3241	Foot	1.8567	0.2580
Shin	5.479	0.3733	-	-	-
Talus[3]	0.125	0	-	-	-
Foot	1	0.2580	-	-	-
Total	**66.1930**	-	-	**6.09**	-

This makes our $n = 19$ and $m = 11$, where m and n assume the same definitions provided in Section 1.6.1. Furthermore, it is imperative to use precise values corresponding to the link lengths(l), masses(m_g), inertia's ($\mathbf{I}$), etc such that physically feasible motions can be realized. While the parameters for the prosthesis are straightforward and extracted from [71], the parameters for the human model are calculated from the mass distribution for an average human male given in [65] and are readily found in Table. 4.1.

4.3 Gait Optimization

With the kinematic and dynamic parameters in hand, obtaining the corresponding models at this point of the discussion becomes trivial and will not be discussed again in this chapter (refer to Chapter 2 for the dynamic model). Our next step, that aligns with the discussion in Section 1.6.1 is the search for a feasible trajectory. This, in turn, translates to finding a

[3]Pseudo Link added to accommodate two DoFs at the ankle joint. It bridges the gap between ankle roll joint and ankle pitch joint.

feasible orbit, $\mathcal{O}$ (refer to Chapter 2). This particular section is presented in a tutorial style, i.e., we go over all the minute details such that it can act as a reference for anyone who would like to recreate the TO problem for the model in this prelim document or any other legged robot model. We make use of FROST [106], primarily because of its direct support towards hybrid dynamic systems, as discussed in Section 1.6.2. The formulation remains the same if the reader were to use any other toolbox. The problem formulation and the justification for the specific choice of domains is then given in Subsection 4.3.1. The equality and inequality constraints are given in Subsection 4.3.2. The reasoning behind the choice of cost function is then provided in Subsection 4.3.3. Finally, we provide the performance metrics that resulted in an optimal gait for this specific problem in Subsection 4.3.4.

4.3.1 Problem Formulation

In order to capture the most relevant nuances of human walking, we make use of a hybrid formulation with 4 domains. Specifically, for human walking, the *heel-rise* plays a significant role in generating energy efficient walking gaits [120]. Hence, we preserve this behavior in our formulation and capture the "push-off" phases in domain, v_1 and domain, v_3 (see Figure 4.3). The remaining domains, v_2 and v_4 are termed as "single-stance" phases where in, the swing leg is undergoing an aerial phase and the stance leg is planted flat into the ground (see again, Figure 4.3). Note that the focus of this exercise is to demonstrate the effectiveness of the developed distributed controllers, hence in this work we restrict ourselves to 4 domains. The extension to more domains is straightforward if one seeks to capture the "heel strike" and "toe strike" behaviors instead of the "flat-footed" impacts with the ground (in domain v_2 and v_4), as considered in this work.

4.3.2 Constraints

As highlighted, in multiple instances through out this document the optimization problem is subjected to equality and inequality constraints, noted as $C_{eq} = 0$ and $C_{ineq} \leq 0$, respectively. One can however, express an inequality constraint as an equality constraint with the help of slack variables or on the contrary express equality constraints as inequality constraints. Therefore, in what follows below, we highlight the most meaningful constraints imposed on the problem considered in Subsection 4.3.1 without concerning ourselves with the segregation of the constraints.

Impact Velocity Constraints: In situations where the swing foot impact the ground too hard, the system destabilizes and fails to converge to a solution. In order to avoid such scenarios, we enforce a constraint on impact velocities to lie within a reasonable range. For example, if $v_x^{max}, v_y^{max}, v_z^{max} > 0$ are the maximum allowable impact velocities along the x, y and z-axis respectively, then the swing foot velocity, $\dot{p}_{nsf}$ should satisfy the following constraints: $|\dot{p}^x_{nsf}(q^-, \dot{q}^-)| \leq v_x^{max}$, $|\dot{p}^y_{nsf}(q^-, \dot{q}^-)| \leq v_y^{max}$ and $|\dot{p}^z_{nsf}(q^-, \dot{q}^-)| \leq v_z^{max}$, respectively.

Foot Constraints

Contact Constraints: As evident from Figure 4.3, our formulation makes use of we have two particular contact conditions. In push-off phases, a combination of flat-foot (planar) and toe (line) contacts, accounting for a total of 11 constraints. This translates to

$$H(q,\dot{q},u) = \begin{bmatrix} A_{\lambda_{\text{planar}}}(q)^\top \lambda_{\text{planar}}(q,\dot{q},u) \\ A_{\lambda_{\text{line}}}(q)^\top \lambda_{\text{line}}(q,\dot{q},u) \end{bmatrix} > 0, \tag{4.1}$$

where $A_{\lambda_{\text{planar}}}(q)^\top \lambda_{\text{planar}}(q,\dot{q},u)$ groups the constraints under the planar contact column in Table 2.1 and $A_{\lambda_{\text{line}}}(q)^\top \lambda_{\text{line}}(q,\dot{q},u)$ groups the constraints under the line contact column in Table 2.1. On the contrary, when the robot is observing single-stance phases, we only have a total of 6 constraints. Additionally, we introduce an unilateral constraint, h that parameterizes the distance to impact. Together the holonomic and the unilateral constraints for single-stance domains are expressed as

$$H(q,\dot{q},u) = \begin{bmatrix} A_{\lambda_{\text{planar}}}(q)^\top \lambda_{\text{planar}}(q,\dot{q},u) \\ h(q) \end{bmatrix} > 0. \tag{4.2}$$

Flat foot Constraint: We constrain the foot that is in aerial phase to be flat with respect to the ground plane in order to ensure that on impact the foot is flat on the ground. This translates to imposing the orientation of the non-stance foot frame denoted by, p_{nsf} to be zero, i.e., $\{p^r_{\text{nsf}}, p^p_{\text{nsf}}, p^y_{\text{nsf}}\}$, should be zero. Additionally, we also impose maximum allowable foot clearance constraint on the non-stance foot, i.e., $p^z_{\text{nsf}} \leq p^{z,\max}_{\text{nsf}}$.

Bounds

We further add bounds on the admissible positions, velocities, and torques as follows

$$q_i^{\min} \leq q_i \leq q_i^{\max},\ \dot{q}_i^{\min} \leq \dot{q}_i \leq \dot{q}_i^{\max},\ u_i^{\min} \leq u_i \leq u_i^{\max}. \tag{4.3}$$

4.3.3 Cost Function

Before we pass on the problem to the NLP solvers, one step remains which is the choice of the cost function. We make use of the torque-squared as our cost function given, in Equation (4.4) with no terminal cost, i.e., $p = 0$.

$$\mathcal{J} = p + \int_0^T \left(\sum_{i=1}^{m} u_i^2(t)\, \mathrm{dt} \right). \tag{4.4}$$

Table 4.2: TO Parameters for the Cooperative Human-Robot Locomotion Problem

# DoFs	Domains	Variables	Inequality Constraints	Equality Constraints	Time (s)
19	4	5562	6563	316	124.836

This particular choice is made as this cost function tends to provide smooth solutions. More particularly, a smooth solution indicates that a piece-wise polynomial spline can be employed to approximate the solution, which implies that the NLPs have a better chance of converging. Moreover, a smooth solution enables better control for the robot. Besides, this cost-function distances the solutions from high, undesirable torques thus resulting in solutions that respect the torque limitations on the actual hardware. Alternatively, one can also use the cost of transport (CoT) as a cost function that operates on the minimizing the mechanical CoT in a gait, which is defined as the ratio of the mechanical work done by the actuators to the total distance traveled in one step.

4.3.4 NLP via FROST

Performance Metrics: We formulate an one-step optimization problem that follows the sequence described in Figure 4.3. The optimization problem was then formulated as a NLP by FROST that employs the IPOPT solver on a computer that posses an AMD Ryzen 7 2700X, eight-core processor with a max clock speed of 3.7 (GHz) and a 64 (GB) RAM. The number of decision variables, constraints along with the performance metrics are readily available in Table 4.2. We report that with a warm start, i.e., properly seeded initial conditions, the NLP was solved in 2 minutes, 4 seconds.

4.4 System Decomposition

In this section we will construct the problem such that it is conducive to the proposed distributed control framework (refer to Chapter 3) by decomposing the system into two subsystems. The model, as highlighted in Section 4.2 has two natural subsystems: the human subsystem and the prosthetic subsystem that will be interchangeably refereed to as Σ_h, and, Σ_r, respectively from here on (see Figure 4.2).

Assumption 4.1. (Global Variables from an IMU): Due to the extrinsic nature of coupling in the problem considered, and under the assumption that the human user is always aware of the prosthesis at all times, to close the loop, we utilize an IMU that provides us with absolute orientation, $\psi : \{r_{\text{imu}}, p_{\text{imu}}, y_{\text{imu}}\}$ and absolute velocity, $\dot{\psi} : \{\dot{r}_{\text{imu}}, \dot{p}_{\text{imu}}, \dot{y}_{\text{imu}}\}$. This will enable orientational awareness to the prosthesis that is considered *essential and minimal global information* required to stabilize the cumulative system.

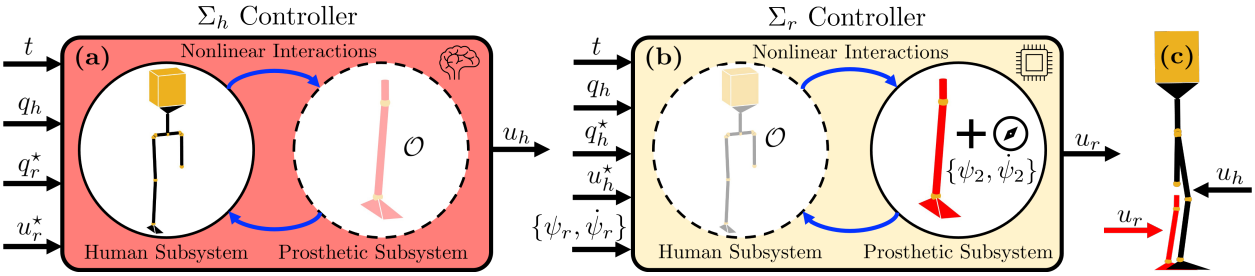


Figure 4.4: illustration of the synthesized distributed controller for the cooperative-human locomotion problem. (a) Controller block for the human subsystem, Σ_h, that assumes the parameters for the prosthetic subsystem, Σ_r, is on the orbit and calculates the control action, u_h. (b) Controller block for the prosthetic subsystem, Σ_r that assumes the parameters for the human subsystem, Σ_h, is on the orbit and calculates the control action, u_r. (c) Distributed control actions that are calculated concurrently and applied to the cumulative model.

This decomposition will result in two different state vectors $q_r(t) \in \mathbb{R}^4$, $q_h(t) \in \mathbb{R}^{15}$ for Σ_r and Σ_h, respectively and are given below

$$q_r(t) \coloneqq \mathrm{col}\big(q_p, q_{lkp}, q_{lap}, q_{lar}\big),$$
$$q_h(t) \coloneqq \mathrm{col}\big(x, y, z, r, p, y, q_{rar}, q_{rap}, q_{rkp}, q_{rhy}, q_{rhr}, q_{rhp}, q_{lhp}, q_{lhr}, q_{lhy}\big).$$

Note the corresponding change in the control vectors, $u_r(t) \in \mathbb{R}^2$ and $u_h(t) \in \mathbb{R}^9$.

Remark 4.2. We remark that the local controller for the Σ_r has access to *at most* 14 state variables. This is a direct result of its 8 internal state variables, $\{q_r(t), \dot{q}_r(t)\}$ and the availability of 6 state variables from the IMU, $\{\psi_r, \dot{\psi}_r\}$. However, Σ_h has access to *only* its 30 state variables, $\{q_h, \dot{q}_h\}$ *but not* the IMU state variables, $\{\psi_h, \dot{\psi}_h\} = 0$. This follows directly from Assumption 4.1.

The entire mechanical system has 8 DoFs during the push-off phases, $\{v_1, v_3\}$ and 13 DoFs during the flat-footed phases, $\{v_2, v_4\}$. The calculation is trivial from the following equation $m_v = n - n_v^h$, where m_v is the number of DoFs and n_v^h is the number of holonomic constraints in that particular domain. This dictates the number of allowed outputs, y such that we have non trivial zero dynamics, i.e., $\dim(\mathcal{Z}_v) \neq 0$. Following this restriction, we choose $n_v^h = 7$ for $\{v_1, v_3\}$ and $n_v^h = 11$ for $\{v_2, v_4\}$. We can now proceed to decompose the outputs y into y_h and y_r such that $\dim(y_h) = 6$ and $\dim(y_r) = 1$ and the corresponding decoupling matrices become $A_h \in \mathbb{R}^{6\times 9}$, $A_r \in \mathbb{R}^{1\times 2}$ for $\{v_1, v_3\}$. For the domains, $\{v_2, v_4\}$, the $\dim(y_h) = 9$, $\dim(y_r) = 2$, $A_h \in \mathbb{R}^{9\times 9}$, $A_r \in \mathbb{R}^{2\times 2}$. Next, we parameterize the local outputs for the robotic subsystem but not the human subsystem by output matrices E_2 over three domains. Here, we assume a local output regulating controller for the human subsystem as [12, 206]. In our parameterization, the Poincaré map is only parameterized by four parameters related to the robotic subsystem, i.e., $\xi \in \mathbb{R}^4$, whereas the synthesis problem in [12, Sec. VII.B] included 78 parameters for a model with less DoFs (i.e., 94.8% reduction in controller parameters). The

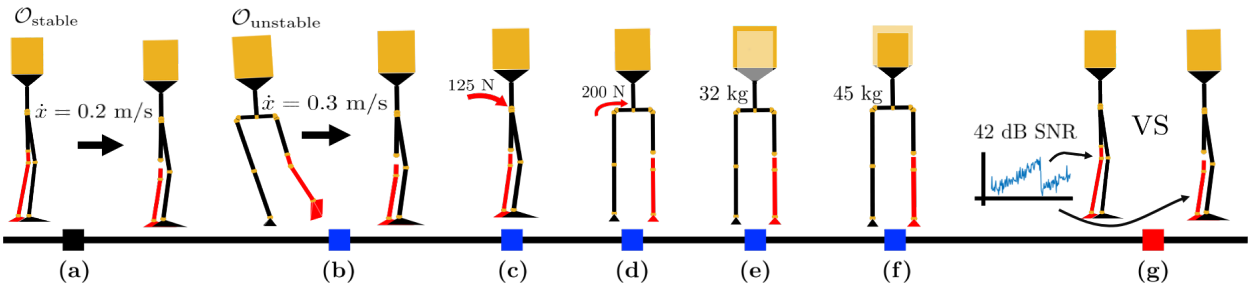


Figure 4.5: Illustration of an elaborate sequence of simulations undertaken to evaluate the performance of the proposed distributed controllers: the black node, (a) depicts the simulation as a result of decomposition from a stable orbit. The blue nodes, (b)-(e) denote the robustness analysis. The red node, (g) indicates the comparative analysis between centralized and distributed controllers when a white Gaussian noise is induced into the prosthesis joints.

local parameters for the robotic leg are then effectively optimized by the NLP (see Equation (3.30)) for exponential stability. Here the Poincaré section is taken at the beginning of domain v_1 and the Poincaré map is $15 = 2 \times 8 - 1$ dimensional as the mechanical system has 8 DoFs in domain v_1. In addition, we make use of a total of four parameters to parameterize the local outputs over three domains.

4.5 Numerical Simulations

We perform a series of simulations to validate the performance and demonstrate the robustness of the proposed distributed controllers (see Figure 4.5). First, we obtain a gait stabilized by the centralized controller and then proceed to develop a decomposition based distributed controller (see Figure 4.5 (a)). Next, we begin various robustness tests to rigorously evaluate the performance of the distributed controllers. For this objective, we first obtain an unstable gait as a result of a poorly tuned centralized controller and then synthesize the distributed controller with this orbit as a reference (see Figure 4.5 (b)). We further evaluate the robustness with the help of two simulations that periodically inject an external force of 125 (N) and 200 (N) along the x-axis and y-axis, respectively (see Figure 4.5 (c)-(d)). To next evaluate the performance in the presence of dynamic uncertainties, we vary the mass of the torso to 32 (kg) in one simulation and 45 (kg) in the other simulation (see Figure 4.5 (e)-(f)). Finally, we undertake a comparative study to evaluate the performance differences of centralized and distributed controllers, by inducing a Gaussian white noise of 42 (dB) signal-to-noise-ratio (SNR) in the Σ_r measurements (see Figure 4.5 (g)). Subsequent subsections will discuss in detail the results of these numerical simulations.

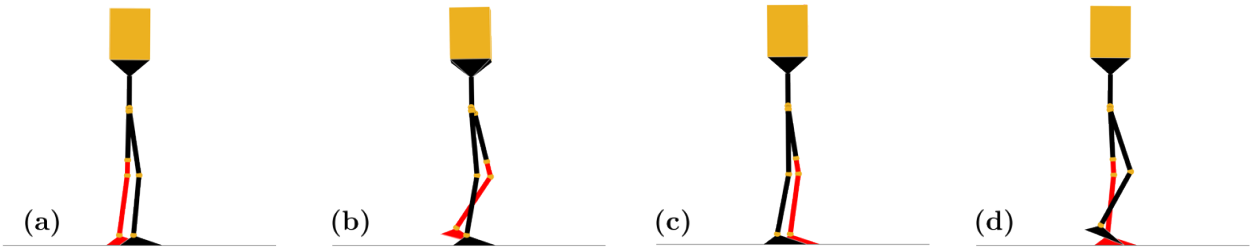


Figure 4.6: Snapshots captured at the 100^{th} step, illustrating stable walking as a result of the distributed controller synthesized from a stable orbit, $\mathcal{O}_{\text{stable}}$. (a) and (c) capture double stance phase, (b) and (d) capture heel-rise phase.

4.5.1 Asymptotically Stable Walking: Decomposition from a Stable Orbit

As a first step, we setup a TO problem as discussed in Section 4.3 for a gait with a forward speed of 0.2 (m/s). We include the outputs as our decision variables but discard them after the solution is obtained and synthesize our own centralized controller. For this particular simulation, we stabilize the gait with a centralized controller first and simulate it for 100 consecutive steps. We then extract the orbit at the 100^{th} step. The resultant orbit, $\mathcal{O}_{\text{stable}}$ is now the reference orbit that will be decomposed per the description in Section 4.4 to design a distributed controller. After the decomposition, we search for the local control parameters through the help of an optimization problem setup in Section 3.5. As highlighted earlier, four parameters are used to parameterize the local outputs over three domains. The four optimal values of E_2 are noted as $\{0.0571, 0.1017, -0.4186, 2.0232\}$ and correspond to the IMU pitch, roll, pitch, and yaw, respectively. The remaining parameters are presumed to be the same as used by the centralized controller. We report that the synthesized distributed controller was able to stabilize the entire system that was again simulated for 100 consecutive steps (see Figure 4.6). The complete sequence of steps is captured in one frame with the help

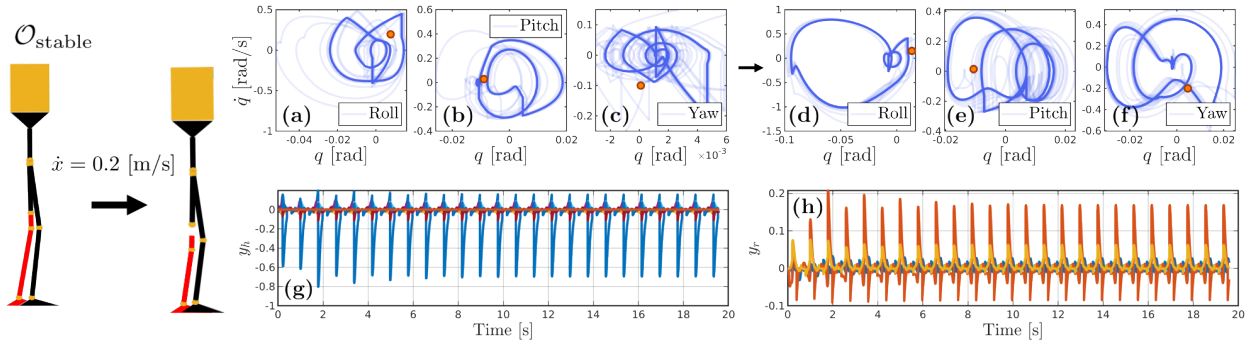


Figure 4.7: Limit cycles and outputs illustrating an asymptotically stable walking gait at 0.2 (m/s). The results are accumulated for cooperative human-robot walking over 100 steps.

of Figure 4.7. Figures 4.7 (a)-(c) show the limit cycles obtained from the simulation that utilizes a centralized controller. Figures 4.7 (d)-(f) demonstrate the limit cycles obtained as a result of simulating with a distributed controller stabilizing the system. In both instances, asymptotic stability is demonstrated with the convergence of the limit cycles to a periodic orbit. The behavior of the decomposed outputs in each subsystem, y_h and y_r are presented in Figures 4.7 (g) and (h), respectively. Further, as expected, outputs of both the subsystems are being driven to zero. The resultant distributed controller will be referred to as the "nominal distributed controller" for the remainder of this chapter.

4.5.2 Robustness Analysis

In order to materialize the developed distributed controller, performance has to be evaluated in non-ideal and uncertain scenarios. Towards that objective, this subsection will describe the details of all the simulations undertaken to demonstrate robustness, captured by the blue knot points in Figure. 4.5 (b)-(f).

Asymptotically Stable Walking: Decomposition from an Unstable Orbit

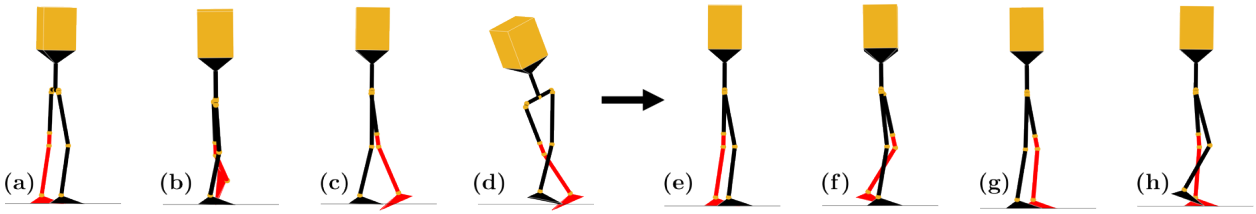


Figure 4.8: (a)-(d) Snapshots captured at 8^{th} step illustrating the destabilizing sequence triggered by an intentionally designed, poorly tuned, centralized controller. (e)-(h) Snapshots captured at 100^{th} step illustrating stable walking as a result of the distributed controller that is synthesized from an unstable orbit, $\mathcal{O}_{\text{unstable}}$, extracted from the simulation in (a)-(d).

For the first demonstration, we would like to highlight that the proposed decomposition based distributed control synthesis does not necessitate the presence of a stable orbit and simultaneously show that the gaits at higher speeds can also be stabilized. To achieve this objective, we reconstruct the TO problem in Section 4.3 for a gait with the desired forward speed, $\dot{x}_d$, of 0.3 (m/s). We then proceed to intentionally engineer an orbit that induces a destabilizing motion by poorly tuning the values of the H_0 matrix that governs the stability of the controller. The outcome of this exercise is a highly unstable gait, that can be observed from the limit cycles of the body's absolute roll, pitch, and, yaw (see Figure 4.9 (a) -(c)). The destabilizing motion sequence starts at step 7 and renders the system completely unstable in step 8 (see Figure 4.8 (a)-(d)). We then proceed to extract this orbit and denote the resultant orbit as $\mathcal{O}_{\text{unstable}}$. We again, systematically decompose the system as described in

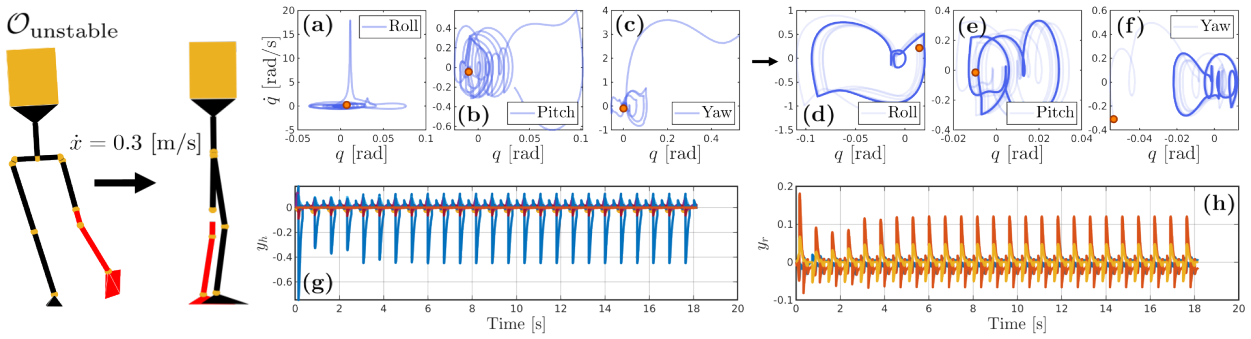


Figure 4.9: Limit cycles depicting convergence to an orbit as a result of distributed controllers synthesized from an *unstable* centralized orbit. The results are accumulated for cooperative human-robot walking over 100 steps.

Section 4.4. The parameters for the human subsystem are kept the same as the destabilizing centralized controller while synthesizing the distributed controller as described in Section 3.5. The four resultant optimal values of E_2 are noted as $\{-0.1026, 0.1124, 2.0233, 0.4066\}$, corresponding to the IMU pitch, roll, pitch, and yaw, respectively. We report that the corresponding distributed controller successfully stabilized the entire system that was simulated for 100 consecutive steps (see Figure 4.9 (e)-(h)). Figures 4.9 (d)-(f) capture the limit cycles obtained as a result of simulating with a distributed controller synthesized from an unstable orbit. The convergence of the limit cycles to the respective periodic orbits illustrates the asymptotic stability induced by the designed distributed controller. Furthermore, the decomposed outputs, y_h and y_r are being driven to zero (see Figure 4.9 (g)-(h)), as expected. This demonstrates that the synthesized distributed controller does *not* bank on the existence of a stable orbit.

Response to the Application of Persistent Forces

In this section, we will evaluate the performance of the nominal distributed controller in the presence of external forces. In particular, we performed two simulations that induce persistent forces of 125 (N) and 200 (N) at the pelvis, along the x-axis and y-axis, respectively. The persistent forces are specifically applied during the prosthetic leg's swing phase (v_2) for every step taken. We distinctively chose the domain, v_2, as this strains the prosthetic subsystem to respond to the external forces and not the human subsystem. For the first simulation, we report that the synthesized distributed controller was able to find and converge to a different periodic orbit that stabilizes the system (see Figure 4.10 (a)-(c)) despite the presence of an unplanned external force of 125 (N). Figure 4.10 (d) captures the calculated torques, u_r in the Σ_r for the first 25 steps of the simulated 100 steps. The nominal controller is again tested with a persistent force of 200 (N) along the positive y-axis. Through Figures 4.10 (e)-(f), we show that the distributed controller was once again able to find and converge to a different periodic orbit to stabilize the cumulative system, despite the application of

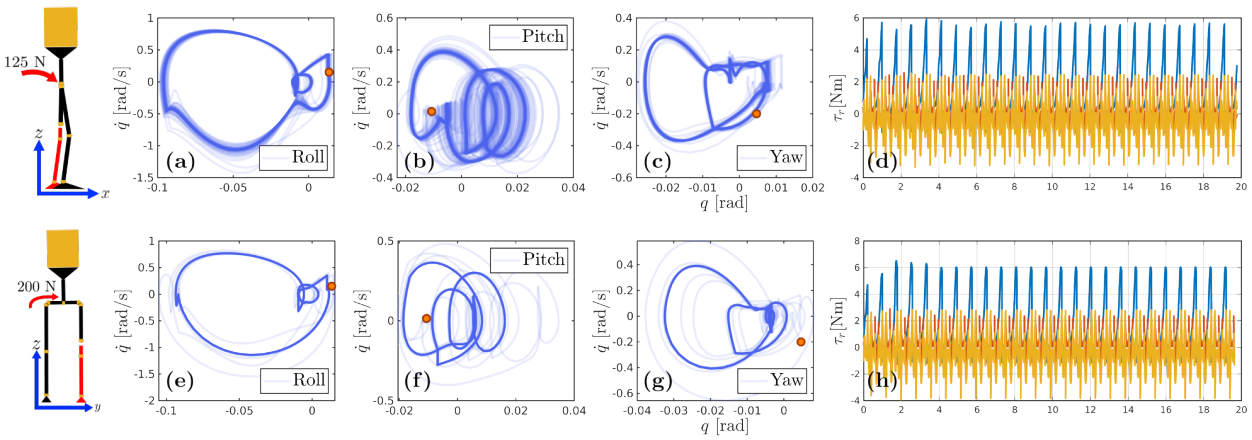


Figure 4.10: (a)-(c) Limit cycles depicting convergence to an orbit albeit an external disturbance force of 125 (N) applied along the x-axis. (d) Torques in the prosthesis subsystem. (e)-(g) Limit cycles depicting convergence to an orbit albeit an external disturbance force applied of 200 (N) along the y-axis. (h) Torques in the prosthesis subsystem.

persistent forces. Furthermore, the torques in the prosthesis subsystem for this set of simulation is captured in Figure 4.10 (h). The torques recorded in both the simulations are with in the reach of the rated motor torque in the PRleg, with a continuous torque of 2.8 (Nm) and a peak torque of 8.3 (Nm) [71], before the gear reduction is applied. We further note that for both scenarios, the distributed controller drives the outputs asymptotically to zero (not pictured here). This demonstrates that the nominal distributed controller is robust to an unknown, persistent force up to 125 (N) in the x-axis and 200 (N) in the y-axis.

Response to Dynamic Uncertainties in the System

We discussed in Section 4.2 that for the human subsystem, we adopted average parameters from [65]. However, if the distributed controller were to materialize, it should not have to be tuned for every user. To demonstrate that parameter tuning is *not* necessary every time the designated user changes, we perform a simulative case study wherein we introduce dynamic uncertainties into the system. More specifically, we vary the torso mass of the system but still make use of the nominal controller designed with a torso mass of 40.2 (kg). In one case study, we change the torso mass of the robot to 32 (kg) and simulate the system for 100 steps. Figures 4.11 (a)-(c) demonstrate the nominal controller successfully found and converged to a periodic limit cycle, despite the presence of dynamic uncertainties in the system. The torques for this simulation are captured in Figure 4.11 (d). Similarly, in a different simulation we vary the torso mass of the robot to 45 (kg), Figures 4.11 (e)-(g) illustrate that the nominal controller was again able to find and converge to a periodic orbit albeit the dynamic uncertainties induced into system. The corresponding torques are given in Figure 4.11 (h). Further, the decomposed outputs are asymptotically driven to zero, as

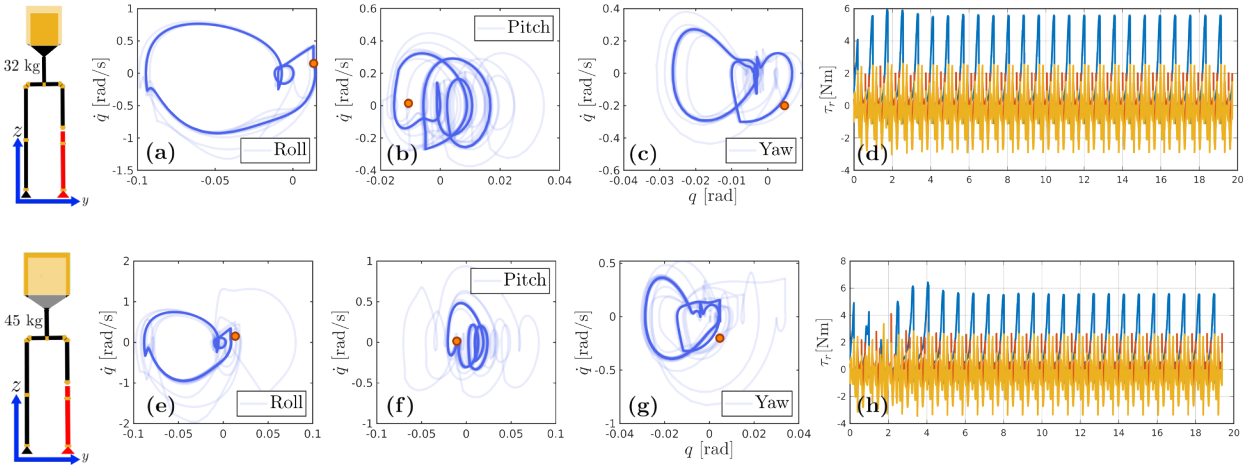


Figure 4.11: (a)-(c) Limit cycles depicting convergence to an orbit when the torso mass is varied to 32 (kg). (d) Torques in the prosthesis subsystem. (e)-(g) Limit cycles depicting convergence to an orbit albeit varying the torso mass to 45 (kg). (h) Torques in the prosthesis subsystem.

expected (not pictured here). This demonstrates the robustness of the developed distributed controllers in the face of dynamic uncertainties.

4.5.3 Centralized vs Distributed Controllers

In this section, we undertake a comparative analysis to draw meaningful differences between the *existing* centralized controllers and the *proposed* distributed controllers. To address this, we consider a common situation occurring in practice, that is the presence of noisy sensor measurements. In particular, we examine noisy signals in the velocity components, $\dot{q}$. Since, Σ_h is essentially modeling the human subsystem, Σ_h, it is not justifiable to induce any noise in to the components of $\dot{q}_h$. On the contrary, Σ_r has optical encoders that can be prone to noisy velocity outputs. To replicate this behavior, in our simulative case study, we induce white Gaussian noise with a SNR of 42 (dB) into the velocity components ($\dot{q}_r$) of Σ_r. Two simulations, with a total of 30 steps for the centralized and distributed controllers, respectively are implemented. However, we start to see distinctively different behaviours in the very first step itself (see Figure 4.12). The key observation is that the noise induced in velocity components of the prosthetic joints is propagated into the human subsystem and this is evident from the noise manifested in the Σ_h torques, τ_h (see Figure 4.12 (a)) under the centralized control architecture. On the contrary, due to the decentralized nature of the model under the distributed control framework, the noise is *rejected* from the human subsystem (see Figure 4.12 (c)) and is only limited to the torques, τ_r in Σ_r. The second observation of interest is that the amplitude of propagated noise into the torque measurements is significantly higher under the centralized control framework as opposed to the distributed control framework (see Figure 4.12 (b) and (d)).

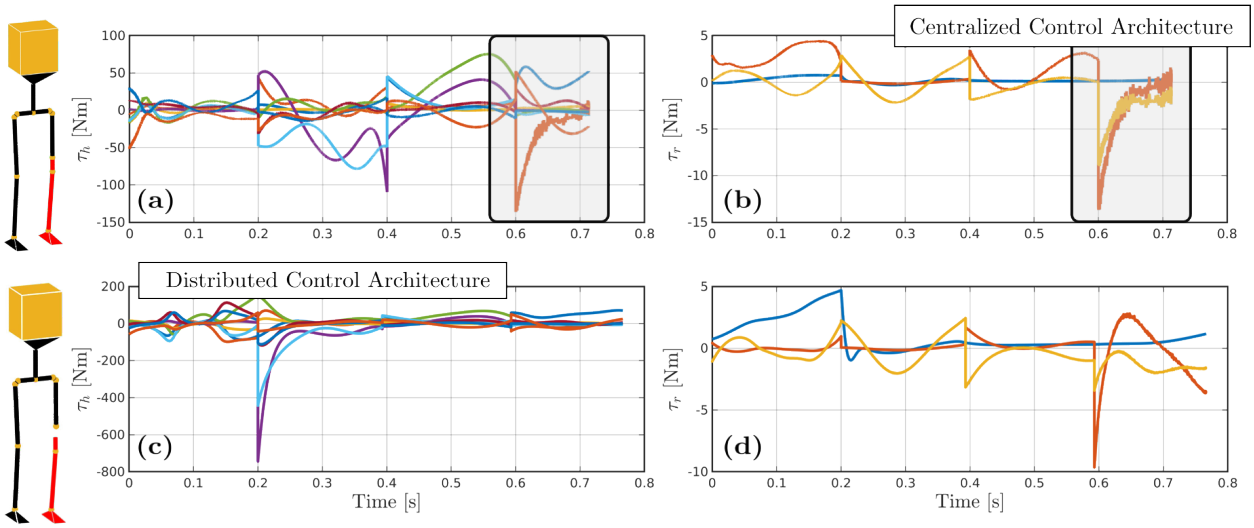


Figure 4.12: Simulation results demonstrating the performance of centralized, (a)-(b) and distributed controllers, (c)-(d) when the velocity components ($\dot{q}_r$) of Σ_r are induced with white Gaussian noise. (a),(c) capture the torques of Σ_h while (b),(d) capture Σ_r torques. Simulation was executed for 40 steps but results are truncated to 1 step to clearly highlight differences.

4.6 Summary and Discussion

In this chapter, we validated the theory developed to synthesize distributed controllers in Chapter 3 with the help of a problem that by nature provides a decentralized structure: human-lower-extremity prosthesis locomotion. We build a multi-domain cooperative locomotion model and demonstrate via numerical simulations that our method can stabilize this system without the need of any additional sensors. To demonstrate practicality, our nominal distributed controller is rigorously tested in the presence of persistent forces and dynamic uncertainties that are injected into the system but are unknown to the controller. We further show that our method does not rely on the existence of a stable orbit by synthesizing a distributed controller from a highly unstable orbit and demonstrating that the corresponding controller successfully stabilizes the cooperative human-robot locomotion model. Moreover, we show that distributed controllers facilitate better disturbance rejection as opposed to centralized controllers with the help of a comparative study.

While these results indicate significant potential and encourage the deployment of distributed controllers for the problem of cooperative human-robot locomotion, the implications go beyond the application itself. As highlighted in Figure 3.1, the framework can be extended to exoskeletons, bipedal robots and quadrupedal robots, alike. We have discussed in Chapter 1 and observed in Section 4.3 that the choice of full-order models at the motion planning stage necessitates offline trajectory optimization. This assumption is justifiable when viewed in the context of human-robot collaborative locomotion, as the human still holds the major share of control authority and can physically guide the prosthesis in navigating the envi-

ronment. On the contrary, for quadrupedal locomotion this choice would necessitate offline trajectory planning for every small change observed in the environment. This is a tedious task and would entail constructing a library of gaits. This is a key *roadblock* for on-board implementation on a physical robot. To address this, we propose a distributed QP-based control algorithm for quadrupedal locomotion. Addressing this objective demands answering few open questions, that forms the subject of investigation for part II of this thesis.

Part II

Distributed Quadratic Programming-based Controllers: A Scalable Approach based on I-O Linearization

Chapter 5

Network of Distributed Quadratic Programming-Based Controllers

This chapter presents the main theoretical contribution for the work done in Part II. It formulates a *network of distributed QPs* aimed at addressing non-sparse and strongly coupled nonlinear systems. We note that the resultant local controllers can be extended to any legged robot: biped or quadruped. A brief motivating survey is provided in Subsection 5.1.1. Next, the specific contributions of this chapter in Subsection 5.1.2. Finally, after providing a concise review of related work addressing distributed QPs, Section 5.2 will formulate the problem considered in this work. It is then followed by the synthesis of a network of distributed QPs in Section 5.3. In Section 5.4, we will perform the asymptotic convergence analysis and conclude the chapter in Section 5.5 with a brief summary.

5.1 Introduction

5.1.1 Motivation

QP-based controllers have received extensive consideration in recent years for the real-time planning and control of legged robots. For example, they have been used in the context of MPC for path planning of reduced-order locomotion models (see e.g., [42, 56, 68]) and whole-body motion control of full-order models (see e.g., [76, 81, 127]). The bedrock of these approaches considers the *centrality* structure. As modern legged and humanoid robots evolve into more complex architectures equipped with high DoFs, centralized QP-based controllers cannot scale with the dimensionality of these increasingly sophisticated machines. In addition, centralized QP-based algorithms cannot be easily transferred into the control of powered prosthetic legs and lower-limb exoskeletons because of the presence of inherent decentralization in the human-robot structure. The curse of dimensionality is a common problem in the field of large-scale systems such as power networks and urban traffic networks [180]. As a solution, a vast majority of work has focused on developing distributed and decentralized control algorithms in large-scale systems, see, e.g., [46, 155, 158, 188]. However, a systematic extension to high-dimensional and underactuated legged robots is lacking. This chapter considers this idea and details the pitfalls that differentiate local controllers for legged robots from local controllers for large-scale systems.

5.1.2 Contributions of this Chapter

The specific outcomes of this chapter are summarized in the following contributions: (1) Development of a network of distributed QPs (see Figure 5.1) with an inherent feedback structure that preserves the steady-state solution of the centralized QP on periodic orbits. (2) We theoretically establish a set of sufficient conditions under which the solutions of the local QPs asymptotically converge to the steady-state solution of the centralized QP.

5.1.3 Related work

In recent years, there have been significant theoretical and numerical advances in developing efficient methods that consider a *distributed* treatment of QPs. Examples include, but are not limited to, active-set methods [112, 142], Lagrangian decomposition or dual decomposition [133], and distributed multiple shooting methods [181]. These methods, however, depend upon the sparse nature of the equality constraints[125]. They translate very well to the problem of network systems or distributed MPC [69] as the subsystems are weakly coupled, hence preserving sparsity. On the contrary, legged robots are underactuated, high-dimensional, and inherently unstable hybrid dynamical systems. By nature, any morphological consideration of subsystems will result in these subsystems being strongly coupled. This further *motivates* the development of distributed QPs for high-DOF legged robots with strong coupling amongst their subsystems.

Steady-state dynamic gaits correspond to periodic trajectories of the hybrid models of locomotion. Different nonlinear control approaches have been developed to stabilize the periodic gaits of these systems. These approaches include but are not limited to HZD [54, 107, 147, 197, 212], transverse linearization [145], and controlled symmetries [194]. The HZD approach has been integrated with the Control Lyapunov Functions (CLFs) [30, 76, 81, 141] and Control Barrier Functions (CBFs) [160] to formulate QP-based nonlinear controllers for legged robots. The HZD approach has also been used to synthesize distributed nonlinear controllers for bipedal and quadrupedal robots [12, 141], prosthetic legs [92, 219], and exoskeletons [11]. References [12, 118, 164] developed decentralized controllers based on bilinear matrix inequalities (BMIs) and decomposition for locomotion models. Reference [141] presented stability guarantees, based on CLFs and input-to-state-stability (ISS), for an interconnected system of bipeds. It then developed model-free local QPs for quadrupedal locomotion.

5.2 Problem Formulation

The objective of this section is to present the problem statement to synthesize distributed QP-based nonlinear controllers for periodic orbits of hybrid models of locomotion. Similar

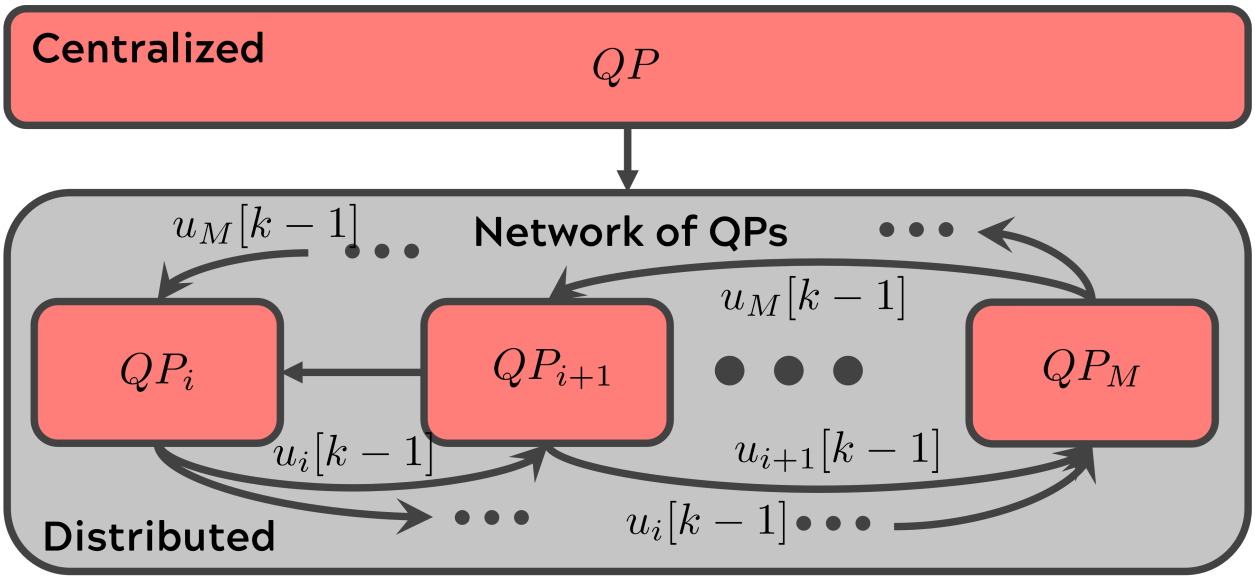


Figure 5.1: Illustration of the proposed network of distributed QPs based on the decomposition approach.

to Chapter 2 and without loss of generality, we consider single-domain hybrid systems as follows:

$$\Sigma : \begin{cases} \dot{x} = f(x) + g(x)\,u, & x \in \mathcal{X} \\ x^+ = \Delta(x^-), & x^- \in \mathcal{X} \cap \mathcal{S}, \end{cases} \tag{5.1}$$

where $x \in \mathcal{X} \subset \mathbb{R}^n$ and $u \in \mathcal{U} \subset \mathbb{R}^m$ denote the states and controls inputs, respectively, for some positive integers n and m. In addition, $\mathcal{X}$ and $\mathcal{U}$ represent the state manifold and a convex polytope of admissible control inputs. The evolution of the system during continuous-time domains is described by the smooth (i.e., $\mathcal{C}^\infty$) dynamics $\dot{x} = f(x) + g(x)\,u$. In addition, $\mathcal{S}$ denotes the guard of the hybrid system on which the evolution of the system is described by a smooth discrete-time transition (i.e., reset law) as $x^+ = \Delta(x^-)$. Here, x^- and x^+ denote the state of the system right before and after the reset law, respectively.

In this work, we consider the following class of real-time centralized QPs for nonlinear control of legged locomotion that can arise from I-O linearization:

$$\min_{u} \quad \frac{1}{2} \sum_{i \in \mathcal{I}} u_i^\top P_{ii}\, u_i \tag{5.2}$$

$$\text{s.t.} \quad \sum_{j \in \mathcal{I}} A_{ij}(x)\, u_j = b_i(x), \quad i \in \mathcal{I} \tag{5.3}$$

$$u_i \in \mathcal{U}_i, \qquad i \in \mathcal{I}. \tag{5.4}$$

Here, we assume that the nonlinear model is composed of $M \geq 2$ interconnected subsystems

Σ_i for $i \in \mathcal{I} := \{1, \cdots, M\}$. The local control inputs of the i-th subsystem is given by $u_i \in \mathcal{U}_i \subset \mathbb{R}^{m_i}$. In particular, $u := \text{col}(u_i \,|\, i \in \mathcal{I})$ and we assume that $\mathcal{U} = \mathcal{U}_1 \times \cdots \times \mathcal{U}_M$ for some convex polytopes $\mathcal{U}_i$ with $\sum_{i\in\mathcal{I}} m_i = m$. In addition, (5.3) represents the corresponding decomposition for some state-dependent coupled equality constraints given by $A(x)\, u = b(x)$. Furthermore, (5.4) denotes the decoupled inequality constraints corresponding to feasibility conditions. Finally, P_{ii}'s for $i \in \mathcal{I}$ are positive definite matrices. Throughout this chapter, we shall assume that by employing the centralized QP-based controller (5.2)-(5.4), there is an asymptotically stable periodic orbit (i.e., gait) $\mathcal{O}$ for the hybrid system model (5.1).

Assumption 5.1. (Periodic Solution): We suppose that there is a unique and periodic optimal solution of the strictly convex QP (5.2)-(5.4) on the orbit $\mathcal{O}$. This solution, referred to as the *steady-state solution*, is denoted by $u_{\text{s}}^\star(t) := \text{col}(u_{i,\text{s}}^\star(t) \,|\, i \in \mathcal{I})$ for $0 \leq t < T$, where T represents the fundamental period of the orbit. It is further assumed that the steady-state solution belongs to the interior of the set $\mathcal{U}_i$, that is, $u_{i,\text{s}}^\star(t) \in \text{int}(\mathcal{U}_i)$ for all $i \in \mathcal{I}$ and every $t \in [0, T)$.

Assumption 5.1 is *not* restrictive in that one can enlarge the admissible set of controls to satisfy the condition $u_{i,\text{s}}^\star(t) \in \text{int}(\mathcal{U}_i)$. Throughout this chapter, we shall assume that the steady-state solution is known for the proposed network of local QPs. This is *not* a limited assumption as one can consider $u_{\text{s}}^\star(t)$ as the *feedforward control inputs* (e.g., joint-level torques) that generate the orbit $\mathcal{O}$ (i.e., gait). For future purposes, the *steady-state Lagrange multipliers* corresponding to the equality constraints (5.3) on the periodic orbit are denoted by $\alpha_{i,\text{s}}^\star(t)$ for $i \in \mathcal{I}$. The steady-state solution and Lagrange multipliers ($u_{\text{s}}^\star(t)$ and $\alpha_{\text{s}}^\star(t)$) will be used to construct the proposed network of local QPs. Since the QPs will be solved digitally, we will continue our analysis in discrete time. In our notation, $k \in \mathbb{Z}_{\geq 0} := \{0, 1, \cdots\}$ represents the discrete time, T_s denotes the sampling time, and we assume that $\frac{T}{T_s} = N$ for some positive integer N.

Assumption 5.2. (One-Step Communication Delay): Motivated by the inherent limitation of the distributed structure, at every time sample k, each local QP can have access to the optimal solution of the other local QPs solved at time $k-1$, but *not* the current time sample. In particular, local QPs can share their optimal solutions from the previous time sample.

Problem 1. (Synthesis of Local QPs): The synthesis problem of distributed controllers consists of designing M local QPs whose optimal solutions asymptotically converge to the steady-state optimal solutions of Assumption 5.1 while meeting the communication protocol of Assumption 5.2.

5.3 Network of Distributed QPs

The objective of this section is to propose a network of local QPs that addresses Problem 1.

5.3.1 Synthesis of Local QPs

As discussed before, the local QPs can communicate and share their previous optimal solutions. Let us denote the optimal solutions of the QP_i at time sample $k-1$ by $u_i[k-1]$. We now propose the following structure for the local QP_i, $i \in \mathcal{I}$

$$\begin{aligned} \min_{u_i} \quad & \frac{1}{2} u_i^\top P_{ii}\, u_i + \eta_i^\top u_i \\ \text{s.t. } & A_{ii}\, u_i = b_i - \zeta_i \\ & u_i \in \mathcal{U}_i, \end{aligned} \tag{5.5}$$

where η_i and ζ_i are *feedback terms* defined as follows:

$$\begin{aligned} \eta_i &:= \sum_{j \in \mathcal{I}\setminus\{i\}} A_{ji}^\top \alpha_{j,\text{s}}^\star[k] - \sum_{j \in \mathcal{I}} L_{ij} \left(u_j[k-1] - u_{j,\text{s}}^\star[k-1] \right) \\ \zeta_i &:= \sum_{j \in \mathcal{I}\setminus\{i\}} A_{ij}\, u_{j,\text{s}}^\star[k] - \sum_{j \in \mathcal{I}} A_{ij}\, K_{ij} \left(u_j[k-1] - u_{j,\text{s}}^\star[k-1] \right) \end{aligned} \tag{5.6}$$

with K_{ij} and L_{ij} for $i, j \in \mathcal{I}$ being proper gain matrices to be determined. For future purposes, the elements of this set of gain matrices can be embedded in a *parameters vector* Θ. In (5.6), "\" represents the set minus. In the proposed structure (see Figure 5.1), the terms $\eta_i^\top u_i$ and ζ_i are added to the cost function and right-hand side of the equality constraints, respectively. These functions consist of feedforward terms $u_{j,\text{s}}^\star[k]$ and $\alpha_{j,\text{s}}^\star[k]$ for $j \in \mathcal{I} \setminus \{i\}$ as well as feedback terms to penalize the deviation from the orbit. In Theorem 5.3, we will show that the proposed network of local QPs preserves the steady-state solution $u_{\text{s}}^\star[k]$. Theorem 5.5 will show that the convergence to the steady-state solution will be achieved via a proper selection of the parameter vector Θ.

Theorem 5.3. (Steady-State Solutions for QPs): *Under Assumptions 5.1-5.2, $u_{i,s}^\star[k]$ for $i \in \mathcal{I}$ are unique optimal solutions for the proposed local QPs in (5.5) on the desired orbit $\mathcal{O}$.*

Proof. Let us assume that at time $k-1$, the solutions for the local QPs coincide with $u_{i,\text{s}}^\star[k-1]$ for all $i \in \mathcal{I}$. This assumption reduces η_i and ζ_i to the feedforward terms, i.e., $\eta_i = \sum_{j \in \mathcal{I}\setminus\{i\}} A_{ji}^\top \alpha_{j,\text{s}}^\star[k]$ and $\zeta_i = \sum_{j \in \mathcal{I}\setminus\{i\}} A_{ij}\, u_{j,\text{s}}^\star[k]$. We then show that $u_{i,\text{s}}^\star[k]$ is the unique solution for the local QP_i at time k. We remark that from Assumption 5.1, $u_i = u_{i,\text{s}}^\star[k]$ satisfies the equality constraint $A_{ii}\, u_i = b_i - \zeta_i$ and the inequality constraint. Hence, it is a feasible solution. We next show that the first and second-order Karush–Kuhn–Tucker (KKT) optimality conditions are met at this point. Since from Assumption 5.1 the inequality constraints are inactive (i.e., $u_i \in \text{int}(\mathcal{U}_i)$), the Lagrangian for the local QPs (5.5) is reduced to

$$\mathcal{L}_i := \frac{1}{2} u_i^\top P_{ii}\, u_i + \eta_i^\top\, u_i + \alpha_i^\top (A_{ii}\, u_i + \zeta_i - b_i), \tag{5.7}$$

where α_i represents the Lagrange multipliers corresponding to the equality constraints of the QP$_i$ with $i \in \mathcal{I}$. The first-order KKT optimality condition then implies that

$$\frac{\partial \mathcal{L}_i}{\partial u_i} = u_i^\top P_{ii} + \sum_{j \in \mathcal{I} \setminus \{i\}} \alpha_{j,\mathrm{s}}^{\star\top}[k]\, A_{ji} + \alpha_i^\top A_{ii} = 0. \tag{5.8}$$

It can be shown that this latter equation coincides with the one obtained from the first-order KKT condition for the Lagrangian of the centralized QP (5.2)-(5.4). In addition, $\frac{\partial^2 \mathcal{L}_i}{\partial u_i^2} = P_{ii}$ is positive definite, and hence, $u_i = u_{i,\mathrm{s}}^\star[k]$ is indeed the unique optimal solution. □

5.4 Asymptotic Convergence Analysis

From Theorem 5.3, the local QPs preserve the steady-state solution. We next study the properties and convergence behavior for the solutions of the proposed local QPs. Let us assume that the system's state evolves on the periodic orbit. According to the feedback structure, the optimal solutions of the local QPs at time $k+1$ depend on the ones from time k. In particular, there is a nonlinear and time-varying function $\mathcal{F}$ that defines the following discrete-time dynamics

$$u[k+1] = \mathcal{F}\left(k, u[k], \Theta\right), \quad k = 0, 1, \cdots. \tag{5.9}$$

According to the construction procedure, $\mathcal{F}$ is periodic in k with period $N > 0$. In addition, for every parameters vector Θ, $u_{\mathrm{s}}^\star[k] = \mathrm{col}(u_{i,\mathrm{s}}^\star[k] \,|\, i \in \mathcal{I})$ is an N-periodic solution to (5.9). Next, we make the following assumption.

Assumption 5.4. The matrices A_{ii}, $i \in \mathcal{I}$ in (5.3) are full row rank on the periodic orbit $\mathcal{O}$.

Now, we are in a position to present the following result.

Theorem 5.5. (Local Asymptotic Stability): *Under Assumptions 5.1-5.4, the following statements hold.*

1) *The function $\mathcal{F}(k, u, \Theta)$ is continuously differentiable (i.e., $\mathcal{C}^1$) with respect to u on $u = u_s^\star[k]$.*

2) *The N-periodic solution $u_s^\star[k]$ is locally asymptotically stable for (5.9) if the eigenvalues of the monodromy matrix lie inside the unit circle, where the monodromy matrix is defined as follows:*

$$\Psi_k(\Theta) := \prod_{\ell=0}^{N-1} \frac{\partial \mathcal{F}}{\partial u}(k+\ell, u_s^\star(k+\ell), \Theta), \tag{5.10}$$

in which the matrices in the product are ordered from right to left for increasing indices ℓ.

Proof. The QPs in (5.5) can be considered as optimization problems whose cost and constraints are parameterized by (η_i, ζ_i). We aim to show that the solutions of these QPs are $\mathcal{C}^1$ with respect to (η_i, ζ_i) on $u = u_{\mathrm{s}}^\star[k]$. From Theorem 5.3, $u_{i,\mathrm{s}}^\star[k]$ is the unique optimal solution for the local QPs. The cost and constraints are smooth (i.e., $\mathcal{C}^\infty$) in (u_i, η_i, ζ_i). Since the inequality constraints are inactive at $u_i = u_{i,\mathrm{s}}^\star[k]$, the active constraints are reduced to the equality ones. From Assumption 5.4, the gradients of the equality constraints with respect to the decision variables are full rank (i.e., regularity condition). Hence, the sufficient conditions of Fiacco's Theorem [77, Theorem 2.1] are satisfied. This guarantees the existence, uniqueness, and $\mathcal{C}^1$ continuity of the solutions with respect to (η_i, ζ_i) on an open neighborhood of $u = u_{\mathrm{s}}^\star[k]$. The fact that the feedback terms (η_i, ζ_i) are smooth in terms of u completes the proof of 1). 2) is an immediate result of Floquet stability theory [51] for periodic and nonlinear discrete-time systems. □

Remark 5.6. Theorem 5.5 presents an analytical tool for tuning local QPs' parameters to guarantee the asymptotic convergence of their solutions to the steady-state solution.

5.5 Summary

This chapter presented a formal foundation to decompose QP-based nonlinear controllers into a network of low-dimensional and distributed QP-based controllers with an inherent feedback structure for the general class of legged locomotion models. The proposed network of local QPs is developed based on a one-step communication delay protocol and preserves the steady-state solution of the centralized QP on periodic orbits. Properties of the local QPs were studied to establish a set of sufficient conditions under which the solutions of the local QPs asymptotically converge to the steady-state solution of the centralized QP. In the next chapter, we will validate the proposed theory by considering the problem of quadrupedal locomotion as an example.

Chapter 6

Application to Quadrupedal Locomotion

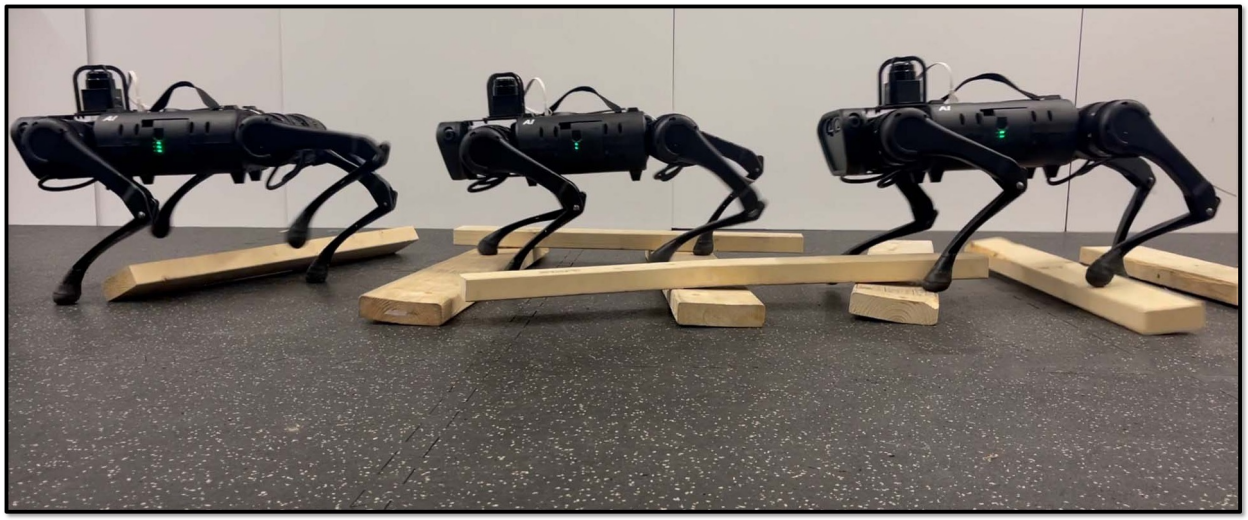


Figure 6.1: A composite of three separate images from the same experimental run illustrating the evolution of the distributed QP-based control algorithm on a quadrupedal platform, A1 [8]. Here, the robot is blindly trotting forward (from right to left) in the presence of arbitrarily dispersed wooden blocks at the speed of 0.5 (m/s).

This chapter aims to *validate* the theoretical contributions established in Chapter 5 by extending the proposed distributed QP-based formulation to the problem of quadrupedal locomotion. Towards that end, a brief summary of contributions is first provided in Subsection 6.1.2. Subsequently, in Section 6.2, a comprehensive description of the quadrupedal robot studied in this realization is presented. Section 6.3 will then condense the network of QPs introduced in Chapter 5 to *two* distributed QPs for the example in consideration. In Section 6.4, the performance of the resulting QP-network is promptly evaluated through rigorous numerical simulations. Further, to demonstrate that the proposed methodology translates equivalently well to reality, we perform a series of blind locomotion experiments on the physical platform (see Figure 6.1) and the corresponding results are presented in detail in Section 6.5. For convenience, a comprehensive video[1] that compiles the experimental results is made

[1]Video: https://youtu.be/VoYYCZrJOps

available. Finally, a brief summary and discussion in Section 6.6 concludes this chapter.

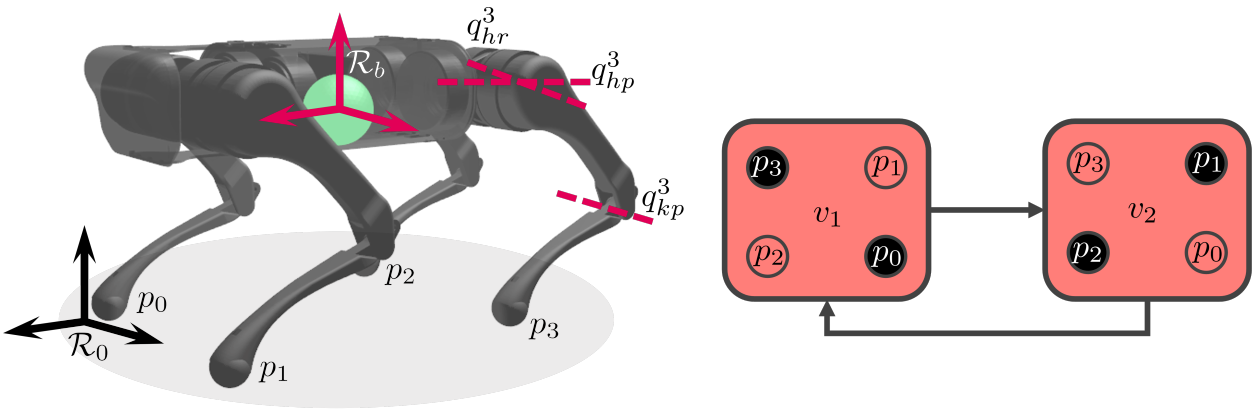
6.1 Introduction

6.1.1 Related Work

Previous efforts towards distributed control was primarily applied to the problems of collaborative, human-lower extremity prosthesis [12, 13, 118], wherein local controllers were developed based on Bilinear Matrix Inequalities (BMIs) and decomposition for hybrid models. Additionally, in [11], decentralized controllers were extended for the problem of locomotion with a robotic exoskeleton suit. In these instances, the choice of a distributed/decentralized control is immediate as the systems in consideration are naturally presented with a decentralized structure. However, the extension of distributed approaches for controlling systems that by nature posses a centralized structure has not been widely exploited in the literature. The notion of local controllers is appealing as they present attractive properties such as noise rejection and result in low-dimensional control problems that noticeably have less computation time. As the design of robots become more elaborate and complicated, such approaches might alleviate computational complexity and provide tractable solutions. Towards that end, for the first time, in [164], the idea of a single quadrupedal robot as a composition of two subsystems was exploited. A minimal communication channel was observed in this work and decentralized QPs (not based on decomposition) were developed for a walking gait. This work explored different morphological combination of subsystems, such as front-hind, left-right, and diagonal, albeit in simulations. In [141], an interconnected system of bipeds was explored and stability guarantees were provided based on CLFs and input-to-state-stability (ISS). However, it uses the resultant control law as a reference and proceeds to develop model-free local QPs to test on hardware experimentally. In this chapter and in [119], we consider the problem of quadrupedal locomotion as representative example to numerically and experimentally validate the theoretical framework established in Chapter 5. This results in some key contributions and are noted in the following subsection.

6.1.2 Contributions of this Chapter

The *specific outcomes* of this chapter are summarized with the following contributions: (1) In this chapter, it is demonstrated that favoring the distributed QP structure presents better noise rejection properties through rigorous numerical simulations. In particular, it is shown that when noise is injected into the front subsystem's velocity components, the noise propagates comparatively less into the hind subsystem's torque components in the distributed architecture. It further quantifies and reports the decrease in signal noise in Subsection 6.4. (2) As a consequence of the virtual separation and the subsequent formulation of two



(a) **(b)**

Figure 6.2: (a) Joint description of a 18-DoF quadrupedal robot. (b) Illustration of the two-domain trot gait considered in this realization. Here, the filled circle indicates contact with ground.

local QPs, the solve time is *reduced* by a factor of 3.82, the corresponding solve times are reported in Subsection 6.5. (3) This chapter demonstrates, for the *first time*, a robust and agile realization of model-based distributed QPs on a quadrupedal platform. An extensive set of blind locomotion experiments is reported in this chapter.

6.2 Robot Model Description

We consider a readily available quadrupedal platform, A1 [8] weighing around 12.45 (kg) with a nominal operating height of 0.26 - 0.28 (m) (see Figure 6.1). 18 DoFs can describe the evolution of the floating-based model of the quadruped. In order to do so, we attach a frame, $\mathcal{R}_b$, to the geometric center of the torso, which introduces 6 unactuated DoFs and is attributed to the absolute position ($\{x, y, z\}$) and orientation ($\{r, p, y\}$) of the robot's torso with respect to the global frame, $\mathcal{R}_0$ (see Figure. 6.2 (a)). The other 12 DoFs are actuated and associated with the legs' motions. For each leg, 2-DoFs capture the hip pitch and hip roll motion (for example, $\{q^3_{hr}, q^3_{hp}\}$), and one additional DoF captures the knee pitch motion (for example, q^3_{kp}) (see Figure. 6.2 (a)). Following the norm in the robotics society, $q(t) \in \mathbb{R}^{18}$ is our minimal coordinate representation that accurately describes the model's state at any point in time. Correspondingly then, the state vector is encoded in $x(t) := \text{col}(q, \dot{q}) \in \mathbb{R}^{36}$. Finally, the actuated joints that shape the control vector are encoded in $\tau(t) \in \mathbb{R}^{12}$.

The physical robot can receive torque commands wherein each joint has a max continuous torque rating of 33.5 (Nm) and a maximum rated velocity of 21 (rad/s). The states and velocities of the actuated joints and the absolute orientation (from an on-board IMU) are available to the user from the robot. The user also has access to the angular velocities associated with the orientation components, which are promptly converted into the world frame. The only missing components are the robots' position ($\{x, y, z\}$) and the correspond-

ing linear velocities, $(\{\dot{x}, \dot{y}, \dot{z}\})$ which the robot cannot directly measure. These components must be estimated. We use a simple kinematic estimator to address this issue and form a complete state vector in this work. We are now in a position to evaluate our model-based controller on the hardware platform.

In this realization, we specifically study the double-domain trotting gait (see Figure. 6.2 (b)) wherein the robot alternates contact with the ground using the diagonal leg pair (for example, p_0 and p_3 in domain 1 (i.e., v_1)) to locomote in any given direction.

6.3 QP-Based I-O Linearizing Controller

Let us quickly recall that the evolution of the mechanical system during continuous-time domains of locomotion can be described by the Euler-Lagrange equations and principle of virtual works as follows:

$$D(q)\,\ddot{q} + H(q,\dot{q}) = B\,\tau + J^{\top}(q)\,\lambda, \tag{6.1}$$

where $q \in \mathcal{Q}$ represents the generalized coordinates, $\mathcal{Q} \subset \mathbb{R}^{18}$ denotes the configuration space, $\tau \in \mathcal{T}$ denotes the joint-level torques, $\mathcal{T} \subset \mathbb{R}^{12}$ represents a closed and convex set of admissible torques, $\lambda := \text{col}\{\lambda_\ell \,|\, \ell \in \mathcal{G}\} \in \mathbb{R}^6$ denotes a vector including individual GRFs, and $\mathcal{G}$ represents the set of contacting leg ends with the environment. In addition, $D(q) \in \mathbb{R}^{18\times 18}$ represents the positive definite mass-inertia matrix, $H(q,\dot{q}) \in \mathbb{R}^{18}$ includes the Coriolis, centrifugal, and gravitational terms, and $B \in \mathbb{R}^{18\times 12}$ is the input distribution matrix. We remark that this model is valid if the individual GRFs at contacting leg ends remain in the linearized friction cone, denoted by $\mathcal{FC}$, that is, $\lambda_\ell \in \mathcal{FC}$ for all $\ell \in \mathcal{G}$. By defining the state vector as $x := \text{col}(q,\dot{q}) \in \mathcal{X} := \mathcal{Q} \times \mathbb{R}^{18}$, continuous-time dynamics can be described by the following state equation

$$\dot{x} = f(x) + g(x)\,\tau + w(x)\,\lambda. \tag{6.2}$$

The continuous-time state equation (6.2) is different from the one in (5.1) and the one in (2.9). However, the decision variables for the control problem in Section 6.3.1 will contain τ and λ allowing us to represent the QP in the form of (5.2) - (5.4). If a leg contacts the ground, the system's state then undergoes a discrete-time transition as in (5.1) and according to rigid impact models [212].

6.3.1 Centralized QP

We consider a set of holonomic output functions, referred to as *virtual constraints* [212], to be regulated for the whole-body motion control. More particularly, we study time-varying

virtual constraints defined as follows:

$$y(t, x) := h_0(q) - h_d(t), \tag{6.3}$$

where $h_0(q)$ encodes a set of controllable quantities, referred to as controlled variables, and $h_d(t)$ denotes the desired evolution of the controlled variables on the gait $\mathcal{O}$ as noted earlier in Chapter 2. Nominally for the trot gait considered here, 12 distinctive outputs are sufficient to have enough control authority in the centralized control scheme [20]. The 12 outputs are chosen as follows: 6 outputs for regulating the torso position, $p^{\text{com}} \in \mathbb{R}^3$ and orientation, $\phi^{\text{com}} \in \mathbb{R}^3$ and the other 6 are reserved for the Cartesian positions of the two swing leg ends, for example, $p_0^{sw} \in \mathbb{R}^3$ and $p_3^{sw} \in \mathbb{R}^3$ (see Figure 6.3 (a)). However, in this work, the choice of virtual constraints is not immediate—decomposing the nominal centralized controller does not leave each subsystem with enough control authority to cumulatively stabilize the quadruped. In order to overcome this roadblock then, we introduce 6 additional virtual constraints such that each subsystem is endowed with sufficient control authority. This makes our output function for the I-O linearizing and QP-based centralized controller 18-dimensional (i.e., $y \in \mathbb{R}^{18}$). Here, just as earlier, 6 outputs are reserved to control the Cartesian positions of two swing leg ends. The other 6 outputs are designated for the Cartesian coordinates of the COMs for the front and hind subsystems. The remaining 6 outputs are redundant and reserved for the absolute orientations of each subsystem (see Figure. 6.3 (b)). This leads to the augmented centralized controller, and henceforth, when we refer to the centralized scheme, we will be referring to this formulation. To address this redundancy and to ensure the validity of Assumption 5.4 (full row rank), we introduce defect variables v to be used later in (6.4). Next, differentiating the output y along the dynamics

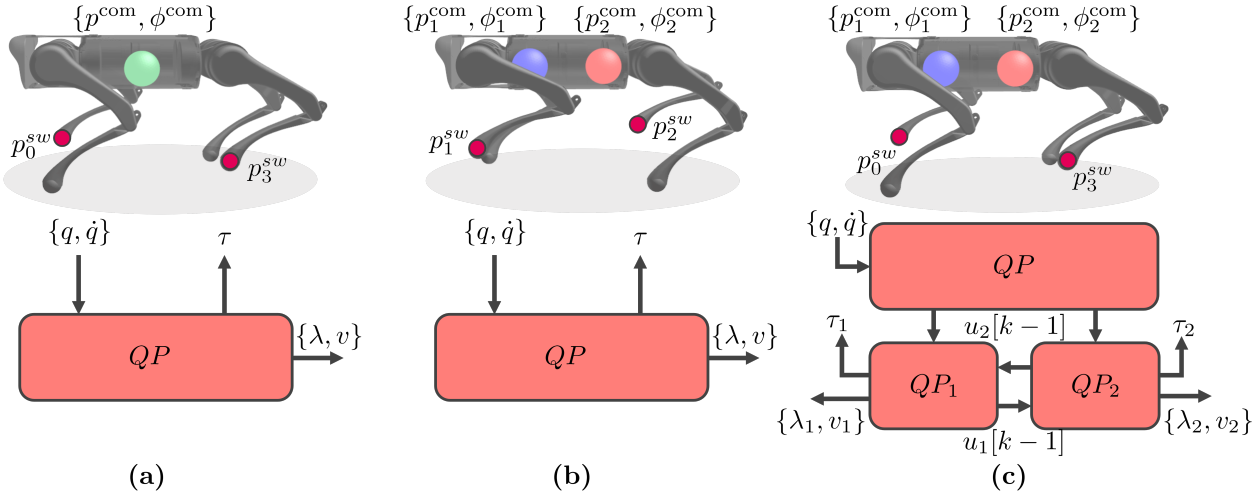


Figure 6.3: An illustration of the outputs and the different structures of QP-based controllers considered: (a) nominal centralized, (b) augmented centralized, and (c) distributed controller adopted from Chapter 5.

(6.2) gives

$$\ddot{y} = \mathrm{L}_g\mathrm{L}_f y\,\tau + \mathrm{L}_w\mathrm{L}_f y\,\lambda + \mathrm{L}_f^2 y - \ddot{h}_d = -K_P\,y - K_D\,\dot{y} + v, \tag{6.4}$$

where "L" denotes the Lie derivative, $\mathrm{L}_g\mathrm{L}_f y(t,x)$ and $\mathrm{L}_w\mathrm{L}_f y(t,x)$ represents the decoupling matrices with respect to τ and λ. Here, the right-hand side term $-K_P\,y - K_D\,\dot{y} + v$ represents the desired output dynamics to be achieved and $v \in \mathbb{R}^{18}$ is the defect variable. The objective is to solve for $(\tau, \lambda, v) \in \mathbb{R}^{36}$ that satisfies the output dynamics (6.4) and contact equations while having feasible torques and GRFs. Contact equations are a set of affine conditions in terms of (τ, λ) that represent the zero acceleration for the stance leg ends. In particular, analogous to the output dynamics, differentiating the Cartesian coordinates of the stance leg ends results in

$$\ddot{p} = \mathrm{L}_g\mathrm{L}_f p\,\tau + \mathrm{L}_w\mathrm{L}_f p\,\lambda + \mathrm{L}_f^2 p = 0, \tag{6.5}$$

where $p := \mathrm{col}\{p_\ell \,|\, \ell \in \mathcal{G}\} \in \mathbb{R}^6$ represents a vector containing the Cartesian coordinates of two stance leg ends.

To solve for a feasible (τ, λ, v) that meets the output dynamics (6.4) and contact equations (6.5), we then set up the following real-time and centralized convex QP

$$\begin{aligned}
\min_{(\tau,\lambda,v)}\ & \frac{\gamma_1}{2}\|\tau\|^2 + \frac{\gamma_2}{2}\|\lambda\|^2 + \frac{\gamma_3}{2}\|v\|^2 \\
\text{s.t.}\ & \mathrm{L}_g\mathrm{L}_f y\,\tau + \mathrm{L}_w\mathrm{L}_f y\,\lambda + \mathrm{L}_f^2 y - \ddot{h}_d = -K_P\,y - K_D\,\dot{y} + v \\
& \mathrm{L}_g\mathrm{L}_f p\,\tau + \mathrm{L}_w\mathrm{L}_f p\,\lambda + \mathrm{L}_f^2 p = 0 \\
& \tau \in \mathcal{T}, \quad \lambda_\ell \in \mathcal{FC}, \quad \forall \ell \in \mathcal{G}.
\end{aligned} \tag{6.6}$$

To minimize the effect of the defect variable v in the output dynamics, we add a penalty term to the cost function as $\frac{\gamma_3}{2}\|v\|^2$. We remark that the cost function tries to minimize a weighted sum of 2-norms of τ, λ, and v. Additionally, we note that the decision variables are represented by $u := \mathrm{col}(\tau, \lambda, v) \in \mathbb{R}^{36}$ for the centralized QP. From (6.6), we can extract the steady-state solution, $u_{\mathrm{s}}^\star(t)$, thus satisfying Assumption 5.1 (Periodic Solution).

6.3.2 Distributed QPs for Quadrupedal Locomotion

We are now in a position to formulate the distributed QPs for the problem of quadrupedal locomotion. Note that the choice of the decomposition topology entirely dictates the subsequent steps. It imposes the number of distinctive outputs one can choose (as discussed in Subsection 6.3.1) and decides the number of local control problems to be solved. For example, the system can be visualized as a virtual composition of 2 bipeds wherein it can either be a combination of front-hind, left-right, or diagonal subsystems. Alternatively, it can also be viewed as a virtual composition of 4 monopods. Since this chapter's primary goal is to validate the theoretical formulation presented in Chapter 5, in this work, we refrain from implementational complexity and consider the *front-hind* decomposition (see Figure 6.3 (c)) for the synthesis of distributed QPs (i.e., $\mathcal{I} = \{1, 2\}$). Consequently, the local outputs for

each subsystem, $y_i \in \mathbb{R}^9$, $i \in \mathcal{I}$, are constructed by appropriately partitioning the components corresponding to the individual subsystems from y. For example, subsystem 1 will have the COM positions and orientation as the first 6 outputs and the Cartesian position of the subsystem's swing leg as the remaining 3 outputs. The local torques, $\tau_i \in \mathbb{R}^6$, $i \in \mathcal{I}$, are similarly separated.

Equivalently, the decision variables for the local QPs can be denoted by $u_i := \text{col}(\tau_i, \lambda_i, v_i) \in \mathbb{R}^{18}$ for $i \in \mathcal{I}$. Note that, $\tau_i \in \mathbb{R}^6$, $\lambda_i \in \mathbb{R}^3$ and $v_i \in \mathbb{R}^9$ for the two-domain trot gait. It can be shown that the gradient of the equality constraints with respect to (τ, λ, v) is full rank as $\mathrm{L}_w\mathrm{L}_f p$ is invertible. The use of the defect variables together with the proper choice of local outputs as described earlier meets Assumption 5.4 (Full Rank). We can now condense the network of local QPs in Equation (5.5) to two distributed QPs with the corresponding local feedback terms in Equation (5.6) for the problem of quadrupedal locomotion. As an example, the QP corresponding to the front subsystem (subsystem 1) is noted as follows

$$\begin{aligned} \min_{u_1} \quad & \frac{1}{2} u_1^\top P_{11}\, u_1 + \eta_1^\top u_1 \\ \text{s.t.} \quad & A_{11}\, u_1 = b_1 - \zeta_1 \\ & u_1 \in \mathcal{U}_1, \end{aligned} \tag{6.7}$$

where η_1 and ζ_1 are expanded as follows:

$$\begin{aligned} \eta_1 :=& A_{21}^\top \alpha^\star_{j,\mathrm{s}}[k] - L_{11}\left(u_1[k-1] - u^\star_{1,\mathrm{s}}[k-1]\right) - L_{12}\left(u_2[k-1] - u^\star_{2,\mathrm{s}}[k-1]\right) \\ \zeta_1 :=& A_{11}\, K_{11}\, \left(u^\star_{1,\mathrm{s}}[k-1] - u_1[k-1]\right) + A_{12}\left(u^\star_{2,\mathrm{s}}[k] - K_{12}\left(u_2[k-1] - u^\star_{2,\mathrm{s}}[k-1]\right)\right). \end{aligned} \tag{6.8}$$

In a similar manner, the QP corresponding to the hind subsystem (subsystem 2) can be constructed. Each local QP takes a similar form as in Equation (6.6) when further expanded with decomposed equality and local inequality constraints.

6.4 Numerical Simulations

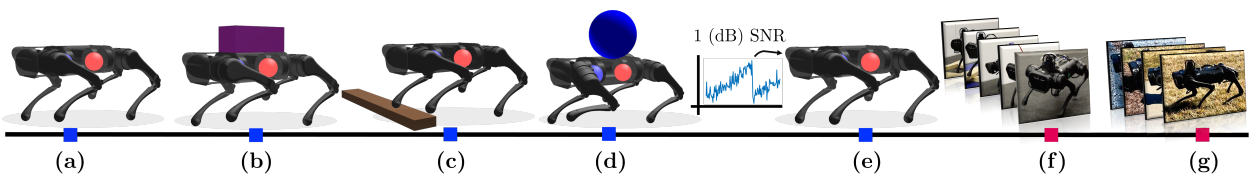


Figure 6.4: Illustration of an extensive set of simulations and experiments undertaken to evaluate the performance of the proposed distributed QP-based controllers. The blue nodes indicate simulation case studies: (a) nominal forward walking, (b)-(d) robustness analysis, and (e) performance of the the proposed distributed controller in the presence of noisy velocities. The red nodes signify experimental evaluation: (f) a series of indoor experiments and (g) a series of outdoor experiments.

To rigorously evaluate the performance of the resultant local controllers, we perform extensive numerical simulations (see Figure 6.4 (a)-(d)). All QP problems described in this thesis are solved using the qpSWIFT[165] solver, and we make use of the RaiSim [113] physics engine for all simulation case studies. The centralized QP in Equation (6.6) comprises of 24 coupled equality constraints and 70 decoupled inequality constraints, and the corresponding decomposed local QPs (see Equation (6.7) for an example of the front subsystem QP) each has 12 equality constraints and 35 inequality constraints. We make use of the same PD gains given by $K_P = 400$ and $K_D = 40$, and the same coefficients of the cost function given by $\gamma_1 = 100$, $\gamma_2 = 1$, and $\gamma_3 = 10^7$ for both centralized and distributed QPs. Further, the coefficients of the gain matrices in the proposed feedback structure (6.8) for the front subsystem QP are then tuned as $L_{11} = 10$, $L_{12} = 50$, $K_{11} = 0.08$, $K_{12} = 0.05$. Similarly, the coefficients for the gain matrices of the hind subsystem are tuned to be $L_{21} = 50$, $L_{22} = 10$, $K_{21} = 0.05$, and $K_{22} = 0.08$.

We consider a time-varying phase variable, $s := \frac{t-t_0}{t_d}$, where $t_d := t_f - t_0$ for the dynamic system to evolve in each domain (see Figure 6.2 (b)), wherein, we enforce each domain to strictly follow, $t_d = 0.2$ (s). We construct our desired time-varying control variables, $h_d(t)$ as follows: For the x and y translational coordinates, we prescribe the current COM position, x^{com} and y^{com} and a height of 0.26 (m). Subsequently, the desired evolution of the roll, pitch and yaw are set to 0 such that the torso is held parallel to the ground. At the beginning of each domain, we employ the Raibert footstep planner [173] to get the target foot step, p_x^d and p_y^d for the swing legs, given as

$$p_x^d = \frac{1}{2}\dot{x}_d^{\text{com}} t_d + K(\dot{x}^{\text{com}} - \dot{x}_d^{\text{com}}). \tag{6.9}$$

Here, $\dot{x}_d^{\text{com}}$ is the desired velocity of the COM and K is some positive gain. Similarly, one can obtain p_y^d. We then use the time-varying Beziér polynomials to generate the desired position and velocity trajectories for the swing leg ends given in [75] as

$$\begin{aligned} C(\alpha(t), s) &= \sum_{i=0}^{n} b_{i,n}\alpha_i \\ \dot{C}(\alpha(t), \dot{\alpha}(t), s) &= \frac{\partial C}{\partial s}(\alpha(t), s)\frac{1}{T} + C(\dot{\alpha}(t), s), \end{aligned} \tag{6.10}$$

where $b_{i,n}$ is the Berstein basis polynomial of degree n. Further, here $\dot{\alpha}$ encodes the desired velocity of the respective hip in the world frame. Finally, we prescribe a step height of 0.12 (m) and make use of the time-varying Beziér polynomial to obtain a trajectory for the z-component of the swing leg, p_z^d. Note that the physical parameters described here (except body height) are used throughout this work without any change.

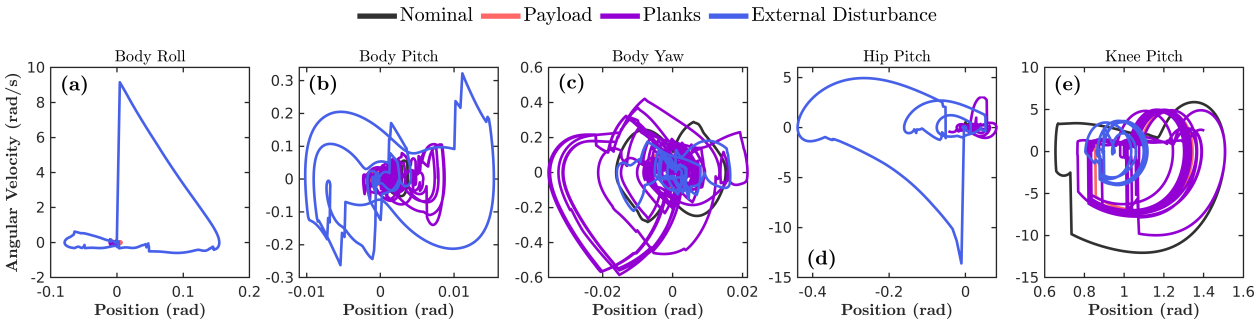


Figure 6.5: Phase plots, (a)-(e) of the body roll, body pitch, body yaw, hip pitch, and knee pitch corresponding to the *simulations* with nominal trot at 0.8 (m/s), payload at 0.3 (m/s), planks at 0.3 (m/s) and external disturbances (see Figure 6.8 (a)-(c)) applied during in-place trotting.

6.4.1 Nominal Trotting and Robustness Analysis

First, we consider the nominal in-place trotting locomotion with the centralized QP-based controller. This will allow us to extract the steady-state orbit, $u_s^\star$. Note that this is the only isolated instance of deploying the centralized controller throughout this work, i.e., we use the same steady-state orbit for all consequent distributed QP-based implementation in simulations and experiments. We then systematically evaluate the distributed QP-based controller in simulation for the forward trot gait while commanding a velocity of up to 1 (m/s). In the latter simulations, we evaluate the performance of the distributed controller for backward and lateral trot gaits. We report that we can successfully reach speeds up to 1 (m/s) in simulation in all directions. Next, to satisfactorily evaluate the robustness of the controller, before we transfer it to reality, we consider three scenarios. In the first scenario, we consider a simulation with a payload of around 5 (kg) (see Figure 6.4 (b)) and command the robot to trot at a velocity of 0.3 (m/s). Next, we consider planks of 0.04 (m) in height, displaced in simulation environment with the gap between each plank is set to 0.5 (m) (see Figure 6.4 (c)) and the mass of each plank is set to 0.9 (kg). Finally, to evaluate robustness to external disturbances, we simulate an object of 3 (kg) with a velocity of 3 (m/s) flying into the robot during an in-place trot along the positive y-axis (see Figure 6.4 (d) and Figure 6.8 (a)-(c)). These case studies have demonstrated that the distributed control scheme described in this work robustly stabilize the gait against mass uncertainty, ground height variations and external disturbances, as evident from the phase plots in Figure 6.5 (a)-(e). A video[2] compilation of all simulations are readily available.

6.4.2 Differentiating Distributed from Centralized Controllers

Finally, to differentiate between the performance of centralized and distributed QPs, we inject noise into the velocity components of the front subsystem with a SNR of 1 (dB). Figure

[2]Simulation Video: https://youtu.be/k-wg1svFVxo

6.6 studies the simulated hip roll torque (right hind leg) in the hind subsystem. It highlights noise propagation into the hind subsystem with the centralized QP-based controller, whereas the distributed QP-based controller significantly rejects the noise. In particular, the computed SNR for one stance domain of the torque signal in Figure 6.6 is increased from 18.35 (dB) in the centralized QP to 37.37 (dB) in the distributed QP.

Additionally, the time-varying Jacobian matrices in (5.10) are numerically computed via finite difference and their spectral radius approximately becomes 0.13. This makes the spectral radius of the monodromy matrix over $N = 400$ samples almost zero which indicates the asymptotic convergence to the steady-state solution.

6.5 Hardware Realization: Real-Time Control on A1

To realize **Task R3** (see Figure 1.3 (c)) and demonstrate the practical footing of the proposed distributed control algorithms, following the successful simulations and the corresponding validation via numerical analysis, we will consider the experimental implementation of the distributed controller with the same gains to stabilize the gaits of the robot. However, due to the insufficient computation power onboard the robot, we bring in additional computation power. In particular, we use an off-board laptop that is equipped with an 11^{th} Gen, 4-core, i7-1185G7 processor running at 3.00 (GHz) with a 16 (GB) RAM. As noted in Section 6.4, we again make use of the qpSWIFT solver [165] to solve the local QPs while enforcing a 1 (kHz) update frequency. During the experimental runs, we note that the centralized QP takes on average 0.130 (ms) and the distributed QPs take on average 0.034 (ms). In particular, the total solve time is reduced by a factor of 3.82 as a consequence of adopting the proposed

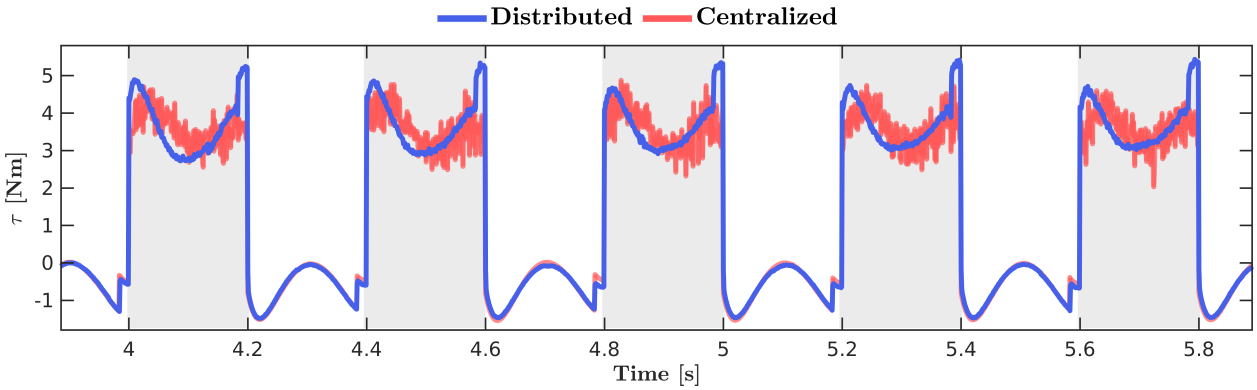


Figure 6.6: Hip roll torque (from simulation) in the distributed control scheme against the centralized control scheme. In this case study, the noise was injected into the velocity components of the front subsystem (front right leg), and plotted here are the torques of the right hind leg. The propagation of noise into the hind subsystem in the centralized control scheme is visibly higher. Here, the gray overlay indicates ground contact.

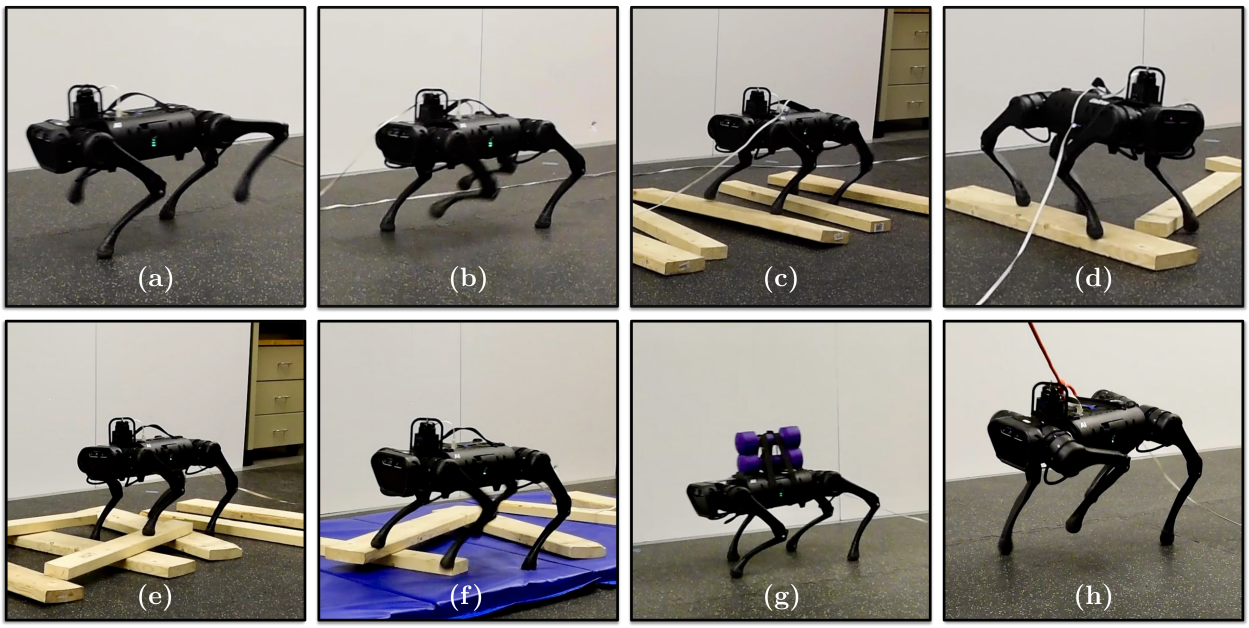


Figure 6.7: Snapshots illustrating a series of blind locomotion experiments (indoor) utilizing the proposed distributed QP-based controllers on the quadrupedal platform, A1. (a) nominal forward trot, (b) nominal backward trot, (c) backward trot in a field of arbitrarily displaced wooden blocks, (d) lateral trot on arbitrarily displace wooden blocks, (e) forward locomotion in a field of arbitrarily dispersed wooden blocks, (f) trotting on a compliant surface (gym mat) scattered with wooden blocks, (g) forward locomotion with a payload of 4.54 (kg) (36% uncertainty), (h) aggressive tethered pulling while trotting in-place.

local QPs.

6.5.1 Indoor Experiments

To rigorously evaluate the performance on hardware, we first undertake a series of indoor experiments in laboratory conditions and slowly increase complexity with each successful run to eliminate multiple uncertainties at once. Specifically, we consider (a) a terrain arbitrarily dispersed with wooden blocks (see Figure 6.7 (a)), (b) a compliant surface (gym mat) arbitrarily scattered with wooden blocks (see Figure 6.7 (b)), and (c) locomotion with a payload of 4.54 (kg) (36% uncertainty in the total mass) (see Figure 6.7 (c)). We consider a speed of 0.5 (m/s) for experiments in (a) and (c) and a speed of 0.4 (m/s) for the one in (b). Additionally, we consider (d) tethered pulling (see Figure 6.7 (d)) and (e) external pushes (see Figure 6.8 (d)-(f)) for in-place trot locomotion. The proposed local controllers can robustly stabilize the gaits, as evident from the phase plots in Figure 6.9 (a)-(e). Video[3] compilation of experiments are available online.

[3]Experiment Video: https://youtu.be/ED4ZfMZJPE8

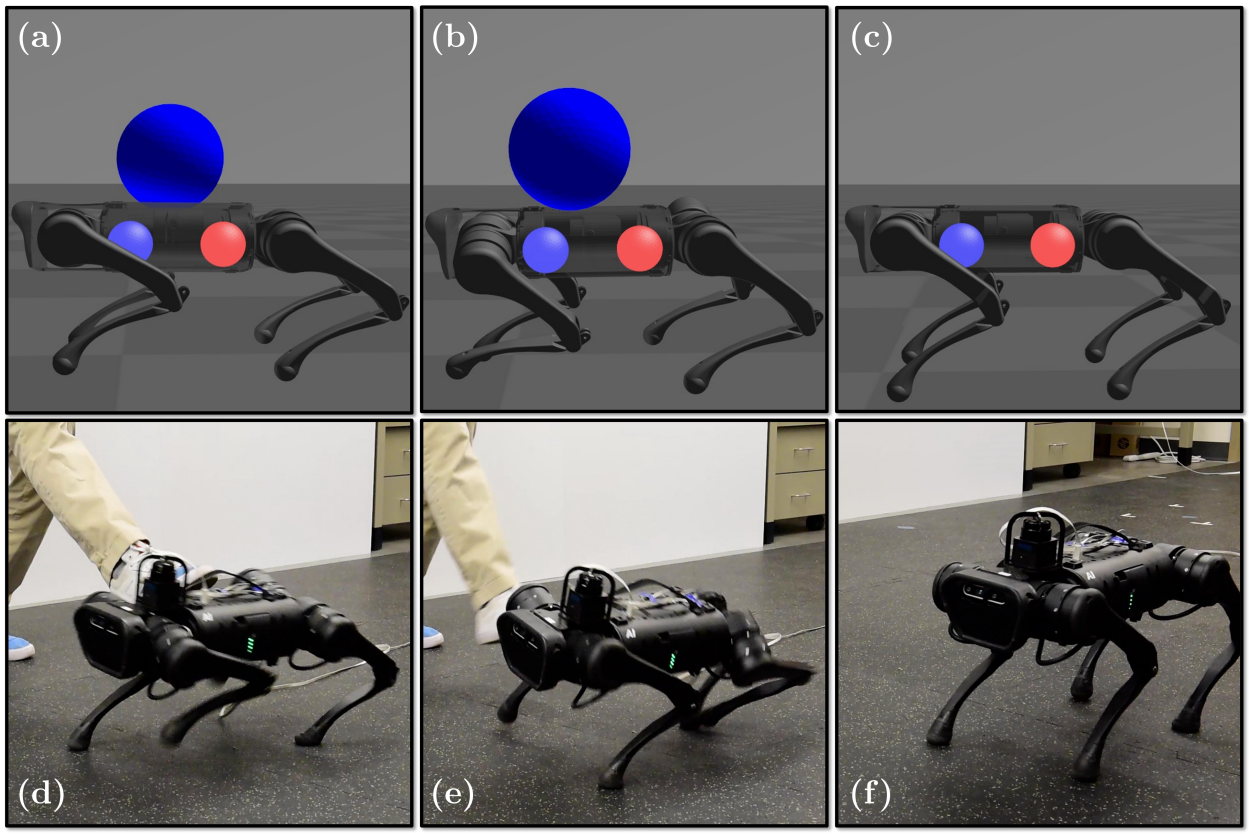


Figure 6.8: (a)-(c) Simulation case-study evaluating the performance of the distributed controller in presence of external disturbance provided by a 3 (kg) object flying at the robot in the positive y-axis. (d)-(f) Experiments demonstrating recovery from an aggressive push during in-place trotting.

One specific experiment that involves application of aggressive external forces (see Figure 6.8 (d)-(e)) during in-place trot locomotion, highlights the local controller's far-reaching capabilities to stabilize gaits of the robot. This is further corroborated by Figure 6.9 (c) wherein the phase plot corresponding to the aggressive push experiment converges back to the nominal orbit.

Figure 6.10 depicts the experimental torque of the hip roll from the distributed QP in contrast to the corresponding steady-state torque (extracted from numerical simulations) and highlights the close convergence of the local torque to the prescribed one.

6.5.2 Outdoor Experiments

To evaluate the performance of the distributed controller after satisfying results in indoor settings, we place A1 in real unstructured environments courtesy of the varying range of terrains outside Goodwin Hall. Specifically, we consider (a) terrain dispersed with loose

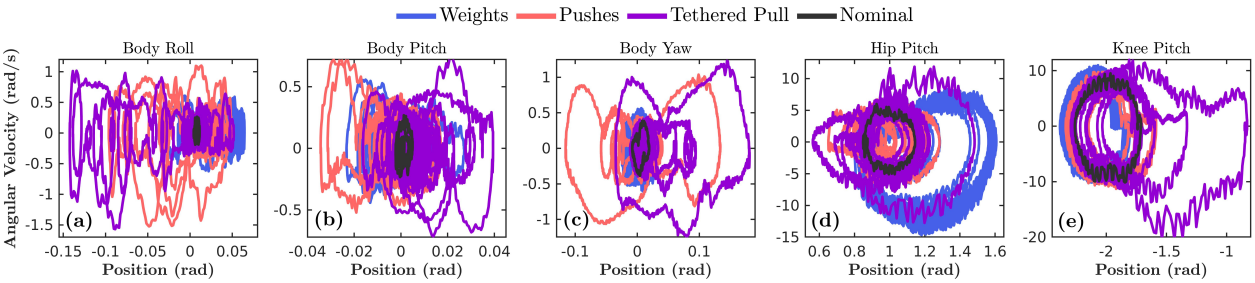


Figure 6.9: Phase plots, (a)-(e) of the body roll, body pitch, body yaw, hip pitch, and knee pitch corresponding to the *experiments* with payloads (see Figure 6.7 (c)), tethered pulling (see Figure 6.7 (d)) and external pushes (see Figure 6.8 (d)-(f)). An overlay of the nominal trot at 0.5 (m/s) is also provided.

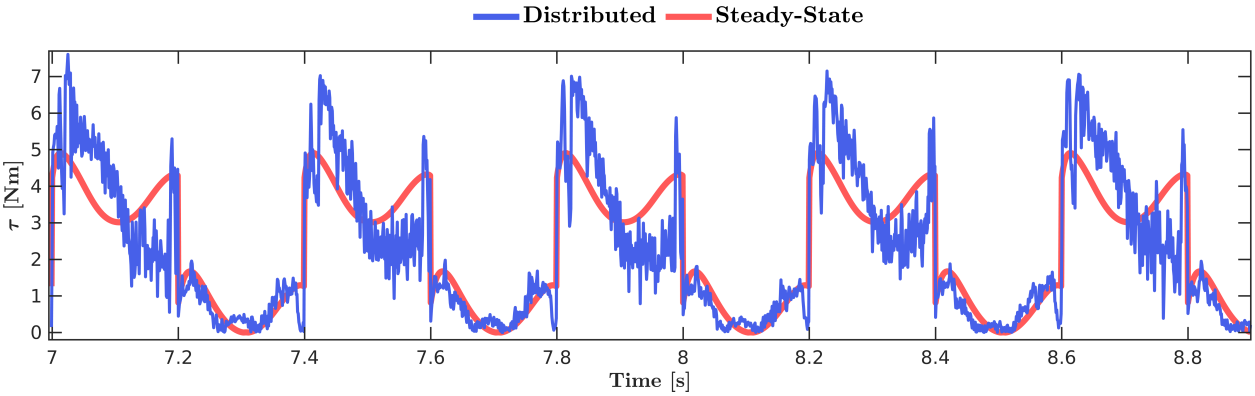


Figure 6.10: Comparison of the hip roll torque of the front right leg against the corresponding steady-state torque during nominal trot experiment, demonstrating close convergence.

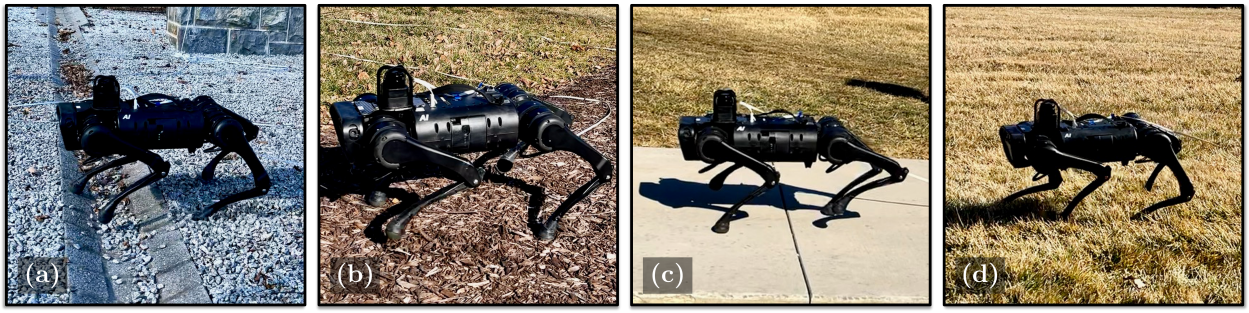


Figure 6.11: Snapshots illustrating various blind locomotion experiments in natural environments (outdoor) utilizing the proposed distributed QP-based controllers on the quadrupedal platform, A1. (a) Gravel with an abrupt change in terrain height at the speed of 0.4 (m/s), (b) mulch at the speed of 0.35 (m/s), (c) concrete at the speed of 0.35 (m/s), (d) grass at the speed of 1.0 (m/s).

gravel and a sudden abrupt change in terrain height facilitated by the water drain (see Figure 6.11 (a)), (b) terrain with small inclinations covered with a layer of mulch (see Figure

6.11 (b)), and (c) concrete pedestrian walkway (see Figure 6.11 (c)). We consider nominal

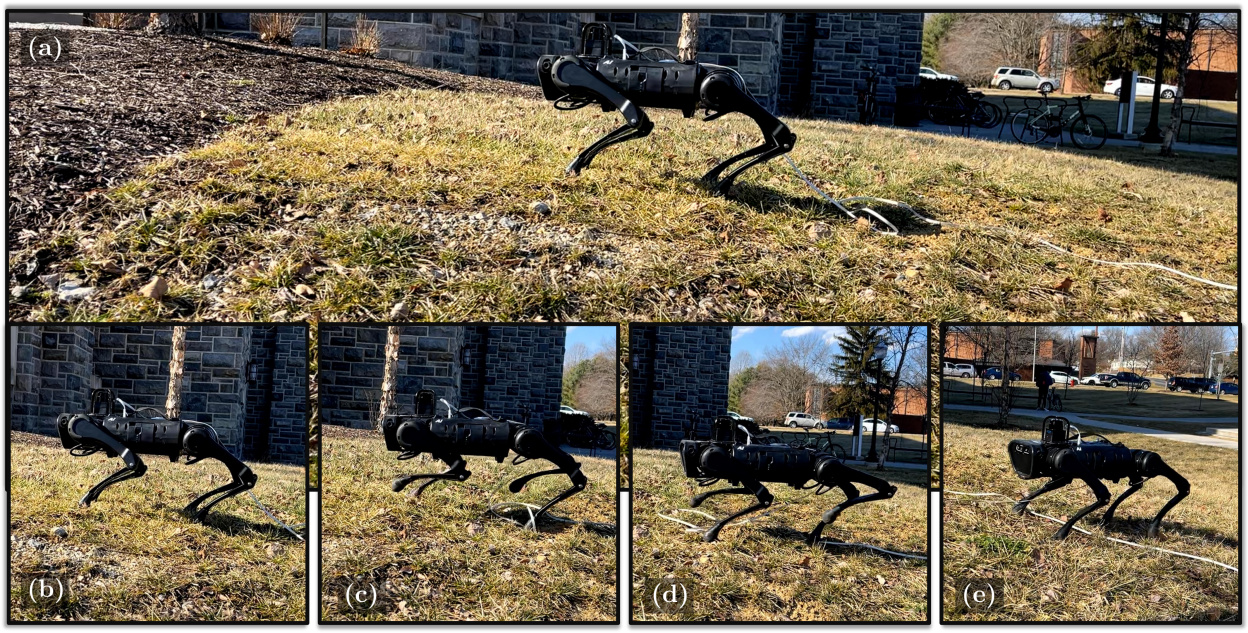


Figure 6.12: (a) A big picture view to clearly illustrate the inclination of the terrain A1 is set to traverse. Inset: (b)-(e) close-up snapshots (view from left to right) illustrating stable downhill locomotion on the inclined landscape while blindly trotting backward at the speed of 0.35 (m/s).

speeds ranging between 0.35 - 0.5 (m/s) for the experiments in Figure 6.11 (a)-(c). (d) Trotting on grassy terrain at the speed of 1.0 (m/s) (see Figure 6.11 (d)). Additionally, we also consider backward trotting on an inclined slope at the speed of 0.35 (m/s) (see Figure 6.12 (a)-(e)). The controller has no prior information of the inclination in terrain nor is it commanded to maintain a body pitch to assist in trotting downwards. The limit cycles corresponding to all outdoor experiments are given in Figure 6.14. Similarly, the local virtual constraints are given in Figure 6.13, the torques and GRFs are given in Figure 6.15.

6.6 Summary and Discussion

As a representative example, this chapter considers the problem of quadrupedal locomotion to test the general, analytical results developed in Chapter 5. It condenses the network of distributed QPs introduced in Section 5.3 to *two* local I-O linearizing, QP-based controllers by considering a front-hind decomposition strategy to realize agile quadrupedal locomotion. The chapter sufficiently demonstrated that the fundamental assumptions made in Chapter 5 are *not* restrictive. More particularly, it first systematically synthesizes an 18-output, centralized QP and simulates the resultant controller in a physics-based engine (RaiSim [113]) while considering an in-place trot gait. Consequently, it extracts the steady-state orbit

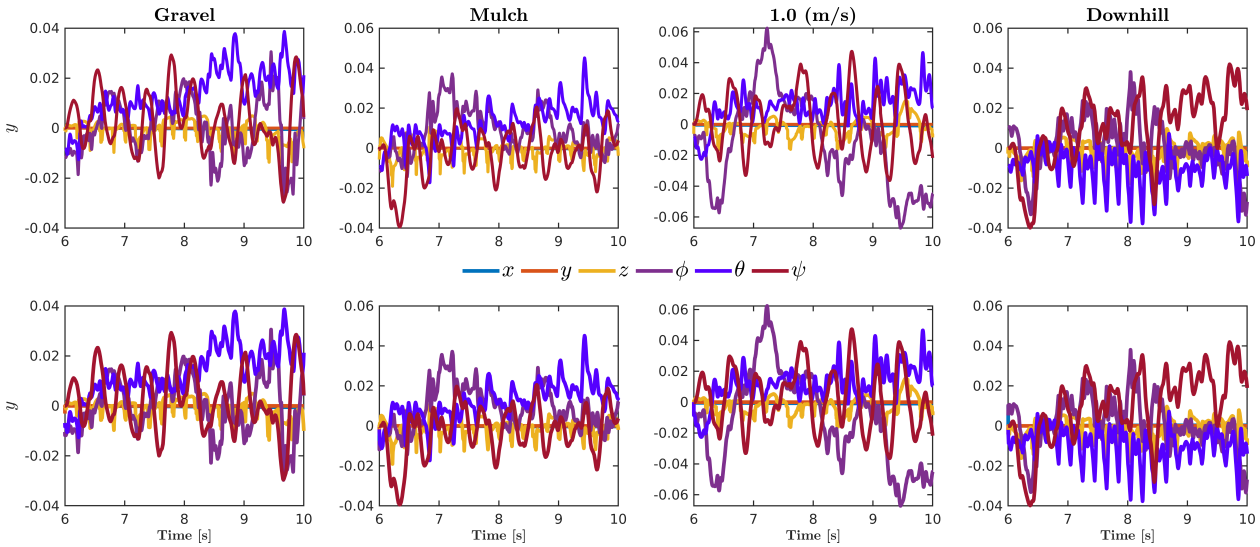


Figure 6.13: Virtual constraints (outputs) for all the outdoor experimental runs on A1. The *top row* indicates the outputs of the front subsystem and the *bottom row* indicates the outputs in the hind subsystem. Here, ϕ, θ and ψ indicate roll, pitch and yaw, respectively.

and synthesizes the distributed QP-based controller, which is similarly simulated in the same physics engine, illustrating that a one-step communication delay protocol between the local QPs is feasible. The performance of the resultant distributed controller is then exhaustively evaluated via various numerical simulations involving unknown terrain uncertainties and unmodelled external disturbances.

Further, in this chapter, a simulation case study that introduced additive white Gaussian noise with an SNR of 1 (dB) into the velocity components of the quadrupeds' front right leg was undertaken to distinguish between the centralized and distributed QP-based controllers. This chapter also demonstrates that the same results are extended to reality when evaluated on the physical quadruped, A1. A rigorous set of indoor experiments involving trotting on a rubber mat, a field of arbitrarily displaced wooden blocks, compliant surfaces or external pushes, tethered pulling, and in the presence of payload demonstrated the inherent robustness. Various speeds ranging from 0.4 - 0.7 (m/s) were evaluated in these experiments. The stabilizing distributed (asymptotic stability) controller was then tested in a varying set of natural terrain outside a controlled lab environment including on an inclined slope and up to 1.0 (m/s) speeds. A notably exciting point to observe here is that all the experiments used the *same* steady-state orbit extracted from the numerical simulation.

As a result, this chapter notes three key takeaways: 1) The simulation case study demonstrated that the distributed QP-based controller is noticeably *less sensitive* to noise than the centralized QP. It was shown that the strength of the noise propagated into the torques of hind subsystem in the distributed scheme amounted to 37.37 (dB) SNR, whereas the centralized scheme recorded an 18.35 (dB) SNR. 2) The total solve time of the local QPs

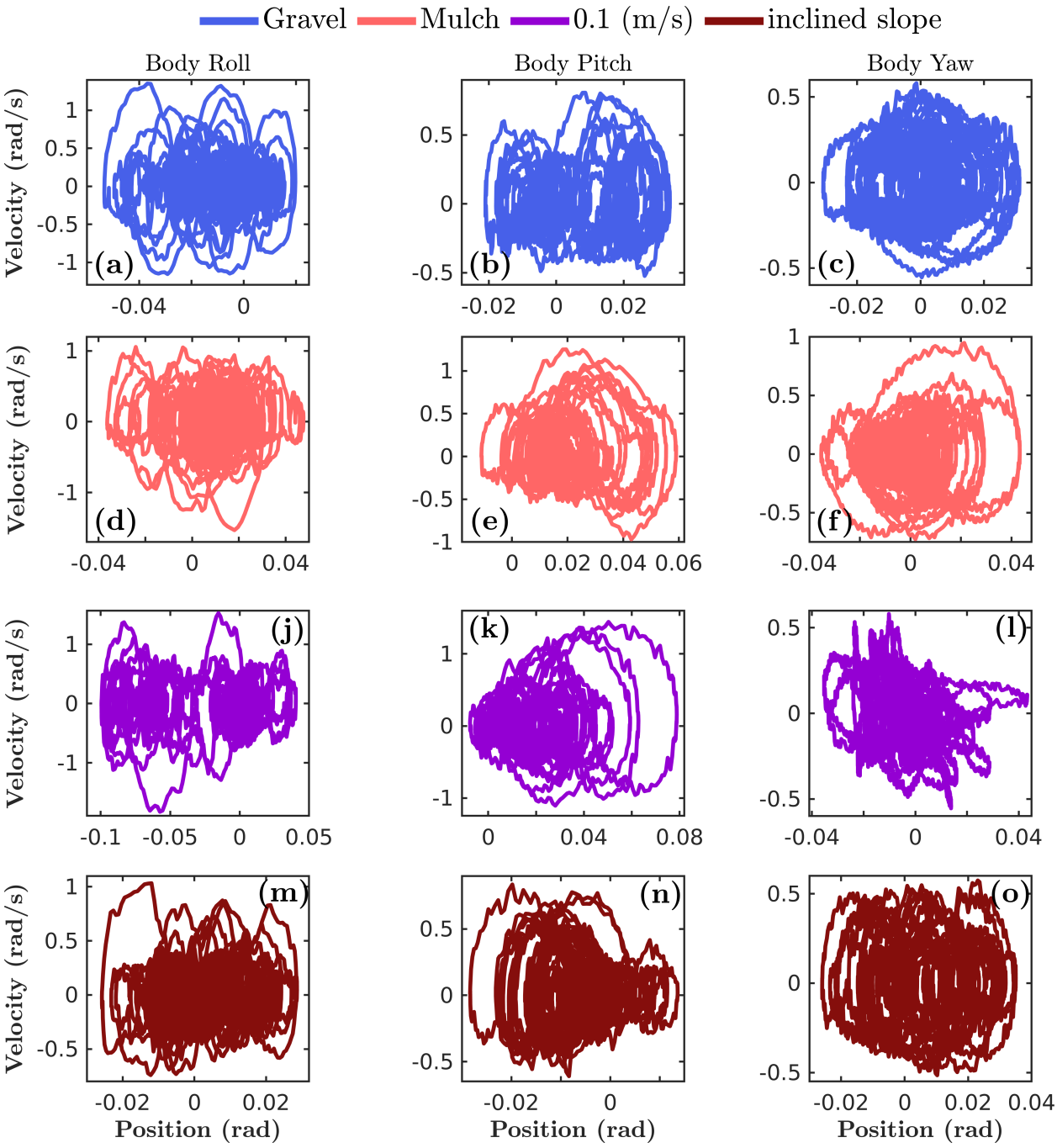


Figure 6.14: Phase plots of the body roll, body pitch and body yaw corresponding to, (a)-(c) gravel, (d)-(e) mulch, (j)-(k) 0.1 (m/s), and (m)-(o) inclined slope. The y-axis denotes the angular velocities.

is *reduced* by 3.82 times compared to the centralized QP. 3) The distributed controller is *practically feasible* as illustrated by the agile and robust locomotion.

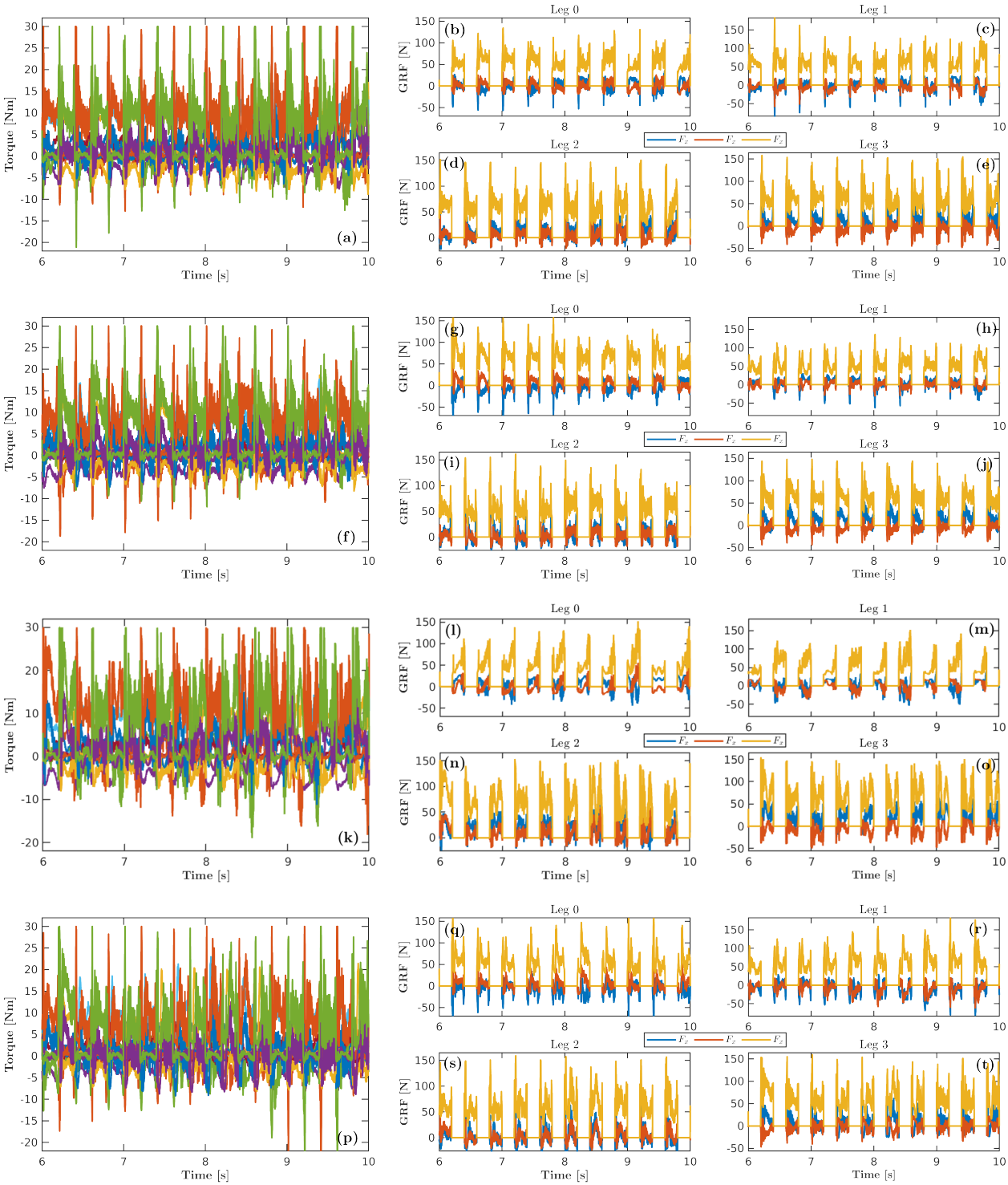


Figure 6.15: Joint torques and GRFs at the leg ends from the outdoor experimental runs: (a)-(e) loosely displaced gravel, (f)-(j) mulch, (k)-(o) grass at the speed of 1.0 (m/s), and (p)-(t) downhill locomotion.

Chapter 7

Conclusions and Future Work

This chapter provides a concise summary of the contributions of this thesis in Section 7.1, followed by a brief discussion on possible future directions in Section 7.2. It then concludes the thesis with final thoughts in Section 7.3.

7.1 Summary of Contributions

7.1.1 Decomposition-Based Distributed Controllers

We proposed a computationally tractable approach (see Chapter 3) to synthesizing local controllers for a general class of legged robots. More particularly, we integrated decomposition and optimization techniques allowing in a significant reduction of the parametric space as we only need to modify and optimize the pre-existing centralized HZD controllers rather than searching for local controllers from scratch. This leads to a small-scale optimization problem. Further, to simplify the search for the optimal local control parameters, we studied the properties of the Jacobian of the Poincaré map of the centralized and distributed controllers and demonstrated that there exists an upper bound on the norm of the difference of the respective Jacobian matrices. As a representative example, the problem of human-lower extremity prosthesis was considered, and it was shown that in comparison to the previous work, the extension of the theory presented in Chapter 3 demonstrated a 94.8% reduction in the parametric search space. The corresponding full-order simulations of a 19 DoF human-prosthesis model have demonstrated the feasibility of low-dimensional local control algorithms, marking the completion of **Task R1** (see Figure 1.3).

Further investigations demonstrated that the proposed controller is inherently robust to persistent external forces up to 200 (N) and can result in stable locomotion amidst unbeknownst changes to the mass parameters of the human model up to a difference of 20% from the original mass. The latter result is compelling as this indicates that the controller need not be tuned every time the human user changes. Additionally, it was also demonstrated with the help of a case study (see Chapter 4) that the proposed distributed controller rejects noise—noisy signal originated in one subsystem, stays in the same subsystem. On the contrary, the same claim is not supported for the centralized controller. For the human-prosthesis example, it is essential to note the roles of each agent: the human maintains the

principal control authority (i.e., guide), and the prosthesis has the minor control authority (i.e., follower). It is also important to note that the reference trajectories for the hybrid model were obtained through a tedious offline NLP optimization. Since the user maintains principal control authority, it is not necessary to regenerate new trajectories in case of minor changes in operating conditions. However, the same cannot be said for systems with no human intervention (e.g., quadrupeds) when a violation of assumptions in the trajectory optimization problems is detected. This necessitates a new offline trajectory optimization problem to be formulated every time a change in operating conditions is observed. In order to mitigate this roadblock, we shift to the use of heuristic reference trajectories and look towards the implementation of distributed QPs to encode control objectives subject to physical constraints.

7.1.2 Decomposition-Based Distributed QP Controllers

We formalized the notion of a network of local QPs (see Chapter 5) by presenting a systematic method to decompose the centralized QP to low-dimensional distributed QPs. We employed a one-step communication delay protocol such that the local QPs can establish a practical channel of data exchange amongst themselves. A feedback structure was introduced to preserve the steady-state solution of the centralized QP on periodic orbits (representing gaits). We theoretically established this fact by studying the properties of local QPs and establish a set of sufficient conditions under which the solutions of local QPs asymptotically converge to the steady-state solution. As a representative example, we consider the problem of quadrupedal locomotion and condensed the aforementioned network of distributed QPs to two local QPs by considering a front-hind decomposition. Through rigorous numerical full-order simulations of an 18 DoF quadruped, A1, we demonstrated the feasibility of this approach, marking the completion of **Task R2** (see Figure 1.3).

Additionally, similar to the case study in Subsection 7.1.1, we introduced noise into the front subsystem of the quadruped in both centralized and distributed control schemes. As opposed to the earlier example of human prosthesis (naturally decentralized structure), the virtual separation of the quadruped is susceptible to noise propagation. However, it was quantified that the distributed QP-based controller presents better noise rejection properties in comparison to the centralized QP. In particular, noisy torques in the hind subsystem with an SNR of 18.35 (dB) were recorded in the centralized QP, whereas the distributed QP formulation recorded only 37.37 (dB). It is important to recall that lower SNR indicates higher noise power in the signal.

7.1.3 Experimental Validation on a Quadrupedal Platform

This thesis supported the theoretical advances (see Chapter 5) on distributed QPs for dense, hybrid locomotion models with strong nonlinear interaction forces through numerous ex-

periments (see Chapter 6), marking the completion of **Task R3** (see Figure 1.3). The successful translation of distributed QPs to hardware amidst a wide range of aggressive external disturbances and adaptation to unknown terrains is promising and illustrates the inherent robustness of the proposed approach. The perception that all decomposition-based approaches need additional legwork—a centralized QP has to be solved every time a change in operational conditions is observed for the distributed QP to be functional without any performance deterioration—was abolished. It was shown that the centralized QP was solved only once and in simulation while performing in-place locomotion. The orbit extracted from this isolated run was used as a reference for all the preceding distributed QP experiments.

However, it is worth noting that a drastic change in operating conditions compared to the extracted reference orbit results in performance deterioration. For example, experiments with a payload of 4.54 (kg) and on a compliant surface (gym mat) were shown to be successful individually. However, when these two scenarios were combined, the robot demonstrated abnormal behavior during locomotion. Specifically, the robot had trouble maintaining the commanded height, resulting in unnecessary crouching mid gait. We will discuss possible approaches to increase robustness in Section 7.2. Finally, it was demonstrated that a significant change in average QP solve time is possible. More particularly, the implementation of distributed QPs reduces the computation time by a factor of 3.82. Furthermore, this is the first work in which a distributed HZD-based control was implemented on a hardware platform without any approximations or the use of optimal reference trajectory tracking techniques.

In conclusion, our numerical simulations in **Task R1** and **Task R2** show that the developed distributed controllers can reduce noise propagation, which agrees with the behavior of weakly coupled systems. For strongly coupled systems, theoretical and formal guarantees towards noise sensitivity form the future work. Approaches that focus on further improving the robustness of the methods presented in this thesis, alternative strategies, and applications are discussed in the next section.

7.2 Possible Directions for Future Work

This thesis's theoretical and experimental results underline the practical advantages of distributed control algorithms for high-dimensional legged robots. However, as highlighted in Subsection 7.1.3, the robustness presented by the decomposition schemes deteriorates when strayed away from the reference orbit. This can occur with drastic changes in physical parameters or commanding higher forward/lateral velocities. A potential solution to this problem may lie in assembling a library of orbits for varying physical parameters (for ex: height, different coefficients of friction) and various speeds and switching between the orbits. While extensive research on capturing the dynamics involved in switching between orbits of different gaits exists, the behavior of the system on switching between orbits of the same gait is an open problem.

An alternative approach to this shortcoming may be synthesizing distributed controllers that only rely on minimal communication. This approach prescribes the synthesis of distributed controllers from scratch instead of decomposition. While this was briefly exploited in one of earlier contributions, the improper choice of outputs imposed restrictions on the performance of the resulting controller. In particular, the controller could only be extended to a 4-domain walking gait and could not be generalized to the trotting gait. Possibly, with the choice of outputs presented in the latter part of this thesis, this problem can be revisited. Additionally, integrating the distributed controller with high-level planning algorithms based on distributed MPC could strengthen the notion of complete distributed control of legged robots.

Finally, the last, but a very interesting direction of research could be the extension of the distributed control framework to the detection of an abrupt failure (fault) of a subsystem. Nature has demonstrated that in the face of disability, humans and animals alike can locomote with the help of the available appendages. For example, dogs with a loss of limb have been observed to perform stable tripod gaits. Beyond fault detection is the modification of the existing distributed control algorithm to accommodate the sudden change in the system such that a statically-stable gait can be executed to steer the robot to a safe place. This can truly influence the autonomous applications of legged robots without human oversight.

7.3 Final Thoughts

This thesis has presented a fundamental contribution towards the synthesis of low-dimensional distributed controllers for high-dimensional legged robots. We have presented computationally tractable methods that preserve the strong interaction forces amongst individual subsystems and demonstrated the feasibility through rigorous numerical simulations. Additionally, we demonstrated the practical significance of the proposed distributed controllers through extensive experimentation on a quadrupedal platform. We hope that this work sufficiently highlighted the advantages of considering a distributed control scheme and that this small step could inspire the extension of local control schemes to even complex and high-dimensional legged robots, such as humanoids and exoskeletons.

Bibliography

[1] ANYbotics, https://www.anybotics.com/.

[2] Agility Robotics, http://www.agilityrobotics.com/.

[3] BostonDynamics, https://www.bostondynamics.com/.

[4] Robots with legs are getting ready to walk among us, https://www.theverge.com/2017/2/22/14635530/bipedal-legged-robots-mobility-advantages.

[5] DARPA Robotics Challenge, http://www.darpa.mil/program/darpa-robotics-challenge.

[6] Ghost Robotics, https://www.ghostrobotics.io/.

[7] MIT News: Cheetah 3, http://news.mit.edu/2018/blind-cheetah-robot-climb-stairs-obstacles-disaster-zones-0705.

[8] Unitree, http://www.unitree.cc/.

[9] Freedom Innovations, https://www.freedom-innovations.com/pacifica/.

[10] Össur. Power Knee, https://www.ossur.com/en-us, accessed on 3/18/2020.

[11] A. Agrawal, O. Harib, A. Hereid, S. Finet, M. Masselin, L. Praly, A. D. Ames, K. Sreenath, and J. W. Grizzle. First steps towards translating hzd control of bipedal robots to decentralized control of exoskeletons. *IEEE Access*, 5:9919–9934, 2017.

[12] K. Akbari Hamed and R. D. Gregg. Decentralized feedback controllers for robust stabilization of periodic orbits of hybrid systems: Application to bipedal walking. *IEEE Transactions on Control Systems Technology*, 25(4):1153–1167, July 2017. ISSN 1063-6536. doi: 10.1109/TCST.2016.2597741.

[13] K. Akbari Hamed and R. D. Gregg. Decentralized event-based controllers for robust stabilization of hybrid periodic orbits: Application to underactuated 3D bipedal walking. *IEEE Transactions on Automatic Control*, 64(6):2266–2281, June 2019. ISSN 0018-9286. doi: 10.1109/TAC.2018.2863184.

[14] K. Akbari Hamed and J.W. Grizzle. Event-based stabilization of periodic orbits for underactuated 3-D bipedal robots with left-right symmetry. *Robotics, IEEE Transactions on*, 30(2):365–381, April 2014. ISSN 1552-3098. doi: 10.1109/TRO.2013.2287831.

[15] K. Akbari Hamed, N. Sadati, W. A. Gruver, and G. A. Dumont. Exponential stabilisation of periodic orbits for running of a three-dimensional monopedal robot. *Control Theory Applications, IET*, 5(11):1304–1320, July 2011. ISSN 1751-8644. doi: 10.1049/iet-cta.2010.0292.

[16] K. Akbari Hamed, N. Sadati, W. A. Gruver, and G. A. Dumont. Stabilization of periodic orbits for planar walking with noninstantaneous double-support phase. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*, 42(3): 685–706, May 2012. ISSN 1083-4427. doi: 10.1109/TSMCA.2011.2169246.

[17] K. Akbari Hamed, B.G. Buss, and J.W. Grizzle. Exponentially stabilizing continuous-time controllers for periodic orbits of hybrid systems: Application to bipedal locomotion with ground height variations. *The International Journal of Robotics Research*, 35(8):977–999, 2016.

[18] K. Akbari Hamed, W. Ma, and A. D. Ames. Dynamically stable 3D quadrupedal walking with multi-domain hybrid system models and virtual constraint controllers. In *2019 American Control Conference (ACC)*, pages 4588–4595, July 2019. doi: 10.23919/ACC.2019.8815085.

[19] K. Akbari Hamed, V. R. Kamidi, A. Pandala, W. Ma, and A. D. Ames. Distributed feedback controllers for stable cooperative locomotion of quadrupedal robots: A virtual constraint approach. In *2020 American Control Conference (ACC)*, pages 5314–5321. IEEE, 2020.

[20] K. Akbari Hamed, J. Kim, and A. G. Pandala. Quadrupedal locomotion via event-based predictive control and QP-based virtual constraints. *IEEE Robotics and Automation Letters*, 5(3):4463–4470, 2020.

[21] Q. Ali and S. Montenegro. Decentralized control for scalable quadcopter formations. *International Journal of Aerospace Engineering*, 2016, 2016.

[22] J. Alonso-Mora, R. Knepper, R. Siegwart, and D. Rus. Local motion planning for collaborative multi-robot manipulation of deformable objects. In *2015 IEEE international conference on robotics and automation (ICRA)*, pages 5495–5502. IEEE, 2015.

[23] J. Alonso-Mora, E. Montijano, T. Nägeli, O. Hilliges, M. Schwager, and D. Rus. Distributed multi-robot formation control in dynamic environments. *Autonomous Robots*, 43(5):1079–1100, 2019.

[24] E. Ambrose, W. Ma, C. Hubicki, and A. D. Ames. Toward benchmarking locomotion economy across design configurations on the modular robot: Amber-3m. In *2017 IEEE Conference on Control Technology and Applications (CCTA)*, pages 1270–1276. IEEE, 2017.

[25] A. D. Ames. Human-inspired control of bipedal walking robots. *Automatic Control, IEEE Transactions on*, 59(5):1115–1130, May 2014. ISSN 0018-9286. doi: 10.1109/TAC.2014.2299342.

[26] A. D. Ames and S. Sastry. Hybrid Routhian reduction of Lagrangian hybrid systems. In *American Control Conference*, pages 6–pp. IEEE, 2006. URL http://ames.caltech.edu/HLR.pdf.

[27] A. D. Ames, R. D. Gregg, E. D. B. Wendel, and S. Sastry. On the geometric reduction of controlled three-dimensional bipedal robotic walkers. In *3rd Workshop on Lagrangian and Hamiltonian Methods for Nonlinear Control*, 2006. URL http://ames.caltech.edu/BipedReduction.pdf.

[28] A. D. Ames, R. D. Gregg, and M. W. Spong. A geometric approach to three-dimensional hipped bipedal robotic walking. In *Decision and Control, 46th IEEE Conference on*, pages 5123–5130, Dec 2007. doi: 10.1109/CDC.2007.4434880.

[29] A. D. Ames, R. D. Gregg, E. D. B. Wendel, and S. Sastry. On the geometric reduction of controlled three-dimensional bipedal robotic walkers. In *Lagrangian and Hamiltonian Methods for Nonlinear Control 2006*, pages 183–196, Berlin, Heidelberg, 2007. Springer. ISBN 978-3-540-73890-9.

[30] A. D. Ames, K. Galloway, K. Sreenath, and J. W. Grizzle. Rapidly exponentially stabilizing control Lyapunov functions and hybrid zero dynamics. *IEEE Transactions on Automatic Control*, 59(4):876–891, April 2014. ISSN 0018-9286. doi: 10.1109/TAC.2014.2299335.

[31] A.D. Ames and S. Sastry. Hybrid geometric reduction of hybrid systems. In *Decision and Control, 45th IEEE Conference on*, pages 923–929, Dec 2006. doi: 10.1109/CDC.2006.377245.

[32] A.D. Ames, R. W. Sinnet, and E. D.B. Wendel. Three-dimensional kneed bipedal walking: A hybrid geometric approach. In Rupak Majumdar and Paulo Tabuada, editors, *Hybrid Systems: Computation and Control*, volume 5469 of *Lecture Notes in Computer Science*, pages 16–30. Springer Berlin Heidelberg, 2009. ISBN 978-3-642-00601-2. doi: 10.1007/978-3-642-00602-9_2.

[33] N. Ayanian and V. Kumar. Decentralized feedback controllers for multiagent teams in environments with obstacles. *IEEE Transactions on Robotics*, 26(5):878–887, 2010.

[34] N. Ayanian, D. Rus, and V. Kumar. Decentralized multirobot control in partially known environments with dynamic task reassignment. *IFAC Proceedings Volumes*, 45 (26):311–316, 2012.

[35] V. Azimi, T. Shu, H. Zhao, E. Ambrose, A. D. Ames, and D. Simon. Robust control of a powered transfemoral prosthesis device with experimental verification. In *2017 American Control Conference (ACC)*, pages 517–522, May 2017. doi: 10.23919/ACC.2017.7963005.

[36] A. F. Azocar, L. M. Mooney, L. J. Hargrove, and E. J. Rouse. Design and characterization of an open-source robotic leg prosthesis. In *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob)*, pages 111–118. IEEE, 2018.

[37] D. D. Bainov and P. S. Simeonov. *Systems With Impulse Effect: Stability, Theory and Applications*. Ellis Horwood Ltd, June 1989.

[38] Lubomír Bakule. Decentralized control: An overview. *Annual Reviews in Control*, 32 (1):87–98, 2008.

[39] J. T. Betts. Survey of numerical methods for trajectory optimization. *Journal of guidance, control, and dynamics*, 21(2):193–207, 1998.

[40] A. V. Birn-Jeffery, C. M. Hubicki, Y. Blum, D. Renjewski, J. W. Hurst, and M. A. Daley. Don't break a leg: running birds from quail to ostrich prioritise leg safety and economy on uneven terrain. *Journal of Experimental Biology*, 217(21):3786–3796, 2014.

[41] G. Bledt, P. M. Wensing, and S. Kim. Policy-regularized model predictive control to stabilize diverse quadrupedal gaits for the mit cheetah. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4102–4109, Sep. 2017.

[42] G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim. MIT Cheetah 3: Design and control of a robust, dynamic quadruped robot. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 2245–2252, Oct 2018. doi: 10.1109/IROS.2018.8593885.

[43] R. Blickhan. The spring-mass model for running and hopping. *Journal of biomechanics*, 22(11-12):1217–1227, 1989.

[44] R. Blickhan and R. J. Full. Similarity in multilegged locomotion: bouncing like a monopode. *Journal of Comparative Physiology A*, 173(5):509–517, 1993.

[45] F. Borrelli, A. Bemporad, and M. Morari. *Predictive control for linear and hybrid systems*. Cambridge University Press, 2017.

[46] F. Bullo, J. Cortés, and Martinez. S. *Distributed Control of Robotic Networks: A Mathematical Approach to Motion Coordination Algorithms*. Princeton University Press, 2009.

[47] K. Byl and R. Tedrake. Approximate optimal control of the compass gait on rough terrain. In *Robotics and Automation. IEEE International Conference on*, pages 1258–1263, May 2008. doi: 10.1109/ROBOT.2008.4543376.

[48] K. Byl and R. Tedrake. Metastable walking machines. *The International Journal of Robotics Research*, 28(8):1040–1064, 2009. doi: 10.1177/0278364909340446.

[49] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar. Distributed model predictive control. *IEEE control systems magazine*, 22(1):44–52, 2002.

[50] Q. Cao and I. Poulakakis. Quadrupedal running with a flexible torso: control and speed transitions with sums-of-squares verification. *Artificial Life and Robotics*, 21(4): 384–392, Dec 2016.

[51] T.P. Chagas, P.-A. Bliman, and K.H. Kienitz. Stabilization of periodic orbits of discrete-time dynamical systems using the prediction-based control: New control law and practical aspects. *Journal of the Franklin Institute*, 355(12):4771–4793, 2018. ISSN 0016-0032. doi: https://doi.org/10.1016/j.jfranklin.2018.04.040.

[52] Amber-Lab YouTube Channel. Durus walks like a human, https://youtu.be/1fC7b2LjVW4, July 2016.

[53] Carlos Mastalli YouTube Channel. Motion planning for quadrupedal locomotion: Coupled planning, terrain mapping and whole-body control, https://youtu.be/KI9x1GZWRwE, September 2018.

[54] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. R. Westervelt, C. Canudas-de Wit, and J.W. Grizzle. RABBIT: A testbed for advanced control theory. *IEEE Control Systems Magazine*, 23(5):57–79, Oct 2003. ISSN 1066-033X. doi: 10.1109/MCS.2003.1234651.

[55] C. Chevallereau, J.W. Grizzle, and C.-L. Shih. Asymptotically stable walking of a five-link underactuated 3-D bipedal robot. *Robotics, IEEE Transactions on*, 25(1): 37–50, Feb 2009. ISSN 1552-3098. doi: 10.1109/TRO.2008.2010366.

[56] M. Chignoli and P. M. Wensing. Variational-based optimal control of underactuated balancing for dynamic quadrupeds. *IEEE Access*, 8:49785–49797, 2020.

[57] H. Choi, L. Brunet, and J. P. How. Consensus-based decentralized auctions for robust task allocation. *IEEE transactions on robotics*, 25(4):912–926, 2009.

[58] S. Collins, A. Ruina, R. Tedrake, and M. Wisse. Efficient bipedal robots based on passive-dynamic walkers. *Science*, 307(5712):1082–1085, 2005. doi: 10.1126/science.1107799.

[59] S. H. Collins, M. Wisse, and A. Ruina. A three-dimensional passive-dynamic walking robot with two legs and knees. *The International Journal of Robotics Research*, 20(7): 607–615, 2001.

[60] S. Corner, C. Sandu, and A. Sandu. Modeling and sensitivity analysis methodology for hybrid dynamical system. *Nonlinear Analysis: Hybrid Systems*, 31:19–40, 2019.

[61] S. Corner, A. Sandu, and C. Sandu. Adjoint sensitivity analysis of hybrid multibody dynamical systems. *Multibody System Dynamics*, 49(4):395–420, 2020.

[62] H. Dai and R. Tedrake. Optimizing robust limit cycles for legged locomotion on unknown terrain. In *IEEE 51st Annual Conference on Decision and Control*, pages 1207–1213, Dec 2012. doi: 10.1109/CDC.2012.6425971.

[63] H. Dai and R. Tedrake. $\mathcal{L}_2$-gain optimization for robust bipedal walking on unknown terrain. In *Robotics and Automation, IEEE International Conference on*, pages 3116–3123, May 2013. doi: 10.1109/ICRA.2013.6631010.

[64] M. A. Daley, J. R. Usherwood, G. Felix, and A. A. Biewener. Running over rough terrain: guinea fowl maintain dynamic stability despite a large unexpected change in substrate height. *Journal of experimental biology*, 209(1):171–187, 2006.

[65] P. de Leva. Adjustments to Zatsiorsky-Seluyanov's segment inertia parameters. *J Biomech*, 29(9):123–1230, 1996.

[66] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim. Dynamic locomotion in the MIT Cheetah 3 through convex model-predictive control. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–9, Oct 2018. doi: 10.1109/IROS.2018.8594448.

[67] Y. Ding, A. Pandala, and H. Park. Real-time model predictive control for versatile dynamic motions in quadrupedal robots. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8484–8490, May 2019. doi: 10.1109/ICRA.2019.8793669.

[68] Y. Ding, A. Pandala, C. Li, Y-H. Shin, and H-W. Park. Representation-free model predictive control for dynamic motions in quadrupeds. *IEEE Transactions on Robotics*, 37(4):1154–1171, 2021. doi: 10.1109/TRO.2020.3046415.

[69] W. B. Dunbar. Distributed receding horizon control of dynamically coupled nonlinear systems. *IEEE Transactions on Automatic Control*, 52(7):1249–1263, July 2007.

[70] M. F. Eilenberg, H. Geyer, and H. Herr. Control of a powered ankle–foot prosthesis based on a neuromuscular model. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 18(2):164–173, 2010.

[71] T. Elery, S. Rezazadeh, C. Nesler, and R. D. Gregg. Design and validation of a powered knee–ankle prosthesis with high-torque, low-impedance actuators. *IEEE Transactions on Robotics*, 36(6):1649–1668, 2020.

[72] J. Englsberger, C. Ott, M. A. Roa, A. Albu-Schäffer, and G. Hirzinger. Bipedal walking control based on capture point dynamics. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4420–4427, Sep. 2011. doi: 10.1109/IROS.2011.6094435.

[73] S. Fahmi, C. Mastalli, M. Focchi, D. G. Caldwell, and C. Semini. Passivity based whole-body control for quadrupedal locomotion on challenging terrain. *IEEE Robot. Automat. Lett.(RA-L)*, 2019.

[74] S. Fahmi, C. Mastalli, M. Focchi, and C. Semini. Passive whole-body control for quadruped robots: Experimental validation over challenging terrain. *IEEE Robotics and Automation Letters*, 4(3):2553–2560, July 2019. ISSN 2377-3766. doi: 10.1109/LRA.2019.2908502.

[75] R. T. Fawcett. Real-time planning and nonlinear control for robust quadrupedal locomotion with tails. Master's thesis, Virginia Tech, 2021.

[76] R. T. Fawcett, A. Pandala, A. D. Ames, and K. Akbari Hamed. Robust stabilization of periodic gaits for quadrupedal locomotion via QP-based virtual constraint controllers. *IEEE Control Systems Letters*, 6:1736–1741, 2022. doi: 10.1109/LCSYS.2021.3133198.

[77] A. V. Fiacco. Sensitivity analysis for nonlinear programming using penalty methods. *Mathematical Programming*, 10(1):287–311, 1976.

[78] L. B. Freidovich, A. S. Shiriaev, and I. R. Manchester. Stability analysis and control design for an underactuated walking robot via computation of a transverse linearization. *IFAC Proceedings Volumes*, 41(2):10166–10171, 2008.

[79] L.B. Freidovich, U. Mettin, A.S. Shiriaev, and M.W. Spong. A passive 2-DOF walker: Hunting for gaits using virtual holonomic constraints. *Robotics, IEEE Transactions on*, 25(5):1202–1208, Oct 2009. ISSN 1552-3098. doi: 10.1109/TRO.2009.2028757.

[80] R. J. Full and D. E. Koditschek. Templates and anchors: Neuromechanical hypotheses of legged locomotion on land. *Journal of Experimental Biology*, 202:3325–3332, December 1999.

[81] K. Galloway, K. Sreenath, A. D. Ames, and J. W. Grizzle. Torque saturation in bipedal robotic walking through control Lyapunov function-based quadratic programs. *IEEE Access*, 3:323–332, 2015. ISSN 2169-3536. doi: 10.1109/ACCESS.2015.2419630.

[82] M. Garcia, A. Chatterjee, A. Ruina, and M. Coleman. The simplest walking model: stability, complexity, and scaling. 1998.

[83] C. Gehring, C. D. Bellicoso, P. Fankhauser, S. Coros, and M. Hutter. Quadrupedal locomotion using trajectory optimization and hierarchical whole body control. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4788–4794. IEEE, 2017.

[84] H. Geyer, A. Seyfarth, and R. Blickhan. Compliant leg behaviour explains basic dynamics of walking and running. *Proceedings of the Royal Society B: Biological Sciences*, 273(1603):2861–2867, 2006.

[85] P. E. Gill, W. Murray, and M. A. Saunders. Snopt: An sqp algorithm for large-scale constrained optimization. *SIAM review*, 47(1):99–131, 2005.

[86] R. Goebel, R.G. Sanfelice, and A.R. Teel. *Hybrid Dynamical Systems: Modeling, Stability, and Robustness*. Princeton University Press, March 2012.

[87] R. Goedel, Ricardo G. S., and A. R. Teel. Hybrid dynamical systems: modeling stability, and robustness, 2012.

[88] A. Goswami, B. Espiau, and A. Keramane. Limit cycles in a passive compass gait biped and passivity-mimicking control laws. *Autonomous Robots*, 4(3):273–286, 1997.

[89] A. Goswami, B. Thuilot, and B. Espiau. A study of the passive gait of a compass-like biped robot: Symmetry and chaos. *The International Journal of Robotics Research*, 17(12):1282–1301, 1998.

[90] R. D. Gregg and M. W. Spong. Reduction-based control of three-dimensional bipedal walking robots. *The International Journal of Robotics Research*, 29(6):680–702, May 2010. ISSN 0278-3649. doi: 10.1177/0278364909104296.

[91] R.D. Gregg and L. Righetti. Controlled reduction with unactuated cyclic variables: Application to 3D bipedal walking with passive yaw rotation. *Automatic Control, IEEE Transactions on*, 58(10):2679–2685, Oct 2013. ISSN 0018-9286. doi: 10.1109/TAC.2013.2256011.

[92] R.D. Gregg and J.W. Sensinger. Towards biomimetic virtual constraint control of a powered prosthetic leg. *IEEE Transactions on Control Systems Technology*, 22(1): 246–254, Jan 2014. ISSN 1063-6536. doi: 10.1109/TCST.2012.2236840.

[93] R.D. Gregg and M.W. Spong. Reduction-based control with application to three-dimensional bipedal walking robots. In *American Control Conference*, pages 880–887, June 2008. doi: 10.1109/ACC.2008.4586604.

[94] R.D. Gregg and M.W. Spong. Reduction-based control of branched chains: Application to three-dimensional bipedal torso robots. In *Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on*, pages 8166–8173, Dec 2009. doi: 10.1109/CDC.2009.5399591.

[95] R.D. Gregg, A.K. Tilton, S. Candido, T. Bretl, and M.W. Spong. Control and planning of 3-D dynamic walking with asymptotically stable gait primitives. *Robotics, IEEE Transactions on*, 28(6):1415–1423, Dec 2012. ISSN 1552-3098. doi: 10.1109/TRO.2012.2210484.

[96] R.D. Gregg, T. Lenzi, L.J. Hargrove, and J.W. Sensinger. Virtual constraint control of a powered prosthetic leg: From simulation to experiments with transfemoral amputees. *IEEE Transactions on Robotics*, 30(6):1455–1471, Dec 2014. ISSN 1552-3098. doi: 10.1109/TRO.2014.2361937.

[97] R. J. Griffin and A. Leonessa. Model predictive control for dynamic footstep adjustment using the divergent component of motion. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1763–1768, 2016. doi: 10.1109/ICRA.2016.7487320.

[98] R. J. Griffin, G. Wiedebach, S. Bertrand, A. Leonessa, and J. Pratt. Walking stabilization using step timing and location adjustment on the humanoid robot, Atlas. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 667–673, Sep. 2017. doi: 10.1109/IROS.2017.8202223.

[99] R. J. Griffin, G. Wiedebach, S. Bertrand, A. Leonessa, and J. Pratt. Straight-leg walking through underconstrained whole-body control. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5747–5754. IEEE, 2018.

[100] J. W. Grizzle, C. Chevallereau, R. W. Sinnet, and A. D. Ames. Models, feedback control, and open problems of 3D bipedal robotic walking. *Automatica*, 50(8):1955–1988, 2014.

[101] J.W. Grizzle, G. Abba, and F. Plestan. Asymptotically stable walking for biped robots: Analysis via systems with impulse effects. *IEEE Transactions on Automatic Control*, 46(1):51–64, Jan 2001. ISSN 0018-9286. doi: 10.1109/9.898695.

[102] T. Gurriet, S. Finet, G. Boeris, A. Duburcq, A. Hereid, O. Harib, M. Masselin, J. Grizzle, and A. D. Ames. Towards restoring locomotion for paraplegics: Realizing dynamically stable walking on exoskeletons. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 2804–2811. IEEE, 2018.

[103] W.M. Haddad, V. Chellaboina, and S.G. Nersesov. *Impulsive and Hybrid Dynamical Systems: Stability, Dissipativity, and Control*. Princeton University Press, July 2006.

[104] K. Akbari Hamed and A. D. Ames. Nonholonomic hybrid zero dynamics for the stabilization of periodic orbits: Application to underactuated robotic walking. *IEEE Transactions on Control Systems Technology*, Under Review, 2018.

[105] O. Harib, A. Hereid, A. Agrawal, T. Gurriet, S. Finet, G. Boeris, A. Duburcq, M. E. Mungai, M. Masselin, A. D. Ames, , K. Sreenath, and J. W. Grizzle. Feedback control of an exoskeleton for paraplegics: Toward robustly stable, hands-free dynamic walking. *IEEE Control Systems Magazine*, 38(6):61–87, 2018.

[106] A. Hereid and A. D. Ames. FROST: Fast robot optimization and simulation toolkit. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 719–726, Sept 2017. doi: 10.1109/IROS.2017.8202230.

[107] A. Hereid, C. M. Hubicki, E. A. Cousineau, and A. D. Ames. Dynamic humanoid locomotion: A scalable formulation for HZD gait optimization. *IEEE Transactions on Robotics*, pages 1–18, 2018. ISSN 1552-3098. doi: 10.1109/TRO.2017.2783371.

[108] I.A. Hiskens and M.A. Pai. Hybrid systems view of power system modelling. In *Circuits and Systems. Proceedings The 2000 IEEE International Symposium on*, volume 2, pages 228–231 vol.2, 2000. doi: 10.1109/ISCAS.2000.856300.

[109] J.K. Holm and M.W. Spong. Kinetic energy shaping for gait regulation of underactuated bipeds. In *Control Applications. IEEE International Conference on*, pages 1232–1238, 2008. doi: 10.1109/CCA.2008.4629638.

[110] J.K. Holm, Dongjun Lee, and M.W. Spong. Time-scaling trajectories of passive-dynamic bipedal robots. In *Robotics and Automation, IEEE International Conference on*, pages 3603–3608, April 2007. doi: 10.1109/ROBOT.2007.364030.

[111] Y. Hurmuzlu and D. B. Marghitu. Rigid body collisions of planar kinematic chains with multiple contact points. *The International Journal of Robotics Research*, 13(1): 82–92, 1994. doi: 10.1177/027836499401300106.

[112] H. Huynh. *A large-scale quadratic programming solver based on block-LU updates of the KKT system*. PhD thesis, Stanford University, 2008.

[113] J. Hwangbo, J. Lee, and M. Hutter. Per-contact iteration method for solving contact dynamics. *IEEE Robotics and Automation Letters*, 3(2):895–902, April 2018. ISSN 2377-3774. doi: 10.1109/LRA.2018.2792536.

[114] A. Isidori. *Nonlinear Control Systems*. Springer; 3rd edition, 1995.

[115] 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 *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*, volume 1, pages 239–246. IEEE, 2001.

[116] 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 *2003 IEEE International Conference on Robotics and Automation*, volume 2, pages 1620–1626 vol.2, Sep. 2003. doi: 10.1109/ROBOT.2003.1241826.

[117] R. E. Kambic, T. J. Roberts, and S. M. Gatesy. Guineafowl with a twist: asymmetric limb control in steady bipedal locomotion. *Journal of Experimental Biology*, 218(23): 3836–3844, 2015.

[118] V. R. Kamidi, J. C. Horn, R. D. Gregg, and K. Akbari Hamed. Distributed controllers for human-robot locomotion: A scalable approach based on decomposition and hybrid zero dynamics. *IEEE Control Systems Letters*, 5(6):1976–1981, Dec. 2021. doi: 10.1109/LCSYS.2020.3045669.

[119] V. R. Kamidi, J. Kim, R. T. Fawcett, A. D. Ames, and K. Akbari Hamed. Distributed quadratic programming-based nonlinear controllers for periodic gaits on legged robots. *IEEE Control Systems Letters*, Feb. 2022. Under Review.

[120] D. C. Kerrigan, U. Della Croce, M. Marciello, and P. O. Riley. A refined view of the determinants of gait: significance of heel rise. *Archives of physical medicine and rehabilitation*, 81(8):1077–1080, 2000.

[121] T. Keviczky, F. Borrelli, K. Fregene, D. Godbole, and G. J. Balas. Decentralized receding horizon control and coordination of autonomous vehicle formations. *IEEE Transactions on control systems technology*, 16(1):19–33, 2007.

[122] H. K. Khalil. *Nonlinear Systems*. Pearson, 3rd edition, 2001.

[123] J. Kim, C. Choi, and M. W. Spong. Passive dynamic walking with knee and fixed flat feet. In *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 2744–2750. IEEE, 2012.

[124] 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*, 2012. doi: 10.1177/0278364912452673.

[125] A. Kozma, C. Conte, and M. Diehl. Benchmarking large-scale distributed convex quadratic programming algorithms. *Optimization methods and software*, 30(1):191–214, 2015.

[126] D. J. Kriellaars, R. M. Brownstone, B. R. Noga, and L. M. Jordan. Mechanical entrainment of fictive locomotion in the decerebrate cat. *Journal of neurophysiology*, 71(6):2074–2086, 1994.

[127] S. Kuindersma, F. Permenter, and R. Tedrake. An efficiently solvable quadratic program for stabilizing dynamic locomotion. In *IEEE International Conference on Robotics and Automation*, pages 2589–2594, June 2014.

[128] S. Kuindersma, F. Permenter, and R. Tedrake. An efficiently solvable quadratic program for stabilizing dynamic locomotion. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2589–2594, May 2014. doi: 10.1109/ICRA.2014.6907230.

[129] A. D. Kuo. Stabilization of lateral motion in passive dynamic walking. *The International journal of robotics research*, 18(9):917–930, 1999.

[130] J. Lack, M. J. Powell, and A. D. Ames. Planar multi-contact bipedal walking using hybrid zero dynamics. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2582–2588, May 2014. doi: 10.1109/ICRA.2014.6907229.

[131] J. Lack, M. J. Powell, and A. D. Ames. Planar multi-contact bipedal walking using hybrid zero dynamics. In *Robotics and Automation, IEEE International Conference on*, pages 2582–2588, May 2014. doi: 10.1109/ICRA.2014.6907229.

[132] D. Lehmann, E. Henriksson, and K. H. Johansson. Event-triggered model predictive control of discrete-time linear systems subject to disturbances. In *2013 European Control Conference (ECC)*, pages 1156–1161, 2013.

[133] C. Lemaréchal. *Lagrangian Relaxation*, pages 112–156. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001. ISBN 978-3-540-45586-8.

[134] N. Lissandrini, C. Verginis, P. Roque, A. Cenedese, and D. V. Dimarogonas. Decentralized nonlinear mpc for robust cooperative manipulation by heterogeneous aerial-ground robots. In *International Conference on Intelligent Robots and Systems (IROS) October 25-November 25, 2020.*, 2020.

[135] J. Liu, D. Munoz de la Peña, B. J. Ohran, P. D. Christofides, and J. F. Davis. A two-tier architecture for networked process control. *Chemical Engineering Science*, 63 (22):5394–5409, 2008.

[136] J. Liu, D. Munoz de la Pena, and P. D. Christofides. Distributed model predictive control of nonlinear process systems. *AIChE journal*, 55(5):1171–1184, 2009.

[137] J. Liu, D. Munoz de la Peña, B. J. Ohran, P. D. Christofides, and J. F. Davis. A two-tier control architecture for nonlinear process systems with continuous/asynchronous feedback. *International Journal of Control*, 83(2):257–272, 2010.

[138] J. Liu, L. Zhou, P. Tokekar, and R. K. Williams. Distributed resilient submodular action selection in adversarial environments. *IEEE Robotics and Automation Letters*, 6(3):5832–5839, 2021. doi: 10.1109/LRA.2021.3080629.

[139] S. G. Loizu, D. V. Dimarogonas, and K. J. Kyriakopoulos. Decentralized feedback stabilization of multiple nonholonomic agents. In *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, volume 3, pages 3012–3017. IEEE, 2004.

[140] W. Ma, K. Akbari Hamed, and A. D. Ames. First steps towards full model based motion planning and control of quadrupeds: A hybrid zero dynamics approach. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5498–5503. IEEE, 2019.

[141] W-L. Ma, N. Csomay-Shanklin, S. Kolathaya, K. Akbari Hamed, and A. D. Ames. Coupled control Lyapunov functions for interconnected systems, with application to quadrupedal locomotion. *IEEE Robotics and Automation Letters*, 6(2):3761–3768, 2021. doi: 10.1109/LRA.2021.3065174.

[142] C. Maes. *A regularized active-set method for sparse convex quadratic programming*. PhD thesis, Stanford University, 2010.

[143] J. M. Maestre, D. Muñoz De La Peña, and E. Camacho. Distributed model predictive control based on a cooperative game. *Optimal Control Applications and Methods*, 32 (2):153–176, 2011.

[144] M. Maggiore and L. Consolini. Virtual holonomic constraints for Euler Lagrange systems. *Automatic Control, IEEE Transactions on*, 58(4):1001–1008, April 2013. ISSN 0018-9286. doi: 10.1109/TAC.2012.2215538.

[145] I. R. Manchester, U. Mettin, F. Iida, and R. Tedrake. Stable dynamic walking over uneven terrain. *The International Journal of Robotics Research*, 30(3):265–279, 2011. doi: 10.1177/0278364910395339.

[146] A. E. Martin and R. D. Gregg. Hybrid invariance and stability of a feedback linearizing controller for powered prostheses. In *American Control Conference*, pages 4670–4676, 2015.

[147] A. E. Martin, D. C. Post, and J. P. Schmiedeler. The effects of foot geometric properties on the gait of planar bipeds walking under HZD-based control. *The International Journal of Robotics Research*, 33(12):1530–1543, 2014. doi: 10.1177/0278364914532391.

[148] A. E. Martin, D. C. Post, and J. P. Schmiedeler. Design and experimental implementation of a hybrid zero dynamics-based controller for planar bipeds with curved feet. *The International Journal of Robotics Research*, 33(7):988–1005, 2014.

[149] J. Martin, V. R. Kamidi, A. Pandala, R. Fawcett, and K. Akbari Hamed. Exponentially stabilizing and time-varying virtual constraint controllers for dynamic quadrupedal bounding. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3914–3921, 2020.

[150] MATLAB. Matlab optimization toolbox, 2020.

[151] T. McGeer. Passive dynamic walking. *I. J. Robotic Res.*, 9(2):62–82, 1990.

[152] T. McGeer. Passive walking with knees. In *Proceedings., IEEE International Conference on Robotics and Automation*, pages 1640–1645. IEEE, 1990.

[153] D. Mellinger, M. Shomin, N. Michael, and V. Kumar. Cooperative grasping and transport using multiple quadrotors. In *Distributed autonomous robotic systems*, pages 545–558. Springer, 2013.

[154] J. Mendez, S. Hood, A. Gunnel, and T. Lenzi. Powered knee and ankle prosthesis with indirect volitional swing control enables level-ground walking and crossing over obstacles. *Science Robotics*, 5(44), 2020.

[155] M. Mesbahi and Egerstedt. M. *Graph Theoretic Methods in Multiagent Networks*. Princeton University Press, 2010.

[156] A. Mohammadi, M. Maggiore, and L. Consolini. Dynamic virtual holonomic constraints for stabilization of closed orbits in underactuated mechanical systems. *Automatica*, 94:112 – 124, 2018. ISSN 0005-1098. doi: https://doi.org/10.1016/j.automatica.2018.04.023.

[157] B. Morris and J.W. Grizzle. Hybrid invariant manifolds in systems with impulse effects with application to periodic locomotion in bipedal robots. *Automatic Control, IEEE Transactions on*, 54(8):1751–1764, Aug 2009. ISSN 0018-9286. doi: 10.1109/TAC.2009.2024563.

[158] N. Motee and A. Jadbabaie. Distributed multi-parametric quadratic programming. *IEEE Transactions on Automatic Control*, 54(10):2279–2289, Sep 2009. doi: 10.1109/TAC.2009.2014916.

[159] M. Neunert, M. Stäuble, M. Giftthaler, C. D. Bellicoso, J. Carius, C. Gehring, M. Hutter, and J. Buchli. Whole-body nonlinear model predictive control through contacts for quadrupeds. *IEEE Robotics and Automation Letters*, 3(3):1458–1465, July 2018. ISSN 2377-3766. doi: 10.1109/LRA.2018.2800124.

[160] Quan Nguyen, Ayonga Hereid, Jessy W Grizzle, Aaron D Ames, and Koushil Sreenath. 3D dynamic walking on stepping stones with control barrier functions. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, pages 827–834. IEEE, 2016. URL http://ames.caltech.edu/3D_Stepping_Stones.pdf.

[161] Massachusetts Institute of Technology YouTube Channel. Mit cheetah robot lands the running jump, https://youtu.be/-Vt5u46WVSI, May 2015.

[162] D. E. Orin, A. Goswami, and S. Lee. Centroidal dynamics of a humanoid robot. *Autonomous robots*, 35(2):161–176, 2013.

[163] D. Panagou, M. Turpin, and V. Kumar. Decentralized goal assignment and trajectory generation in multi-robot networks: A multiple lyapunov functions approach. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6757–6762. IEEE, 2014.

[164] A. Pandala, V. R. Kamidi, and K. Akbari Hamed. Decentralized control schemes for stable quadrupedal locomotion: A decomposition approach from centralized controllers. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3975–3981, 2020.

[165] A. G. Pandala, Y. Ding, and H. Park. qpSWIFT: A real-time sparse quadratic program solver for robotic applications. *IEEE Robotics and Automation Letters*, 4(4):3355–3362, Oct 2019. ISSN 2377-3774. doi: 10.1109/LRA.2019.2926664.

[166] H.-W. Park, A. Ramezani, and J.W. Grizzle. A finite-state machine for accommodating unexpected large ground-height variations in bipedal robot walking. *Robotics, IEEE Transactions on*, 29(2):331–345, April 2013. ISSN 1552-3098. doi: 10.1109/TRO.2012.2230992.

[167] I. Poulakakis and J.W. Grizzle. The spring loaded inverted pendulum as the hybrid zero dynamics of an asymmetric hopper. *Automatic Control, IEEE Transactions on*, 54(8):1779–1793, Aug 2009. ISSN 0018-9286. doi: 10.1109/TAC.2009.2024565.

[168] I. Poulakakis, E. Papadopoulos, and M. Buehler. On the stability of the passive dynamics of quadrupedal running with a bounding gait. *The International Journal of Robotics Research*, 25(7):669–687, 2006.

[169] 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*, 20(2):129–143, 2001. doi: 10.1177/02783640122067309.

[170] J. Pratt, J. Carff, S. Drakunov, and A. Goswami. Capture point: A step toward humanoid push recovery. In *2006 6th IEEE-RAS International Conference on Humanoid Robots*, pages 200–207, Dec 2006. doi: 10.1109/ICHR.2006.321385.

[171] 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*, 31(10):1117–1133, 2012. doi: 10.1177/0278364912452762.

[172] D. Quintero, A. E. Martin, and R. D. Gregg. Unifying the gait cycle in the control of a powered prosthetic leg. In *2015 IEEE International Conference on Rehabilitation Robotics (ICORR)*, pages 289–294. IEEE, 2015.

[173] M. H. Raibert, H. B. Brown Jr, and M. Chepponis. Experiments in balance with a 3d one-legged hopping machine. *The International Journal of Robotics Research*, 3(2): 75–92, 1984.

[174] A. Ramezani, J. W. Hurst, K. Akbai Hamed, and J. W. Grizzle. Performance analysis and feedback control of ATRIAS, a three-dimensional bipedal robot. *Journal of Dynamic Systems, Measurement, and Control December, ASME*, 136(2), December 2013. doi: 10.1115/1.4025693.

[175] J. Reher, E. A. Cousineau, A. Hereid, C. M. Hubicki, and A. D. Ames. Realizing dynamic and efficient bipedal locomotion on the humanoid robot DURUS. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1794–1801, May 2016. doi: 10.1109/ICRA.2016.7487325.

[176] N. Sadati, G. A. Dumont, K. Akbari Hamed, and W. A. Gruver. Two-level control scheme for stabilisation of periodic orbits for planar monopedal running. *Control Theory Applications, IET*, 5(13):1528–1543, Sept 2011. ISSN 1751-8644. doi: 10.1049/iet-cta.2010.0512.

[177] N. Sadati, G. A. Dumont, K. Akbari Hamed, and Gruver W. A. *Hybrid Control and Motion Planning of Dynamical Legged Locomotion*. Wiley-IEEE Press, October 2012.

[178] C. O. Saglam and K. Byl. Meshing hybrid zero dynamics for rough terrain walking. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5718–5725, May 2015. doi: 10.1109/ICRA.2015.7140000.

[179] C.O. Saglam and K. Byl. Switching policies for metastable walking. In *Decision and Control, IEEE 52nd Annual Conference on*, pages 977–983, Dec 2013. doi: 10.1109/CDC.2013.6760009.

[180] N. Sandell, P. Varaiya, M. Athans, and M. Safonov. Survey of decentralized control methods for large scale systems. *IEEE Transactions on Automatic Control*, 23(2): 108–128, April 1978. doi: 10.1109/TAC.1978.1101704.

[181] C. Savorgnan, C. Romani, A. Kozma, and M. Diehl. Multiple shooting for distributed systems with applications in hydro electricity production. *Journal of Process Control*, 21(5):738–745, 2011.

[182] 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. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 225(6):831–849, 2011.

[183] C.-L. Shih, J. W. Grizzle, and C. Chevallereau. From stable walking to steering of a 3D bipedal robot with passive point feet. *Robotica*, 30:1119–1130, 12 2012. ISSN 1469-8668. doi: 10.1017/S026357471100138X.

[184] A. S. Shiriaev, A. Sandberg, and C. Canudas de Wit. Motion planning and feedback stabilization of periodic orbits for an acrobot. In *Decision and Control. 43rd IEEE Conference on*, volume 1, pages 290–295 Vol.1, Dec 2004. doi: 10.1109/CDC.2004.1428645.

[185] A. S. Shiriaev, J. W. Perram, and C. Canudas-de Wit. Constructive tool for orbital stabilization of underactuated nonlinear systems: Virtual constraints approach. *IEEE Transactions on Automatic Control*, 50(8):1164–1176, Aug 2005. ISSN 0018-9286. doi: 10.1109/TAC.2005.852568.

[186] A. S. Shiriaev, L. B. Freidovich, and I. R. Manchester. Periodic motion planning and analytical computation of transverse linearizations for hybrid mechanical systems. In *2008 47th IEEE Conference on Decision and Control*, pages 4326–4331. IEEE, 2008.

[187] A.S. Shiriaev, L.B. Freidovich, and S.V. Gusev. Transverse linearization for controlled mechanical systems with several passive degrees of freedom. *IEEE Transactions on Automatic Control*, 55(4):893–906, April 2010. ISSN 0018-9286. doi: 10.1109/TAC.2010.2042000.

[188] D. D. Siljak. *Decentralized Control of Complex Systems*. Dover Publications, December 2011.

[189] A. M. Simon, K. A. Ingraham, N. P. Fey, S. B. Finucane, R. D. Lipschutz, A. J. Young, and L. J. Hargrove. Configuring a powered knee and ankle prosthesis for transfemoral amputees within five specific ambulation modes. *PloS ONE*, 9(6):e99387, 2014.

[190] A. M. Simon, K. A. Ingraham, N. P. Fey, S. B. Finucane, R. D. Lipschutz, A. J. Young, and L. J. Hargrove. Configuring a powered knee and ankle prosthesis for transfemoral amputees within five specific ambulation modes. *PloS one*, 9(6):e99387, 2014.

[191] R. W. Sinnet and A. D. Ames. Extending two-dimensional human-inspired bipedal robotic walking to three dimensions through geometric reduction. In *American Control Conference (ACC), 2012*, pages 4831–4836. IEEE, 2012. URL http://ames.caltech.edu/Reduction_ACC12_final.pdf.

[192] G. Song and M. Zefran. Underactuated dynamic three-dimensional bipedal walking. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 854–859, May 2006. doi: 10.1109/ROBOT.2006.1641816.

[193] M. W. Spong. Passivity based control of the compass gait biped. *IFAC Proceedings Volumes*, 32(2):506–510, 1999.

[194] M. W. Spong and F. Bullo. Controlled symmetries and passive walking. *IEEE Transactions on Automatic Control*, 50(7):1025–1031, July 2005. ISSN 0018-9286. doi: 10.1109/TAC.2005.851449.

[195] M. W. Spong, J. K. Holm, and D. Lee. Passivity-based control of bipedal locomotion. *Robotics Automation Magazine, IEEE*, 14(2):30–40, June 2007. ISSN 1070-9932. doi: 10.1109/MRA.2007.380638.

[196] K. Sreenath and V. Kumar. Dynamics, control and planning for cooperative manipulation of payloads suspended by cables from multiple quadrotor robots. *rn*, 1(r2):r3, 2013.

[197] K. Sreenath, H.-W. Park, I. Poulakakis, and J. W. Grizzle. Compliant hybrid zero dynamics controller for achieving stable, efficient and fast bipedal walking on MABEL. *The International Journal of Robotics Research*, 30(9):1170–1193, August 2011.

[198] K. Sreenath, H.-W. Park, I. Poulakakis, and J.W. Grizzle. Embedding active force control within the compliant hybrid zero dynamics to achieve stable, fast running on MABEL. *The International Journal of Robotics Research*, 32(3):324–345, 2013. doi: 10.1177/0278364912473344.

[199] G. W. Stewart. *Matrix Algorithms: Volume 1: Basic Decompositions*, volume 1. Siam, 1998.

[200] F. Sup, H. A. Varol, and M. Goldfarb. Upslope walking with a powered knee and ankle prosthesis: Initial results with an amputee subject. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 19(1):71–78, 2011.

[201] F. Tan, C. Fu, and K. Chen. Biped blind walking on changing slope with reflex control system. In *2010 IEEE International Conference on Robotics and Automation*, pages 1709–1714. IEEE, 2010.

[202] R. Tedrake, T.W. Zhang, and H.S. Seung. Stochastic policy gradient reinforcement learning on a simple 3D biped. In *Intelligent Robots and Systems. Proceedings 2004 IEEE/RSJ International Conference on*, volume 3, pages 2849–2854 vol.3, Sept 2004. doi: 10.1109/IROS.2004.1389841.

[203] B. R. Trilaksono. Distributed consensus control of robot swarm with obstacle and collision avoidance. In *2015 2nd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, pages 2–2. IEEE, 2015.

[204] M. R. Tucker, J. Olivier, A. Pagel, H. Bleuler, M. Bouri, O. Lambercy, J. del R Millán, R. Riener, H. Vallery, and R. Gassert. Control strategies for active lower extremity prosthetics and orthotics: a review. *Journal of neuroengineering and rehabilitation*, 12 (1):1–30, 2015.

[205] A. N. Venkat, J. B. Rawlings, and S. J. Wright. Stability and optimality of distributed model predictive control. In *Proceedings of the 44th IEEE Conference on Decision and Control*, pages 6680–6685. IEEE, 2005.

[206] D. J. Villarreal, H. A. Poonawala, and R. D. Gregg. A robust parameterization of human gait patterns across phase-shifting perturbations. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 25(3):265–278, March 2017. ISSN 1534-4320. doi: 10.1109/TNSRE.2016.2569019.

[207] O. Villarreal, V. Barasuol, P. Wensing, and C. Semini. MPC-based controller with terrain insight for dynamic legged locomotion. *arXiv preprint arXiv:1909.13842*, 2019.

[208] A. Wächter and L. T. Biegler. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1):25–57, 2006.

[209] S.-H. Wang and E. Davison. On the stabilization of decentralized control systems. *IEEE Transactions on Automatic Control*, 18(5):473–478, Oct 1973. ISSN 0018-9286. doi: 10.1109/TAC.1973.1100362.

[210] Y. Wang and S. Boyd. Fast model predictive control using online optimization. *IEEE Transactions on control systems technology*, 18(2):267–278, 2009.

[211] E. R. Westervelt, J. W. Grizzle, and C. C. De Wit. Switching and pi control of walking motions of planar biped walkers. *Automatic Control, IEEE Transactions on*, 48(2): 308–312, Feb 2003. ISSN 0018-9286. doi: 10.1109/TAC.2002.808489.

[212] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek. Hybrid zero dynamics of planar biped walkers. *IEEE Transactions on Automatic Control*, 48(1):42–56, Jan 2003. ISSN 0018-9286. doi: 10.1109/TAC.2002.806653.

[213] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris. *Feedback Control of Dynamic Bipedal Robot Locomotion*. Taylor & Francis/CRC, 2007.

[214] E. R. Westervelt, B. Morris, and K. D. Farrell. Analysis results and tools for the control of planar bipedal gaits using hybrid zero dynamics. *Autonomous Robots*, 23 (2):131–145, 2007.

[215] R. K. Williams, A. Gasparri, and G. Ulivi. Decentralized matroid optimization for topology constraints in multi-robot allocation problems. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 293–300. IEEE, 2017.

[216] A. W. Winkler, C. Mastalli, I. Havoutis, M. Focchi, D. G. Caldwell, and C. Semini. Planning and execution of dynamic whole-body locomotion for a hydraulic quadruped on challenging terrain. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5148–5154. IEEE, 2015.

[217] P. Yang, R. A. Freeman, and K. M. Lynch. Multi-agent coordination by decentralized estimation and control. *IEEE Transactions on Automatic Control*, 53(11):2480–2496, 2008.

[218] H. Ye, A. N. Michel, and L. Hou. Stability theory for hybrid dynamical systems. *Automatic Control, IEEE Transactions on*, 43(4):461–474, Apr 1998. ISSN 0018-9286. doi: 10.1109/9.664149.

[219] H. Zhao, J. Horn, J. Reher, V. Paredes, and A. D. Ames. Multicontact locomotion on transfemoral prostheses via hybrid system models and optimization-based control. *IEEE Transactions on Automation Science and Engineering*, 13(2):502–513, 2016.

[220] H. Zhao, E. Ambrose, and A. D. Ames. Preliminary results on energy efficient 3D prosthetic walking with a powered compliant transfemoral prosthesis. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1140–1147, 2017.