Modeling, Planning, and Control of Quadrupedal Locomotion on Dynamic Rigid Surfaces with Unknown Vertical Motions

Dissertation by

Amir Iqbal

Submitted in partial fulfillment of the requirements for the
degree of
Doctor of Philosophy in Mechanical Engineering

Francis College of Engineering
Department of Mechanical Engineering
University of Massachusetts Lowell

February 2024
Defended November 9, 2023

Modeling, Planning, and Control of Quadrupedal Locomotion on Dynamic Rigid Surfaces with Unknown Vertical Motions

Dissertation by

Amir Iqbal

Submitted in partial fulfillment of the requirements for the
degree of
Doctor of Philosophy in Mechanical Engineering
Francis College of Engineering
Department of Mechanical Engineering
University of Massachusetts Lowell

Author . . . . . .
November 09, 2023

Certified by . . . . . .
Christopher Niezrecki, Ph. D.
Distinguished University Professor, Department of Mechanical Engineering
Dissertation Advisor

Certified by . . . . . .
Yan Gu, Ph. D.
Associate Professor, School of Mechanical Engineering, Purdue University
Dissertation Co-Advisor

Certified by . . . . . .
Kshitij Jerath, Ph. D.
Assistant Professor, Department of Mechanical Engineering
Dissertation Committee Member

Certified by . . . . . .
Kelilah Wolkowicz, Ph. D.
Assistant Professor, Department of Mechanical Engineering
Dissertation Committee Member

Certified by . . . . . .
Sushant Veer, Ph. D.
Senior Research Scientist, NVIDIA Research
Dissertation Committee Member

Modeling, Planning, and Control of Quadrupedal Locomotion on Dynamic Rigid Surfaces with Unknown Vertical Motions

by

Amir Iqbal

Abstract of a dissertation submitted to the faculty of the
Francis College of Engineering
Department of Mechanical Engineering
in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy
University of Massachusetts Lowell
2023

Dissertation Advisor: Christopher Niezrecki, Ph. D.
Title: Distinguished University Professor, Department of Mechanical Engineering

Dissertation Co-Advisor: Yan Gu, Ph. D.
Title: Associate Professor, School of Mechanical Engineering, Purdue University

Dissertation Committee Member: Kshitij Jerath, Ph. D.
Title: Assistant Professor, Department of Mechanical Engineering

Dissertation Committee Member: Kelilah Wolkowicz, Ph. D.
Title: Assistant Professor, Department of Mechanical Engineering

Dissertation Committee Member: Sushant Veer, Ph. D.
Title: Senior Research Scientist, NVIDIA Research

Abstract

Legged robots move around by using their legs to make and break contact with the ground. Thanks to this unique form of locomotion, legged robots have the great potential to navigate over difficult terrains and assist humans in performing demanding tasks in hazardous environments. However, the existing approaches of dynamic modeling, motion planning, and control design do not explicitly address the challenges of legged locomotion on dynamic rigid surfaces (i.e., rigid surfaces that move in the inertial frame). This dissertation aims to bridge the existing knowledge gaps in the modeling, planning, and control of legged robot locomotion on dynamic rigid surfaces (DRS). To achieve the overarching goal, this dissertation comprises the following three main studies.

The first study derives a full-order dynamic model of legged walking on DRS, proposes a provably stabilizing controller, and validates the modeling and control framework in simulations and hardware experiments. The control approach is synthesized based on the formulation of the full-order robot model as a hybrid, time-varying system. The stability analysis of the closed-loop control system is performed through the construction of multiple Lyapunov functions. The validation results in simulations and hardware experiments confirm the effectiveness of the proposed control approach in guaranteeing the stability and robustness of a quadrupedal robot walking on DRS with known periodic motions. Still, this study relies on computationally expensive offline trajectory planning, which is unsuitable for real-world applications where frequent replanning is typically demanded to ensure locomotion robustness under uncertainties. To that end, the subsequent contributions aim to realize efficient trajectory planning for real-time applications.

The second study derives a reduced-order dynamic model of legged walking on DRS and introduces an approximate analytical solution to the model under vertical sinusoidal DRS motion (e.g., ship motion in regular sea waves). Furthermore, the study designs a hierarchical planner that exploits the proposed analytical solution to enable real-time, physically feasible motion generation for locomotion on DRS. The validation results support the efficiency and accuracy of the proposed solution in simulations, as well as the efficiency and physical feasibility of the proposed planner through 3-D realistic simulations and hardware experiments. Yet, this framework only solves the real-time locomotion planning problem for DRS with sinusoidal vertical movements, which may not be suitable for DRS with general motions. To overcome this limitation, the last focus of this dissertation research is on legged locomotion under unknown general (periodic or aperiodic) vertical DRS movements.

The final study of this dissertation presents a hierarchical control framework to enable robust legged locomotion on DRSes with general unknown vertical motions. The key novelty of the framework lies in its higher layer, which is a discrete-time, robustly stabilizing footstep controller. One basis of the footstep controller is a new hybrid, time-varying, linear inverted pendulum (HT-LIP) model that is computationally efficient and accurately captures the essential hybrid dynamics of DRS locomotion. Also, a new set of sufficient stability conditions is derived to directly guide the controller design for ensuring the robust stability of the HT-LIP model under general, uncertain, vertical DRS motions. Further, the footstep controller is cast as a quadratic program, integrating the proposed HT-LIP model and essential stability conditions. The middle layer of the framework takes the

desired footstep locations as input to produce kinematically feasible whole-body reference trajectories, which are then accurately tracked by a lower-layer full-body torque controller. Hardware experiments on a Unitree Go1 quadrupedal robot confirm the robustness of the proposed framework under various general vertical DRS motions and uncertainties (e.g., slippery and uneven surfaces, external solid and liquid load, unknown sway motion, and sudden pushes). The studies presented in this dissertation showcase the evolution of research to solve the theoretical knowledge gaps and practical challenges in designing and implementing a control framework that enables robust quadrupedal robot locomotion on dynamic surfaces with unknown general vertical motion.

Acknowledgments

I would like to extend my deepest gratitude to my primary research advisor, Professor Yan Gu. My academic journey would not have been possible without your guidance, support, and unwavering commitment. Your dedication to our research, willingness to challenge and inspire me, and commitment to excellence have left an indelible mark on my academic experience. Your ability to provide constructive feedback and encouragement to explore new ideas has been invaluable. Despite administrative constraints that prevent me from formally mentioning Professor Gu as my advisor on this dissertation, she is at the core of everything I have achieved in my Ph.D., and her impact on my research cannot be overstated. I would also like to thank my advisor, Professor Christopher Niezrecki, whose support has been instrumental in navigating through the final phase of my Ph.D. program. His prompt responses to my queries, expert feedback on my research, and astute guidance and support in overcoming critical transitional hurdles have been invaluable.

My sincere thanks go to my committee members, Professor Kshitij Jerath, Professor Kelilah Wolkowicz, and Dr. Sushant Veer, for their critical examination of my dissertation research impact and valuable feedback on my research presentations. Specifically, I would like to thank Dr. Veer for his guidance in shaping my research direction, giving constructive and critical comments on my research writing, and mentoring me to strike a balance between the present tasks and future goals.

Thanks to the University of Massachusetts Lowell and Purdue University for infrastructure and administrative support. Additionally, I extend my gratitude to the National Science Foundation and the Office of Naval Research for their funding that supported this research.

I express my gratitude to my fellow Ph.D. lab mates, including Yuan, Zenan, Zijian, and Ruochen, and the undergraduate and master's students – Matt, Luisa, James, Aaron, and Leo – for their invaluable support and contributions to our research. A special thanks to Yuan for creating a welcoming lab environment and being a true friend in both good and challenging times. Zenan, I want to express my gratitude for your unwavering friendship and reliable support. I thank Alok for feedback, discussions, and planning the trip to California National Parks. Hossein, my go-to neighbor for Ph.D.-related matters, deserves my thanks. Finally, I extend my appreciation to countless friends, professors, and colleagues outside of my lab who have been sources of encouragement and support.

I would like to extend my deepest appreciation to my wife, Fahina, for her immense sacrifice, unwavering solace, and constant inspiration throughout this academic journey. I am also grateful for the precious gift of our son, Azfar – a blessing that she brought into our lives with remarkable strength and grace, even in the face of my limited availability to support her during this time. Her love, understanding, and resilience have been my greatest source of motivation and strength, and for that, I am profoundly thankful. My heartfelt appreciation goes to my mother, Nizhat Amna, whose love and strength served as my constant support pillars, and to my sisters and brothers, who consistently supported my pursuit of knowledge. In loving memory, I dedicate this dissertation to my father, Sarfaraz Ahmad, who, though no longer with us, continues to be a source of inspiration and a driving force behind my pursuit of knowledge. He not only celebrated my achievements but also instilled in me the ethos of making a positive impact in the lives of others, a lesson I hold close to my heart.

Publications

Peer-Reviewed Journal Publications:

(1) Iqbal, A., Gao, Y. and Gu, Y., 2020. Provably Stabilizing Controllers for Quadrupedal Robot Locomotion on Dynamic Rigid Platforms. *IEEE/ASME Transactions on Mechatronics*, 25(4), pp.2035-2044.

(2) Iqbal, A., Mao, Z. and Gu, Y., 2021. Modeling, Analysis, and Control of SLIP Running on Dynamic Platforms. *ASME Letters in Dynamic Systems and Control*, 1(2): 021007.

(3) Iqbal, A., Veer, S. and Gu, Y., 2023. Analytical Solution to a Time-Varying LIP Model for Quadrupedal Walking on a Vertically Oscillating Surface. *IFAC Mechatronics*, 96(103073).

(4) Iqbal, A., Veer, S., Niezrecki, C. and Gu, Y., 2023. HT-LIP Model based Robust Control of Quadrupedal Robot Locomotion under Unknown Vertical Ground Motion. *In preparation*.

Conference Papers and Poster Presentations:

(1) Iqbal, A., Veer, S., Niezrecki, C. and Gu, Y., 2023. HT-LIP based Walking Pattern Generation for Robust Quadrupedal Locomotion on Rigid Surfaces with General Vertical Motion. In Proc. of *IEEE/RSJ International Conference on Intelligent Robots and Systems*.

(2) Iqbal, A., Veer, S. and Gu, Y., 2023. Robust Quadrupedal Locomotion through Asymptotic Stabilization of H-LIP on Dynamic Rigid Surfaces with General Vertical

Motion. In Proc. of *IEEE/ASME International Conference on Advance and Intelligent Mechatronics*.

(3) Iqbal, A., Veer, S. and Gu, Y., 2023. Asymptotic stabilization of aperiodic trajectories of a hybrid-linear inverted pendulum walking on a vertically moving surface. In Proc. of *American Control Conference, pp. 3030-3035*.

(4) Iqbal, A., Mao, Z. and Gu, Y., 2019, October. Modeling, Analysis, and Control of SLIP Running on Dynamic Platforms. In Proc. of *ASME Dynamic Systems and Control Conference*.

(5) Iqbal, A., Gao, Y. and Gu, Y., 2020. Provably Stabilizing Controllers for Quadrupedal Robot Locomotion on Dynamic Rigid Platforms. In Proc. of *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*.

(6) Iqbal, A. and Gu, Y., 2021. Extended Capture Point and Optimization-Based Control for Quadrupedal Robot Walking on Dynamic Rigid Surfaces. In Proc. of *IFAC Modeling, Estimation and Control Conference, 54(20), pp. 72-77*.

(7) Iqbal, A. and Gu, Y., 2021. Extending Capture Point to Dynamic Rigid Surfaces. In Proc. of *Dynamic Walking Conference*.

(8) Iqbal, A., Veer, S. and Gu, Y., 2022. Analytical Approximate Solution to Mathieu's Equation Enables Real-Time Motion Planning for Legged Robot Walking on a Vertically Moving Surface. In Proc. of *Dynamic Walking Conference*.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Legged robots are robots that move around by using their legs to make and break contact with the ground. Because of this unique form of locomotion, legged robots could potentially negotiate terrains that are prohibitively challenging for wheeled or tracked robots, such as ladders, gaps, and stairs. For this reason, legged robots capable of robust locomotion in complex environments could benefit a wide range of critical applications such as firefighting, emergency response [2, 3], maintenance [4, 5], and inspection [6, 7].

While dynamic modeling, motion planning, and control have been extensively studied for legged locomotion on stationary surfaces [8, 9, 10, 11], locomotion on a rigid surface that moves in the inertial frame, which we call a "dynamic rigid surfaces (DRS)," remains a new legged robot functionality that has not been fully solved. Real-world examples of DRS include ships, oil platforms, public transportation vehicles, trains, and aircraft. Legged robots that can reliably move on such surfaces could be used to assist humans in performing challenging and dangerous missions such as maintenance and service operations on vessels, firefighting and fire suppression on ships and oil platforms [12, 13, 14], and safety and surveillance tasks on moving public transit vehicles [15, 16, 17].

However, modeling, planning, and control for legged locomotion on DRS are fundamentally challenging due to (a) high-dimensional, hybrid, and nonlinear walking dynamics [18, 19] and (b) time-varying unknown movement of the robot's contact points on the

surface [20].

Towards solving this fundamental challenge for achieving reliable legged locomotion performance on DRS, this dissertation draws upon dynamic modeling, nonlinear control theory, theory of hybrid systems, and optimization to create new methods of dynamic modeling, motion planning, and control design for DRS locomotion. Furthermore, this dissertation extensively validates the proposed methods in hardware experiments.

1.2 Literature Review and Knowledge Gaps

Enabling reliable legged locomotion on DRS requires appropriate methods of dynamic modeling, motion planning, and controller design. To that end, this chapter briefly reviews existing modeling, planning, and control approaches to legged robot locomotion on stationary surfaces and summarizes the associated knowledge gaps for enabling reliable DRS locomotion.

1.2.1 Dynamic Modeling

This section presents various existing dynamic models of legged locomotion on static surfaces that have been utilized to develop model-based robot planners and controllers. A model is a simplified representation of a real-world system expressed by mathematical equations. In legged robotics, models are utilized to describe or predict system behaviors, develop planning and control algorithms, and analyze stability and safety. A full-order model is a detailed model of a legged robot's kinematics and dynamics and is often computationally expensive and unsuitable for real-time planning and control. In contrast, reduced-order models are simplified models of the system with reduced degrees of freedom. Although reduced-order models are generally less accurate than full-order models, the low dimensionality of reduced-order models makes them more computationally efficient than full-order models and thus more suitable for real-time planning and control.

In purely mathematical terms, the dynamic model of legged locomotion during continuous phases (when the swing foot moves in the air) is described as the following nonlinear

dynamical process with state $\mathbf{x}$ and input $\mathbf{u}$:

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)). \tag{1.1}$$

The model in (1.1) predicts the evolution of state $\mathbf{x}$ in response to the input $\mathbf{u}$. This general mathematical representation can be used as the basis for all the models discussed in the following subsections.

(1) Linear Inverted Pendulum Model

A widely used reduced-order model of legged robot locomotion is the classical three-dimensional (3D) inverted pendulum model illustrated in Fig. 1-1.

The model represents the dynamics of a point mass atop a massless leg and can be transformed into a 3D linear inverted pendulum (LIP) model [21] under the following assumptions:

(A1) The point mass of the LIP maintains a constant vertical height with respect to the support point of the LIP.

(A2) The centroidal angular momentum of the robot is negligible (or the angular velocity and angular accelerations are negligibly small).

(A3) The walking surface is rigid and flat (i.e., all the successive steps of the robot are at constant height).

Under assumptions (A1)-(A3), the LIP dynamics become:

$$\ddot{x} = \frac{g}{z_0} x \text{ and } \ddot{y} = \frac{g}{z_0} y, \tag{1.2}$$

where x- and y- are the horizontal coordinates of the center of mass (CoM) with respect to the support point S, g is the magnitude of the acceleration due to gravity, and z_0 is the vertical (aligned with gravity vector) height between the support point and the CoM. This linear model is an approximate representation of legged robot dynamics on static surfaces.

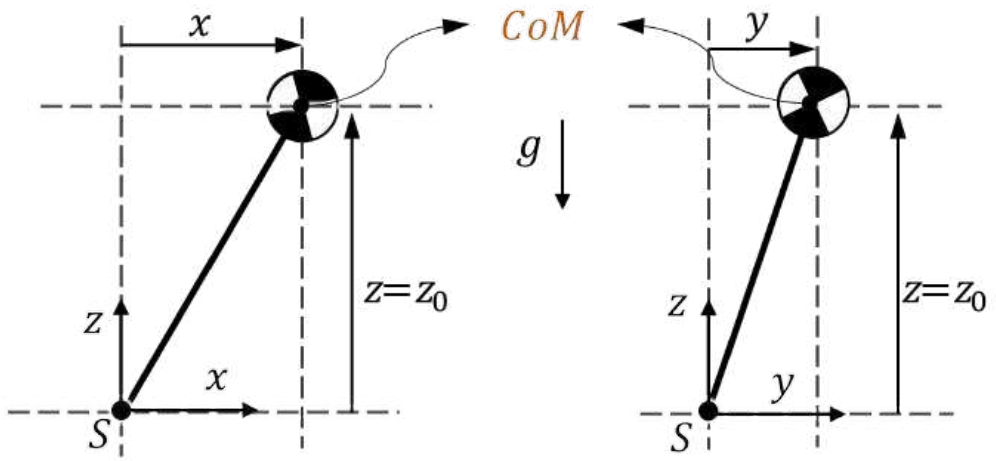

Figure 1-1: LIP model on a rigid surface.

The LIP model has been extensively used for walking pattern generation and control design of legged robots such as bipeds [21, 10, 22] and quadrupeds [23, 24]. Due to the simplicity of the LIP, it has been used as a basis to form balance [10] and stability [25] criteria, and its inherent connection with another widely studied reduced-order model, hybrid zero dynamics [26], has recently been analyzed [27]. Moreover, the simplicity of the model helps in elegantly defining key concepts of walking dynamics, e.g., capture point [25], capturability [28, 29], the divergent component of motion (DCM) [30], etc. These definitions are exploited for motion panning of the actual robot. The LIP model has been extended to more complex scenarios by considering a varying CoM height [31], CoM movement on a 3-D plane [32], nontrivial centroidal angular momentum [25], hybrid robot dynamics [33], and walking on DRS [34].

(2) Spring-Loaded Inverted Pendulum Model

The spring-loaded inverted pendulum (SLIP) model [35, 36, 37] is a classical reduced-order model for bouncy-legged animal locomotion [38, 39]. The spring of the SLIP model eliminates impulsive impact at touchdown and recycles kinetic energy during the stance phase, which helps explain the efficient legged animal locomotion [35, 37]. The SLIP model can also be used to explain insect locomotion dynamics [40, 41]. The running and hopping models in the sagittal plane of locomotion on a rigid flat surface can be expressed based on

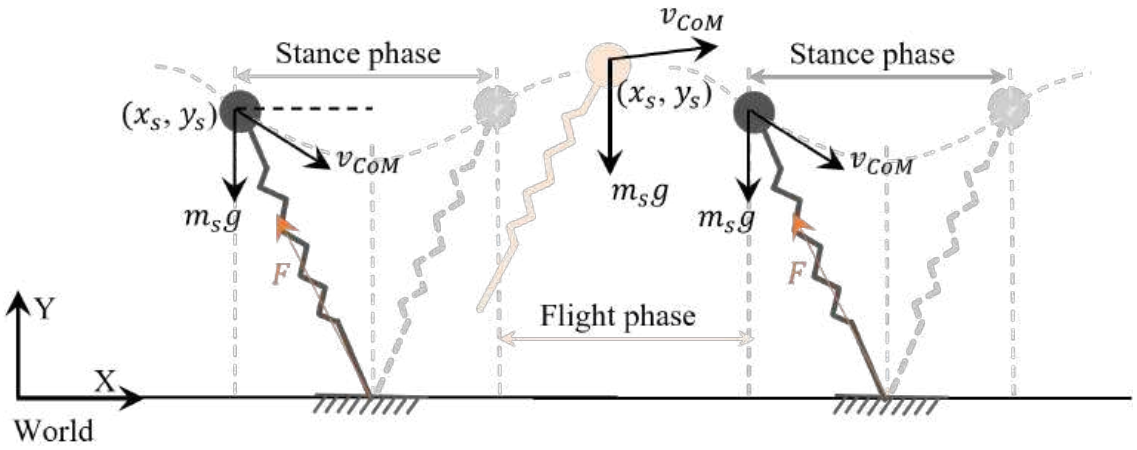

Figure 1-2: SLIP model on a rigid surface.

a SLIP model [42, 43]. The SLIP dynamics are simplified under the following assumption:

(A4) The spring leg of the SLIP model is massless.

During a flight phase, the SLIP dynamics are only affected by gravity (see Fig. 1-2) and under the assumption (A4), it is given as:

$$\frac{d^2x_s(t)}{dt^2} = 0 \text{ and } \frac{d^2y_s(t)}{dt^2} = -g\ , \tag{1.3}$$

where x_s and y_s represent the x- and y- positions of the SLIP center of mass, respectively, g is the magnitude of gravitational acceleration, and v_{CoM} in Fig. 1-2 shows the direction of CoM velocity. The stance-phase dynamics of the SLIP are given by:

$$\frac{d^2x_s(t)}{dt^2} = \frac{-F_x(t)}{m_s} \text{ and } \frac{d^2y_s(t)}{dt^2} = -g + \frac{-F_y(t)}{m_s}, \tag{1.4}$$

where F_x and F_y are the components of spring force along x and y direction of the world frame.

(3) Centroidal Dynamics Model

The centroidal dynamics model is simplified under the following assumptions:

(A5) All the segments of the robot are rigid bodies.

(A6) The contact surface is rigid.

Under the assumptions (A5) and (A6), the centroidal dynamics model expresses the rate of change of momentum about a reference frame anchored at the centroid (i.e., current CoM) in terms of all external forces as:

$$\mathbf{A}\ddot{\mathbf{q}} + \dot{\mathbf{A}}\dot{\mathbf{q}} = \begin{bmatrix} m\mathbf{g} + \sum_{c=1}^{n_c} \mathbf{f}_c \\ \sum_{c=1}^{n_c} (\mathbf{p}_c - \mathbf{r}) \times \mathbf{f}_c \end{bmatrix}, \tag{1.5}$$

where $\mathbf{A} \in \mathbb{R}^{6\times n}$ is the centroidal momentum matrix [44], m is the total mass of the robot, $\mathbf{g}$ is the gravitational vector, n is the number of generalized coordinates, $\mathbf{f}_c$ is the contact force at contact point c, n_c is the number of contact points, $\mathbf{r}$ is the position vector of the current CoM with respect to a world frame, and $\mathbf{p}_c$ is the position vector of contact points with respect to a world frame. $\mathbf{q}, \dot{\mathbf{q}}$, and $\ddot{\mathbf{q}}$ are the generalized position, velocity, and acceleration of the robot, respectively.

The centroidal dynamics model is a more general and reliable description of the robot dynamics compared to the LIP as it accounts for orientation dynamics and does not constrain the CoM position trajectory. However, the accuracy comes at a cost, as the centroidal dynamics model can be nonlinear and computationally expensive [44]. This model has been used for locomotion planning and control of bipeds [45, 46] and quadrupeds [47, 48].

For high-dimensional robots, the planning and control based on the complete centroidal dynamics model are slower than the LIP model [49], which may make this model unsuitable for a dynamic real-world application with limited computing capacity. This is because the model transfers the momentum contribution of each rigid body of the robots to the centroid, which is computationally expensive. The reduced-order model introduced next is a simplification of the centroidal dynamics model and has been widely used for quadrupedal planning and control.

(4) Single Rigid Body Dynamics Model

The single rigid body dynamics (SRBD) model is a simplification of the centroidal dynamics model as it assumes that the centroidal momentum contributions of joint velocity are

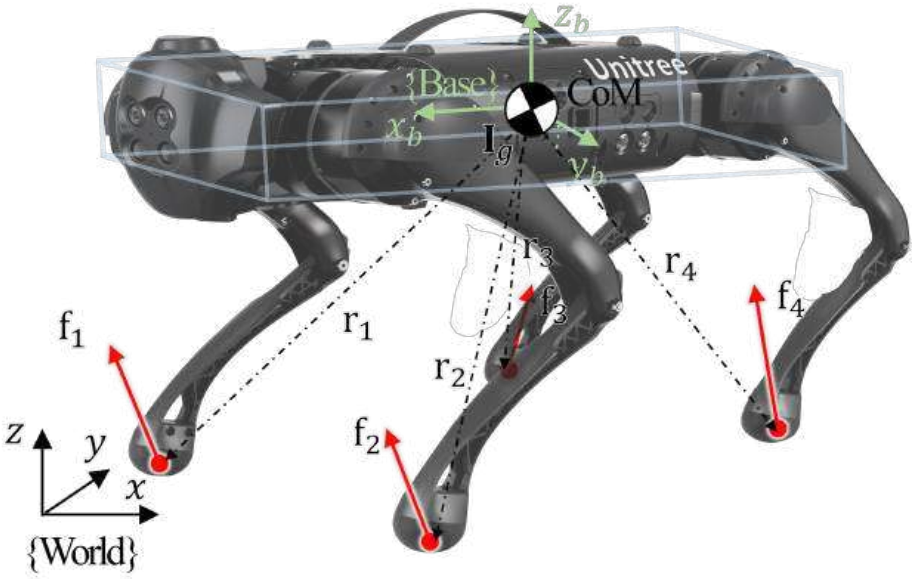

Figure 1-3: An illustration of the SRBD model of a quadrupedal robot.

negligible and the whole-body inertia of the robot remains similar to one of its nominal configurations [50]. An illustration of the SRBD model is shown in Fig. 1-3. The simplifying assumptions of the SRBD model are the following:

(A7) The centroidal momentum contributions of joint velocity are negligible.

(A8) The whole-body inertia of the robot remains similar to one of its nominal configurations.

The SRBD model of a robot can be given as [51]:

$$\begin{bmatrix} \sum_{c=1}^{n_c} \mathbf{f}_c \\ \sum_{c=1}^{n_c} (\mathbf{p}_c - \mathbf{r}) \times \mathbf{f}_c \end{bmatrix} = \begin{bmatrix} m(\mathbf{g} + \ddot{\mathbf{r}}) \\ \mathbf{I}_g \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}_g \boldsymbol{\omega} \end{bmatrix}, \tag{1.6}$$

where $\mathbf{I}_g$ is the centroidal moment of inertia of the robot, m is the total mass of the robot, $\mathbf{g}$ is the gravitational vector, $\mathbf{f}_c$ is the contact force at contact point c, n_c is the number of contact points, $\mathbf{r}$ is the position vector of current CoM with respect to a world frame, $\mathbf{p}_c$ is the position vector of contact points with respect to the world frame, and $\dot{\boldsymbol{\omega}}$ is the vector of the angular acceleration of the base.

Such a model is low-dimensional and more accurate than the LIP model. A quadruped

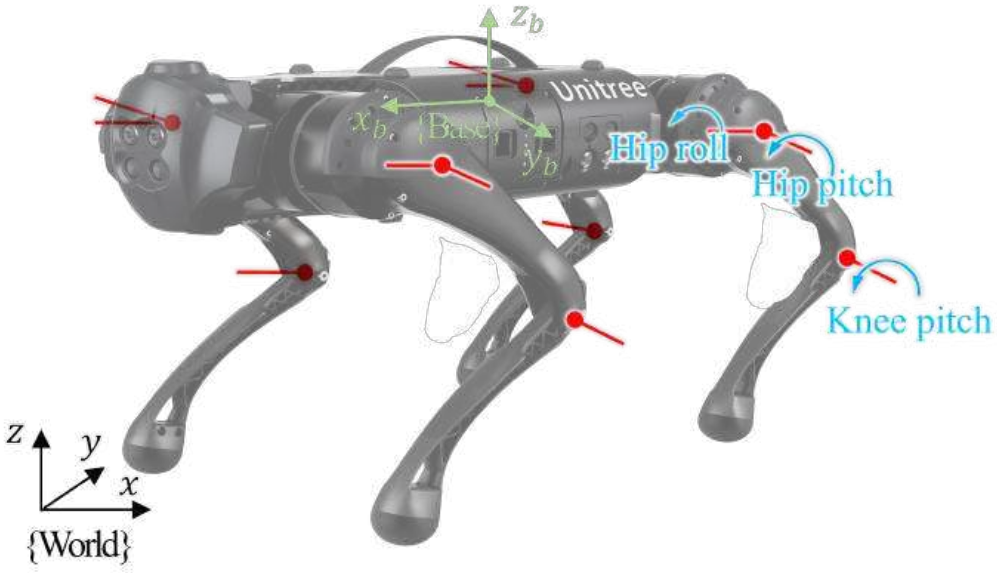

Figure 1-4: Full-order quadrupedal robot model.

with lighter limbs and a heavier trunk tends to faithfully satisfy assumptions (A7) and (A8). Thus, many quadrupeds modeled as SRBD show impressive real-world performance in static, flat, or uneven terrain [1, 51].

(5) Full-Order Dynamic Model

Full-order models are a detailed representation of a legged robot's kinematics and dynamics. As many of today's legged robots are made of highly stiff links, their full-order models typically assume that all the links of a multi-body robot are rigid, as outlined in the assumption (A5). An illustration of a full-order quadrupedal model is shown in Fig. 1-4. The continuous-time equation of motion of such a model is derived using Lagrange's method [52, 20]. The unconstrained full-order robot dynamics can be given as:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) := \mathbf{B}\mathbf{u} + \mathbf{J}_c^T\mathbf{F}_c, \tag{1.7}$$

where $\mathbf{M}(\mathbf{q}) : \mathcal{Q} \to \mathbb{R}^{(n+6)\times(n+6)}$ is the inertia matrix, $\mathbf{C}(\mathbf{q},\dot{\mathbf{q}}) : \mathcal{TQ} \to \mathbb{R}^{(n+6)\times 1}$ represents the sum of Coriolis, centrifugal, and gravitational terms, $\mathbf{B} \in \mathbb{R}^{(n+6)\times m}$ is the actuator selection matrix, $\mathbf{F}_c \in \mathbb{R}^{n_{ct}}$ is the generalized external force acting at the contact points

between the robot and its environment, and $\mathbf{J}_c(\mathbf{q})$ is the Jacobian of contact points. Here, $\mathcal{Q}$ is the configuration space and $\mathcal{TQ}$ is the tangent bundle of $\mathcal{Q}$.

For locomotion on stationary rigid surfaces, the contact constraints enforce the no-slip condition between the contact points of the robot and the surface by the holonomic constraints expressed as:

$$\mathbf{J}_c\ddot{\mathbf{q}} + \dot{\mathbf{J}}_c\dot{\mathbf{q}} = \mathbf{0}. \tag{1.8}$$

Discrete impact dynamics: During a swing-foot landing event, an impact between the foot and the ground leads to an instantaneous jump in velocity $\dot{\mathbf{q}}$. The position $\mathbf{q}$ stays continuous due to the instantaneous nature of the jump event. By integrating the dynamics (1.7) and constraints (1.8) across the impact event, we get:

$$\mathbf{M}(\dot{\mathbf{q}}^+ - \dot{\mathbf{q}}^-) = \mathbf{J}_c^T \delta\mathbf{F}_c \text{ and } \mathbf{J}_c\dot{\mathbf{q}}^+ = \mathbf{0}, \tag{1.9}$$

where $\star^-$ and $\star^+$ represent the values of $\star$ right before and after the impact, respectively, $\delta\mathbf{F}_c$ is the impulsive impact force. Rearranging the above equation gives:

$$\begin{bmatrix} \dot{\mathbf{q}}^+ \\ \delta\mathbf{F}_c \end{bmatrix} = \begin{bmatrix} \mathbf{M}(\mathbf{q}) & -\mathbf{J}_c^T(\mathbf{q}) \\ \mathbf{J}_c(\mathbf{q}) & \mathbf{0}_{n_{ct}\times n_{ct}} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{M}(\mathbf{q})\dot{\mathbf{q}}^- \\ \mathbf{0} \end{bmatrix}, \tag{1.10}$$

where $\mathbf{0}_{n_{ct}\times n_{ct}}$ is a $n_{ct} \times n_{ct}$ zero matrix. Equation (1.10) can then be used to obtain the jump in the generalized velocities as:

$$\dot{\mathbf{q}}^+ := \mathbf{R}(\mathbf{q})\dot{\mathbf{q}}^-. \tag{1.11}$$

where $\mathbf{R} : \mathbb{R}^+ \times Q \to \mathbb{R}^{(n+6)\times(n+6)}$. The (continuous-time and hybrid) full-order rigid body dynamic model has been widely utilized for planning and control of legged robots [53, 54, 20, 23, 55, 56, 57] and manipulators [58, 59, 60]. As reviewed earlier, full-order robot models provide an accurate representation of a robotic system's behavior by capturing all degrees of freedom. However, their high computational demands may limit their utility for real-time planning in dynamic environments with varying parameters.

(1) Knowledge Gap 1

Dynamic modeling for legged locomotion on stationary surfaces has been well studied, as reviewed earlier. However, these models may not be valid for describing the dynamics of legged robot locomotion on a dynamically moving rigid surface, especially when the surface motion is significant. This is because these models all assume that the surface is stationary. The existing robot dynamics models for locomotion on moving surfaces (reviewed in Chapters 2.1.1, 3.1.1, and 4.1.1) do not explicitly consider surface motion for DRS with large inertia or rigid actuation such as ships and aircraft. To date, dynamic modeling of locomotion accounting for independent motion of the surface remains an open question.

1.2.2 Trajectory Planning

Trajectory planners generate the desired robot motions that a controller should reliably track [61, 62, 63, 23]. The trajectory planning problems of legged robots are complex partly due to the robot's underactuated base (e.g., trunk), which cannot be directly affected by the joint motors. Furthermore, the legged robot's dynamics are high-dimensional, hybrid, and nonlinear, further compounding the planning challenges.

(1) Decoupled Planning Based on Reduce-Order Models

The base movement of a legged robot is directly dependent on the foot's or the end effector's interaction with the environment. For this reason, traditionally, the planning problem of a legged robot has been decoupled into footstep planning and base planning. In this existing planning approach, the footstep location is computed first [64, 65, 66], and then the base trajectories are generated based on the corresponding zero moment points [67, 68], capture point [25, 69, 70], or divergent components of motion [30] of the robot. These planning approaches are reviewed in Section 3.1.1 of Chapter 3. This approach of decoupled planning is relatively fast as the base trajectory planning is based upon a reduced-order model such as LIP [10].

(2) Trajectory Optimization Based on Full-Order Models

The other important approach for gait or trajectory planning of legged robots uses the full-order robot model [53, 71] and aims to optimize some of the performance metrics (e.g., motor torque and cost of transport) while respecting all the feasibility constraints of the robot. The general formulation for such a planning approach can be mathematically described as:

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & h(\boldsymbol{\alpha}) \\ \text{subject to} \quad & \mathbf{f}_{eq}(\boldsymbol{\alpha}) = \mathbf{0},\ \mathbf{g}_{ineq}(\boldsymbol{\alpha}) \leq \mathbf{0}, \end{aligned} \tag{1.12}$$

where $\boldsymbol{\alpha}$ is the vector of optimization variables, $h(\boldsymbol{\alpha})$ is the cost function that needs to be minimized, and the vector functions $\mathbf{f}_{eq}$ and $\mathbf{g}_{ineq}$ are the sets of all equality and inequality constraints, respectively. The constraints function ensures kinematics feasibility (e.g., joints limit), dynamic feasibility (e.g., actuator and velocity limits, friction cone, unilateral constraint, and power limit), and stability of the generated trajectory. The cost function is set up to optimize a desired performance metric (e.g., input torque or cost of transport). Such a formulation for trajectory planning problems is also referred to as trajectory optimization [72, 73].

(3) Trajectory Planning Based on Model Predictive Control

The trajectory planning problem can also be formulated as a model predictive control (MPC) problem. The MPC-based planner generally shows more robust locomotion performance [74, 75] compared to the conventional footstep planner or trajectory optimization due to its capability of continuously adjusting the desired trajectories based on the current robot state. The MPC-based planner can be used for reduced-order model-based footstep planning [76] or full-order model-based trajectory planner [77, 78]. Depending on the constraints in the MPC problem, the formulation can be a linear or nonlinear problem. A linear MPC problem can be cast as a quadratic program, which can be solved much faster than a nonlinear MPC [79]. Advances in MPC solution finding algorithms allow speeding up nonlinear MPC for planning and control applications [80].

Knowledge Gap 2

Generating a feasible and optimal trajectory for a real-world scenario is still a challenge as we often need to balance the trade-off between optimality and speed [81, 82, 83]. While existing robot planners can achieve reasonable computational efficiency and feasibility for stationary surface locomotion, it is still an open research problem to extend these methods to dynamic surface locomotion due to the lack of suitable reduced-order robot models.

1.2.3 Controller Design

The top priority of controller design for legged robot locomotion is to ensure locomotion stability. The stability of legged locomotion can be loosely defined as the robot's ability to sustain locomotion without falling over to the ground. This loose definition has been translated into various stability criteria to guide the design of legged robot controllers. The following paragraphs briefly summarize some of the most widely used stability criteria and control methods for legged locomotion on static surfaces.

(1) Zero Moment Point

The zero moment point (ZMP) is a point on the ground where the influence of all the forces acting on the robot can be replaced by a single force [84]. This criterion is widely used for the balance and control of legged robots [67, 68, 85]. The ZMP-based controllers have realized walking [86], running [87], and balancing [88] of legged robots. Controller design based on the ZMP criteria often relies on a reduced-order model such as LIP [89, 86]. The ZMP controllers are versatile but typically require full actuation of the robot. The full-actuation requirement is restrictive because underactuation will occur in the event of foot slipping and rolling, and cannot be addressed by the ZMP controllers.

(2) Capture Point

The concept of the capture point has also been utilized for control design. At any instant, a capture point is a point on the walking surface where the robot needs to instantaneously step to in order to come to a complete stop [25]. Capture point trajectories are typically

computed for a reduced-order LIP model of the robot [25, 90]. The capture point-based controller design has been used for enabling the locomotion of legged robots on static flat surfaces [25, 29] and uneven surfaces [91].

(3) Hybrid Zero Dynamics

The hybrid zero dynamics (HZD) approach is a popular framework utilized to design a provably stable controller for the hybrid model of legged robot locomotion [92, 9, 20, 52, 18, 93, 54]. The distinct feature of the HZD framework lies in that it explicitly treats the hybrid dynamics of legged robot locomotion based on nonlinear control theory and the theory of hybrid dynamical systems. Legged locomotion dynamics are naturally hybrid, involving discrete-time behaviors of foot landings and continuous-time motions of foot swinging. By explicitly addressing the hybrid locomotion behaviors, the HZD framework has realized remarkable locomotion performance for the periodic gait of fully actuated [94] and underactuated [95, 96] robot locomotion on static surfaces.

(4) Whole-Body Controller

The whole-body control approach has been extensively utilized to coordinate the full-body movements of a legged robot during the execution of multiple tasks [97, 98, 99]. Such a control approach has been proven effective for real-world multi-task scenarios [99, 100, 101] and robust locomotion on static uneven surfaces, stairs, rocks, and sand [102, 101].

(5) Learning-based Controller

Recently, learning-based control methods [103, 104, 105] have shown promise in achieving remarkably robust locomotion on static and locally deformable surfaces under a broad spectrum of unstructured real-world environments (e.g., sand, grass, hiking trails, and creeks). A contact-adaptive learning-based controller policy for quadrupedal locomotion [106] leads to efficient and robust walking on static surfaces. Integrating perception modalities to learning [107] improves the terrain awareness of the learning-based control and thus enhances locomotion robustness. However, the perturbations induced by DRS (e.g.,

moving ships) act as persistent and continuous disturbances to legged locomotion, which are fundamentally different from those induced by ground with a stationary base but uncertain surface properties such as unevenness, compliance, and inclination.

Knowledge Gap 3

As reviewed earlier, various existing control frameworks have produced remarkably stable legged locomotion on static surfaces. However, they may not sustain stable walking on DRS because they are typically designed assuming that the surface is stationary. Both theoretical and experimental investigations are needed to formulate a control framework that stabilizes legged locomotion on DRS.

Performance Gap

Based on the literature review and the testing of state-of-the-art proprietary controller developed for legged robots, we observe that the existing control framework design for static surface locomotion does not perform reliably and can even fail to sustain walking when the robot moves on DRS. For instance, the results of our tracking control simulations in Fig. 2-8 show that a controller designed for a static surface is unreliable in tracking the desired trajectory on DRS and eventually the robot falls over. Therefore, a new control approach is needed to provably attain reliable performance for legged locomotion on DRS.

1.3 Contributions and Impacts

This dissertation aims to achieve robust quadrupedal locomotion on vertically moving rigid surfaces by creating new models, planners, and control algorithms. Specifically, the objective of this dissertation is to address the identified knowledge gaps pertaining to the advancement of modeling, planning, and control frameworks for legged locomotion on DRS, as well as to mitigate the observed performance limitations present in state-of-the-art proprietary controllers when operating on DRS. The contributions of this dissertation are based on the following articles [20, 34, 108, 109, 110].

The main contributions are:

(a) Deriving a provably stabilizing controller for quadrupedal locomotion on DRS by formulating the full-order robot model as a hybrid, time-varying system and analyzing the closed-loop control system through the construction of multiple Lyapunov functions [20]. This contribution uncovers the high complexity of the full-order models of legged locomotion on DRS and introduces a provably stabilizing controller for sustaining walking motion.

(b) Proposing reduced-order models and approximate analytical solutions of legged walking under a vertical sinusoidal DRS motion and designing a hierarchical planner based on the proposed analytical solutions for real-time motion planning [108]. This effort reveals the essential dynamic behaviors of DRS locomotion and contributes to solving the problem of online, physically feasible planning for locomotion on DRS experiencing a vertical sinusoidal motion.

(c) Developing a robust control framework for legged locomotion on DRS under unknown general vertical motion by establishing stability conditions for a hybrid time-varying linear inverted pendulum (HT-LIP) model and utilizing it in the design of a robust footstep controller through a quadratic program [109]. Validation experiments of this framework demonstrate its effectiveness in addressing the challenges posed by unknown surface motion and common external disturbances encountered in real-world applications.

The above contributions address scientific knowledge gaps in the control of legged locomotion on dynamically moving rigid surfaces and propose an effective solution to a practical problem of legged robot deployment on rigid surfaces with dynamic motion, such as ships and vessels that support the transport of 80% of the world trade volume [111]. The impacts of this work are grouped into the following categories.

Theoretical or scientific impacts: This dissertation aims to solve the control problem for quadrupedal locomotion on DRS by deriving new theorems and algorithms to fill the Knowledge Gaps 1-3 in the literature. The work in this dissertation has filled the following theoretical knowledge gaps.

- Deriving nonlinear hybrid time-varying full-order dynamic model of legged locomotion on DRS
- Proposing reduced-order and full-order dynamic models of legged locomotion over rigid surfaces with general dynamic motion
- Developing physically feasible real-time trajectory planning methods for legged locomotion on dynamic rigid surfaces
- Developing control frameworks that ensure provable stability and robust performance under unknown general vertical surface motion and real-world uncertainties
- Deriving provable stability conditions of the overall full-order and reduced-order time-varying models

The above contributions provide the basis for further development of modeling, analysis, and control frameworks for legged locomotion on rigid surfaces with general dynamic motion. **Practical or engineering impacts on society:** The dissertation research aims to solve real-world legged robot deployment and performance concerns on rigid surfaces with dynamic motion, e.g., ships. This problem is of practical value because more than 80% of the world trade volume is through sea routes [111]. Existing literature does not explicitly address the legged locomotion control problem for DRS such as vessels and aircraft. To that end, we aim to develop new control frameworks that ensure provable locomotion performance on DRS and validate the framework in hardware experiments.

Developing a control framework that ensures robust locomotion and provable performance on such surfaces can lead to legged robot deployment on ships or any other DRS for autonomous firefighting, inspection, services, search and rescue, and surveillance. The experimental validations of the robust control framework, presented in Chapter 4 showcase robust performance under unknown vertical surface motion, uncertain sway motion, and various uncertainties (e.g., slippery and uneven surfaces, external solid and liquid load, and sudden pushes) that a robot may encounter in real-world operation. These results enable the application of the research in this dissertation to address high-value engineering challenges associated with deploying legged robots for ship inspection, services, and maintenance.

This dissertation has uncovered inconsistent and unstable performance of the proprietary quadruped controllers on DRS, developed the world's first robust control framework, and experimentally enabled the world's first robust quadrupedal walking on rigid surfaces with unknown motions that emulate ships in sea waves.

The following is a brief outline of the subsequent chapters in this dissertation:

Chapter 2 reports the results corresponding to item (a) in the aforementioned contributions list. The work has been published as a journal paper titled "Provably Stabilizing Controllers for Quadrupedal Robot Locomotion on Dynamic Rigid Platforms [20]."

Chapter 3 presents the work associated with item (b) of the contributions. The results have been published as a journal paper titled "Analytical Solution to a Time-Varying LIP Model for Quadrupedal Walking on a Vertically Oscillating Surface [108]."

Chapter 4 describes the work associated with the item (c) of the contributions list. Part of the results in this chapter is published as a peer-reviewed conference paper titled " Asymptotic Stabilization of Aperiodic Trajectories of a Hybrid-Linear Inverted Pendulum Walking on a Vertically Moving Surface [109]."

Chapter 5 presents the conclusions, lessons learned, and future works for further enhancing the impact of this dissertation.

Chapter 2

Full-Order Modeling and Provably Stabilizing Control Under Known Periodic Surface Motions

This chapter presents the derivation of a nonlinear hybrid time-varying full-order dynamic model and the design of a provably stabilizing controller for quadrupedal robot locomotion on dynamic rigid surfaces (DRS) with known periodic motion. The contents of this chapter have been published in the following peer-reviewed journal article.

- Iqbal, A., Gao, Y. and Gu, Y., 2020. Provably Stabilizing Controllers for Quadrupedal Robot Locomotion on Dynamic Rigid Platforms. *IEEE/ASME Transactions on Mechatronics*, 25(4), pp.2035-2044.

2.1 Introduction

Legged robots have the potential to assist human operators in critical real-world applications such as emergency response and disaster relief [112, 113]. These applications may demand the capabilities of a legged robot in reliably traversing platforms that exhibit dynamic behaviors. Real-world examples of dynamic platforms include unstable buildings on disaster sites, vessels, and aircraft.

However, as reviewed in Chapter 1, dynamic modeling and controller design for quadrupedal robot locomotion on dynamic platforms pose a challenging problem due to the high complexity of the associated robot dynamics. Essentially, walking robot dynamics are inherently hybrid involving complex discrete behaviors [114, 115, 116, 117, 118, 119, 120, 121, 122, 123] (i.e., uncontrolled sudden jumps in a robot's joint velocities when the swing foot strikes the platform surface). Furthermore, the robot dynamics may become explicitly time-varying when the robot moves on dynamic platforms.

This chapter reports our research that aims to fill Knowledge Gaps 1 and 3 (reviewed in Chapter 1). This work has been published in a journal article [20] entitled "Provably Stabilizing Controllers for Quadrupedal Robot Walking on Dynamic Rigid Surfaces."

2.1.1 Related Work

Previous research on control design for legged robot locomotion has been mainly focused on static platforms, including flat and uneven terrains [10, 26, 124, 125, 126, 127, 128]. However, these approaches cannot reliably sustain stable legged locomotion on dynamic platforms, as demonstrated by the simulation results in Fig. 2-8, because such approaches do not explicitly account for the motion of the stance feet induced by the movement of the platform surface.

Control design for legged locomotion on dynamic platforms is an active research topic. Recently, locomotion on granular terrains, such as sand and gravels, has been increasingly intensively studied [129, 130]. Granular terrains are considered dynamic platforms because the contact surface between a robot's stance foot and the terrain is subject to constant, and often significant, movements. Based on the previous modeling result, control approaches have been derived to enable impulsive robot hopping on granular terrains [131]. A stability criterion for planar bipedal robot walking on a granular terrain has been developed based on a general model of the leg-terrain interaction force [132], which can be used to guide the controller design for locomotion stabilization on granular terrains.

Besides granular terrains, floating-base platforms with a rigid surface, which are referred to as dynamic rigid platforms in this chapter, form another class of dynamic platforms

that commonly exist in real-world environments, such as airplanes, vessels, and ground vehicles. Recently, control design for legged locomotion over dynamic rigid platforms has been increasingly researched [133, 30]. However, these control approaches only address the robot's continuous-phase dynamics but not the discrete foot-landing behaviors. The resulting model discrepancy may deteriorate the control performance, particularly when the landing impacts are significant during dynamic walking [26].

In addition to the hybrid nature, robot dynamics associated with locomotion over dynamic rigid platforms are also time-varying, due to the time-varying movement of the foot-platform contact points induced by the platform motion. To date, control design that explicitly addresses such hybrid, time-varying behaviors remains unsolved and underexplored.

Motivated by the current research needs, the objective of this chapter is to derive a control approach that provably realizes stable quadrupedal robot locomotion on dynamic rigid platforms by explicitly addressing the associated hybrid, time-varying robot dynamics. We have previously introduced a model-based controller design for hybrid systems with state-triggered jumps that include legged robots walking on static platforms [54, 134]. The controller explicitly addresses the robot's hybrid dynamics associated with static-terrain walking. Yet, the previous control approach is not directly applicable to legged locomotion on a dynamic platform, because it does not consider the time-varying global movement of the platform-robot contact points induced by the platform motion.

2.1.2 Contributions

The main contribution of this chapter is the derivation of a control approach that provably achieves stable quadrupedal robot locomotion on dynamic rigid platforms by explicitly addressing the associated hybrid, time-varying walking dynamics.

As the initial step of our ongoing research on dynamic-platform locomotion, this research focuses on tackling the controller design challenge and assumes that the platform motion has been sensed or estimated. The specific contributions of this chapter include:

(a) Formulating the model of a quadrupedal robot that walks on a dynamic rigid surface as a hybrid, time-varying system with state-triggered jumps.

(b) Extending our previous model-based control approach for static-terrain walking with the proposed system model to provably stabilize the hybrid, time-varying control system for realizing stable locomotion on dynamic rigid platforms.

(c) Demonstrating through simulations that control laws synthesized for static-terrain walking may not guarantee stable locomotion on dynamic platforms.

(d) Validating the effectiveness of the proposed control approach in guaranteeing the stability and robustness of locomotion on dynamic rigid platforms, both through simulations and experimentally on a physical quadrupedal robot.

2.2 Full-Order Modeling of Hybrid Time-Varying Walking Dynamics

This section introduces the modeling of the hybrid, time-varying dynamics of a quadrupedal robot that walks on a dynamic rigid platform. The model serves as a basis for the proposed controller design.

Quadrupedal robot walking naturally involves hybrid dynamic behaviors. A complete quadrupedal walking cycle can be decomposed into four continuous phases connected by four discrete events, as illustrated in Fig. 2-1. When the swing leg moves in the air, the robot dynamics are continuous. When the swing leg strikes the walking surface, a foot-landing impact occurs and causes a sudden jump in the robot's joint velocities [26], which is sometimes referred to as a state-triggered jump.

Let $\mathbf{q}$ be the generalized coordinates of the robot, which is defined as:

$$\mathbf{q} := \left[\mathbf{p}_b^T,\ \boldsymbol{\gamma}_b^T,\ q_1,\ q_2,\ ...,\ q_n\right]^T \in \mathcal{Q}, \tag{2.1}$$

where $\mathbf{p}_b := [x_b, y_b, z_b]^T$ is the vector of a robot's floating-base position with respect to the world coordinate frame, $\boldsymbol{\gamma}_b := [\phi_b, \theta_b, \psi_b]^T$ is the vector of the floating-base roll, pitch, and yaw angles with respect to the world coordinate frame, $\left[q_1\ q_2\ ...\ q_n\right]^T$ is the vector of the robot's joint angles, and $\mathcal{Q} \subset \mathbb{R}^{n+6}$ is the configuration space of the robot.

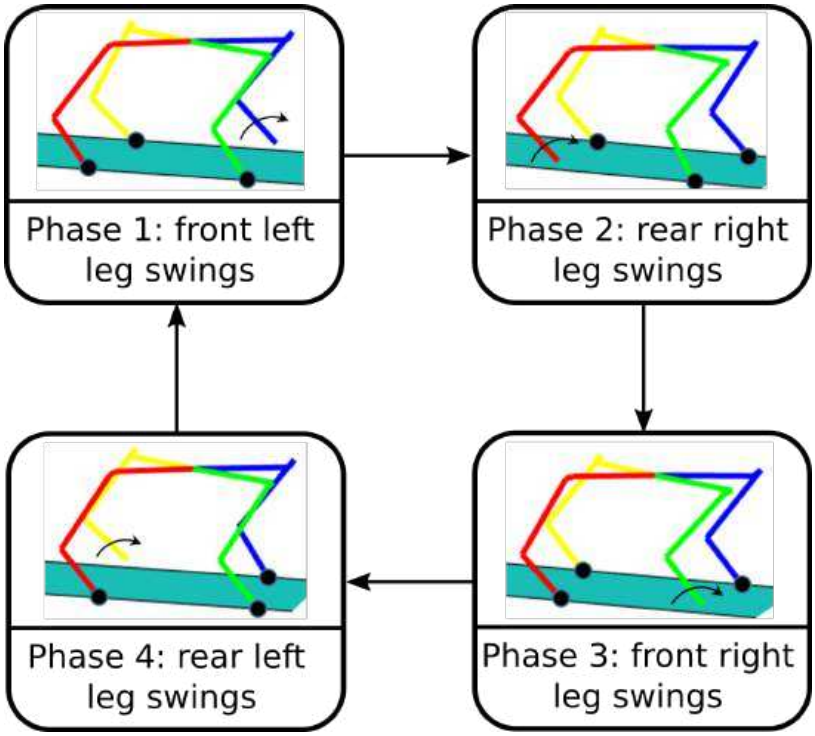

Figure 2-1: Illustration of a complete quadrupedal walking cycle. Swing and stance legs are marked with arrows and circles, respectively.

Let $\mathbf{u} \in U \subset \mathbb{R}^m$ be a vector of the robot's joint torques, where U is the set of admissible joint torques.

The degree of underactuation (DOU) of a walking quadrupedal robot can be computed as [135]:

$$\text{DOU} = n + 6 - (n_{ct} - n_{ci}) - m, \tag{2.2}$$

where n_{ct} is the number of all holonomic constraints and n_{ci} is the number of internal constraints.

During three-dimensional (3-D) quadrupedal walking, $n_{ct} = 9$ and $n_{ci} = 3$. Suppose that all of the robot's joints are independently actuated. Then, $\text{DOU} = 0$. Thus, the robot is fully actuated, and all of its degrees of freedom (DOFs) can be directly commanded. The DOF can be calculated as:

$$\text{DOF} = n + 6 - n_{ct}. \tag{2.3}$$

The class of dynamic platforms considered in this chapter are dynamic platforms with rigid surfaces, such as vessels, airplanes, and ground vehicles. As platform dynamics may be

unknown or difficult to estimate during real-world robot operations, the motion of a platform instead of its dynamics is considered in the derivation of the robot's dynamic model.

Modeling of the robot dynamics during a continuous phase and under a foot-landing impact is introduced next along with a mathematical definition of the impact event.

2.2.1 Continuous-Phase Dynamics

During a continuous phase of quadrupedal robot walking, three of the four legs contact the platform surface at their far ends (i.e., stance feet). Let $\mathbf{p}_c(\mathbf{q}) \in \mathbb{R}^{n_{ct}}$ be the position vector of the three stance feet in the world coordinate frame. When the stance feet do not slip on the platform surface, the associated holonomic constraints can be expressed as:

$$\mathbf{J}_c\ddot{\mathbf{q}} + \dot{\mathbf{J}}_c\dot{\mathbf{q}} = \mathbf{A}_p(t), \tag{2.4}$$

where $\mathbf{J}_c(\mathbf{q}) := \frac{\partial \mathbf{p}_c}{\partial \mathbf{q}}(\mathbf{q})$. Note that the expressions of $\mathbf{J}_c$ during different continuous phases within a complete walking cycle are different because the stance feet are different. The variable $\mathbf{A}_p \in \mathbb{R}^{n_{ct}}$ is the vector of the platform accelerations at the contact points, which is explicitly time-dependent when the platform is dynamic.

With Lagrange's method, the continuous-phase dynamics of a quadrupedal robot that walks on a dynamic platform can be obtained as:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{B}\mathbf{u} + \mathbf{J}_c^T\mathbf{F}_c, \tag{2.5}$$

where $\mathbf{M}(\mathbf{q}) : \mathcal{Q} \to \mathbb{R}^{(n+6)\times(n+6)}$ is the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) : \mathcal{TQ} \to \mathbb{R}^{(n+6)\times 1}$ represents the sum of Coriolis, centrifugal, and gravitational terms, $\mathbf{B} \in \mathbb{R}^{(n+6)\times m}$ is the actuator selection matrix, and $\mathbf{F}_c \in \mathbb{R}^{n_{ct}}$ is the generalized external force induced by the contact between the robot's stance feet and the platform surface. Here, $\mathcal{TQ}$ is the tangent bundleof $\mathcal{Q}$.

From Eqs. (2.4) and (2.5), $\mathbf{F}_c$ can be obtained as:

$$\mathbf{F}_c = -(\mathbf{J}_c\mathbf{M}^{-1}\mathbf{J}_c^T)^{-1}(\mathbf{J}_c\mathbf{M}^{-1}(\mathbf{B}\mathbf{u} - \mathbf{C}) + \dot{\mathbf{J}}_c\dot{\mathbf{q}} - \mathbf{A}_p(t)). \tag{2.6}$$

Substituting Eq. (2.6) into Eq. (2.5) yields:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \bar{\mathbf{C}}(t, \mathbf{q}, \dot{\mathbf{q}}) = \bar{\mathbf{B}}(\mathbf{q})\mathbf{u}, \tag{2.7}$$

where

$$\bar{\mathbf{C}} := \mathbf{C} - \mathbf{J}_c^T(\mathbf{J}_c\mathbf{M}^{-1}\mathbf{J}_c^T)^{-1}(\mathbf{J}_c\mathbf{M}^{-1}\mathbf{C} - \dot{\mathbf{J}}_c\dot{\mathbf{q}} + \mathbf{A}_p(t))$$

and

$$\bar{\mathbf{B}} := \mathbf{B} - \mathbf{J}_c^T(\mathbf{J}_c\mathbf{M}^{-1}\mathbf{J}_c^T)^{-1}(\mathbf{J}_c\mathbf{M}^{-1}\mathbf{B}).$$

Note that $\bar{\mathbf{C}}$ is explicitly time-dependent during dynamic-platform walking because $\mathbf{A}_p$ is explicitly time-dependent.

2.2.2 Switching Surface

When a swing foot strikes the platform surface, an impact occurs, causing a sudden jump in the generalized velocities $\dot{\mathbf{q}}$. Thus, the foot-landing event connects a discrete impact and a continuous phase, which can be mathematically defined as the following switching surface S_q:

$$\begin{aligned} S_q := \{(t, \mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^+ \times \mathcal{TQ} : d_{sw}(t, \mathbf{q}) = 0, \\ \dot{d}_{sw}(t, \mathbf{q}, \dot{\mathbf{q}}) < 0\}, \end{aligned} \tag{2.8}$$

where $d_{sw} : \mathbb{R}^+ \times \mathcal{Q} \to \mathbb{R}$ can be chosen as the shortest distance between the swing foot and the platform surface.

2.2.3 Discrete Impact Dynamics

Upon a swing-foot landing event, an impact occurs between the foot and the platform surface. Due to the instantaneous nature of the impact, the value of $\mathbf{q}$ remains continuous under an impact. However, the value of $\dot{\mathbf{q}}$ jumps.

Under an impulsive impact, the continuous dynamics in Eq. (2.5) and the holonomic

constraint in Eq. (2.4) become:

$$\mathbf{M}(\dot{\mathbf{q}}^{+} - \dot{\mathbf{q}}^{-}) = \mathbf{J}_c^T \, \delta\mathbf{F}_c \text{ and } \mathbf{J}_c \dot{\mathbf{q}}^{+} = \mathbf{V}_p^{+}, \tag{2.9}$$

where $\star^{-}$ and $\star^{+}$ represent the values of $\star$ right before and after the impact, respectively, $\delta\mathbf{F}_c$ is the impulsive impact force, and $\mathbf{V}_p$ is the vector of the platform velocities at the three contact points. The value of $\mathbf{V}_p$ also jumps because the platform velocities at the new foot-contact points right after the impact are not necessarily the same as the platform velocities at the previous foot-contact points right before the impact.

Rearranging the above equation gives:

$$\begin{bmatrix} \dot{\mathbf{q}}^{+} \\ \delta\mathbf{F}_c \end{bmatrix} = \begin{bmatrix} \mathbf{M}(\mathbf{q}) & -\mathbf{J}_c^T(\mathbf{q}) \\ \mathbf{J}_c(\mathbf{q}) & \mathbf{0}_{n_{ct} \times n_{ct}} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{M}(\mathbf{q})\dot{\mathbf{q}}^{-} \\ \mathbf{V}_p^{+} \end{bmatrix}, \tag{2.10}$$

where $\mathbf{0}_{n_{ct} \times n_{ct}}$ is a $n_{ct} \times n_{ct}$ zero matrix. Equation (2.10) can then be used to obtain the jump in the generalized velocities as:

$$\dot{\mathbf{q}}^{+} := \mathbf{R}(t, \mathbf{q})\dot{\mathbf{q}}^{-}. \tag{2.11}$$

where $\mathbf{R} : \mathbb{R}^{+} \times Q \to \mathbb{R}^{(n+6)\times(n+6)}$. Note that the expression of $\mathbf{R}$ is explicitly time-dependent because $\mathbf{R}$ contains the explicitly time-dependent function $\mathbf{V}_p$. Also, the expressions of $\mathbf{R}$ during different continuous phases within a complete walking cycle are different because the expressions of the corresponding Jacobian matrices $\mathbf{J}_c$ are different.

The model derived in this section clearly shows that the dynamics of a robot that walks on a dynamic platform are hybrid, time-varying, and involve state-triggered jumps that cannot be directly controlled due to their infinitesimally short periods of duration.

2.3 Model-Based Feedback Control During Continuous Phases

This section introduces a continuous-phase control law that provably stabilizes continuous-phase quadrupedal walking over dynamic rigid platforms. To guarantee that the control

law also provably stabilizes the overall hybrid walking process, a Lyapunov-based stability analysis is performed as explained in Section 2.4.

2.3.1 Continuous Model-Based Control

In this work, stable walking over dynamic platforms is achieved through the provable stabilization of the hybrid, time-varying control system.

Let $\mathbf{h}(\mathbf{q})$ and $\mathbf{h}_d(t,\mathbf{q})$ be the control variables and their reference trajectories, respectively. Suppose that all of the robot's joints are independently actuated, i.e., $m = n$, and that $\mathbf{h}_d(t,\mathbf{q})$ is smooth in t within each continuous domain. Then, by Eq. (2.2), the robot is fully actuated. Thus, the number of control variables can be chosen as the same as the robot's DOF. From Eq. (2.3), we have $\mathbf{h} \in \mathbb{R}^{n+6-n_{ct}}$.

The trajectory tracking error can be defined as:

$$\boldsymbol{\varepsilon}_h(t,\mathbf{q}) := \mathbf{h}(\mathbf{q}) - \mathbf{h}_d(t,\mathbf{q}). \tag{2.12}$$

The control objective is then: to drive $\|\boldsymbol{\varepsilon}_h\|$ to a bounded small number at the steady state for the overall hybrid closed-loop system.

Since the discrete impact dynamics cannot be directly controlled due to their infinitesimally short periods of duration, we choose to derive a continuous control law to directly stabilize the continuous-phase dynamics.

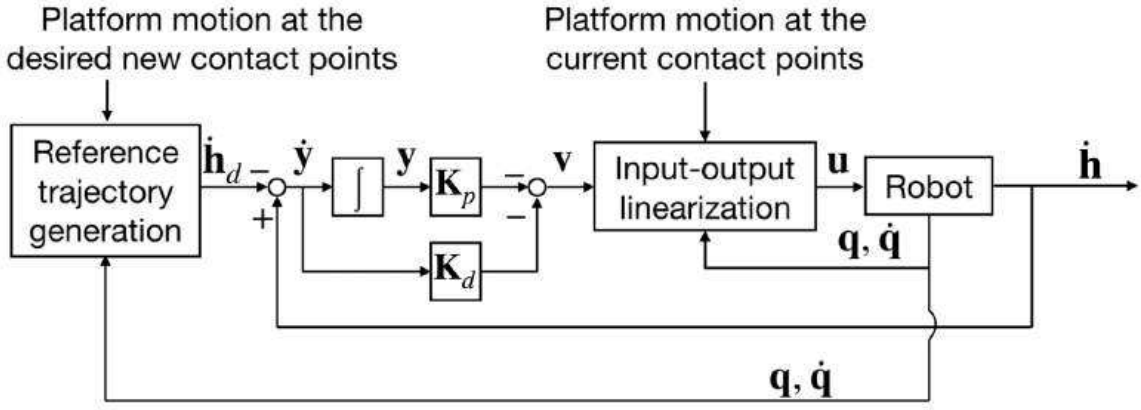

Figure 2-2: A block diagram of the proposed continuous-phase control law.

The proposed continuous control law is derived based on the full-order model of the

continuous-phase walking dynamics as derived in Section 2.2. A block diagram of the proposed control law is shown in Fig. 2-2.

To simplify the controller design, input-output linearization [136] is utilized to transform the continuous-phase nonlinear dynamics into a linear system.

With the trajectory tracking error $\boldsymbol{\varepsilon}_h$ chosen as the output function $\mathbf{y}$, we have $\dot{\mathbf{y}} = \frac{\partial \boldsymbol{\varepsilon}_h}{\partial \mathbf{q}}\dot{\mathbf{q}} + \frac{\partial \boldsymbol{\varepsilon}_h}{\partial t}$ and $\ddot{\mathbf{y}} = \frac{\partial}{\partial \mathbf{q}}(\frac{\partial \boldsymbol{\varepsilon}_h}{\partial \mathbf{q}}\dot{\mathbf{q}})\dot{\mathbf{q}} + \frac{\partial \boldsymbol{\varepsilon}_h}{\partial \mathbf{q}}\ddot{\mathbf{q}} + \frac{\partial^2 \boldsymbol{\varepsilon}_h}{\partial t^2}$. From Eq. (2.7), we have $\ddot{\mathbf{q}} = \mathbf{M}^{-1}(\bar{\mathbf{B}}u - \bar{\mathbf{C}})$. Substituting the expression of $\ddot{\mathbf{q}}$ into the expression of $\ddot{\mathbf{y}}$, we have $\ddot{\mathbf{y}} = \frac{\partial}{\partial \mathbf{q}}(\frac{\partial \boldsymbol{\varepsilon}_h}{\partial \mathbf{q}}\dot{\mathbf{q}})\dot{\mathbf{q}} + \frac{\partial \boldsymbol{\varepsilon}_h}{\partial \mathbf{q}}\mathbf{M}^{-1}(\bar{\mathbf{B}}u - \bar{\mathbf{C}}) + \frac{\partial^2 \boldsymbol{\varepsilon}_h}{\partial t^2}$.

Thus, by choosing a continuous control law as

$$\mathbf{u} = (\frac{\partial \boldsymbol{\varepsilon}_h}{\partial \mathbf{q}}\mathbf{M}^{-1}\bar{\mathbf{B}})^*(\frac{\partial \boldsymbol{\varepsilon}_h}{\partial \mathbf{q}}\mathbf{M}^{-1}\bar{\mathbf{C}} - \frac{\partial}{\partial \mathbf{q}}(\frac{\partial \boldsymbol{\varepsilon}_h}{\partial \mathbf{q}}\dot{\mathbf{q}})\dot{\mathbf{q}} - \frac{\partial^2 \boldsymbol{\varepsilon}_h}{\partial t^2} + \mathbf{v}), \tag{2.13}$$

where $(\cdot)^*$ denotes the pseudoinverse of a matrix $(\cdot)$, we obtain the linearized output function dynamics as $\ddot{\mathbf{y}} = \mathbf{v}$.

Define $\mathbf{v}$ as a proportional-derivative (PD) control law:

$$\mathbf{v} = -\mathbf{K}_p\mathbf{y} - \mathbf{K}_d\dot{\mathbf{y}}, \tag{2.14}$$

where $\mathbf{K}_p$ and $\mathbf{K}_d$ are PD gains. Then, the closed-loop dynamics during continuous phases are $\ddot{\mathbf{y}} = -\mathbf{K}_p\mathbf{y} - \mathbf{K}_d\dot{\mathbf{y}}$.

By properly assigning values for the PD gains, the proposed input-output linearizing control law can provably guarantee the stability of the closed-loop control system during continuous phases.

However, due to the presence of the uncontrolled impact dynamics, the proposed control law cannot automatically guarantee the provable stabilization of the overall hybrid, time-varying walking dynamics. To derive sufficient conditions under which the proposed control law can provably solve the stabilization problem, the stability of the closed-loop system is analyzed in Section 2.4.

Before introducing the stability analysis, we first introduce the construction of impact invariance that can be used to satisfy a necessary condition for asymptotic stabilization.

2.3.2 Impact Invariance Construction

If $\mathbf{y}^+ = \mathbf{0}$ and $\dot{\mathbf{y}}^+ = \mathbf{0}$ hold when $\mathbf{y}^- = \mathbf{0}$ and $\dot{\mathbf{y}}^- = \mathbf{0}$ under an impact, then $\mathbf{y}$ and $\dot{\mathbf{y}}$ are called "impact invariant" [26]. The impact invariance condition essentially requires that the reference trajectory should respect the impact dynamics. It is a necessary condition for achieving asymptotic stabilization of hybrid systems with state-triggered jumps that include the robot dynamics as derived in Section 2.2.

To meet the impact invariance condition, we have previously derived a method of constructing impact invariance [54] for the design of trajectory tracking controllers for legged locomotion on static platforms. The method translates the condition into equality constraints and enforces the constraints through trajectory generation. In this research, we extend the previous method from the controller design for legged locomotion on static platforms to dynamic platforms.

Consider a complete gait cycle of quadrupedal robot walking. Let $\mathbf{q} = \mathbf{q}_d(t) \in \mathbb{R}^{n+6}$ be the solution to the equation $\mathbf{h}(\mathbf{q}) - \mathbf{h}_d(t, \mathbf{q}) = \mathbf{0}$. By definition, to meet the impact invariance condition, the following equations should hold:

$$\mathbf{q}_d(\tau_{k,i}^+) = \mathbf{q}_d(\tau_{k,i}^-);\ \mathbf{q}_d(\tau_{k,0}^+) = \mathbf{q}_d(\tau_{k,4}^-); \tag{2.15}$$

$$\begin{aligned} \dot{\mathbf{q}}_d(\tau_{k,i}^+) &= \mathbf{R}(\tau_{k,i}, \mathbf{q}_d(\tau_{k,i}^-))\dot{\mathbf{q}}_d(\tau_{k,i}^-); \\ \dot{\mathbf{q}}_d(\tau_{k,0}^+) &= \mathbf{R}(\tau_{k,4}, \mathbf{q}_d(\tau_{k,4}^-))\dot{\mathbf{q}}_d(\tau_{k,4}^-). \end{aligned} \tag{2.16}$$

Here, $i \in \{1,2,3\}$, and $0 < \tau_{k,0} < \tau_{k,1} < \tau_{k,2} < \tau_{k,3} < \tau_{k,4}$. The variables $\tau_{k,0}^+$ and $\tau_{k,4}^-$ are the initial and the final instants of the complete k^{th} gait cycle ($k \in \{1,2,3,...\}$). The variable $\tau_{k,i}$ ($i \in \{1,2,3\}$) is the planned instant of a foot-landing event between $t = \tau_{k,0}^+$ and $t = \tau_{k,4}^-$.

Equation (2.15) always holds due to the continuity of the generalized coordinates $\mathbf{q}$ under an impact. Thus, to construct impact invariance, only Eq. (2.16) needs to be enforced through trajectory generation as explained in Section V.

2.4 Closed-Loop Stability Analysis Based on Lyapunov Theory

This section presents the stability analysis of the closed-loop hybrid, time-varying system under the proposed continuous control law. The analysis result is a set of sufficient conditions that the proposed controller should satisfy in order to provably stabilize the hybrid closed-loop control system for achieving stable walking on dynamic rigid platforms.

Define $\mathbf{x} := [\mathbf{y}^T \ \dot{\mathbf{y}}^T]^T$. Under the proposed continuous-phase control law in Eqs. (2.13) and (2.14), the closed-loop output function dynamics can be compactly expressed as:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x}, & \text{if } (t, \mathbf{x}^-) \notin S(t, \mathbf{x}); \\ \mathbf{x}^+ = \boldsymbol{\Delta}(t, \mathbf{x}^-), & \text{if } (t, \mathbf{x}^-) \in S(t, \mathbf{x}), \end{cases} \tag{2.17}$$

where $\mathbf{A} := \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{K}_p & -\mathbf{K}_d \end{bmatrix}$, $\mathbf{0}$ and $\mathbf{I}$ are zero and identity matrices with appropriate dimensions, and the expressions of S and $\boldsymbol{\Delta}$ can be obtained from the switching surface S_q in Eq. (2.8), the reset map $\mathbf{R}$ in Eq. (2.11), and the trajectory tracking error $\boldsymbol{\varepsilon}_h$ in Eq. (2.12). For notational simplicity, one continuous phase and the subsequent discrete jump of a complete walking cycle are given in Eq. (2.17).

The continuous-phase dynamics can be provably stabilized by properly selecting the PD gains such that $\mathbf{A}$ is Hurwitz. However, the instantaneous, uncontrolled impact dynamics cannot be directly regulated. Hence, we will utilize the construction of multiple Lyapunov functions [137] to derive sufficient conditions for the overall hybrid, time-varying closed-loop system.

The proposed stability analysis is an extension of our previous work on trajectory tracking control of static-platform locomotion [54]. The previous analysis cannot be directly performed on the hybrid, time-varying system in Eq. (2.17) because it does not consider the time dependence of the system dynamics induced by the platform motion. To this end, this chapter extends the previous analysis with the consideration of platform motions as summarized in the following assumption:

(A9) The platform acceleration at the foot-contact points, $\mathbf{A}_p(t)$, is differentiable in t within a continuous phase.

The assumption (A9) is valid for real-world dynamic rigid platforms such as vessels moving in regular waves [138].

Before explaining the proposed stability analysis, several related variables and concepts are first introduced.

Let τ_k and t_k be the planned and the actual k^{th} impact timings, respectively. The variables t_k^- and t_k^+ denote the time instants right before and after the k^{th} impact, respectively. For notational simplicity, $\star(t_{k-1}^-)$ and $\star(t_{k-1}^+)$ are denoted as $\star|_{k-1}^-$ and $\star|_{k-1}^+$, respectively, for the rest of this chapter.

A fictitious system is introduced and defined as $\dot{\bar{\mathbf{x}}} := \mathbf{A}\bar{\mathbf{x}}$ such that under the initial condition $\bar{\mathbf{x}}(t_0) = \mathbf{x}_0$, a solution of the system is given by $\bar{\mathbf{x}}(t;t_0,\mathbf{x}_0)$, $\forall\, t > t_0$. Then, the solution to the actual continuous-phase system between the $(k-1)^{th}$ and the k^{th} impacts satisfies:

$$\mathbf{x}(t) = \bar{\mathbf{x}}(t;t_{k-1}^+,\mathbf{x}|_{k-1}^+),\ \ \forall t \in (t_{k-1},\, t_k]. \tag{2.18}$$

Main Theorem (Closed-Loop Stability Conditions): The closed-loop control system in Eq. (2.17) is locally asymptotically stable if the following two conditions are met:

(C1) Reference trajectories are planned with the impact invariance condition met to respect the impact dynamics.

(C2) The PD gains are chosen such that $\mathbf{A}$ is Hurwitz and that the state $\mathbf{x}$ converges to zero sufficiently fast during continuous phases. ■

Proof: By the theory of multiple Lyapunov functions [137], a hybrid system is asymptotically stable if there exists a Lyapunov function candidate and a positive number r such that for all $\mathbf{x}|_0^+ \in B_r(\mathbf{0}) := \{\mathbf{x} : \|\mathbf{x}\| \leq r\}$, the following two conditions hold: 1) the Lyapunov function candidate asymptotically decreases during each continuous phase and 2) its values at the initial instants of all continuous phases form a strictly decreasing sequence.

We begin the stability analysis with continuous phases.

Suppose that the PD gains are chosen such that the matrix $\mathbf{A}$ is Hurwitz. Then, there exists a Lyapunov function candidate $V(\mathbf{x})$ and positive numbers c_1, c_2, and c_3 such that

$$c_1\|\mathbf{x}\|^2 \leq V(\mathbf{x}) \leq c_2\|\mathbf{x}\|^2 \text{ and } \dot{V}(\mathbf{x}) \leq -c_3\|\mathbf{x}\|^2 \tag{2.19}$$

hold for all $\mathbf{x}$ during continuous phases [136]. Then,

$$V|_k^+ \leq e^{-\frac{c_3}{c_2}(t_k - t_{k-1})} V|_{k-1}^+,\ k \in \{1,2,...\}. \tag{2.20}$$

Next, the evolution of the Lyapunov function candidate V across a state-triggered jump is analyzed. By the triangle inequality principle, the norm of the state $\mathbf{x}$ right after the k^{th} impact can be approximated as:

$$\begin{aligned}\|\mathbf{x}|_k^+\| = \|\boldsymbol{\Delta}(t_k, \mathbf{x}|_k^-)\| \leq & \|\boldsymbol{\Delta}(t_k, \mathbf{x}|_k^-) - \boldsymbol{\Delta}(\tau_k, \mathbf{x}|_k^-)\| \\ & + \|\boldsymbol{\Delta}(\tau_k, \mathbf{x}|_k^-) - \boldsymbol{\Delta}(\tau_k, \mathbf{0})\| + \|\boldsymbol{\Delta}(\tau_k, \mathbf{0})\|.\end{aligned} \tag{2.21}$$

By the condition (C1), we have $\boldsymbol{\Delta}(\tau_k, \mathbf{0}) = \mathbf{0}$.

Under the assumption (A9), the platform velocity at the foot-contact points, $\mathbf{V}_p(t)$, is continuously differentiable in t. Then, Eq. (2.11) and the definition of $\mathbf{x}$ indicate that $\boldsymbol{\Delta}(t, \mathbf{x})$ is locally continuously differentiable in t and $\mathbf{x}$. Thus, there exist Lipschitz constants, L_{Δ_t} and L_{Δ_x}, and a positive number r_Δ such that for any $\mathbf{x}|_0^+ \in B_{r_\Delta}(\mathbf{0})$, we have:

$$\|\Delta(t_k, \mathbf{x}|_k^-) - \Delta(\tau_k, \mathbf{x}|_k^-)\| \leq L_{\Delta_t}\|t_k - \tau_k\|; \tag{2.22}$$

$$\|\Delta(\tau_k, \mathbf{x}|_k^-) - \Delta(\tau_k, \mathbf{0})\| \leq L_{\Delta_x}\|\mathbf{x}|_k^-\|. \tag{2.23}$$

Since the continuous-phase dynamics of the hybrid system in Eq. (2.17) is defined by a function (i.e., $\mathbf{Ax}$) that is continuously differentiable in $\mathbf{x}$, there exists a Lipschitz constant L_t and a positive number r_t such that the difference between the actual and the planned k^{th} impact timings can be approximated as [139]:

$$\|t_k - \tau_k\| \leq L_t\|\bar{\mathbf{x}}(\tau_k; t_{k-1}^+, \mathbf{x}|_{k-1}^+)\| \tag{2.24}$$

for all $\mathbf{x}|_0^+ \in B_{r_t}(\mathbf{0})$.

Combining Eqs. (2.22) and (2.24) yields

$$\|\mathbf{\Delta}(t_k, \mathbf{x}|_k^-) - \mathbf{\Delta}(\tau_k, \mathbf{x}|_k^-)\| \leq L_{\Delta_t} L_t \|\bar{\mathbf{x}}(\tau_k; t_{k-1}^+, \mathbf{x}|_{k-1}^+)\|. \tag{2.25}$$

From Eqs. (2.21)-(2.25), we have

$$\|\mathbf{x}_k^+\| \leq L_{\Delta_t} L_t \|\bar{\mathbf{x}}(\tau_k; t_{k-1}^+, x_{k-1}^+)\| + L_{\Delta_x} \|\mathbf{x}_k^-\| \tag{2.26}$$

for any $\mathbf{x}_0^+ \in B_r(0)$, where $r := \min(r_\Delta, r_t)$.

From Eqs. (2.19) and (2.20), we have

$$\|\mathbf{x}|_k^-\| \leq \sqrt{\frac{c_2}{c_1}} e^{-\frac{c_3}{2c_2}(t_k - t_{k-1})} \|\mathbf{x}|_{k-1}^+\|; \tag{2.27}$$

$$\|\tilde{\mathbf{x}}|_k^-\| \leq \sqrt{\frac{c_2}{c_1}} e^{-\frac{c_3}{2c_2}(\tau_k - t_{k-1})} \|\mathbf{x}|_{k-1}^+\|. \tag{2.28}$$

From Eqs. (2.26)-(2.28), we have $\|\mathbf{x}|_k^+\| \leq \sqrt{\frac{c_2}{c_1}}(L_{\Delta_t} L_t + L_{\Delta_x} e^{-\frac{c_3}{2c_2}(t_k - \tau_k)}) e^{-\frac{c_3}{2c_2}(\tau_k - t_{k-1})} \|\mathbf{x}|_{k-1}^+\|$ By the condition (C2), the PD gains can be assigned to allow a sufficiently high convergence rate. Then, suppose that the PD gains are chosen such that $e^{-\frac{c_3}{2c_2}(t_k - \tau_k)} \leq 1 + \varepsilon$ holds for some positive number ε. Then, for any $\mathbf{x}|_0^+ \in B_r(\mathbf{0})$, we have $\|\mathbf{x}|_k^+\| \leq \sigma \|\mathbf{x}|_{k-1}^+\|$ with

$$\sigma := \sqrt{\frac{c_2}{c_1}}(L_{\Delta_t} L_t + L_{\Delta_x}(1 + \varepsilon)) e^{-\frac{c_3}{2c_2}\Delta\tau_k}$$

and $\Delta\tau_k := \tau_k - t_{k-1}$. Therefore, $V|_k^+ \leq \Omega V|_{k-1}^+$, $\Omega := \frac{c_2\sigma^2}{c_1}$.

If the PD gains are chosen such that $\mathbf{A}$ is Hurwitz and that $\frac{c_2\sigma^2}{c_1} < 1$ holds (i.e., $\Omega < 1$), then $V|_0^+$, $V|_1^+$, $V|_2^+$... form a strictly decreasing sequence. By the definition of σ, if the continuous-phase convergence rate $\frac{c3}{c2}$ can be chosen sufficiently large, then $\sigma < \sqrt{\frac{c_1}{c_2}}$ holds (i.e., $\frac{c_2\sigma^2}{c_1} < 1$ is satisfied). Then, by the stability theory based on the construction of multiple Lyapunov functions, the closed-loop hybrid system is locally asymptotically stable for any $\mathbf{x}|_0^+ \in B_r(\mathbf{0})$. ■

2.5 Reference Trajectory Generation

This section explains the proposed trajectory generation method for planning the desired quadrupedal walking motions on dynamic rigid platforms.

2.5.1 Formulation of a Model-Based Optimization Problem

In this chapter, the problem of trajectory generation is formulated as an optimization problem. The solution of the optimization problem is the reference trajectories, $\mathbf{h}_d$, which represent kinematically and dynamically feasible walking motions over dynamic rigid platforms.

The formulation of the optimization problem begins with properly defining $\mathbf{h}_d$. Without loss of generality, $\mathbf{h}_d$ is defined based on walking pattern encoding [26]. A walking pattern represents the relative evolution of configuration-based variables with respect to a phase variable that presents how far or how long a robot has walked within a gait cycle. Let θ be the phase variable. Let θ^+ and θ^- be the planned values of θ at the beginning and the end of a complete gait cycle, respectively. Then, the normalized phase variable s is defined as $s(\theta) := \frac{\theta-\theta^+}{\theta^- - \theta^+}$. The desired walking pattern of the control variable $\mathbf{h}$ can be encoded by $s(\theta)$ as $\mathbf{h}(\mathbf{q}) - \mathbf{h}_d(s(\theta)) = \mathbf{0}$. The N^{th}-order Bézier curves can be used to define $\mathbf{h}_d$ as $\mathbf{h}_d(s) := \sum_{i=0}^{N} \boldsymbol{\alpha}_i \frac{N!}{i!(N-i)!} s^i (1-s)^{N-i}$. Here, $\boldsymbol{\alpha}_i \in \mathbb{R}^{n+6-n_{ct}}$ are the coefficients of the Bézier curves, and are in turn used as optimization variables.

The cost function of the optimization problem can be chosen as the energy consumed during walking [26].

Necessary constraints are considered to ensure that the optimized reference trajectories would correspond to kinematically and dynamically feasible quadrupedal locomotion over dynamic rigid platforms. These constraints include: 1) switching surfaces as derived in Section 2.2.2; 2) the impact invariance condition derived in Section 2.3.2; 3) joint position and velocity limits; 4) joint torque limits with the torque from Eq. (2.13); 5) platform-contact constraints (e.g., unilateral constraints and friction-cone constraints) with the contact force from Eq. (2.6); and 6) desired gait features (e.g., the desired duration of the gait cycle).

Both continuous-phase and discrete dynamics are contained in the constraints. The continuous-phase dynamics in Eq. (2.5) are contained in the joint torque limits and the

platform-contact constraints. The discrete impact dynamics in Eq. (2.11) are contained in the impact invariance condition.

The platform motions (e.g., $\mathbf{V}_p$ and $\mathbf{A}_p$) are also contained in the constraints because the platform motion directly affects the contact force between the robot and the platform, the occurrence of a swing-foot landing event, and the impact dynamics, as explained in Section 2.2.

2.5.2 Optimization Setup

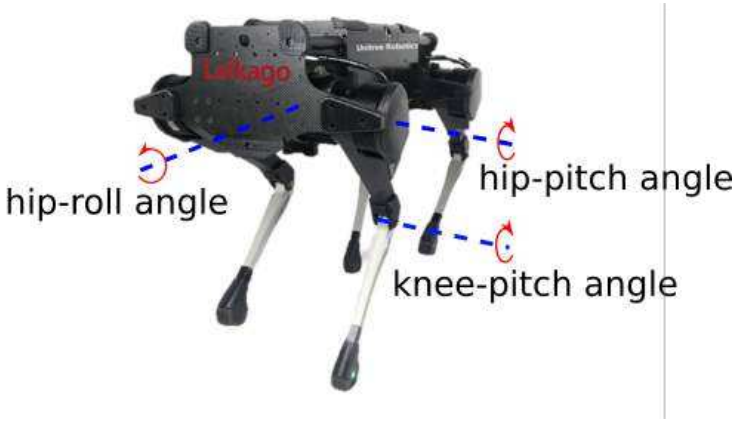

Figure 2-3: A Laikago quadrupedal robot (developed by Unitree) used for experimental validation. Each leg of the robot has three actuated joints, which are hip-roll, hip-pitch, and knee-pitch joints.

The optimization is set up as a nonlinear programming (NLP) problem. The software tool used to solve the NLP is MATLAB's `fmincon` function. As the focus of this chapter is on controller design, the planning is solved offline.

The control variables $\mathbf{h}$ are chosen as the base pose (i.e., position and orientation) and swing-foot position; that is, $\mathbf{h}(\mathbf{q}) := \left[\mathbf{p}_b^T(\mathbf{q}), \boldsymbol{\gamma}_b^T(\mathbf{q}), \mathbf{p}_{sw}^T(\mathbf{q})\right]^T$.

In general, the phase variable θ can be defined based on either generalized coordinates $\mathbf{q}$ [26, 134] or time t [54]. In this chapter, θ is defined as time-dependent; i.e., $\theta(t) := t - \tau_k, \;\; k \in \{1,2,...\}$ during the k^{th} walking cycle (i.e., $t \in (t_k, t_{k+1}]$). Thus, θ represents the current instant relative to the planned initial instant of the k^{th} walking cycle. With the phase variable chosen as time-based, the reference trajectories $\mathbf{h}_d$ become functions of t alone, i.e., $\mathbf{h}_d(t,\mathbf{q}) = \mathbf{h}_d(t)$.

2.6 Simulations and Experiments

This section presents the validation results obtained through simulations and experiments.

2.6.1 Simulation and Experimental Setup

In this chapter, MATLAB simulations are used to validate the theoretical control law derived in Section 2.3. Simulations on PyBullet, which is a 3-D realistic robot simulator [140], are conducted to gain preliminary insight into the controller implementation on hardware. Experiments on a physical quadrupedal robot (see Fig. 2-3) are used to evaluate the effectiveness of the proposed control approach as well.

The dynamic rigid platform used for validation is:

(P1) A platform with a whole-body pitching motion and a surface translating motion (Nominal pitching amplitude = $\pm 5^{\circ}$. Nominal pitching frequency = 0.5 Hz).

In experiments, the platform (P1) is chosen as a pitching treadmill with translating belts on the surface.

To validate the proposed control approach during walking motions with different gait characteristics, three sets of reference trajectories are used in simulations and experiments. They are generated using the optimization-based planning method in Section 2.5. The gait characteristics are:

(G1) Step length = 10 cm. Maximum swing-foot height = 6 cm. (Treadmill belt speed = 5 cm/s).

(G2) Step length = 16 cm. Maximum swing-foot height = 6 cm. (Treadmill belt speed = 8 cm/s).

(G3) Step length = 16 cm. Maximum swing-foot height = 9 cm. (Treadmill belt speed = 8 cm/s).

For simplicity, the optimization enforces the exact correspondence between the duration of the reference gait cycle and one period of the platform pitching motion. These reference trajectories are displayed in Figs. A-4 to A-6 in Appendix A.

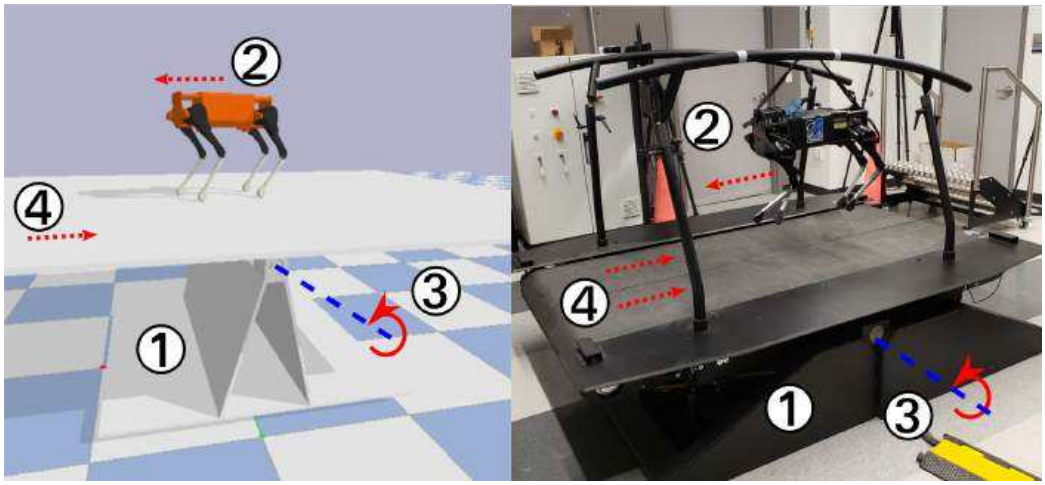

Figure 2-4: PyBullet simulation and experimental setup for assessing the performance of the proposed control strategy. ①: treadmill. ②: Laikago robot and its walking direction. ③: rotating axis of the treadmill. ④: moving direction of the treadmill belt.

As it is assumed that the platform motion is known (i.e., sensed or estimated) in the theoretical controller design, we added perturbations to the platform motion to assess the robustness of the proposed control approach under uncertainties. Two sets of perturbations are implemented with the platform (P1), including:

(U1) Uncertainties in the pitching amplitude.

(U2) Uncertainties in the belt speed.

More details on the setups of MATLAB and PyBullet simulations as well as experiments are given next.

MATLAB Simulation Setup: The robot model derived in Section 2.2 is used to simulate the closed-loop system under the control law in Eqs. (2.13) and (2.14). The PD gains for gait (G1) are chosen as $\mathbf{K}_p = \text{diag}(100, 36, 110, 100, 36, 110, 36, 64, 110)$ and $\mathbf{K}_d = \text{diag}(20, 12, 21, 20, 12, 21, 12, 16, 21)$, yielding closed-loop poles with negative real parts between -10.5 and -6. The PD gains for the uncertainty case (U1) are chosen the same as gait (G1). The PD gains for other cases are included in Appendix A.

PyBullet Simulation Setup: A 3-D realistic robot model that closely emulates the physical and geometric properties of the physical quadrupedal robot is used in the simulation, as shown in Fig. 2-4. The "PD" control gains are set as 1.0 and 0.25, respectively. Note that

these values may not reflect the true PD gains implemented in PyBullet due to the intrinsic gain multipliers used in the simulator.

Experimental Setup: The experimental setup consists of a quadrupedal robot and a treadmill, as shown in Fig. 2-4. The "PD gains" of the joint-level controllers are set as 5.5 and 0.2, respectively. Similar to PyBullet, these values may not reflect the true PD gains implemented on the physical robot's joint motors due to the intrinsic gain multipliers of the hardware. These PD gains in MATLAB, PyBullet, and experiments are tuned to produce similar continuous-phase convergence rates.

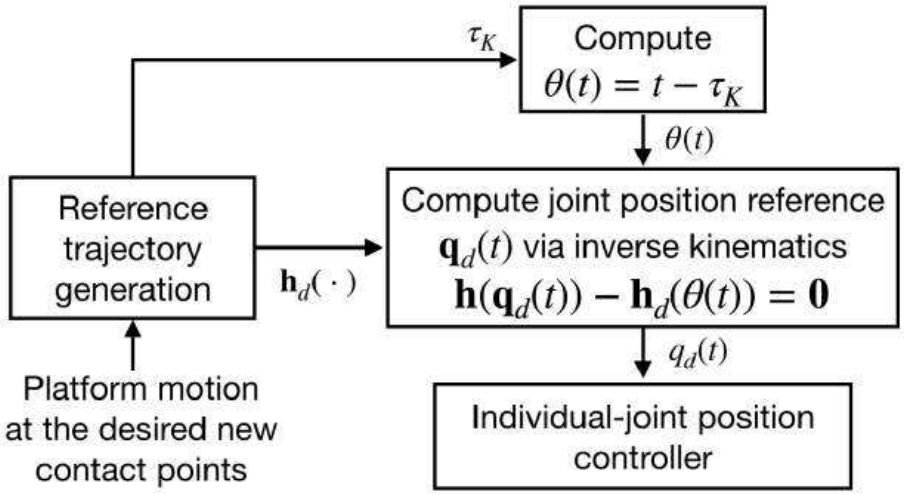

Figure 2-5: A flow chart of the controller implementation procedure used in PyBullet simulations and hardware experiments.

The treadmill used in the experiment is a split-belt Motek M-gait treadmill[141], which is capable of performing pre-programmed sinusoidal pitching motions. Its dimension is: 2.3 m (length) by 1.82 m (width) by 0.5 m (height). Its total mass is 750 kg. Each of its two belts is powered by a 4.5 kW servo motor. During experiments, the motion of the two belts is always synchronized.

The physical robot used for experimental validation is a quadrupedal Laikago robot developed by Unitree [142] (see Fig. 2-3). Its dimension is 0.56 m (length) by 0.35 m (width) by 0.6 m (height). Its total mass is 25 kg, and each leg weighs 2.9 kg. Each leg has three independently actuated joints, i.e., hip-roll, hip-pitch, and knee-pitch joints, with a power density of 0.80 kW/kg and a torque limit of 20 Nm, 55 Nm, and 55 Nm, respectively. These

torque limits are incorporated into the robot models used for the controller implementation in MATLAB and PyBullet simulations.

2.6.2 Controller Implementation in PyBullet and Experiments

In PyBullet simulations and experiments, the proposed control law in Eqs. (2.13) and (2.14), which is a torque command, is implemented as an individual-joint PD controller, which is preferred in robot walking experiments in the presence of model uncertainties [26]. Accordingly, the original reference trajectories $\mathbf{h}_d$ are converted into joint-position reference trajectories $\mathbf{q}_d(t)$, which are then sent to the robot's individual-joint controllers, as illustrated in Fig. 2-5.

2.6.3 Validation Results with a Periodically Pitching Rigid Platform

Validation results obtained from MATLAB, PyBullet, and experiments with the reference gait (G1) and the platform (P1) are displayed in Fig. 2-6. The joint trajectory tracking results in Fig. 2-6 (a) demonstrate the reliable tracking performance of the proposed control approach across simulations and experiments. The base roll and pitch trajectories in Fig. 2-6 (b) show that the robot maintains a relatively steady base pose, indicating stable walking on the pitching platform.

Joint torque profiles in Fig. 2-6 (c) overall show relatively consistent trends between simulations and experiments. Note that the hip-pitch torque profile obtained through experiments shows peaks of -10 Nm near 2 sec and 4 sec, whereas MATLAB and PyBullet results do not exhibit such jumps near those time instants. This might be caused by the impact modeling discrepancies between the physical robot and the simulated robots. During the experiment, the rear-left leg of the physical robot slightly rebounds near those time instants right after the rear-right leg strikes the platform surface. However, such rebounding behavior is not captured by the dynamic model as derived in Section 2.2.

The validation results obtained with the gaits (G2) and (G3) demonstrate similar effectiveness of the proposed control approach. These results are displayed in Figs. A-1 and A-2 in Appendix A.

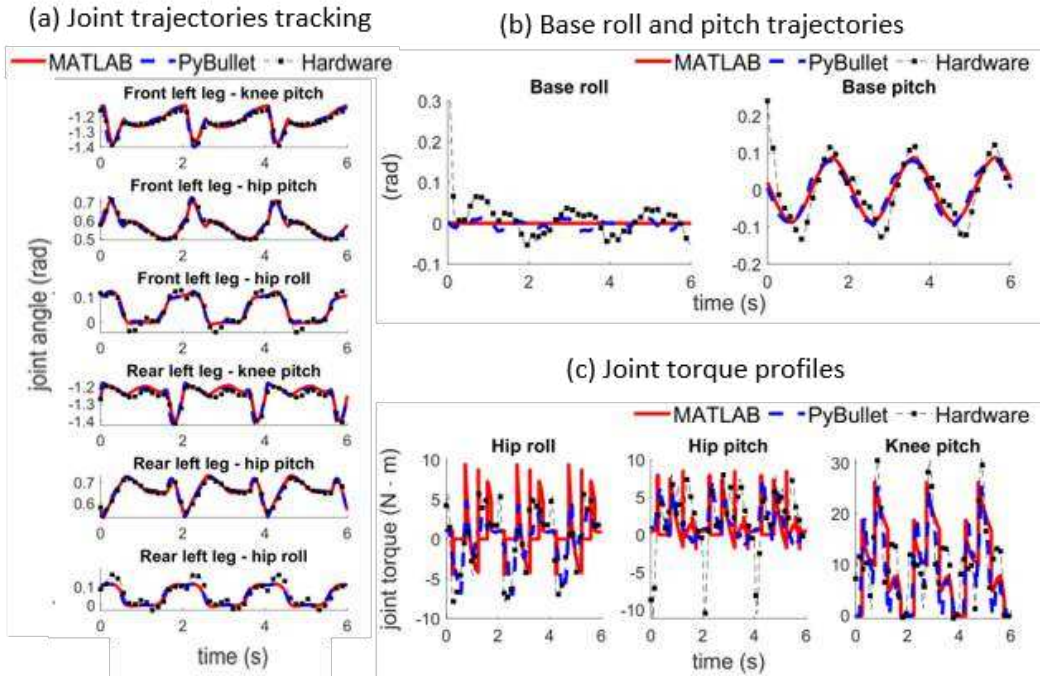

Figure 2-6: Trajectory tracking results with gait (G1): (a) joint position trajectories, (b) base roll and pitch trajectories, and (c) joint torque profiles of the rear-left leg.

2.6.4 Validation Results on Robustness

To assess the robustness of the proposed control approach under uncertainties such as sensor noise and estimation errors, perturbations in the platform motions are implemented in both simulations and experiments.

With the uncertainties (U1), up to ± 20 % uncertainties, which approximately correspond to a variation of 4 cm in the stance-foot height, are added to the nominal pitching amplitude of the platform. Figure 2-7 displays the walking control results obtained from MATLAB, PyBullet, and experiments with the reference gait (G1) and under the uncertainties (U1). The results match relatively closely with those obtained without the uncertainties (U1) in Fig. 2-6, which demonstrates the robustness of the proposed controller in mitigating a relatively moderate level of uncertainties.

With the uncertainties (U2), up to ± 20 % uncertainties, which approximately correspond to a variation of 8 cm in the stance-foot height over 10 gait cycles, are added to the belt speed of the treadmill. The robot's motion was shaky during experiments, but the robot was able to sustain motion for over twenty steps, which indicates that the inherent robustness of the

proposed control approach is able to tackle the implemented uncertainties in the treadmill belt speed. These results are displayed in Fig. A-3 of the Appendix A and match closely with the nominal validation experiments in Fig. A-1 of the Appendix A.

Under a 40% uncertainty in the DRS motion, the robot walking becomes unstable during experiments. This observation can be partly attributed to the reliance of the input-output linearizing controller on the accuracy of the robot model. Handling an error of 40% proves challenging for the controller because it significantly deviates from the nominal DRS motion. Consequently, the stability of the walking motion is compromised, indicating the sensitivity of the controller to the model accuracy.

2.6.5 Comparative Simulations of a Static-Platform Controller

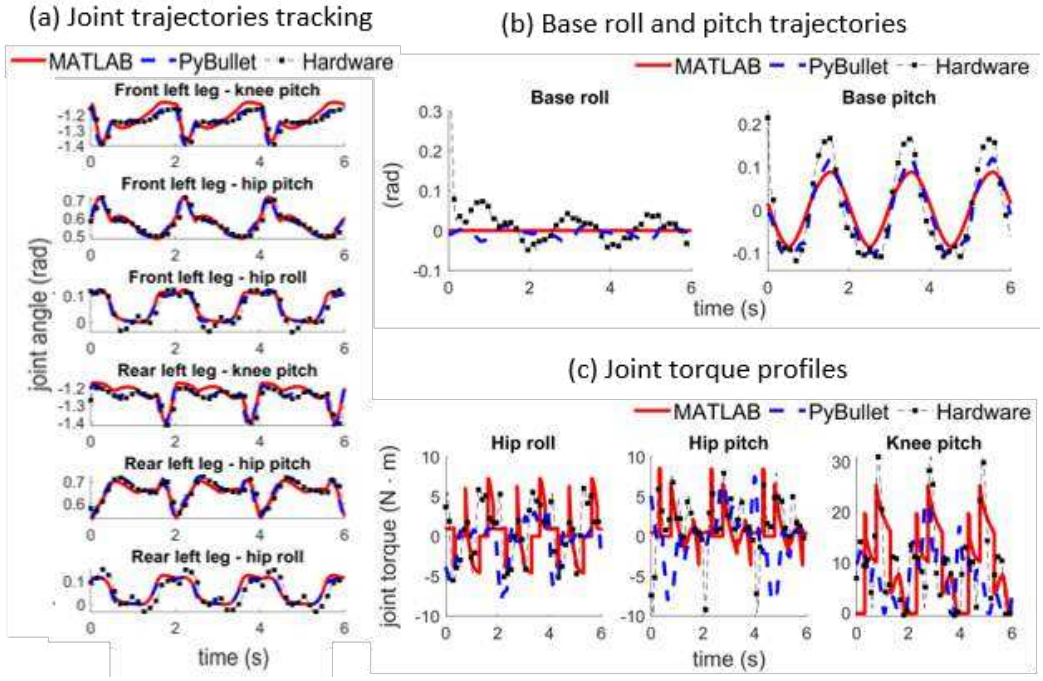

Figure 2-7: Trajectory tracking results with gait (G1) under uncertainties (U1): (a) joint position trajectories, (b) base roll and pitch trajectories, and (c) joint torque profiles of the rear-left leg.

A controller designed for static-platform locomotion is simulated to demonstrate the necessity of explicitly accounting for the time-varying robot dynamics induced by platform movement through controller design. The controller is chosen as our previous control

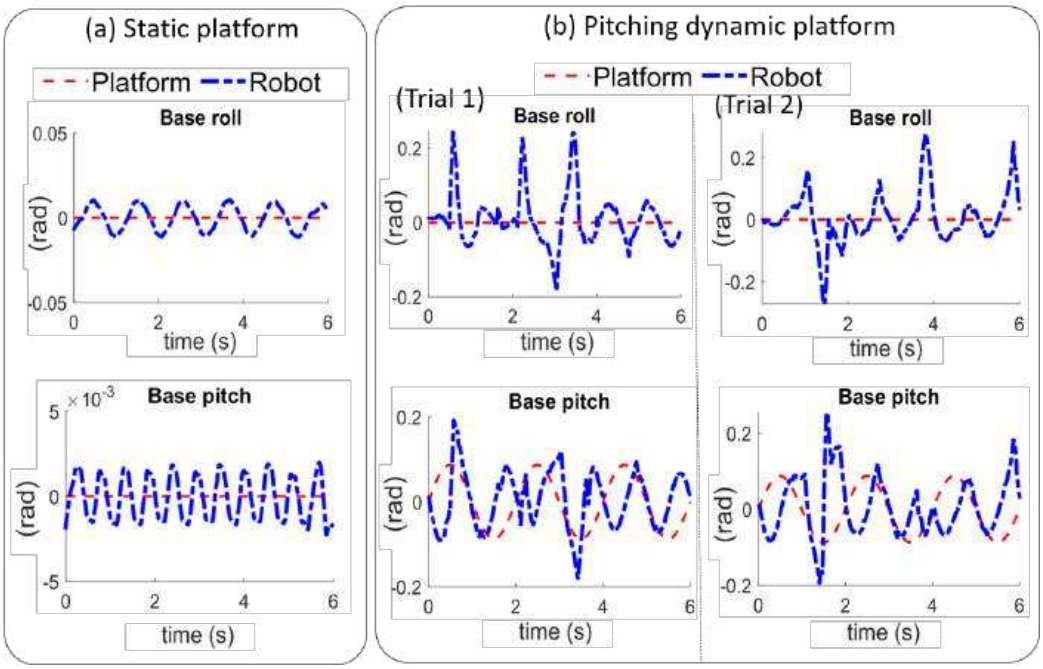

Figure 2-8: Comparative simulation results of the robot's base pose trajectories under a controller designed for static platforms, obtained during (a) walking on a static platform and (b) walking on the dynamic platform (P1).

approach for static-platform locomotion [54], which does not account for the time-varying robot dynamics induced by platform movement. The "PD" gains for PyBullet simulations are set as 1.0 and 0.25, respectively.

Figure 2-8 (a) shows the robot's base pose trajectories during static-terrain walking in PyBullet, which indicates a relatively steady base pose and thus demonstrates stable walking motions on the static terrain. However, the controller fails to sustain stable walking when the robot walks on a pitching platform (i.e., the platform (P1)), as revealed by the irregular base motion in the two trials in Fig. 2-8 (b).

2.7 Discussion

This chapter has derived a model-based control approach that achieves stable quadrupedal locomotion over dynamic rigid platforms by explicitly addressing the associated hybrid, time-varying robot dynamics. Thanks to the inherent robustness of feedback control, the controller demonstrates robustness under moderate levels of platform motion uncertainties, which

indicates that the proposed control could be effective even in the presence of uncertainties caused by platform motion estimation. Despite the impressive estimation accuracy achieved by recent studies on static-platform legged locomotion [143, 144], state estimation for dynamic-platform locomotion remains an open question. To this end, we will investigate state estimator design for dynamic-platform locomotion and integrate the estimator with the proposed control approach in our future work.

To enhance the robustness of the proposed control approach for real-world robot applications, we will extend the construction of multiple Lyapunov functions to synthesize robust control laws for hybrid, time-varying systems that include quadrupedal robots traversing dynamic rigid platforms. Uncertainties that we plan to address include model discrepancies, state estimation errors, and disturbances.

To enable robot locomotion over dynamic platforms with complex, nonperiodic motions, online motion planning techniques will be demanded in addition to a reliable control approach. Online motion planning for legged robots is a challenging problem because of the associated high computational burden. To reduce the computation burden, we will explore the possibility of using a reduced-order robot model instead of a full-order one in online motion planning. This approach is potentially promising because a physical quadrupedal robot typically has a heavy trunk and lightweight legs and thus may be relatively accurately described by a reduced-order model.

2.8 Conclusion

In this chapter, we have introduced a control approach that realizes stable quadrupedal robot locomotion on dynamic rigid platforms by provably stabilizing the associated hybrid, time-varying control system. The model of a quadrupedal robot that walks on a dynamic rigid platform was formulated as a hybrid, time-varying system consisting of continuous phases and state-triggered jumps. A continuous control law was derived to provably stabilize the system during the continuous phases. Lyapunov-based stability analysis was performed to derive sufficient conditions that can be used to directly guide the controller design for provably stabilizing the overall hybrid, time-varying control system. MATLAB and PyBullet

simulations, as well as experiments on a physical quadrupedal robot and a pitching treadmill, were performed to validate the proposed control approach. The validation results demonstrated the effectiveness of the proposed approach in realizing stable quadrupedal locomotion on dynamic rigid platforms even in the presence of moderate levels of uncertainties.

Chapter 3

Reduced-Order Modeling and Analytical Solution for Online Planning Under Periodic Surface Motions

This chapter introduces a time-varying linear inverted pendulum (LIP) locomotion model on DRS, an analytically approximate solution for sinusoidal surface motion, and a real-time hierarchical planning framework utilizing the proposed analytical solution. The contents of this chapter have been published in the following peer-reviewed journal and conference articles.

- Iqbal, A., Veer, S. and Gu, Y., 2023. Analytical Solution to a Time-Varying LIP Model for Quadrupedal Walking on a Vertically Oscillating Surface. *IFAC Mechatronics*, 96(103073).
- Iqbal, A. and Gu, Y., 2021. Extended Capture Point and Optimization-Based Control for Quadrupedal Robot Walking on Dynamic Rigid Surfaces. In Proc. of *IFAC Modeling, Estimation and Control Conference, 54(20), pp. 72-77.*

3.1 Introduction

This chapter presents the modeling and analysis of the essential dynamic behaviors of a legged robot that walks on a vertically oscillating DRS, and the design of an efficient and feasible pattern generator of quadrupedal walking that exploits the analytical results. There has been ample work on reduced-order modeling and trajectory generation of legged locomotion on stationary surfaces, but not for DRS. The research in this chapter constitutes one of the first attempts to build a reduced-order model and leverage such a model in motion planning for DRS locomotion. However, reduced-order modeling and analysis of DRS locomotion are fundamentally complex due to the nonlinear robot dynamics [145, 146, 147] and the time-varying movement of surface-foot contact points [20, 148]. This chapter aims to derive a new reduced-order model for locomotion on DRS and its analytical solution to facilitate real-time trajectory planning, which can be infeasible to achieve with the full-order model introduced in Chapter 2.

3.1.1 Related Work

(1) Reduced-Order Models of Legged Locomotion on Stationary or Dynamic Surfaces

For locomotion on DRS whose motions are affected by the robot (e.g., passive and relatively lightweight surfaces), several reduced-order robot dynamics models have been recently introduced, including an extended LIP [149], centroidal dynamics [150], and rimless-wheel models [151]. Still, it is unclear how to expand these models to DRS whose motion cannot be affected by the robot (e.g., trains, vessels, and elevators). For such substantially heavy or rigidly actuated DRS, the effects of the DRS motion on a spring-loaded inverted pendulum model have been numerically studied [43]. However, the stability conditions and analytical solution of the model remain unknown.

Beyond the scope of legged locomotion, the modeling and analysis of an inverted pendulum with vertically oscillating support, i.e., the Kapitza pendulum [152], is a classical physics problem. The Kapitza pendulum has an intriguing property that under high-frequency support oscillations, the pendulum's upper equilibrium becomes stable whereas its lower one is unstable. Yet, it is an open question whether and when the Kapitza pendulum is a reasonable

approximation of DRS locomotion. Furthermore, the motion frequencies of real-world DRS (e.g., vessels [153]) are commonly too low to meet the conditions underlying the Kapitza pendulum.

(2) Walking Pattern Generation Based on Pendulum Models

Since the LIP model represents the low-dimensional CoM dynamics of robot walking, it has been utilized to efficiently plan physically feasible walking motions on a static surface. Given the user-specified footstep and CoP positions, the exact closed-form analytical solution of the classical LIP [10, 27, 154, 155] has been used to enable real-time planning of feasible CoM trajectories for static-surface walking. This analytical solution has been augmented with the discrete-time jump of the CoM position (relative to the CoP) at a foot-landing event, which is then used to derive the desired footstep locations that provably stabilizes the hybrid LIP model [156, 157]. Recently, the exact capturability conditions of a LIP model with a time-varying CoM height have been derived based on the closed-form solution of the model's time-varying damping function at a robot's desired final CoM state [91]. These conditions are then used to plan the desired CoM and CoP trajectories with provable capturability guarantees. As reviewed in Chapter 1, the underlying LIP models of these planners assume a stationary walking surface, and therefore the planner may not be directly used for DRS locomotion when the surface motion is significant.

3.1.2 Contributions

This chapter focuses on addressing the open questions in reduced-order dynamic modeling and walking pattern generation for locomotion on DRS, as discussed in Section 3.1.1. Motivated by these research needs, this chapter aims to theoretically extend the classical LIP model [21] from stationary surfaces to substantially heavy or rigidly actuated DRS (e.g., ships), introduce an analytical approximate solution to the extended LIP model (termed as "DRS-LIP"), and develop and experimentally validate a real-time walking pattern generator that uses the proposed solution to ensure planning efficiency and feasibility. The contributions of this chapter are:

(a) Deriving the DRS-LIP model to explicitly consider the time-varying movement of the foot-surface contact region.

(b) Forming the analytical approximate solution of the DRS-LIP under a vertical, sinusoidal DRS motion.

(c) Providing the sufficient and necessary stability conditions of the DRS-LIP model based on the Floquet theory, and performing stability analysis under common ranges of surface motions and gait parameters.

(d) Designing a hierarchical walking pattern generator that utilizes the proposed analytical solution to efficiently plan feasible robot motions.

(e) Validating the accuracy and computational efficiency of the proposed analytical approximate solution through comparison with a highly accurate numerical solution.

(f) Demonstrating the planner efficiency, feasibility, and robustness through both realistic PyBullet simulations and hardware experiments under various surface movements, gait parameters, and uncertainties.

The chapter is organized as follows. Section 3.2 introduces the derivation of the proposed reduced-order model of locomotion on DRS. Section 3.3 presents the analytical approximate solution of the proposed model under a vertical, sinusoidal surface motion. Section 3.4 explains the stability condition and analysis of the proposed model. Section 3.5 develops an efficient walking pattern generator based on the analytical results. Section 3.6 reports the validation outcomes of the proposed analytical solution and planner. Section 3.7 discusses the capabilities and limitations of the proposed methods. Finally, Section 3.8 provides the concluding remarks.

3.2 Reduced-Order Model of DRS Locomotion

This section introduces an analytically tractable and computationally efficient reduced-order model that captures the essential robot dynamics associated with legged walking on DRS.

The model is derived by extending the classical LIP model [21] from a static surface to DRS, and is called "DRS-LIP".

Since today's legged robots typically have a heavy upper body and lightweight legs, their CoM dynamics can be approximately described by a LIP, i.e., a point mass atop a massless leg [21], under the assumptions (A1)-(A3). These assumptions are relisted here for ease of referencing.

(A1) The point mass of the LIP maintains a constant vertical height with respect to the support point of the LIP.

(A2) The centroidal angular momentum of the robot is negligible (or the angular velocity and angular accelerations are negligibly small).

(A3) The walking surface is rigid and flat (i.e., all the successive steps of the robot are at constant height).

In this chapter, we use a 3-D LIP to capture the essential dynamics of a 3-D legged robot walking on DRS, as shown in Fig. 4-2. The point mass and support point S in Fig. 4-2 correspond to the robot's CoM and CoP.

Let

$$\mathbf{r}_{wc} = [x_{wc},\ y_{wc},\ z_{wc}]^T \tag{3.1}$$

and

$$\mathbf{r}_{ws} = [x_{ws},\ y_{ws},\ z_{ws}]^T \tag{3.2}$$

respectively denote the positions of the CoM and point S in the world frame.

Then, the CoM position relative to point S, denoted as $\mathbf{r}_{sc}$, is defined as:

$$\mathbf{r}_{sc} = \mathbf{r}_{wc} - \mathbf{r}_{ws} =: [x_{sc},\ y_{sc},\ z_{sc}]^T. \tag{3.3}$$

The CoM dynamics during DRS locomotion can be obtained as:

$$\begin{aligned} \ddot{x}_{wc} &= \frac{f_a x_{sc}}{mr} \sin\theta, \\ \ddot{y}_{wc} &= \frac{f_a y_{sc}}{mr} \sin\theta, \quad \text{and} \\ \ddot{z}_{wc} &= \frac{f_a}{m} \cos\theta - g, \end{aligned} \tag{3.4}$$

where m is the robot's total mass, θ is the angle of the position vector $\mathbf{r}_{sc}$ relative to the vertical axis, g is the norm of the gravitational acceleration, r is the projected length of $\mathbf{r}_{sc}$ on the horizontal plane, and f_a is the norm of the ground contact force pointing from point S to the CoM.

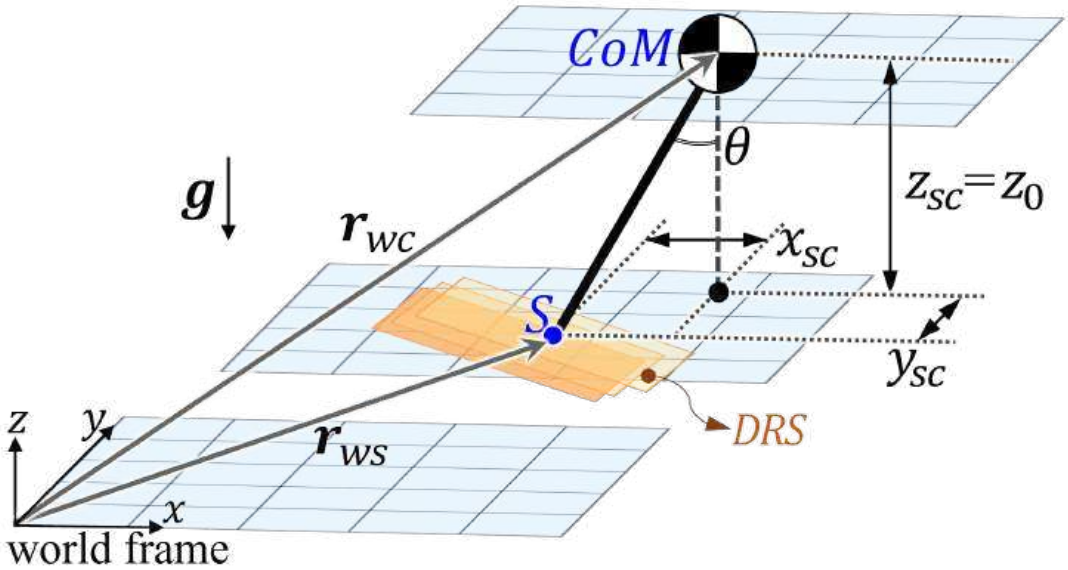

Figure 3-1: Illustration of the proposed DRS-LIP model. All three grid planes are horizontal. The top and middle ones pass through the CoM and the leg's far end S, respectively. The bottom one is fixed to the world frame.

3.2.1 DRS-LIP under a General Vertical Surface Motion

Under the assumption (A1) on the vertical distance z_{sc} between the CoM and point S (see Fig. 4-2), we have:

$$z_{sc} = z_0.$$

This assumption is the simplifying assumption of the classical LIP model that the point-mass height over the stationary surface is constant [21].

Under assumption (A1), the relationships

$$\dot{z}_{wc} = \dot{z}_{ws} \quad \text{and} \quad \ddot{z}_{wc} = \ddot{z}_{ws}$$

hold, and thus the axial force f_a becomes

$$f_a = \frac{m(\ddot{z}_{ws} + g)}{\cos\theta}. \tag{3.5}$$

Accordingly, the horizontal components of the LIP dynamics in Eq. (3.4) become:

$$\begin{aligned} \ddot{x}_{wc} &= (\ddot{z}_{ws} + g)\frac{x_{sc}}{z_0} \quad \text{and} \\ \ddot{y}_{wc} &= (\ddot{z}_{ws} + g)\frac{y_{sc}}{z_0}. \end{aligned} \tag{3.6}$$

From Eq. (3.3), we know $\ddot{x}_{wc} = \ddot{x}_{ws} + \ddot{x}_{sc}$ and $\ddot{y}_{wc} = \ddot{y}_{ws} + \ddot{y}_{sc}$. Substituting them into Eq. (3.6) yields the following horizontal LIP dynamics:

$$\begin{aligned} \ddot{x}_{sc} - \frac{(\ddot{z}_{ws} + g)}{z_0} x_{sc} &= -\ddot{x}_{ws} \quad \text{and} \\ \ddot{y}_{sc} - \frac{(\ddot{z}_{ws} + g)}{z_0} y_{sc} &= -\ddot{y}_{ws}. \end{aligned} \tag{3.7}$$

Note that the DRS' acceleration at S equals the acceleration of point S, i.e., $(\ddot{x}_{ws},\ \ddot{y}_{ws},\ \ddot{z}_{ws})$ in Eq. (3.7), when there is no slippage between the support point S and the surface.

To account for the effects of the DRS motion on the LIP dynamics in Eq. (3.7), we choose to treat the DRS acceleration $(\ddot{x}_{ws}, \ddot{y}_{ws}, \ddot{z}_{ws})$ as explicit time functions, instead of building a separate dynamics model for the DRS [150, 151]. The rationale for this design choice is twofold. First, for real-world rigidly actuated and/or heavy DRS such as vessels, the physical interaction between a robot and DRS has a negligible effect on the dynamics of DRS. Second, in real-world applications, the time profiles of DRS motion are commonly sensed, estimated, and predicted by real-time monitoring systems [158].

With the surface motion modeled as time functions, the LIP dynamics in Eq. (3.7)

are linear, nonhomogeneous, and time-varying. Since DRS, such as cruising ships in regular sea waves, have relatively small horizontal acceleration compared with vertical acceleration [159, 160, 148], we assume:

(A10) The horizontal accelerations of point S (i.e., $\ddot{x}_{ws}$ and $\ddot{y}_{ws}$) are negligible.

Under assumption (A10), the forcing terms in Eq. (3.7) (i.e., $-\ddot{x}_{ws}$ and $-\ddot{y}_{ws}$) vanish, and the horizontal LIP dynamics in Eq. (3.7) become linear, time-varying, and homogeneous:

$$\begin{aligned} \ddot{x}_{sc} - \frac{(\ddot{z}_{ws}+g)}{z_0} x_{sc} = 0 \quad \text{and} \\ \ddot{y}_{sc} - \frac{(\ddot{z}_{ws}+g)}{z_0} y_{sc} = 0. \end{aligned} \tag{3.8}$$

Recall that the vertical CoM position relative to the support point S, z_{sc}, is constant under assumption (A1), and given by:

$$z_{sc} = z_0. \tag{3.9}$$

Remark 1. ***(DRS-LIP)*** *The LIP model in Eqs.* (3.8) *and* (3.9) *describe the simplified dynamics of DRS walking under the assumptions (A1)-(A3) and (A10), which we call "DRS-LIP".*

3.2.2 DRS-LIP under a Vertical Sinusoidal Surface Motion

A real-world DRS, such as a vessel in regular sea waves, typically exhibits a vertical, sinusoidal motion with a constant amplitude and frequency [160]. Thus, we focus on such motions for further analysis of the DRS-LIP.

Under a vertical, sinusoidal surface motion, the vertical acceleration $\ddot{z}_{ws}$ of point S is sinusoidal, and Eq. (3.8) becomes the well-known Mathieu's equation [161], as explained next.

For generality, the vertical sinusoidal motion of the DRS at the surface-foot contact point can be expressed as:

$$z_{ws} = A \sin \omega t, \tag{3.10}$$

where the real scalar parameters A and ω are the amplitude and frequency of the vertical surface motion, respectively.

Then, the surface acceleration $\ddot{z}_{ws}$ at the support point is $\ddot{z}_{ws} := -A\omega^2 \sin \omega t$, with which the horizontal CoM dynamics in Eq. (3.8) becomes:

$$\begin{aligned} \ddot{x}_{sc} - \frac{(g - A\omega^2 \sin \omega t)}{z_0} x_{sc} = 0 \quad \text{and} \\ \ddot{y}_{sc} - \frac{(g - A\omega^2 \sin \omega t)}{z_0} y_{sc} = 0. \end{aligned} \tag{3.11}$$

In Eq. (3.11), the two equations of motion in the x- and y-directions are decoupled and share the same structure. Therefore, their solutions have the same form. For brevity, we focus on the DRS-LIP model in the x-direction for solution derivation and stability analysis in the rest of this chapter.

With a new time variable defined as

$$\tau := \frac{\pi + 2\omega t}{4}, \tag{3.12}$$

the DRS-LIP dynamics in Eq. (3.11) can be transformed into the standard Mathieu's equation as:

$$\frac{d^2 x_{sc}}{d\tau^2} + (c_0 - 2c_1 \cos 2\tau) x_{sc} = 0, \tag{3.13}$$

where the real scalar coefficients c_0 and c_1 are defined as

$$c_0 := -\frac{4g}{\omega^2 z_0} \quad \text{and} \quad c_1 := \frac{2A}{z_0}.$$

3.3 Approximate Analytical Solution

This section introduces a reasonably accurate and computationally efficient approximate analytical solution of the proposed DRS-LIP dynamics model under a vertical, sinusoidal surface motion.

3.3.1 Exact Analytical Solution

The DRS-LIP model in Eq. (3.13) generally does not have an exact, closed-form analytical solution. One straightforward approach to derive an approximate analytical solution is to

utilize the fundamental solution matrix based on the Floquet theory [161]. Alternatively, we choose to exploit the existing analytical results of the well-studied Mathieu's equation to obtain a more computationally efficient solution.

There are various existing analytical approximate solutions of Mathieu's equation, including periodic solutions [162] and those expressed through power series [161]. In this chapter, we adopt the exact analytical solution from [163] because of its generality and accuracy:

Theorem 1. ***(Exact solution of Mathieu's equation)*** *The exact, general (periodic or non-periodic) analytical solution of Mathieu's equation in Eq.* (3.13) *is as follows:*

$$x_{sc}(\tau) = \alpha_1 e^{\mu\tau} \sum_{n=-\infty}^{\infty} C_{2n} e^{i2n\tau} + \alpha_2 e^{-\mu\tau} \sum_{n=-\infty}^{\infty} C_{2n} e^{-i2n\tau}. \tag{3.14}$$

Here, μ is the characteristic exponent of Eq. (3.13)*, α_1 and α_2 are real scalar coefficients, n is an integer, i is a unit imaginary number, and C_{2n}'s are complex scalar coefficients.*

The proof of Theorem 1 is omitted for brevity, which can be readily obtained based on derivations in [163].

Despite its generality and exact accuracy, the analytical solution in Eq. (3.14) may demand an overly high computational load for real-time trajectory planning. Therefore, we use the exact solution to obtain an approximate solution that is reasonably accurate with a low computational cost.

To compute such an approximate solution, we need to determine the number of terms to keep in the approximate solution as well as the values of the parameters μ, α_1, α_2, and C_{2n}'s, which is explained in the rest of this section.

3.3.2 Recurrence relationship between characteristic exponent μ and solution parameters β_n's

To compute the characteristic exponent μ for obtaining the approximate solution, we first derive the recurrence relationship between μ and the solution parameters β_n's.

The solution of Mathieu's equation can be assumed as [164]:

$$x_{sc}(\tau) = e^{\mu\tau} \sum_{n=-\infty}^{\infty} C_{2n} e^{i2n\tau}. \tag{3.15}$$

Substituting Eq. (3.15) into Mathieu's equation in Eq. (3.13), we obtain:

$$\sum_{n=-\infty}^{\infty} [(\mu^2 + (i2n)^2 + 4i\mu n)C_{2n} + (c_0 - 2c_1\cos 2\tau)C_{2n}]e^{(i2n+\mu)\tau} = 0.$$

With $\cos(2\tau) = \frac{e^{i2\tau}+e^{-i2\tau}}{2}$, this equation becomes:

$$\sum_{n=-\infty}^{\infty} [(\mu^2 - (2n)^2 + 2(i\mu)(2n) + c_0)C_{2n} - c_1(e^{i2\tau} + e^{-i2\tau})C_{2n}]e^{(i2n+\mu)\tau} = 0, \tag{3.16}$$

which can be further rearranged as:

$$\sum_{n=-\infty}^{\infty} [((i\mu)^2 + (2n)^2 - 2(i\mu)(2n) - c_0)C_{2n}e^{(i2n+\mu)\tau} + c_1C_{2n}e^{(i2(n+1)+\mu)\tau} + c_1C_{2n}e^{(i2(n-1)+\mu)\tau}] = 0. \tag{3.17}$$

Since the sum in Eq. (3.17) is over $n \in (-\infty, \infty)$, we can relabel all n's in $c_1C_{2n}e^{(i2(n+1)+\mu)\tau}$ and $C_{2n}e^{(i2(n-1)+\mu)\tau}$ as $n-1$ and $n+1$, respectively.

This relabeling transforms Eq. (3.17) into:

$$\sum_{n=-\infty}^{\infty} [((2n - i\mu)^2 - c_0)C_{2n} + c_1C_{2(n+1)} + c_1C_{2(n-1)}]e^{(i2n+\mu)\tau} = 0. \tag{3.18}$$

From Eq. (3.18), we obtain the recurrence relationship between the characteristic exponent μ and the solution parameters β_n's as follows:

$$\beta_n(\mu)C_{2(n+1)} + C_{2n} + \beta_n(\mu)C_{2(n-1)} = 0, \tag{3.19}$$

where the complex scalar function β_n is defined as

$$\beta_n(\mu) := \frac{c_1}{(2n - i\mu)^2 - c_0}. \tag{3.20}$$

3.3.3 Analytical expression of characteristic exponent μ

Equation (3.19) for all $n \in \mathbb{Z}^+$ generates the following infinite set of linear homogeneous equations with the coefficients C_{2n}'s as the unknown variables:

$$\mathbf{\Delta}(\mu) \begin{bmatrix} \cdots, C_{-6}, C_{-4}, C_{-2}\, C_0, C_2, C_4, C_6, \cdots \end{bmatrix}^T = \mathbf{0}, \tag{3.21}$$

where $\mathbf{0}$ is a zero column vector with an infinite dimension and

$$\mathbf{\Delta}(\mu) := \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \iddots \\ \cdots & 0 & \beta_{-1} & 1 & \beta_{-1} & 0 & 0 & 0 & \cdots \\ \cdots & 0 & 0 & \beta_0 & 1 & \beta_0 & 0 & 0 & \cdots \\ \cdots & 0 & 0 & 0 & \beta_1 & 1 & \beta_1 & 0 & \cdots \\ \iddots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{3.22}$$

This set of linear equations have nontrivial solutions for the unknown coefficients C_{2n}'s if the determinant of $\mathbf{\Delta}(\mu)$, denoted as $|\mathbf{\Delta}(\mu)|$, equals zero. From [164], we know $|\mathbf{\Delta}(\mu)| = 0$ can be compactly expressed as:

$$2|\mathbf{\Delta}(0)| \sin^2\left(\frac{\pi\sqrt{c_0}}{2}\right) = 1 - \cosh(\mu\pi). \tag{3.23}$$

Solving Eq. (3.23) provides the exact analytical expression of μ as:

$$\mu = \pm\frac{1}{\pi}\cosh^{-1}(1 - 2|\mathbf{\Delta}(0)| \sin^2\left(\frac{\pi\sqrt{c_0}}{2}\right)). \tag{3.24}$$

Remark 2. ***(Offline computation of parameters μ and $\mathbf{\Delta}(\mu)$)*** *Recall that $c_0 := -\frac{4g}{\omega^2 z_0}$ and $c_1 := \frac{2A}{z_0}$. Therefore, the values of c_0 and c_1 are known if the user-specified CoM height z_0 is known and if the surface motion frequency ω and magnitude A are measured, estimated, or predicted in real-time (e.g., by a surface motion monitoring system [158]). With known c_0*

and c_1, the values of $\beta_n(0)$ (for all $n \in \mathbb{Z}^+$) and $|\boldsymbol{\Delta}(0)|$ are known. Then, we can pre-compute μ using its analytical expression in Eq. (3.24)*, which could then be used to compute the analytical solution during online walking pattern generation.*

3.3.4 Truncation of infinite series

Since the two infinite series in the exact solution in Eq. (3.14) are absolutely and uniformly convergent for any $0 < \tau < \infty$ [165], the exact solution is also convergent and can be approximated as a sum of finite terms.

We use $\hat{x}_{sc}(\tau)$ to denote the approximate solution. With N terms kept, the approximate solution is given by:

$$\hat{x}_{sc}(\tau) = \alpha_1 e^{\mu\tau} \sum_{n=-N}^{N} C_{2n} e^{i2n\tau} + \alpha_2 e^{-\mu\tau} \sum_{n=-N}^{N} C_{2n} e^{-i2n\tau}. \tag{3.25}$$

To simultaneously ensure sufficient accuracy and efficiency for the solution computation, we can determine the number of terms kept, N, offline for the considered range of DRS motion parameters and the user-specified solution tolerance. Specifically, we can numerically compute the minimum number of terms kept that results in a series truncation error less than the tolerance for the given DRS parameter range, which can then be used as the value of N.

3.3.5 Computation of coefficients C_{2n}'s

With the characteristic exponent, μ, and the number of terms kept, N, determined, we can obtain the value of the coefficient C_{2n} ($n \in \{0, 1, ..., N\}$) recursively based on Eq. (3.19), by setting $C_{2N} = 0$ and $C_0 = A$ [163].

The recurrence relationship in Eq. (3.19) indicates that the coefficient satisfies

$$|C_{2n}| << |C_{2(n-1)}| \tag{3.26}$$

for sufficiently large index n (e.g., $n > N$).

Hence, the coefficients with sufficiently large indices can be neglected; that is, $C_{2(N+1)} \approx 0$.

With $C_{2(N+1)} = 0$, solving the recurrence relation in Eq. (3.19) for all indices $n \in \{0,1,...,N\}$ gives:

$$
\begin{aligned}
&\text{for } n = N: \\
&\qquad \beta_N C_{2(N+1)} + C_{2N} + \beta_N C_{2(N-1)} = 0, \\
&\qquad \Rightarrow C_{2N} = -\beta_N C_{2(N-1)}, \text{ since } C_{2(N+1)} = 0 \\
&\text{for } n = N-1: \\
&\qquad \beta_{N-1} C_{2N} + C_{2(N-1)} + \beta_{N-1} C_{2(N-2)} = 0, \\
&\qquad \Rightarrow C_{2(N-1)} = \frac{-\beta_{N-1}}{1-\beta_N \beta_{N-1}} C_{2(N-2)} \\
&\text{for } n = N-2: \\
&\qquad \beta_{N-2} C_{2(N-1)} + C_{2(N-2)} + \beta_{N-2} C_{2(N-3)} = 0, \\
&\qquad \Rightarrow C_{2(N-2)} = \frac{-\beta_{N-2}}{1-\frac{\beta_{N-2}\beta_{N-1}}{1-\beta_{N-1}\beta_N}} C_{2(N-3)} \\
&\qquad\qquad \cdots
\end{aligned}
\tag{3.27}
$$

Hence, C_{2n} ($n \in \{0,1,...N\}$) can be expressed as:

$$
C_{2n} = \frac{-\beta_n}{1-\frac{\beta_n\beta_{n+1}}{1-\frac{\beta_{n+1}\beta_{n+2}}{1-\frac{\beta_{n+2}\beta_{(n+3)}}{1-\cdots}}}} C_{2(n-1)}. \tag{3.28}
$$

By setting $C_0 = A$ in Eq. (3.28), all other coefficients can be determined using Eq. (3.28). The relation in Eq. (3.28) can also be used to find the coefficients C_{-2n}, by replacing the index n with its additive inverse $-n$.

3.3.6 Computation of coefficients α_1 and α_2

The last set of parameters that need to be determined for computing the approximate analytical solution $\hat{x}_{sc}$ are α_1 and α_2, which can be obtained based on the given initial condition of $\hat{x}_{sc}$ as introduced next.

Recall that the coefficient β_n is defined in Eq. (3.20). This definition indicates that β_{-n} is the complex conjugate of β_n, and accordingly C_{-2n} is the complex conjugate of C_{2n}.

We can express the scalar, complex coefficient C_{2n} using the following generic form:

$$C_{2n} = r_{2n} e^{i\theta_{2n}}, \tag{3.29}$$

where r_{2n} and θ_{2n} are real constants.

Then, substituting $C_{2n} = r_{2n} e^{i\theta_{2n}}$ into the approximate analytical solution in Eq. (3.25) gives:

$$\begin{aligned}
\hat{x}_{sc}(\tau) &= \alpha_1 e^{\mu\tau} \sum_{n=-N}^{N} r_{2n} e^{i\theta_{2n}} e^{i2n\tau} \\
&\quad + \alpha_2 e^{-\mu\tau} \sum_{n=-N}^{N} r_{2n} e^{i\theta_{2n}} e^{-i2n\tau} \\
&= \alpha_1 e^{\mu\tau} \sum_{n=1}^{N} [r_0 + r_{2n}(e^{i(2n\tau+\theta_{2n})} + e^{-i(2n\tau+\theta_{2n})})] \\
&\quad + \alpha_2 e^{-\mu\tau} \sum_{n=1}^{N} [r_0 + r_{2n}(e^{-i(2n\tau-\theta_{2n})} + e^{i(2n\tau-\theta_{2n})})] \\
&= \alpha_1 e^{\mu\tau} \sum_{n=1}^{N} [r_0 + 2r_{2n}\cos(2n\tau + \theta_{2n})] \\
&\quad + \alpha_2 e^{-\mu\tau} \sum_{n=1}^{N} [r_0 + 2r_{2n}\cos(2n\tau - \theta_{2n})].
\end{aligned} \tag{3.30}$$

Recall $\tau := \frac{\frac{\pi}{2}+\omega t}{2}$. Replacing τ with $\frac{\frac{\pi}{2}+\omega t}{2}$ in Eq. (3.30) yields:

$$\begin{aligned}
\hat{x}_{sc}(t) &= \alpha_1 e^{\mu \frac{\frac{\pi}{2}+\omega t}{2}} \sum_{n=1}^{N} [r_0 + 2r_{2n}\cos(\frac{n\pi}{2} + n\omega t + \theta_{2n})] \\
&\quad + \alpha_2 e^{-\mu \frac{\frac{\pi}{2}+\omega t}{2}} \sum_{n=1}^{N} [r_0 + 2r_{2n}\cos(\frac{n\pi}{2} + n\omega t - \theta_{2n})].
\end{aligned} \tag{3.31}$$

Given initial condition $(\hat{x}_{sc}(0), \dot{\hat{x}}_{sc}(0))$, we can compute the coefficients α_1 and α_2 based on the solution in Eq. (3.31).

Finally, with the values of the parameters μ, C_{2n}'s, α_1, and α_2 obtained as explained earlier, we can readily compute the approximate solution by substituting those parameter values in the solution expression in Eq. (3.25).

3.4 Stability Analysis

This section presents the stability conditions of the DRS-LIP model, along with the associated stability analysis based on the proposed approximate analytical solution.

3.4.1 Sufficient and necessary stability condition

Before introducing the stability condition, we first define the notion of stability for the DRS-LIP model.

Definition 1. ***(Stability of the DRS-LIP model)*** *By the Floquet theory [166], the DRS-LIP in Eq.* (3.11) *is called "stable" if all its solutions are bounded for any $t > 0$, and is "unstable" if an unbounded solution exists for $t > 0$.*

The stability of the DRS-LIP, as defined in Definition 1, can be determined with the characteristic exponents μ. Since the DRS-LIP is a linear, second-order ordinary differential equation, it has two characteristic exponents, denoted as μ_1 and μ_2.

Theorem 2. ***(Stability condition)*** *Let $Re(\mu_1)$ and $Re(\mu_2)$ respectively denote the real parts of μ_1 and μ_2. Suppose that $Re(\mu_1) \leq Re(\mu_2)$. Then, by the Floquet theory, the model is stable if and only if $Re(\mu_1), Re(\mu_2) < 0$.*

3.4.2 Numerical stability analysis

For typical ship motions in regular sea waves, the parameters (i.e., displacement magnitude A and frequency ω) of the DRS-LIP model in Eq. (3.13) take values within [153]:

$$A \leq 100 \text{ cm} \quad \text{and} \quad \omega \leq 2\pi \text{ rad/s}.$$

Also, the kinematically feasible CoM height z_0 of a typical quadrupedal robot (e.g., Unitree's Laikago) is within:

$$0.3 \text{ m} < z_0 < 0.55 \text{ m}.$$

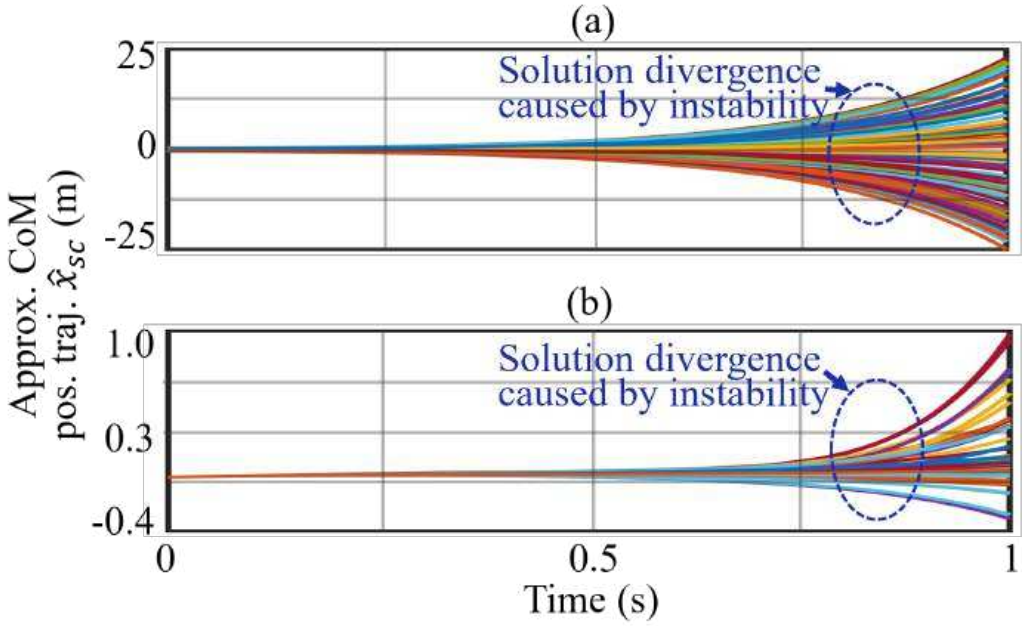

Figure 3-2: Unbounded time evolution of solution $\hat{x}_{sc}(t)$ of the DRS-LIP model under: (a) the same model parameters ($A = 7$ cm, $\omega = \pi$ rad/s, and $z_0 = 42$ cm) but 100 different initial conditions satisfying $|x_{sc}(0)| < 0.4$ m and $|\dot{x}_{sc}(0)| < 0.4$ m/s and (b) different parameters ($0 < \omega \leq 2\pi$ rad/s, $0 < A \leq 100$ cm, and $30 \leq z_0 \leq 55$ cm) but the same initial condition ($x_{sc}(0) = 0.02$ m and $\dot{x}_{sc}(0) = 0.1$ m/s).

Under these parameter ranges, we use Eq. (3.24) to numerically compute the characteristic exponents and obtain that

$$\mathrm{Re}(\mu_2) > 0 \quad \text{and} \quad \mathrm{Re}(\mu_1) < 0.$$

Therefore, by the Floquet theory [166], the DRS-LIP is unstable (i.e., an unbounded solution exists on $t > 0$) under the considered operating condition.

To illustrate this physical insight, Fig. 3-2 presents the corresponding approximate analytical solutions. Subplot (a) displays the approximate solutions under different initial conditions ($|x_{sc}(0)| < 0.4$ m and $|\dot{x}_{sc}(0)| < 0.4$ m/s) and DRS-LIP parameters ($\omega = \pi$ rad/s, $A = 7$ cm, and $z_0 = 42$ cm). Subplot (b) shows the solutions under the same initial condition ($x_{sc}(0) = 0.02$ m and $\dot{x}_{sc}(0) = 0.1$ m/s) but different model parameters ($0 < \omega \leq 2\pi$ rad/s, $0 < A \leq 100$ cm, and $30 \leq z_0 \leq 55$ cm). In all cases except for the trivial initial condition $x_{sc}(0),\ \dot{x}_{sc}(0) = 0$, the solutions grow towards infinity as time t increases, confirming that

the DRS-LIP model is unstable under the considered operating condition.

Remark 3. ***(Effects of DRS-LIP model stability on robot walking stability)*** *Despite the instability of the DRS-LIP model during continuous stance phases, the model is useful for the planning and control of a full-order robot to ensure robot walking stability. This is essentially because as long as the desired CoM motion is feasible during continuous phases, there exists a wide class of nonlinear control approaches (e.g., our prior input-output linearizing controller [20, 167]) that can provably guarantee the walking stability for the overall hybrid full-order robot model. In this chapter, we implement such a controller to indirectly validate the feasibility of the proposed planner (see Section 3.6.5).*

3.5 DRS-LIP Model based Walking Pattern Generation

To demonstrate the practical uses of the DRS-LIP model and its analytical solution, this section presents a hierarchical walking pattern generator that exploits these theoretical results to enable efficient and feasible planning for quadrupedal walking on a vertically oscillating DRS.

The planner is designed for quadrupedal walking [20, 24] whose gait cycle comprises four continuous foot-swinging phases and four discrete foot-landing events (see Fig. 3-3). The planner has two layers as illustrated in Fig. 3-4: (i) the higher layer generates kinematically and dynamically feasible CoM position trajectories, by using the proposed analytical solution of the DRS-LIP model and incorporating necessary feasibility constraints, and (ii) the lower layer utilizes trajectory interpolation to efficiently translate the CoM trajectories into the desired motion for all degrees of freedom of the full-order robot model.

Since the time profiles of the DRS motion are commonly sensed or estimated by real-time motion monitoring systems (e.g., shipboard sensors [158]), the planner design assumes that the nominal profile of the surface motion is known with bounded *inaccuracy*. Hardware experiment results that demonstrate the inherent robustness of the proposed framework under inaccurate surface motion knowledge is provided in Section 3.6.

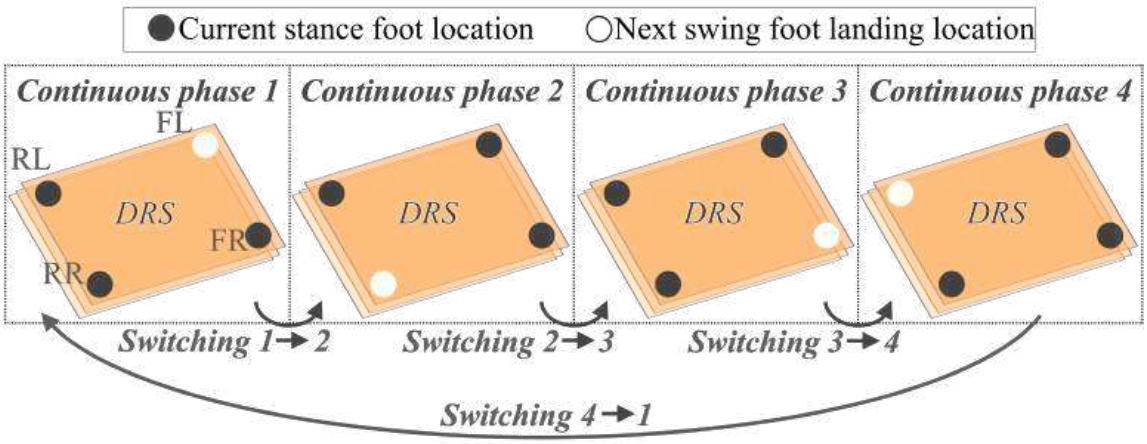

Figure 3-3: A complete quadrupedal walking cycle, with the four feet marked as Front Left (FL), Front Right (FR), Rear Left (RL), and Rear Right (RR).

3.5.1 Higher-Layer CoM Trajectory Planner

The higher-layer planner uses the DRS-LIP as a basis to efficiently generate feasible reference trajectories of the CoM position $\mathbf{r}_{sc}(t)$ through nonlinear optimization.

As the planner is introduced primarily for highlighting the usefulness of the analytical results, we construct a simple higher-layer planner that is computationally efficient for real-world implementations, by reducing the number of optimization variables through the pre-specification of common gait parameters (which characterize the desired gait features). While a simple CoM trajectory planner is developed in this chapter based on the proposed model and analytical solution, the model and solution could be incorporated in a more complex planner (e.g., [168]) that demands fewer user-defined gait parameters and is more versatile for generating complex locomotion tasks.

(1) User-defined gait parameters

Similar to [169], the pre-specified gait parameters in this research are chosen as: (i) average walking velocity (i.e., horizontal CoM velocity), (ii) foot contact sequence (see Fig. 3-3), (iii) stance foot positions, (iv) constant CoM height z_0 above the surface (for respecting assumption (A1)), and (v) gait period. The values of parameters (i)-(iv) are typically set to help ensure a kinematically feasible gait. The value of the parameter (v) is selected such that the quotient of the DRS' nominal motion period and the desired gait period is an integer

(i.e., the desired CoM motion complies with the DRS motion).

(2) Optimization variables

We choose the optimization variables $\boldsymbol{\alpha}$ of the planner as the initial CoM position (x_{sc}, y_{sc}) and velocity ($\dot{x}_{sc}$, $\dot{y}_{sc}$) within each continuous phase. The rationale for this choice is that these variables, along with the DRS-LIP model parameters, completely determine the horizontal CoM position trajectories. The vertical CoM position z_{sc} is not included as an optimization variable because it can be readily obtained from the user-defined CoM height z_0.

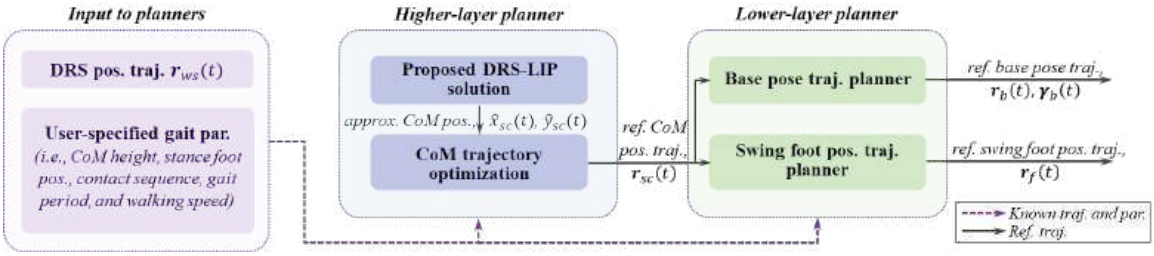

Figure 3-4: Overview of the proposed hierarchical walking pattern generator. The higher layer exploits the proposed analytical solution of the DRS-LIP model to ensure efficient and physically feasible planning of the desired CoM position trajectories $\mathbf{r}_{sc}(t)$. The lower layer converts the reference CoM trajectories $\mathbf{r}_{sc}(t)$ into full-body reference motions ($\mathbf{r}_b(t)$, $\boldsymbol{\gamma}_b(t)$, and $\mathbf{r}_f(t)$) through trajectory interpolation.

(3) Constraints

We design the constraints to enforce physical feasibility [170] and desired gait features. Note that these constraints are formed based on the proposed analytical approximate solution $\hat{x}_{sc}$. The equality constraints include: (i) continuity of the CoM trajectories at the foot-landing events and (ii) the desired walking velocity. The inequality constraints are: (i) friction cone constraint for avoiding foot slipping, (ii) confinement of CoM trajectories within the polygon of support for approximately respecting the CoP constraint, and (iii) upper and lower bounds on $\boldsymbol{\alpha}$.

To meet the constraints, $\boldsymbol{\alpha}$ is obtained by solving the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{\alpha}} \quad & h(\boldsymbol{\alpha}) \\ \text{subject to} \quad & \mathbf{f}_{eq}(\boldsymbol{\alpha}) = \mathbf{0},\ \mathbf{g}_{ineq}(\boldsymbol{\alpha}) \leq \mathbf{0}, \end{aligned} \tag{3.32}$$

where $h(\boldsymbol{\alpha})$ is a scalar cost function (e.g., energy cost of transport), and the vector-valued functions $\mathbf{f}_{eq}$ and $\mathbf{g}_{ineq}$ are the sets of all aforementioned equality and inequality constraints, respectively. The expressions of $\mathbf{f}_{eq}$ and $\mathbf{g}_{ineq}$ are omitted for space consideration.

3.5.2 Lower-Layer Full-Body Trajectory Generator

The lower-layer planner is essentially trajectory interpolation that translates the reference CoM trajectory $\mathbf{r}_{sc}(t)$ (supplied by the higher-layer planner) into the robot's full-body trajectories. To impose a steady trunk/base pose and to avoid swing foot scuffing on the surface, we choose these full-order trajectories to be the absolute base pose (position $\mathbf{r}_b$ and orientation $\boldsymbol{\gamma}_b$) and the swing foot position $\mathbf{r}_f$ relative to the base.

The input to the lower-layer planner (see Fig. 3-4) are: the nominal DRS motion that is vertical and sinusoidal, the CoM position trajectories provided by the higher-layer planner, and the user-defined parameters (e.g., CoM height, stance foot locations, and maximum swing foot height).

(1) Base pose trajectories

The CoM of the robot is approximated as the base (i.e., the geometric center of the trunk) because a quadruped's trunk typically has a symmetric mass distribution and is substantially heavier than the legs. Thus, we set the desired base position trajectories $\mathbf{r}_b(t)$ same as the desired CoM position trajectories $\mathbf{r}_{sc}(t)$. As real-world locomotion tasks are typically encoded by a robot's absolute global/base position, we choose to transform these relative position trajectories into the absolute ones.

With the DRS position $\mathbf{r}_{ws}(t)$ at the support point S, we obtain the absolute base position trajectories $\mathbf{r}_b(t)$ as:

$$\mathbf{r}_b(t) = \mathbf{r}_{sc}(t) + \mathbf{r}_{ws}(t). \tag{3.33}$$

To avoid overly stretched leg joints for ensuring kinematic feasibility, the desired base orientation trajectories $\boldsymbol{\gamma}_b(t)$ are designed to comply with the DRS orientation.

(2) Swing foot position trajectories

The desired swing foot trajectories $\mathbf{r}_f(t)$ (relative to the support point S) are designed to agree with the desired stance foot locations and to respect the kinematic limits of the robot's leg joints. Specifically, we plan the desired swing foot trajectory during a continuous phase by using Bézier polynomials [20] to connect the adjacent desired stance foot positions.

We use s to denote the scalar normalized phase variable that represents how far a walking step has progressed. Let $\mathbf{r}_{f,i}$ and $\mathbf{r}_{f,e}$ respectively denote the desired swing foot locations at the initial and end instants of a continuous phase. We assign the values of $\mathbf{r}_{f,i}$ and $\mathbf{r}_{f,e}$ to match the desired stance foot locations for the given continuous phase.

Then, we use the following Bézier curve to express the desired swing foot position $\mathbf{r}_f$ within the given phase:

$$\mathbf{r}_f(s) = \mathbf{r}_{f,i} + \mathbf{P}(s)(\mathbf{r}_{f,e} - \mathbf{r}_{f,i}), \tag{3.34}$$

where $\mathbf{P}(s)$ is a 3×3 diagonal matrix function with each diagonal term an n^{th}-order Bézier polynomial interpolation.

For walking along a straight line, we can design the lateral swing foot position as constant for simplicity. We design the forward and vertical trajectories to have a relatively fast initial velocity within Continuous Phases 1 and 3, as illustrated in Fig. 3-5 (a). This relatively fast initial velocity allows the robot's full body to have sufficient momentum to leave the previous support polygon and enter the planned current polygon, thereby indirectly meeting the CoP constraints under the desired contact sequence.

Also, as inspired by previous quadrupedal robot planning [24], a brief four-leg-in-support phase is inserted upon a foot-landing event when the two consecutive polygons of support only share a common edge (i.e., "*Switching* $1 \rightarrow 2$" and "*Switching* $3 \rightarrow 4$" in Fig 3-3), so as to ensure smooth and feasible transitions during these events. This transitional phase is highlighted with a grey background in Fig. 3-5. Thanks to this transitional phase, the initial

forward and vertical swing foot velocities within Phases 2 and 4 do not need to be as fast as Phases 1 and 3 (see Fig. 3-5 (b)).

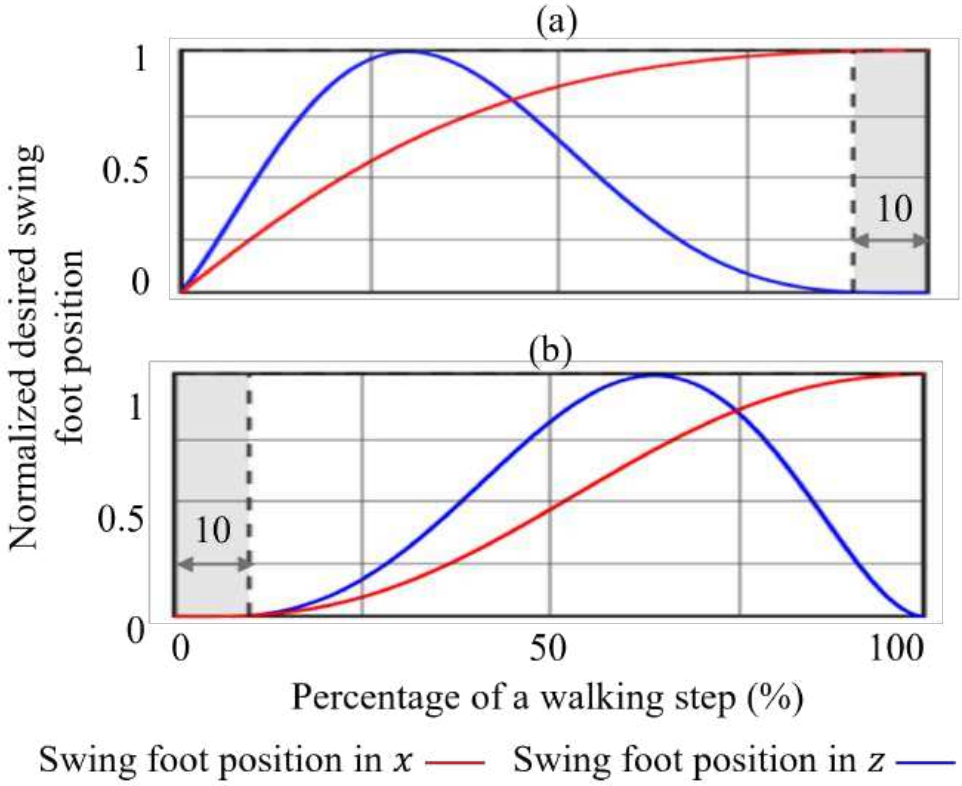

Figure 3-5: Normalized swing foot position trajectories in x- and z-directions during (a) Continuous Phases 1 and 3 and (b) Continuous Phases 2 and 4. The grey background highlights the transitional four-leg-in-support phase.

Remark 4. ***(Effects of LIP model accuracy on trajectory generation feasibility)*** *The dynamic feasibility of the planned trajectories partly depends on the closeness between the DRS-LIP model and the actual robot dynamics. The DRS-LIP model is a relatively faithful representation of an actual DRS-robot system when the robot and DRS behaviors meet the assumptions (A1)-(A3) and (A10) that underlie the proposed model and its solution. Indeed, assumption (A10) holds when the surface motion is vertical and sinusoidal, and the planner explicitly imposes assumption (A1). Moreover, as the planner enforces the desired base orientation to comply with the surface orientation for kinematic feasibility, the planned motion will reasonably respect assumption (A2) for surfaces that translate without rotary motions. Even for real-world DRS that rotate (e.g., vessels), the rate of the robot's centroidal*

angular momentum will be negligible under the typical angular movement range of those DRS [153], thus still respecting assumption (A2).

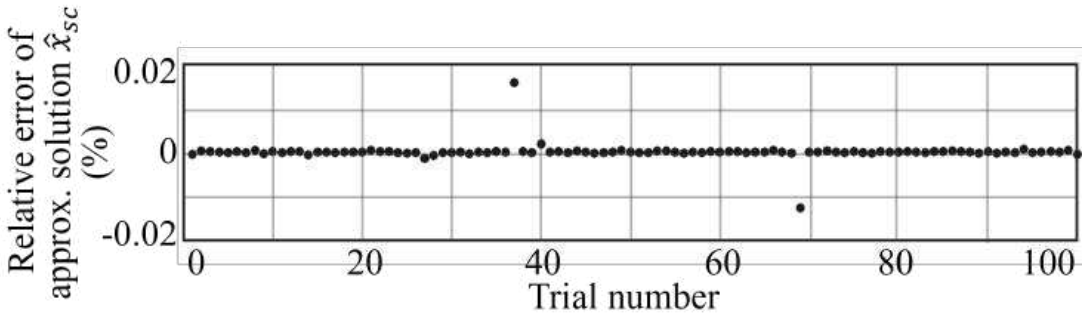

Figure 3-6: Mean percentage error of the proposed analytical approximate solution compared with the high-accuracy numerical solution under model parameters $A = 7$ cm, $\omega = \pi$ rad/s, and $z_0 = 42$ cm for 100 random initial conditions satisfying $|x_{sc}(0)| < 0.2$ m and $|\dot{x}_{sc}(0)| < 0.2$ m/s.

Table 3.1: Average computation time of analytical and numerical solutions for 1000 trials in MATLAB (mean $\pm$ SD)

Solution method	Computation time (ms)
Numerical	2.61 ± 0.43
Analytical (proposed)	0.16 ± 0.02

3.6 Simulation and Experiment Validation

This section presents the simulation results that validate the proposed approximate analytical solution and the walking pattern generator.

3.6.1 Validation of Solution Accuracy and Efficiency

The accuracy and computational efficiency of the proposed analytical approximate solution in Eq. (3.25) is assessed through comparison with a highly accurate numerical solution. For fairness of comparison, both solutions are computed in MATLAB on $t \in [0,\ 0.5]$ sec. The approximate solution has ten terms kept (i.e., $N = 10$) for a reasonable trade-off between

accuracy and computational efficiency. The comparative numerical solution is computed using MATLAB's `ODE45` solver with an error tolerance of 10^{-9} and at a time interval of 0.5 ms.

To validate the proposed solution under different initial conditions, 1000 sets of initial conditions are randomly chosen within a common movement range of quadrupedal walking [24]: $|x_{sc}(0)| < 0.2$ m and $|\dot{x}_{sc}(0)| < 0.2$ m/s. The DRS-LIP model parameters are chosen to be within realistic ranges of DRS motions [160, 153] and quadrupedal robot dimensions [20]: $A = 7$ cm, $\omega = \pi$ rad/s, and $z_0 = 42$ cm.

Figure 3-6 shows the accuracy of the approximate analytical solution (with ten terms kept) compared with the numerical solution for 100 out of the 1000 trials. Within those 100 trials, the maximum value of the absolute percentage error is lower than 0.02% in magnitude, indicating the reasonable accuracy of the proposed approximate solution. For all 1000 trials, the absolute percentage error, measured by mean $\pm$ one standard deviation (SD), is $(0.0012 \pm 0.005)\%$.

Table 3.1 displays the comparison of the average computational time cost (measured by mean$\pm$SD) for the aforementioned 1000 trials. The proposed approximate analytical solution is about 15 times faster to compute than the numerical one.

3.6.2 Simulation and Experimental Setup for Planner Validation

The setup of simulations and hardware experiments is shown in Fig. 3-7. The experiment video is available at `https://youtu.be/F9LH8mdhedg`.

(1) Robot

The validation of the planner utilizes a Laikago quadruped (see Fig. 3-7) developed by Unitree Robotics. The dimension of the robot is 55 cm $\times$ 35 cm $\times$ 60 cm. The robot's total mass is 25 Kg and a power density of 0.8 kW/kg. The robot is powered by a 650 Wh lithium-ion battery weighing 4.4 kg. It has twelve independently actuated joints that can produce up to 18 kW of instantaneous power. Each leg weighs 2.9 kg and has three brushless DC motors located close to the trunk. The torque limits of the three hip-roll, hip-pitch, and

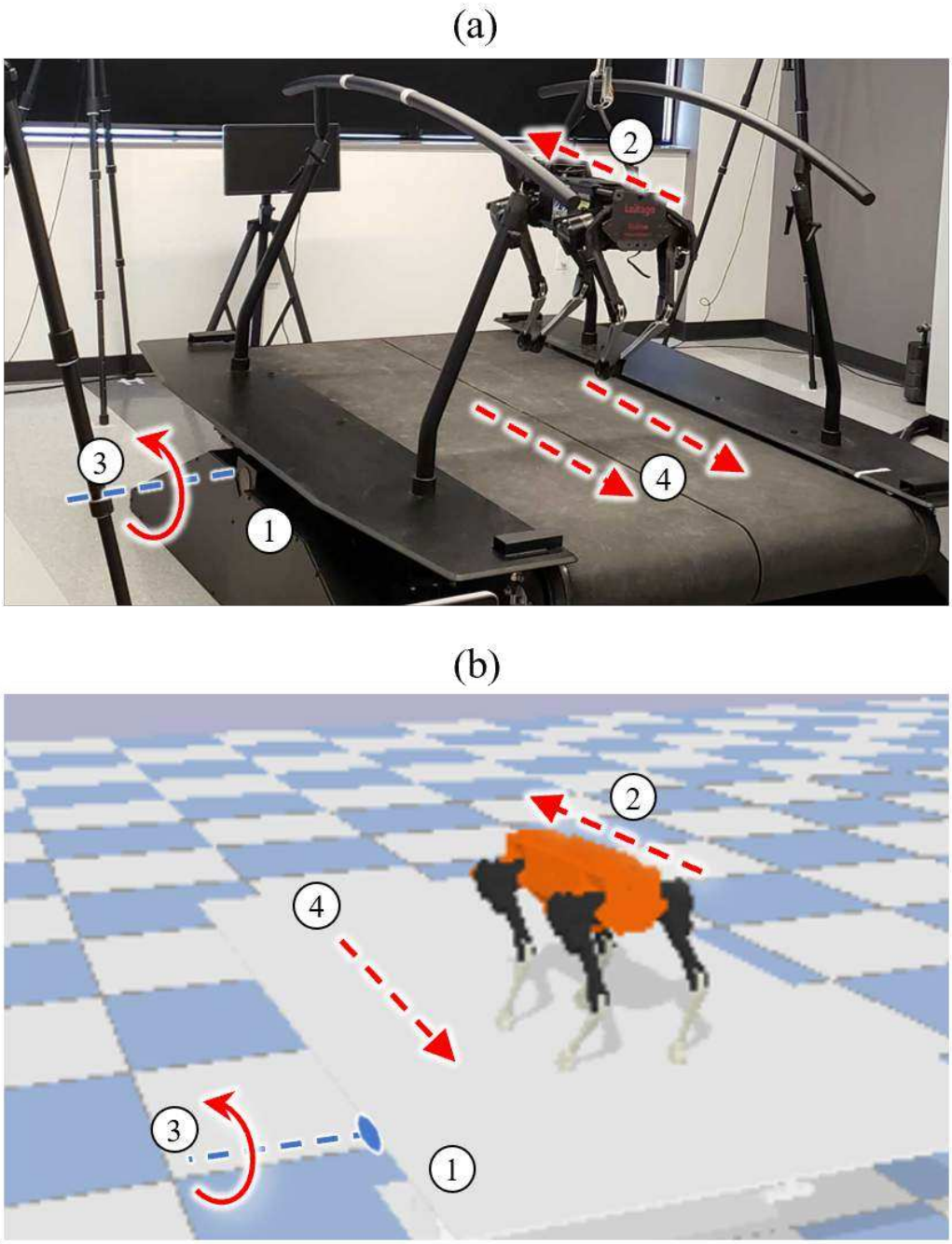

Figure 3-7: Setup of (a) experiments and (b) PyBullet simulations for testing the planner effectiveness using a pitching Motek treadmill (①) and a Laikago quadruped (②). The treadmill has a split belt (④) that moves at a constant speed while the treadmill rocks about the horizontal axis (③).

knee-pitch motors are 20 Nm, 55 Nm, and 55 Nm, respectively. The robot is equipped with an IMU at the trunk, 12 joint encoders, and a contact sensor at each foot.

Table 3.2: User-defined gait parameters in walking pattern generation.

Gait parameter	(G1)	(G2)
Friction coefficient	0.5	0.5
Robot's base height z_0 (cm)	42	42
Gait duration (s)	2	2
Average walking velocity (cm/s)	5	6
Step length (cm)	10	12
Max. step height (cm)	5	5

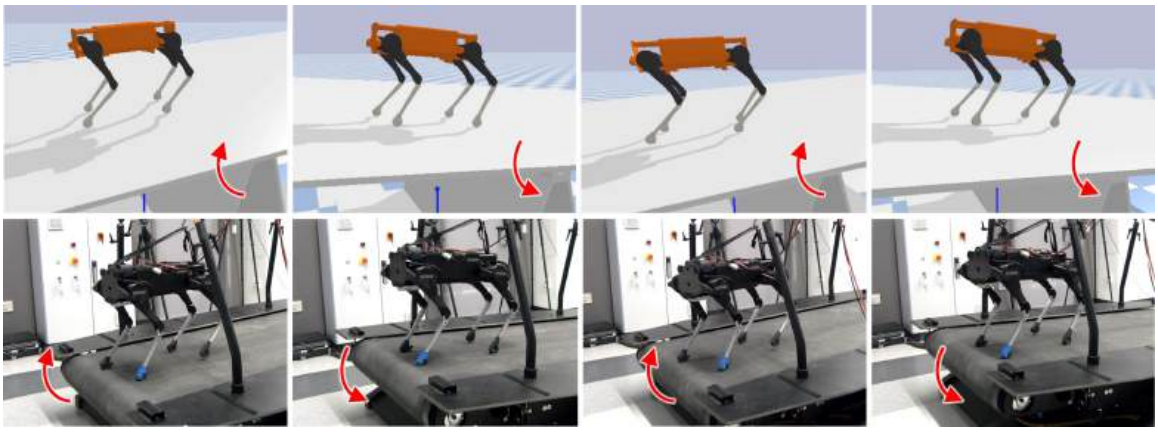

Figure 3-8: Image tiles of quadrupedal walking on a pitching treadmill under the proposed walking pattern generator synthesized based on the DRS-LIP model and its analytical solution. The top and bottom rows respectively show PyBullet simulations and hardware experiments.

(2) Nominal and uncertain DRS motion

Three DRS motions are tested to assess the efficiency, feasibility, and robustness of the planner under different surface motions that emulate vessel movements in regular sea waves [160].

Due to our limited equipment access to programmable, actuated DRS that exclusively exhibits vertical motion, we focus on PyBullet simulations for the planner validation on a vertically moving surface with the following nominal profile without movement uncertainties:

(DRS1) The DRS motion is vertical and sinusoidal with $A = 10$ cm and $\omega = \pi$ rad/s.

Meanwhile, we use a Motek M-Gait treadmill that performs a pitching motion (see Fig. 3-

7) to approximate a vertically moving DRS both in hardware experiments and PyBullet simulations. The Motek treadmill can be pre-programmed to perform user-defined pitching (but not vertical) motions and belt translation. The treadmill weighs 750 kg with a dimension of 2.3 m $\times$ 1.82 m $\times$ 0.5 m. A 4.5 kW servo motor powers each of the treadmill's two belts. During the hardware experiments, the robot is placed approximately 1 m away from the treadmill's axis of rotation, and the belt speed is set to be the same as the desired walking speed. Figure 3-8 shows images of the Laikago robot walking on the rocking treadmill in simulations and experiments.

The actual movement of the pitching treadmill at the foot-surface contact points is different from the nominal, vertical DRS motion used in planning and control, since the pitching treadmill naturally possesses non-negligible horizontal motion and the robot's actual location on the treadmill directly affects the actual surface motion at the support feet. Both the nominal and actual surface motions are listed below for two surface profiles tested:

(DRS2) The nominal, vertical surface motion used in planning and control is sinusoidal with $A = 7$ cm and $\omega = \pi$ rad/s. The actual, pitching DRS motion is a sinusoidal function with an amplitude of 5° and frequency of 0.5 Hz.

(DRS3) The nominal, vertical surface motion used in planning and control is sinusoidal with $A = 11$ cm and $\omega = \pi$ rad/s. The actual, pitching DRS motion is a sinusoidal function with an amplitude of 7° and frequency of 0.5 Hz.

For all surface motions (DRS1)-(DRS3), the surface accelerations in the vertical direction are relatively significant for planner validation, with peak contact-point accelerations approximately at 100 cm/s^2, 70 cm/s^2, and 110 cm/s^2 in magnitude, respectively, when the robot stands about 1 m away from the treadmill's axis of pitching. The corresponding contact-point displacements are 10 cm, 7 cm, and 11 cm, respectively.

For the uncertain surface motions (DRS2) and (DRS3), the upper bounds of the uncertainty in the absolute acceleration of the surface-robot contact points are approximately 20 cm/s^2 and 10 cm/s^2 in vertical and horizontal directions, respectively. Thus, the maximum absolute uncertainties are about 10-20% of the peak vertical acceleration at the contact points.

(3) Validation cases

We validate the efficiency, feasibility, and robustness of the proposed planner under four different combinations of surface motions and desired gait features, both through hardware experiments and simulations. The gait features are sampled from the two parameter sets (G1) and (G2) as specified in Table 3.2.

To assess the planning efficiency, computations in MATLAB and C++ are conducted under:

(Case 1) Combination of surface motion (DRS2) and gait parameters (G1).

To validate the planner feasibility under no surface motion uncertainty, we perform PyBullet simulations under:

(Case 2) Combinations of vertical surface motion (DRS1) and gait (G1).

Finally, both hardware experiments and Pybullet simulations are used to evaluate the planner robustness in the presence of uncertain surface motions:

(Case 3) Combination of uncertain surface motion (DRS2) and gait parameters (G1) and

(Case 4 Combination of uncertain surface motion (DRS3) and gait pameters (G2).

3.6.3 Planner Efficiency Validation

To demonstrate that utilizing the proposed analytical solution improves the planner efficiency compared with the numerical solution, the higher-layer CoM trajectory planning problem is solved based on both solutions under Case 1.

For simplicity, the cost function h in Eq. (3.32) is chosen as trivial. A 6^{th}-order Bézier curve is used to design the desired swing foot trajectory for allowing adequate freedom in trajectory design. Also, we choose to lower the load of computing the proposed analytical solution by pre-computing its solution parameters μ and C_{2n}, as discussed in Remark 2.

To assess the planner efficiency under different common solvers, both MATLAB and C++ are used to solve the optimization-based planning problem in Eq. (3.32) for 1000 runs with the same initial guess of the optimization variable $\boldsymbol{\alpha}$. For fairness of comparison, the

Table 3.3: Average time cost of 1000 runs of higher-layer planning (mean±SD) under Case 1.

Solution method	MATLAB (fmincon)	C++ (Ipopt)
Numerical (ms)	1320.7 ± 13.8	72.3 ± 6.7
Analytical (ms)	269.1 ± 12.9	8.6 ± 2.2

optimality and constraint tolerances are set as 10^{-6} in all runs. In MATLAB, `fmincon` is used with an interior-point solver. For the C++ optimization, the nonlinear optimization solver of the Ipopt package [171] is utilized. We solve the optimization problem on a Windows 10 PC with 32 GB DDR4 RAM and an Intel Xeon W-10855M processor running at a base speed of 2.8 GHz.

For those 1000 runs, Table 3.3 shows that the mean time costs of the analytical solution based higher-layer planning is approximately 7 and 4 times shorter than the numerical solution based one in MATLAB and C++, respectively.

Furthermore, Table 3.3 indicates that the higher-layer planner takes 8.6 ± 2.2 ms to generate the desired CoM trajectory when it is solved by C++ using the approximate analytical solution. The median time cost of those 1000 runs of computations is 8.4 ms. Also, solving the lower-layer planner is typically fast (e.g., MATLAB can solve it within 2 ms) since the planning is essentially trajectory interpolation. Therefore, the mean time cost for solving both higher and lower layers will be less than 11 ms. Since such a time cost is much smaller than the typical quadrupedal walking gait period (i.e., about 2 sec [24]) and real-world DRS motion periods (e.g., 1-100 sec for vessels [153]), the proposed planner would be adequately fast to timely regenerate the desired full-order trajectories in case of any significant changes in the DRS motion.

3.6.4 Comparing Planner Efficiency

To highlight the significant reduction in planning time of the proposed walking pattern generator thanks to the use of our proposed time-varying LIP model, we compare the time cost of the proposed generator for planning 2 s of walking motion with the existing

planning method that utilizes a full-order robot model [52, 20]. This comparative planner is formulated as an optimization problem similar to the proposed pattern generator.

The optimization variables of the full-order model based planner is the vector of Bézier coefficients $\boldsymbol{\beta}$ associated with the base pose and swing foot trajectories.

The equality constraints include: (i) continuity of the desired base pose trajectories across walking phases and (ii) the desired walking velocity. The inequality constraints are: (i) bounds on the joint configuration, (ii) bounds on the CoM position trajectories, and (iii) upper and lower bounds on $\boldsymbol{\beta}$. The cost function is set as a scalar cost function with the same physical meaning as the proposed walking pattern generator.

Under the same settings for the optimization problem, computing unit, and user-defined parameters as in Section 3.6.3, the MATLAB planning time for 2 s of walking pattern based on the full-order robot model, among 10 trials, is 162.4 ± 13.7 s while the time cost is 0.27 ± 0.01 s for the proposed planner based on the time-varying LIP model. This comparative simulation demonstrates a substantial improvement in planning efficiency based on the proposed model simplification, enabling real-time walking pattern generation.

3.6.5 Planner Feasibility Validation

Besides efficiency, the proposed DRS-LIP and its solution can also be used to guarantee planning feasibility, which is validated in PyBullet simulations under Case 2.

To test the feasibility of the planned motion, we choose to implement our previous nonlinear feedback controller [20] that does not explicitly ensure the feasibility of ground contact forces during hardware experiments (Remark 3). If the controller turns out to be effective in sustaining stable walking on DRS, we can infer that the planned trajectory is at least approximately physically feasible.

This controller is derived based on the hybrid full-order robot model and proportional derivative (PD) control. Given feasible desired trajectories, it provably guarantees the walking stability. To help ensure a reasonable tracking performance, the PD gains are tuned as 0.7 and 1.0 in simulations, and 5.5 and 0.15 on hardware.

As shown in Fig. 3-9, the robot sustains stable walking for the entire testing period of 50

gait cycles. The base and joint trajectories closely track their reference values, as shown in subplots (a) and (b). Also, subplot (c) indicates that the actual robot motion indeed respects the torque limits.

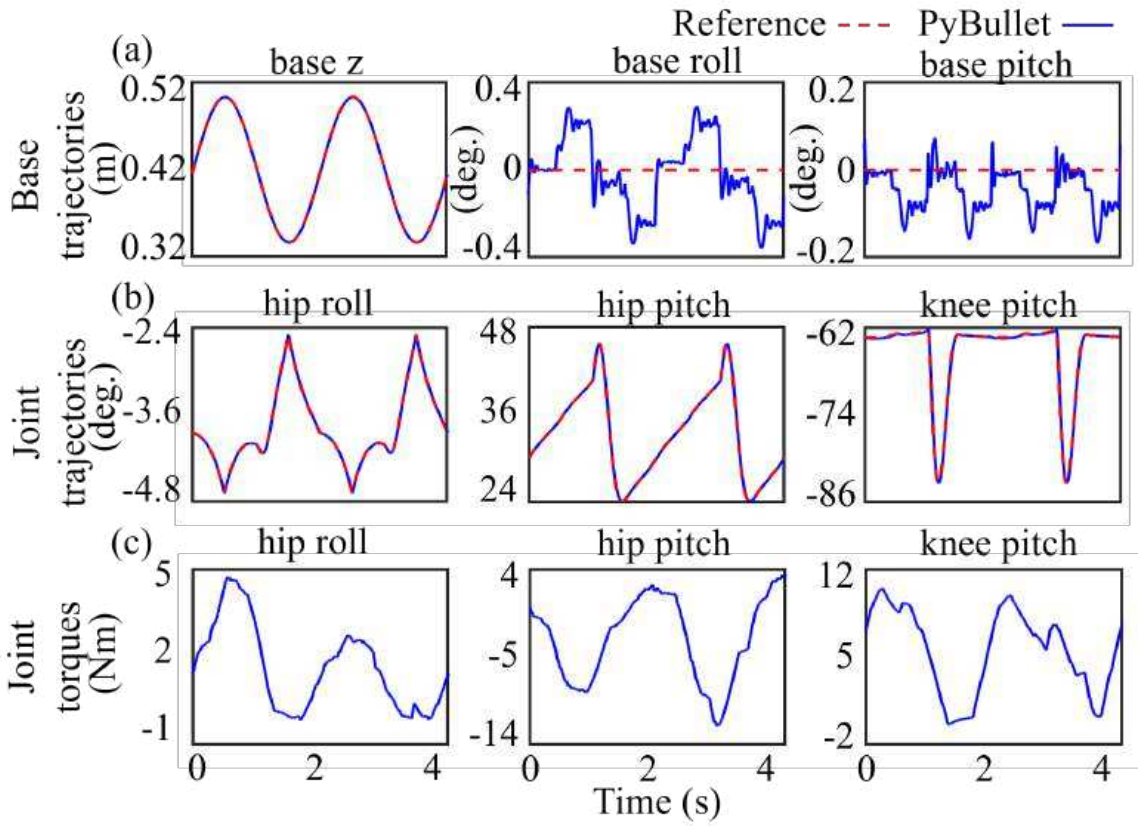

Figure 3-9: PyBullet simulation results for the robot's base and front-right leg under Case 2.

3.6.6 Robustness Validation under Surface Motion Uncertainty

To evaluate the inherent robustness of the proposed planner in the presence of moderate levels of uncertain DRS motions as specified in Section 3.6.2, we test the planner under Cases 3 and 4, with results respectively presented in Figs. 3-10 and 3-11.

In both simulations and experiments, the robot walking is stable, as indicated by the trajectory tracking accuracy in subplots (a) and (b) of Figs. 3-10 and 3-11 as well as the experiment video. Moreover, subplots (c) of Figs. 3-10 and 3-11 confirm that the joint torque limits are met in both simulations and experiments.

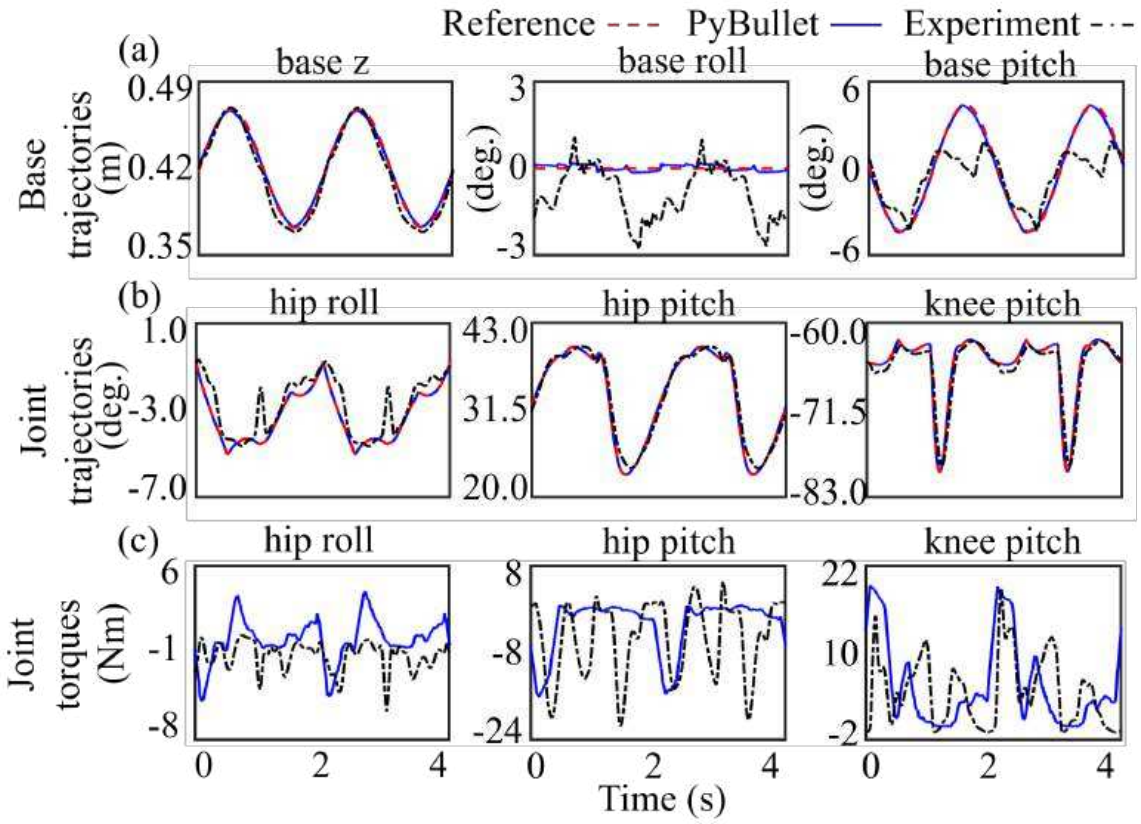

Figure 3-10: Hardware experiment and PyBullet simulation results for the robot's base and front-right leg under Case 3.

Yet, the torque profiles of the front-right leg's three joints display notable discrepancies between PyBullet and experiment results, possibly due to the differences between the simulated and actual robot dynamics as well as the different inherent meanings of their effective PD gains.

Also, the experiment video shows that the robot experiences relatively notable rebounding and slipping at contact-switching events when a rear leg lands on the surface. This violation of the planned contact sequence is directly due to the temporary loss of contact force feasibility, and could be mitigated through improved controller design as discussed in Section 3.7.

The robot controller utilized in this chapter is similar to Chapter 2, which is highly model-dependent and performs poorly under overly large uncertainty. We observed failure for levels of uncertainties exceeding those in Cases 3 or 4.

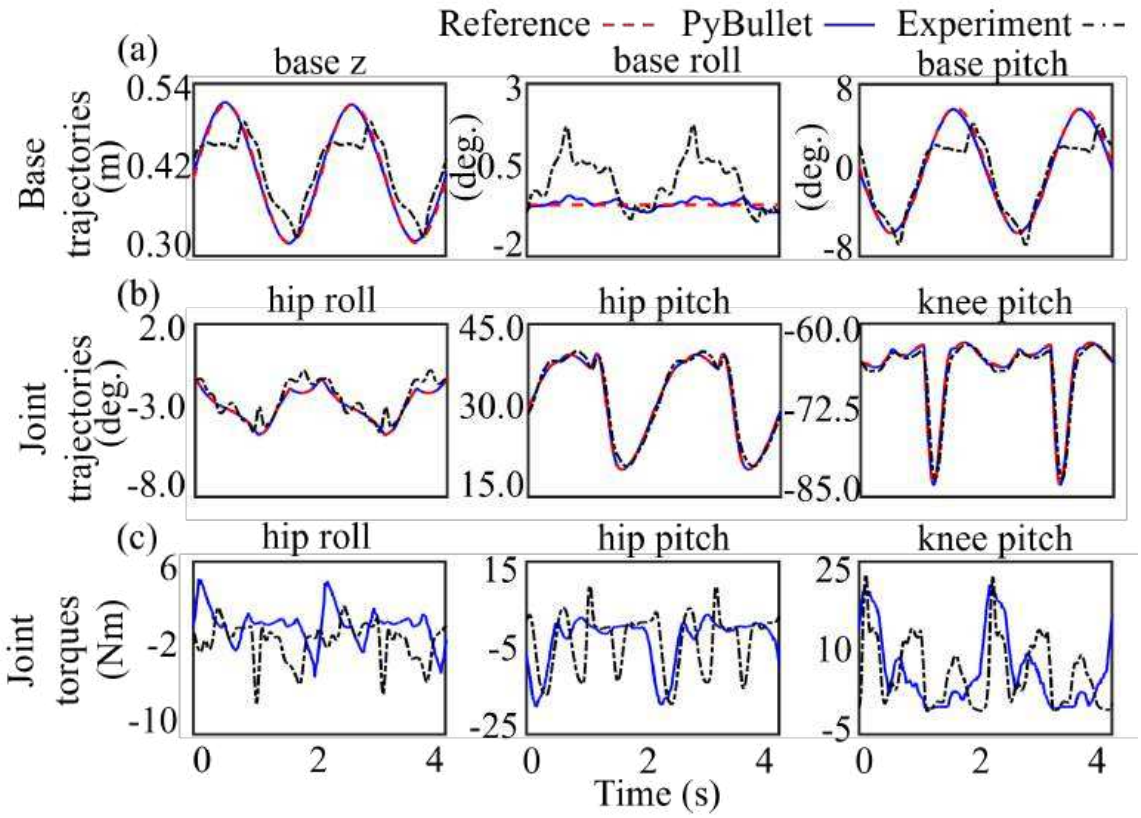

Figure 3-11: Hardware experiment and PyBullet simulation results for the robot's base and front-right leg under Case 4.

3.7 Discussion

This chapter has introduced a reduced-order dynamic model of a legged robot that walks on DRS, by analytically extending the classical LIP model from stationary surfaces [21] to DRS (e.g., ships). The resulting DRS-LIP model in Eq. (3.8) is a linear, second-order differential equation, similar to the classical LIP. However, the DRS-LIP is explicitly time-varying whereas the classical LIP is time-invariant. This fundamental difference is due to the time-varying movement of the surface at the surface-foot contact points. This chapter also investigates the stability of the DRS-LIP based on the Floquet theory. Similar to the classical LIP that describes stationary-surface locomotion [21], the DRS-LIP is unstable under the usual movement range of real-world DRS such as vessels [153].

The DRS-LIP is valid under the assumption (A1) that the actual robot's rate of whole-body angular momentum about the CoM is negligible. To relax this assumption, the point

mass of the proposed DRS-LIP could be augmented with a flywheel [25, 45] to account for the nonzero rate of angular momentum. Moreover, the DRS-LIP can be generalized from a constant CoM height (as enforced by assumption (A2)) to a varying height by integrating with the variable-height LIP for stationary surfaces [91].

In this chapter, we also derive the approximate analytical solution of the DRS-LIP for vertical, sinusoidal surface motions. Its sufficient accuracy and improved computational efficiency compared with numerical solutions are confirmed through MATLAB simulations (Fig. 3-6 and Table 3.1). Although the proposed reduced-order model in Eq. (3.7) does not assume a specific form of surface motion, the proposed analytical solution is derived for vertical and sinusoidal surface motions, which are typical for real-world ship motions in regular sea waves [159, 153]. To address surface motions that are vertical and nonperiodic with their time profiles pieced together by periods of different sinusoidal waves, which cover a wide range of DRS motions [153], the proposed analytical solution could be extended by: (i) forming the individual analytical solutions for those different periods based on the proposed solution derivation method and then (ii) piecing them together to form the needed overall solution. Moreover, if the vertical nonperiodic surface motion comprises periods of general periodic functions instead of sinusoidal waves, we could potentially use the Floquet theory [166] to derive the analytical solution by numerically precomputing the fundamental matrix of the reduced-order model and then constructing the analytical solution using the fundamental matrix. Our future work will also tackle the modeling and planning problem for legged locomotion under general surface motions that contain horizontal movements [172].

To highlight the practical usefulness of the analytical results, the DRS-LIP model and its solution have been used as a basis to synthesize a two-layer walking pattern generator that efficiently produces desired, physically feasible motions for quadrupedal DRS walking. The feasibility of the planned motion is validated by using our previous tracking controller [20] to command a quadrupedal robot to follow the planned motion during DRS walking. As discussed in Section 3.6.1, simulation and experiment results indicate the reasonable feasibility and robustness of the proposed planner under different gait parameters and surface motions (Figs. 3-9-3-11). Furthermore, validations through hardware experiments demonstrated its robustness to uncertainties in DRS motions. To mitigate the temporary violation of the

planned gait sequence observed in experiments, which is partly induced by the discrepancies between the DRS-LIP and the actual robot dynamics, the planned motion could be tracked by an optimization-based controller [1] that explicitly ensures physical feasibility.

With the peak absolute acceleration of the surface-foot contact points at around 1 m/s^2, the quadrupedal walking speed relative to the rocking treadmill is about 5 cm/s in the hardware experiments, which, to our best knowledge, is the fastest quadruped walking speed on a vigorously rotating surface for hardware experiments [20]. To achieve faster walking despite the inevitably higher inaccuracy of the reduced-order model induced by the higher nonlinearity of actual robot dynamics, we can augment the controller described in Section 3.6.1, which does not reason about the feasibility of necessary constraints (e.g., ground contact forces), with an optimization-based controller [173, 24] that explicitly guarantee the feasibility for actual walking. Furthermore, while we consider a constant average walking speed (relative to the dynamic surface) in the validation of the proposed walking pattern generator, the proposed planner can be readily extended from constant to variable speed walking, because the proposed reduced-order model and its analytical solution are valid in describing the robot dynamics within any walking cycle of constant or variable speed walking.

3.8 Conclusion

This chapter has presented an analytically tractable and computationally efficient reduced-order robot dynamics model, the approximate analytical solution of the model, and a real-time motion generator for legged locomotion on a dynamic rigid surface (DRS). The proposed model was derived by theoretically extending the classical linear inverted pendulum (LIP) model from a stationary surface to DRS, and describes the essential time-varying robot dynamics associated with DRS walking, which is fundamentally different from the classical time-invariant LIP model. The analytical solution of the extended LIP model was obtained based on the conversion of the model into the well-studied Mathieu's equation. Exploiting these analytical results, a real-time walking pattern generator was developed to efficiently plan feasible robot motions for quadrupedal walking on a vertically oscillating

surface. Simulation results revealed the continuous-phase stability property of the proposed time-varying LIP model, and demonstrated the efficiency and accuracy of the analytical solution under a common range of real-world DRS movement. Finally, both 3-D realistic PyBullet simulations and hardware experiments on a physical Laikago quadrupedal robot confirmed the computational efficiency, physical feasibility, and inherent robustness of the proposed framework under various gait parameters and surface motions.

Chapter 4

HT-LIP Model based Robust Control of Quadrupedal Robot Locomotion under Unknown Vertical Ground Motion

Our studies on quadruped control reported in Chapters 2 and 3 have primarily focused on provable stabilization [20] and real-time planning [34, 174, 175, 176] for locomotion on a vertically moving DRS with rigid actuation or large inertia, such as vessels in sea waves that exhibit periodic vertical motion. Yet, those studies assume an accurately known surface motion profile, and the controller design presented in Chapter 3 also assumes that the surface motion profile is sinusoidal. Thus, it remains an open question to stabilize legged locomotion on rigidly actuated or heavyweight dynamic surfaces with general or uncertain vertical motions. Building on the success of employing reduced-order models in real-time planning as demonstrated in Chapter 3, this chapter focuses on creating a reduced-order model-based control framework to ensure stability and robustness for quadrupedal locomotion on DRS with unknown vertical motions.

This chapter introduces a hybrid time-varying linear inverted pendulum (termed as "HT-LIP") model that describes the essential robot dynamics during locomotion on DRS with general vertical motion, derives the stability conditions for the model, and develops a quadratic programming (QP)-based robust footstep planning method. The HT-LIP model-based footstep planner is utilized to design a robust control framework for underactuated

quadrupedal robot locomotion on DRS with unknown vertical motion. The proposed control framework is thoroughly validated through hardware experiments on a quadruped robot under various uncertainties.

The contents of this chapter are from a published peer-reviewed conference paper and a journal article in preparation.

- Iqbal, A., Veer, S. and Gu, Y., 2023. Asymptotic stabilization of aperiodic trajectories of a hybrid-linear inverted pendulum walking on a vertically moving surface. In Proc. of *American Control Conference, pp. 3030-3035*.
- Iqbal, A., Veer, S., Niezrecki, C. and Gu, Y., 2023. HT-LIP Model based Robust Control of Quadrupedal Robot Locomotion under Unknown Vertical Ground Motion. *In preparation*.

4.1 Introduction

4.1.1 Related Work

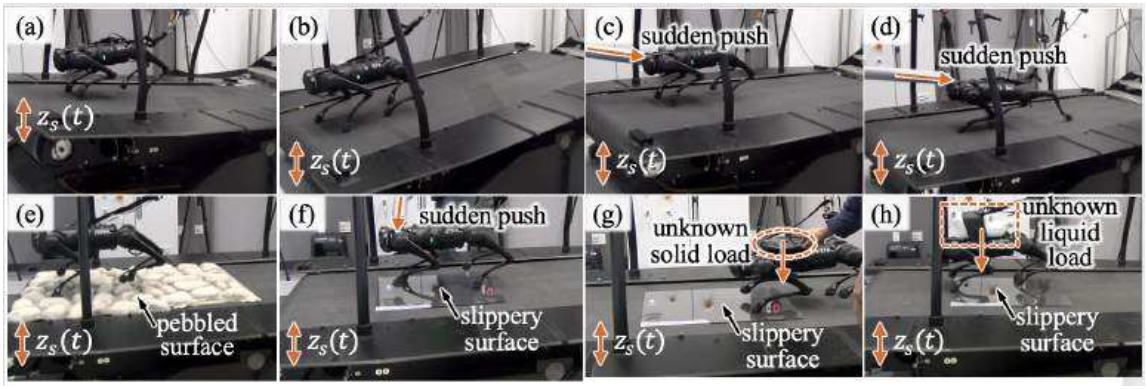

Figure 4-1: Snapshots of experiments. All experiments are under an unknown and aperiodic vertical surface motion $z_s(t)$ as shown in (a) and (b). The robot also experiences additional unknown disturbances, which include: (c) sudden pushes that result in (d) an irregular robot posture just after a push; (e) rocky surface with a peak height of 10 cm; (f) smooth glass surface; (g) solid load (36% of the robot's mass); and (h) liquid load (32% of the robot's mass).

Due to the prevalence of uncertainties in real-world environments, robustness is a crucial performance measure of legged robot control. Various control approaches [1, 177, 178, 103]

have achieved remarkably robust locomotion in a wide variety of unstructured, *static* environments (e.g., sand, grass, hiking trails, and creeks). Yet, since the previous approaches typically assume a static ground, they may not be effective for DRS, which is a rigid surface moving in the inertial frame and can persistently and continuously perturb the robot dynamics.

To enable robust locomotion, reduced-order models can be used to serve as the basis of motion generators to efficiently plan desired trajectories, enabling quick reactions to disturbances and uncertainties. To achieve real-time generation of the desired footsteps for robust locomotion, the classical LIP model has been expanded to capture the hybrid dynamical behaviors of legged locomotion on a stationary surface [179, 27, 180, 155], which include continuous leg-swinging dynamics and discrete foot-switching behaviors. Using the stability theory of hybrid linear time-invariant systems, the asymptotic stability conditions for the hybrid LIP (H-LIP) model under a discrete-time footstep controller have been constructed to enable robust locomotion under external pushes [179, 27]. However, the H-LIP model may not be accurate under a significant DRS motion because it does not explicitly consider the time-varying movement of the foot-surface contact region. Although our studies on quadrupedal locomotion reported in Chapter 3 have analytically extended the classical continuous-time LIP model [21] from static to dynamic surfaces [174, 181, 34] under *accurately known*, *periodic* DRS motions, they do not explicitly consider the hybrid robot dynamics and cannot directly address *unknown* and *general* surface motions.

4.1.2 Contributions

This chapter presents a reduced-order model based control approach that achieves provably robust stabilization of quadrupedal locomotion on a DRS with aperiodic and unknown vertical motions. This chapter makes the following new, substantial contributions:

(a) Derivation of the HT-LIP model that explicitly considers the time-varying surface motion and the hybrid robot behaviors.

(b) Development of generalized stability conditions to stabilize the HT-LIP model locomotion on DRS under unknown vertical DRS motions.

(c) Formulation of a robust footstep controller as a computationally efficient quadratic program (QP) that enforces stability conditions.

(d) Derivation of a hierarchical control approach that incorporates the proposed QP.

(e) Stability analysis of the full-order model under the proposed controller.

(f) Experimental validation under various uncertainties (Fig. 4-1).

4.2 Stabilization of a Hybrid Time-Varying LIP

This section introduces a reduced-order model that captures the essential hybrid robot dynamics associated with quadrupedal trotting on a DRS with a general vertical motion, along with its stabilizing control law.

4.2.1 Open-Loop Reduced-Order Model

To derive the proposed reduced-order model, we extend the classical H-LIP model [179] from static to DRS by combining it with our previous continuous-phase LIP model derived for moving surfaces. Due to the surface movement, the resulting model, as illustrated in Fig. 4-2, is a hybrid, *time-varying* LIP model, which we call "HT-LIP" and is fundamentally different from the *time-invariant* H-LIP model.

This section introduces the derivation of reduced-order model based on the standard hybrid, linear inverted pendulum model (H-LIP) [179]. We use the H-LIP model as the basis of our model derivation because it is computationally efficient, analytically tractable, and relatively accurately captures the essential hybrid dynamics of legged locomotion that include continuous leg-swinging behaviors and discrete foot-switching events. The resulting model, as illustrated in Fig. 4-2, is explicitly time-varying, due to the surface movement, which is fundamentally different from the time-invariant H-LIP model. We term the proposed hybrid, time-varying LIP model as HT-LIP.

(1) Model assumption

The proposed model derivation considers the following simplifying assumptions in addition to the assumption (A1):

(A11) The absolute vertical acceleration of DRS is bounded by the magnitude of gravitational acceleration and is locally Lipschitz in time.

(A12) The desired duration of the continuous phase for the HT-LIP stepping is bounded for all walking steps.

(A13) The foot switching events occur at fixed time instants.

The assumption (A1) helps avoid an overly stretched knee joint that can induce kinematic singularity and also ensures the linearity of the inverted pendulum model [21]. The acceleration bounds in the assumption (A11) hold for common real-world dynamic platforms such as moving vessels and aircraft, thanks to the large magnitude of the bounds. The Lipschitz continuity in time arises from the fact that the acceleration of real-world DRS is continuous, has finite magnitude, and does not change abruptly for all time [153, 182]. The assumption (A12) is reasonable as it ensures a finite duration for each continuous phase during HT-LIP walking and prevents Zeno behavior [183]. Additionally, the assumption (A13) simplifies the stability analysis [184, 179].

(2) Continuous phases

Under the assumption (A1), the continuous-phase dynamics of a 3-D inverted pendulum model along the x- and y-axes of the world frame are linear and share the same form, as explained in our previous work on continuous-time LIP modeling for DRS [109]. Without loss of generality and for brevity, the subsequent analysis considers the HT-LIP model in the x-direction (see Fig. 4-2).

We use $\ddot{z}_s(t)$, g, and z_0 to respectively denote the vertical acceleration of the support point S, the magnitude of gravitational acceleration, and the CoM height above S. Here, the time argument of the surface acceleration is kept in its notation (i.e., $\ddot{z}_s(t)$) to highlight its explicit time dependence.

Denoting the horizontal CoM position relative to point S as x, we express the continuous-phase equation of motion for the HT-LIP in the x-direction as the following continuous-time, time-varying, linear, homogeneous system:

$$\ddot{x} = \frac{\ddot{z}_s(t) + g}{z_0} x. \tag{4.1}$$

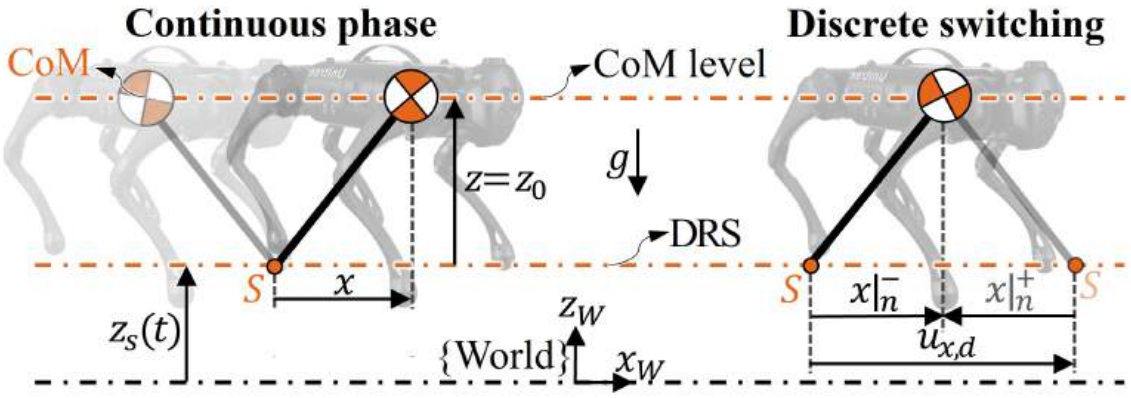

Figure 4-2: An illustration of the proposed HT-LIP model in the sagittal plane. The model describes the time-varying dynamics of the point mass (located at the CoM) under the vertical DRS displacement $z_s(t)$. It also captures the hybrid nature of legged locomotion, including both the continuous foot-swinging phase and the discrete foot-switching behavior.

(3) Discrete foot switching

Besides continuous dynamics, the proposed HT-LIP also considers the discrete foot-landing event when the stance and support feet switch roles. We use τ_n to denote the n^{th} switching instant with $n \in \mathbb{N}$. Further, we denote the time instant just before and after the n^{th} switching instant as τ_n^- and τ_n^+, respectively. For notational brevity, we introduce $\star|_n^- := \star(\tau_n^-)$ and $\star|_n^+ := \star(\tau_n^+)$.

At the switching timing, the location of the support point S on DRS is reset, resulting in a sudden jump in the relative CoM position x. As illustrated in Fig. 4-2, the relative CoM position just after the switching, $x|_n^+$, is given by:

$$x|_n^+ = x|_n^- - u_{x,d}, \tag{4.2}$$

where $u_{x,d}$ is the new support-foot position relative to the previous one in the x-direction.

The CoM velocity stays continuous at the switching instant, that is, $\dot{x}|_n^+ = \dot{x}|_n^-$, because the angular momentum of the CoM about the contact point S is conserved and the CoM height remains constant above S within continuous phases (i.e., assumption (A1)) [27]. Combining the continuous-phase dynamics in (4.1) and the discrete-time jump in (4.2), we obtain the state-space representation of the proposed HT-LIP model as:

$$\begin{cases} \dot{\mathbf{X}} = \boldsymbol{\alpha}(t)\mathbf{X} & \text{if } t \neq \tau_n^-; \\ \mathbf{X}(\tau_n^+) = \mathbf{X}(\tau_n^-) + \boldsymbol{\beta} u_{x,d} & \text{if } t = \tau_n^-, \end{cases} \tag{4.3}$$

where $\mathbf{X} := [x,\ \dot{x}]^T$ and $\boldsymbol{\beta} := [-1,\ 0]^T$. The matrix $\boldsymbol{\alpha}(t)$ is defined as $\boldsymbol{\alpha}(t) := \begin{bmatrix} 0 & 1 \\ f(t) & 0 \end{bmatrix}$ with $f(t) := \frac{\ddot{z}_s(t)+g}{z_0}$. Similar to $z_s(t)$, we keep the time argument t in the notation of $f(t)$ to highlight its explicit time dependence.

(4) Open-loop step-to-step (S2S) model

The S2S model of the HT-LIP compactly describes the hybrid evolution of the HT-LIP during a gait cycle, which is used to construct the proposed stability conditions of the HT-LIP later.

Integrating the continuous dynamics and iterating the discrete jump map based on (4.3) yields the S2S model as:

$$\mathbf{X}|_{n+1}^- = \boldsymbol{\Phi}(f(t); \tau_{n+1}^-, \tau_n^+)(\mathbf{X}|_n^- + \boldsymbol{\beta} u_{x,d}), \tag{4.4}$$

where $\boldsymbol{\Phi}(f(t); \tau_{n+1}^-, \tau_n^+) := \int_{\tau_n^+}^{\tau_{n+1}^-} e^{\boldsymbol{\alpha}}(t)dt$ is the state-transition matrix of the n^{th} continuous phase from τ_n^+ to τ_{n+1}^-.

4.2.2 Discrete Footstep Control for HT-LIP

While the continuous-time portion of the HT-LIP model is unstable [34] and uncontrolled, the discrete-time footstep behavior is directly commanded by the foot displacement $u_{x,d}$. Thus, we design a discrete-time footstep control law based on the HT-LIP model that aims to asymptotically stabilize the desired state trajectory, denoted as $\mathbf{X}_r(t)$, i.e., to drive the state trajectory $\mathbf{X}(t)$ to track the desired trajectory $\mathbf{X}_r(t)$ as time goes to infinity.

The tracking error is defined as $\mathbf{e} := \mathbf{X} - \mathbf{X}_r =: [e,\ \dot{e}]^T$, and x_r and $\dot{x}_r$ represent the elements of $\mathbf{X}_r$, i.e., $\mathbf{X}_r = [x_r, \dot{x}_r]^T$. By incorporating the error $\mathbf{e}$, the discrete HT-LIP stepping controller $u_{x,d}$ at the switching instant τ_n^- is designed as:

$$u_{x,d} = u_{x,r} + \mathbf{K}\mathbf{e}|_n^-. \tag{4.5}$$

Here $u_{x,r} := x_r|_n^- - x_r|_n^+$ is the desired foot-landing position of the desired trajectory $\mathbf{X}_r(t)$, and $\mathbf{K} := [k_1,\ k_2]$ is the feedback gain to be designed later for asymptotic stabilization.

From the feedback control law (4.5) and the open-loop S2S dynamics in (4.4), the closed-loop S2S error dynamics become:

$$\mathbf{e}|_{n+1}^- = \mathbf{A}_{d,n}\mathbf{e}|_n^-, \tag{4.6}$$

where $\mathbf{A}_{d,n}$ is the S2S error state-transition matrix and is defined as $\mathbf{A}_{d,n} := \boldsymbol{\Phi}(f(t); \tau_{n+1}^-, \tau_n^+)(\mathbf{I} + \boldsymbol{\beta}\mathbf{K})$ with $\mathbf{I}$ an identity matrix with an appropriate dimension.

4.3 HT-LIP Based Control Framework

This section presents the overall structure of the proposed hierarchical control framework. The framework aims to achieve robust quadrupedal trotting on DRS with unknown vertical motions.

Ensuring robust quadrupedal trotting during vertical DRS motions poses a challenge due to the complex robot dynamics. One effective way to realize locomotion robustness is to plan the desired, physically feasible footstep locations in real-time [179, 27]. Yet, the computational cost of employing the full-order robot model can be excessive for online planning, given the hybrid, nonlinear, time-varying, and high-dimensional nature of the actual robot dynamics.

Further, underactuation in quadrupedal trotting complicates the control framework design. With 13 DoFs and 12 independently actuated joints, a typical quadrupedal robot (e.g., Unitree's Go1) has one degree of underactuation during trotting. This leads to two-dimensional unactuated dynamics, which can be approximated by the proposed HT-LIP model in (4.3). Underactuation is challenging because (a) while the directly actuated portion of the actual robot dynamics can be well regulated, the unactuated subsystem may not be

directly altered by joint torque commands [27] and (b) the HT-LIP model is unstable during continuous phases under real-world DRS motions (e.g., ship motions in sea waves) [174].

To achieve robust locomotion, the proposed framework adopts the classical hierarchical structure of legged robot control approaches and contains three layers. The key novelty of the framework lies in its higher-layer footstep planner.

The subsequent part of this section describes the proposed control framework structure and presents essential details of each layer of the framework.

4.3.1 Framework Structure

This section presents the overall structure of the proposed hierarchical control framework. The proposed hierarchical control approach ensures robust trotting motions of the actual robot through real-time footstep planning and full-body control, even in the presence of uncertain conditions such as unknown DRS motions. The hierarchical control framework comprises three essential layers as illustrated in Fig. 4-3.

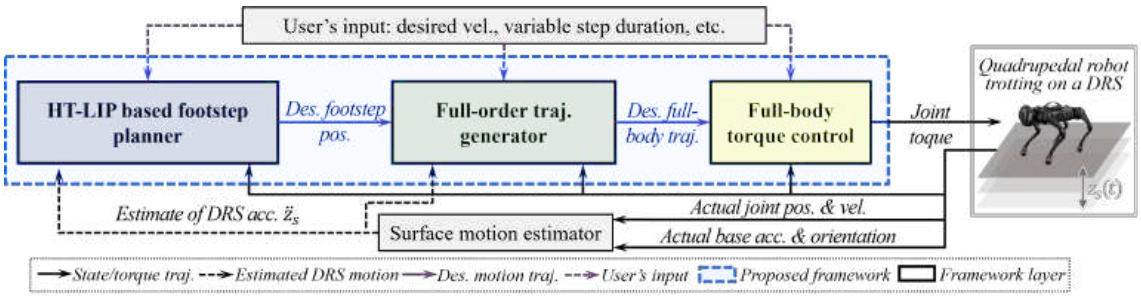

Figure 4-3: Illustration of the proposed hierarchical control framework. The higher layer generates the desired footstep locations. The middle layer employs a full-order model to plan kinematically feasible desired trajectories. The lower-layer controller tracks the desired full-body trajectories.

(1) Higher-layer footstep planning

To effectively reject uncertainties for ensuring robust locomotion, the objective of the higher layer is to efficiently generate the desired, physically feasible footstep positions in real-time that provably stabilize the proposed HT-LIP model. The key to realizing this objective is

the utilization of the proposed HT-LIP model as well as its provably stabilizing footstep controller formulated as an efficient quadratic program (QP).

Representing robot dynamics with the HT-LIP is reasonably accurate for guaranteeing the planner's feasibility since today's legged robots typically have heavy trunks and lightweight limbs [174]. Meanwhile, the linearity and low dimension of the HT-LIP contribute to the computational efficiency of the planner for real-time planning.

Further, to guarantee the stability of the hybrid, time-varying, nonlinear, and under-actuated robot dynamics, we use the proposed HT-LIP footstep controller to indirectly stabilize the unactuated dynamics by provably stabilizing the HT-LIP model. We introduce the stability condition for the footstep controller in Section 4.3.2. To enhance planning efficiency, we construct the controller as a QP as presented in Section 4.3.2.

(2) Middle-layer full-body trajectory generation

Based on the robot's full-order kinematics model, the middle layer translates the output from the higher layer (i.e., the desired footstep location and CoM trajectories) into the desired full-body trajectories while agreeing with assumptions (A1) and (A11)-(A13) underlying the HT-LIP model. This helps ensure the feasibility and efficiency of the trajectory generation and reduces the discrepancy between the actual robot dynamics and the HT-LIP. The details of the middle-layer planner are presented in Section 4.3.3.

(3) Lower-layer full-body control

Considering its high performance in ensuring gait feasibility and tracking performance, the lower layer adopts the existing quadruped controller [1] that outputs the joint torque to track the desired full-body trajectories based on a single rigid body model. The details of the lower-layer full-body control are presented in Section 4.3.4.

4.3.2 Higher Layer: HT-LIP Based Footstep Planning

This section introduces the formulation of the foot-stepping controller design that, even in the presence of uncertain DRS motions, stabilizes a trotting robot's unactuated dynamics

based on the HT-LIP model.

(1) Stability Conditions under Unknown DRS Motions

The design of the proposed higher-layer footstep planner begins with the construction of the asymptotic stability condition of the HT-LIP model under unknown DRS motions.

(a) Supreme model of HT-LIP

The proposed asymptotic stability condition is built on a supreme model of the S2S error dynamics in (4.6), which is derived next.

By definition, the function $f(t)$ is both positive and bounded for $t \in \mathbb{R}^+$ and locally Lipschitz under the assumption (A1). We use $\overline{f}_n$ to represent any positive constant parameter no less than the supremum of $f(t)$ over $t \in (\tau_n, \tau_{n+1}]$ (i.e., $\overline{f}_n$ should satisfy $\overline{f}_n \geq \sup f(t)$ on $t \in (\tau_n, \tau_{n+1}]$.

Since the continuous-phase error system is $\ddot{e} = f_n e$, we define its supreme model as

$$\ddot{\bar{e}} = \overline{f}_n \bar{e}, \tag{4.7}$$

where $\bar{e}$ is the solution of this model. Because the supremum model is linear and time-invariant, its state-transition matrix, denoted as $\overline{\mathbf{\Phi}}$, satisfies $\overline{\mathbf{\Phi}}(\overline{f}_n; \tau_{n+1}^-, \tau_n^+) = \overline{\mathbf{\Phi}}(\overline{f}_n; \Delta\tau_{n+1}, 0)$, where $\Delta\tau_{n+1} := \tau_{n+1}^- - \tau_n^+$ denotes the duration of the n^{th} continuous phase. Accordingly, the S2S state-transition matrix of the supreme model is defined as $\overline{\mathbf{A}}_{d,n} := \overline{\mathbf{\Phi}}(\overline{f}_n; \Delta\tau_{n+1}, 0)(\mathbf{I} + \boldsymbol{\beta}\mathbf{K})$.

(b) Asymptotic stability condition on S2S dynamics

We first introduce the sufficient condition for the asymptotic stability of the closed-loop S2S error model in (4.6).

(i) Lyapunov function candidate V: We consider a Lyapunov function candidate defined as $V(\mathbf{e}) := \frac{1}{2}\|\mathbf{e}\|^2$. According to the existing stability theory [185] of general discrete-time systems that include the S2S error dynamics, the error dynamics are asymptotically stable if: (i) $V(\mathbf{e})$ satisfies the positive definiteness and boundedness conditions mentioned in [185] and (ii) V strictly decreases as n increases. It can be readily proven that condition (i) is met for the selected Lyapunov function candidate V. The rest of this section shows that V meets condition (ii) if the stability condition in Theorem 3 holds.

(ii) Boundedness of error state norm: To prove the sufficient stability condition in Theorem 3, we first establish the boundedness of the norm of the error state at the $(n+1)^{\text{th}}$ switching instant, i.e., $\left\|\mathbf{e}|_{n+1}^{-}\right\|$, in terms of $\left\|\mathbf{e}|_{n}^{-}\right\|$, as summarized in Lemma 1.

Lemma 1 (Boundedness of error state norm). *Consider the assumptions (A11) and (A12) of this dissertation and the S2S error dynamics in* (4.6)*. Recall* $a_{d,n} := \|\overline{\mathbf{A}}_{d,n}\|_\infty := \|\overline{\mathbf{\Phi}}(\overline{f}_n;\Delta\tau_n,0)(\mathbf{I}+\boldsymbol{\beta}\mathbf{K})\|_\infty$*. Then, for all* $n \in \mathbb{N}$*, the following inequality holds*

$$\left\|\mathbf{e}|_{n+1}^{-}\right\| \leq a_{d,n}\left\|\mathbf{e}|_{n}^{-}\right\|. \tag{4.8}$$

Proof: We prove Lemma 1 by first establishing the bounds on the error state for the continuous phase on $t \in [\tau_n^+, \tau_{n+1}^-]$. Since the continuous-phase error model is $\ddot{e} = f(t)e$, which is time-varying, we establish the bounds on the continuous-phase error evolution by considering a supremum time-invariant system.

Recall that $f(t) := \frac{\ddot{z}_s(t)+g}{z_0}$ and that the positive variable $\overline{f}_n$ is no less than the supremum of $f(t)$ over $t \in (\tau_n, \tau_{n+1}]$ (i.e., $\overline{f}_n$ should satisfy $\overline{f}_n \geq \sup f(t)$ on $t \in (\tau_n, \tau_{n+1}]$.

As introduced in (4.7) of the previous section, the supreme model of the continuous-phase error system is given as:

$$\ddot{\overline{e}} = \overline{f}_n \overline{e}, \tag{4.9}$$

where $\overline{e}$ is the solution of (4.9).

Then, according to the results of the Strong Comparison Theorem in Sec. 2 of [186], the solutions e and $\overline{e}$ satisfy the following inequality for all $t \in (\tau_n, \tau_{n+1}]$

$$|e(t)| \leq |\overline{e}(t)|, \tag{4.10}$$

when they share identical initial conditions, i.e, $e(\tau_n^+) = \overline{e}(\tau_n^+)$. Note that the absolute value inequality in (4.10) holds because $f(t)$ is positive for all $t \in \mathbb{R}^+$.

Recall that $\overline{\mathbf{\Phi}}(\overline{f}_n;\tau_{n+1}^-,\tau_n^+)$ represents the state-transition matrix of the supremum time-invariant model in (4.9) on $t \in (\tau_n^+, \tau_{n+1}^-]$ and that $\overline{\mathbf{\Phi}}(\overline{f}_n;\tau_{n+1}^-,\tau_n^+) = \overline{\mathbf{\Phi}}(\overline{f}_n;\Delta\tau_{n+1},0)$.

By using (4.10), the error state $\mathbf{e}|_{n+1}^{+}$ is bounded by:

$$\left|\mathbf{e}|_{n+1}^{-}\right| \leq \left|\overline{\mathbf{\Phi}}(\overline{f}_n;\tau_{n+1}^{-},\tau_n^{+})\mathbf{e}|_n^{+}\right|. \tag{4.11}$$

Next, we apply the discrete switching map in (4.2) to the equation above and obtain:

$$\left|\mathbf{e}|_{n+1}^{-}\right| \leq \left|\overline{\mathbf{\Phi}}(\overline{f}_n;\Delta\tau_{n+1},0)(\mathbf{I}+\boldsymbol{\beta}\mathbf{K})\mathbf{e}|_n^{-}\right|. \tag{4.12}$$

Recalling

$$\overline{\mathbf{A}}_{d,n} := \overline{\mathbf{\Phi}}(\overline{f}_n;\Delta\tau_{n+1},0)(\mathbf{I}+\boldsymbol{\beta}\mathbf{K}) \tag{4.13}$$

and rewriting the right-hand side of the inequality in (4.12) yield:

$$\left|\mathbf{e}|_{n+1}^{-}\right| \leq |\overline{\mathbf{A}}_{d,n}\mathbf{e}|_n^{-}|. \tag{4.14}$$

With the sub-multiplicative property of $|\overline{\mathbf{A}}_{d,n}\mathbf{e}|_n^{-}|$, (4.14) becomes:

$$\left|\mathbf{e}|_{n+1}^{-}\right| \leq \left|\overline{\mathbf{A}}_{d,n}\right|\left|\mathbf{e}|_n^{-}\right|. \tag{4.15}$$

Adding an induced matrix norm $\|.\|$ to both sides of (4.15)and using the properties of $\|.\|$, we get:

$$\begin{aligned}\left\|\mathbf{e}|_{n+1}^{-}\right\| &\leq \left\|(|\overline{\mathbf{A}}_{d,n}||\mathbf{e}|_n^{-}|)\right\| \leq \left\|\overline{\mathbf{A}}_{d,n}\right\|_{\infty}\left\|\mathbf{e}|_n^{-}\right\| \\ &= a_{d,n}\left\|\mathbf{e}|_n^{-}\right\|,\end{aligned} \tag{4.16}$$

which completes the proof. ■

Next, we introduce the sufficient stability condition on S2S dynamics and present the proof based on Lemma 1

Theorem 3 (**Sufficient stability condition on S2S dynamics**). *Consider the assumptions (A11) and (A12). Define* $a_{d,n} := \|\overline{\mathbf{A}}_{d,n}\|_{\infty}$. *The closed-loop S2S error dynamics in* (4.6) *is globally asymptotically stable if* $a_{d,n} < 1$ *holds for all* $n \in \mathbb{N}$.

Proof: We analyze an upper bound of the change in the Lyapunov function $\Delta V(\mathbf{e}|_n^{-})$ across

two adjacent foot landings as:

$$\begin{aligned}\Delta V(\mathbf{e}|_n^-) &:= V(\mathbf{e}|_{n+1}^-) - V(\mathbf{e}|_n^-) = \frac{1}{2}\left\|\mathbf{e}|_{n+1}^-\right\|^2 - \frac{1}{2}\left\|\mathbf{e}|_n^-\right\|^2 \\ &\leq \frac{1}{2}a_{d,n}^2\left\|\mathbf{e}|_n^-\right\|^2 - \frac{1}{2}\left\|\mathbf{e}|_n^-\right\|^2 =: -\sigma_n\left\|\mathbf{e}|_n^-\right\|^2,\end{aligned} \tag{4.17}$$

where σ_n is defined as $\sigma_n := \frac{1}{2}(1-a_{d,n}^2)$. If the positive variable $a_{d,n}$ satisfies $a_{d,n} < 1$ for all $n \in \mathbb{N}$, then $-\sigma_n < 0$ holds on $n \in \mathbb{N}$. Accordingly, the Lyapunov function V meets the sufficient stability conditions for the discrete-time model.This completes the proof. ■

(3) Stability condition on footstep control

Based on Theorem 3, the following theorem provides the sufficient condition under which the footstep controller in (4.5) asymptotically stabilizes the HT-LIP model in (4.3).

Theorem 4 (**Sufficient stability condition on footstep control gain**). *Consider the assumptions (A11) and (A12). The feedback footstep controller gain* $\mathbf{K}$ *(i.e.,* k_1 *and* k_2*) guarantees the closed-loop asymptotic stability of the desired trajectory* $\mathbf{X}_r(t)$ *for the HT-LIP model if*

$$\begin{aligned}&\left|(1-k_1)\cosh(\xi_n)\right| + \left|\frac{\sinh(\xi_n)}{\sqrt{\overline{f}_n}} - k_2\cosh(\xi_n)\right| < 1 \text{ and} \\ &\left|(1-k_1)\sqrt{\overline{f}_n}\sinh(\xi_n)\right| + \left|\cosh(\xi_n) - k_2\sqrt{\overline{f}_n}\sinh(\xi_n)\right| < 1\end{aligned} \tag{4.18}$$

hold for any n^{th} *gait cycle (*$n \in \mathbb{N}$*). Here,* $\xi_n := \Delta\tau_n\sqrt{\overline{f}_n}$.

Proof: The rationale of the proof is to show if (4.18) is valid for all $n \in \mathbb{N}$ then the stability condition in Theorem 3 holds.

By definition, the state-transition matrix $\overline{\mathbf{\Phi}}(\overline{f}_n;\Delta\tau_n,0)$ for the state-space representation of the time-invariant supremum model $\ddot{\bar{e}} = \overline{f}_n\bar{e}$ is given as:

$$\begin{aligned}\overline{\mathbf{\Phi}}(\overline{f}_n;\Delta\tau_n,0) &= e^{\begin{bmatrix}0 & 1\\ \overline{f}_n & 0\end{bmatrix}\Delta\tau_n} =: \begin{bmatrix}\overline{\mathbf{\Phi}}_{11} & \overline{\mathbf{\Phi}}_{12}\\ \overline{\mathbf{\Phi}}_{21} & \overline{\mathbf{\Phi}}_{22}\end{bmatrix} \\ &=: \begin{bmatrix}\cosh(\xi_n) & \frac{\sinh(\xi_n)}{\sqrt{\overline{f}_n}}\\ \sqrt{\overline{f}_n}\sinh(\xi_n) & \cosh(\xi_n)\end{bmatrix}.\end{aligned} \tag{4.19}$$

Using the expressions of the state-transition matrix in (4.19) and those of $\boldsymbol{\beta}$ and $\mathbf{K}$, we can express $\overline{\mathbf{A}}_{d,n}$ as:

$$\overline{\mathbf{A}}_{d,n} = \begin{bmatrix} (1-k_1)\overline{\boldsymbol{\Phi}}_{11} & \overline{\boldsymbol{\Phi}}_{12} - k_2\overline{\boldsymbol{\Phi}}_{11} \\ (1-k_1)\overline{\boldsymbol{\Phi}}_{21} & \overline{\boldsymbol{\Phi}}_{22} - k_2\overline{\boldsymbol{\Phi}}_{21} \end{bmatrix}. \tag{4.20}$$

From (4.20) and the definition of the infinity norm of a matrix, we obtain:

$$\begin{aligned} a_{d,n} = \|\overline{\mathbf{A}}_{d,n}\|_\infty := \max(&|\overline{\boldsymbol{\Phi}}_{11}(1-k_1)| + |\overline{\boldsymbol{\Phi}}_{12} - \overline{\boldsymbol{\Phi}}_{11}k_2|, \\ &|\overline{\boldsymbol{\Phi}}_{21}(1-k_1)| + |\overline{\boldsymbol{\Phi}}_{22} - \overline{\boldsymbol{\Phi}}_{21}k_2|). \end{aligned} \tag{4.21}$$

If the footstep controller satisfies (4.18), then $|\overline{\boldsymbol{\Phi}}_{11}(1-k_1)| + |\overline{\boldsymbol{\Phi}}_{12} - \overline{\boldsymbol{\Phi}}_{11}k_2| < 1$ and $|\overline{\boldsymbol{\Phi}}_{21}(1-k_1)| + |\overline{\boldsymbol{\Phi}}_{22} - \overline{\boldsymbol{\Phi}}_{21}k_2| < 1$ hold for any $n \in \mathbb{N}$. Accordingly, $a_{d,n} = \|\overline{\mathbf{A}}_{d,n}\|_\infty < 1$ holds on $n \in \mathbb{N}$, meeting the stability condition in Theorem 3. ■

Remark 5 (Applicability of Theorems 3 and 4). *The stability conditions in Theorems 3 and 4 are valid for a variable continuous-phase duration and a general (periodic and aperiodic) vertical DRS motion. Also, applying these conditions does not require an accurate knowledge of the DRS motion but an upper bound of its acceleration.*

(2) Formulation of QP-based Footstep Control

To ensure online footstep planning, we formulate a computationally efficient QP that calculates the controller gain $\mathbf{K}$ in real-time, maximizes the error convergence rate, and enforces feasibility and stability conditions of the HT-LIP.

(a) Ensuring real-time update of control gain

Because the stability condition in Theorem 4 relies on the values of the system parameter ξ_n that can vary across different gait cycles, it is necessary to update the control gain $\mathbf{K}$ at least once per gait cycle in order to meet the stability condition. The variance of ξ_n across different gait cycles is due to changes in the gait cycle duration $\Delta\tau_n$ and the parameter $\overline{f}_n$. The varying value of $\Delta\tau_n$ across gait cycles can be induced by users or a high-level path planner, while that of $\overline{f}_n$ can be caused by the constantly changing DRS motion.

For timely mitigation of uncertainties in real-world applications, updating the planned footstep position every time step is necessary [27]. Although Theorem 4 ensures the system stability under the once-per-gait-cycle update of $\mathbf{K}$ and ξ_n, instead of an update every time

step, Theorem 4 can be readily extended to guarantee the stability even when $\mathbf{K}$ and ξ_n are updated every time step. This is essentially because the supremum system used to construct the stability conditions is time-invariant and accordingly its S2S state-ttransition matrix $\overline{\mathbf{A}}_{d,n}$ enjoys the associative property in terms of time t within each continuous phase.

(b) Ensuring fast convergence rate

Lemma 1 shows that for all $n \in \mathbb{N}$ we have $\left\| \mathbf{e}|_{n+1}^- \right\| \leq a_{d,n} \left\| \mathbf{e}|_n^- \right\|$. Thus, minimizing $a_{d,n}$ ensures a fast convergence rate of the error $\mathbf{e}$. Based on (4.21), this can be achieved by minimizing the sum of the squares of $|\overline{\mathbf{\Phi}}_{11}(1-k_1)| + |\overline{\mathbf{\Phi}}_{12} - \overline{\mathbf{\Phi}}_{11}k_2|$ and $|\overline{\mathbf{\Phi}}_{21}(1-k_1)| + |\overline{\mathbf{\Phi}}_{22} - \overline{\mathbf{\Phi}}_{21}k_2|$), which is used as the cost function $J(\mathbf{K})$:

$$J(\mathbf{K}) = \frac{1}{2}\mathbf{K}\mathbf{S}\mathbf{K}^T + \mathbf{K}\mathbf{c}. \tag{4.22}$$

Here $\mathbf{S}$ and $\mathbf{c}$ are respectively the Hessian matrix and gradient vector of the cost function $J(\mathbf{K})$ and are defined as:

$$\begin{aligned} \mathbf{S} &= \begin{bmatrix} 2(\overline{\mathbf{\Phi}}_{11}^2 + \overline{\mathbf{\Phi}}_{21}^2) & 0 \\ 0 & 2(\overline{\mathbf{\Phi}}_{11}^2 + \overline{\mathbf{\Phi}}_{21}^2) \end{bmatrix} \text{ and} \\ \mathbf{c} &= [-2(\overline{\mathbf{\Phi}}_{11}^2 + \overline{\mathbf{\Phi}}_{21}^2),\ -2(\overline{\mathbf{\Phi}}_{11}\overline{\mathbf{\Phi}}_{12} + \overline{\mathbf{\Phi}}_{21}\overline{\mathbf{\Phi}}_{22})]^T. \end{aligned} \tag{4.23}$$

(c) Enforcing stability conditions

The asymptotic stability condition of the HT-LIP model under the proposed footstep control law, given in (4.18), can be rewritten as:

$$\begin{bmatrix} -\overline{\mathbf{\Phi}}_{11} & -\overline{\mathbf{\Phi}}_{11} \\ \overline{\mathbf{\Phi}}_{11} & \overline{\mathbf{\Phi}}_{11} \\ -\overline{\mathbf{\Phi}}_{21} & -\overline{\mathbf{\Phi}}_{21} \\ \overline{\mathbf{\Phi}}_{21} & \overline{\mathbf{\Phi}}_{21} \end{bmatrix} \mathbf{K} < \begin{bmatrix} 1-\overline{\mathbf{\Phi}}_{11}-\overline{\mathbf{\Phi}}_{12} \\ 1+\overline{\mathbf{\Phi}}_{11}+\overline{\mathbf{\Phi}}_{12} \\ 1-\overline{\mathbf{\Phi}}_{21}-\overline{\mathbf{\Phi}}_{22} \\ 1+\overline{\mathbf{\Phi}}_{21}+\overline{\mathbf{\Phi}}_{22} \end{bmatrix}. \tag{4.24}$$

(d) Satisfying kinematic limits and ground-contact constraints

The physical feasibility of a planned footstep is guaranteed by respecting (i) the kinematic bounds on the trotting step length and (ii) the friction cone and unilateral ground-contact constraint. The kinematic limit of the step length $u_{x,d}$ can be expressed as $u_{x,d} \in [u_{min},\ u_{max}]$, where u_{max} and u_{min} are the maximum and minimum step lengths of the HT-LIP, respectively. Meanwhile, the step length should be set to respect the following friction cone and unilateral constraint at the foot-surface contact point: $u_{x,d} \in [-2\mu z_0, 2\mu z_0]$, where μ is the friction coefficient.

In summary, the stability condition and the feasibility constraints can be compactly expressed as:

$$\mathbf{E}\mathbf{K}^T < \mathbf{d} \tag{4.25}$$

with

$$\mathbf{E} := \begin{bmatrix} e & \dot{e} \\ -e & -\dot{e} \\ -\overline{\mathbf{\Phi}}_{11} & -\overline{\mathbf{\Phi}}_{11} \\ \overline{\mathbf{\Phi}}_{11} & \overline{\mathbf{\Phi}}_{11} \\ -\overline{\mathbf{\Phi}}_{21} & -\overline{\mathbf{\Phi}}_{21} \\ \overline{\mathbf{\Phi}}_{21} & \overline{\mathbf{\Phi}}_{21} \end{bmatrix} \text{ and } \mathbf{d} := \begin{bmatrix} l_{max} - u_{r,n} \\ -l_{min} + u_{r,n} \\ 1 - \overline{\mathbf{\Phi}}_{11} - \overline{\mathbf{\Phi}}_{12} \\ 1 + \overline{\mathbf{\Phi}}_{11} + \overline{\mathbf{\Phi}}_{12} \\ 1 - \overline{\mathbf{\Phi}}_{21} - \overline{\mathbf{\Phi}}_{22} \\ 1 + \overline{\mathbf{\Phi}}_{21} + \overline{\mathbf{\Phi}}_{22} \end{bmatrix}, \tag{4.26}$$

where the scalar, real constants l_{max} and l_{min} are defined as $l_{max} := \max(u_{max},\ \mu z_0)$ and $l_{min} := \min(u_{min},\ -\mu z_0)$.

With the cost function and constraints designed, the proposed QP that produces the footstep controller gain $\mathbf{K}$ is given in the following theorem.

Theorem 5 (**QP-based control gain optimization**). *The control gain* $\mathbf{K}$ *that maximizes the convergence rate, guarantees stability, and ensures feasibility for an HT-LIP model is given as a solution to the following QP problem:*

$$\begin{aligned} \min_{\mathbf{K}} \quad & J(\mathbf{K}) \\ \textit{subject to} \quad & \mathbf{E}\mathbf{K}^T < \mathbf{d}. \end{aligned} \tag{4.27}$$

Proof: Minimizing the cost function $J(\mathbf{K})$ leads to the minimization of variable $a_{d,n}$ while the physical feasibility and asymptotic stability of the closed-loop HT-LIP system are guaranteed by enforcing the constraint $\mathbf{E}\mathbf{K}^T < \mathbf{d}$. Hence, the optimal solution to the QP problem corresponds to the optimal convergence rate, feasibility, and stability. ■

Remark 6 (**Solution feasibility and optimality of the proposed QP**). *Note that the cost function in* (4.22) *is convex. Meanwhile, the feasibility and stability constraints of the QP in* (4.27) *are non-conflicting if the feasible region for the constraints* $\mathbf{E}\mathbf{K}^T < \mathbf{d}$ *remains non-empty. Accordingly, the solution feasibility and optimality for the QP problem in* (4.27) *is guaranteed. In practice, the non-emptiness of the feasible region can be numerically evaluated under the admissible range of system parameters* $\overline{f}$ *and* $\Delta\tau_n$.

Remark 7 (**Solving the QP in real-time**). *Solving the proposed QP requires the knowledge of the upper bound of $f(t)$ during any n^{th} gait cycle, as indicated by the stability condition in Theorem 4. Since the needed upper bound can be any upper bound of $f(t)$ during any n^{th} gait cycle, we can solve the proposed QP, in principle, by using a sufficiently large value of the upper bound $\overline{f}_n$ that is valid across any n^{th} gait cycles. Yet, using such a bound might be overly conservative, reducing locomotion robustness. Thus, we choose to estimate the upper bound of the surface acceleration in real-time and update $\overline{f}_n$ at every time step.*

To estimate the surface acceleration, one feasible way is to use forward kinematics to transform the acceleration reading of an on-board inertial measurement unit (IMU) from the IMU/robot frame to the world frame. Using the roughly estimated surface acceleration, we can then approximate an upper bound of the surface acceleration and update the values of $\overline{f}_n$, $a_{d,n}$, and $\overline{\mathbf{\Phi}}$.

(4) Higher-layer output

The output from the higher layer is: (i) desired footstep location $u_{x,d}$ in the forward direction and its lateral counterpart, denoted as $u_{y,d}$, and (ii) desired CoM position trajectories.

To obtain the reference step length $u_{x,r}$ and $u_{y,r}$ of the HT-LIP model, we utilize the HT-LIP model itself, its desired average velocity in the x- and y-directions provided by the user (denoted as $v_{x,d}$ and $v_{y,d}$), and the desired step duration $\Delta\tau_n$. Specifically, we compute $u_{x,r}$ and $u_{y,r}$ as:

$$u_{x,r} := v_{x,d}\Delta\tau_n \text{ and } u_{y,r} := v_{y,d}\Delta\tau_n. \tag{4.28}$$

When the HT-LIP model evolves exactly along its state trajectory $x_r(t)$, the model's footstep location along the x-direction is $u_{x,r}$. As explained in Section 4.2.2, to stabilize the HT-LIP model (for indirectly stabilizing the robot's unactuated dynamics), the proposed discrete footstep controller of the HT-LIP produces the actual footstep locations for the HT-LIP (i.e., $u_{x,d}$ and $u_{y,d}$), which serves as the desired footstep locations for the actual robot.

To account for the actual robot movement in the computation of the footstep lengths $u_{x,d}$

and $u_{y,d}$, we compute them based on the following modification of (4.5):

$$u_{x,d} = u_{x,r} + \mathbf{K}(\hat{\mathbf{X}}_b|_n^- - \mathbf{X}_r|_n^-) \text{ and } u_{y,d} = u_{y,r} + \mathbf{K}(\hat{\mathbf{Y}}_b|_n^- - \mathbf{Y}_r|_n^-), \tag{4.29}$$

where $\mathbf{K}$ is obtained based on (4.27). The vectors $\hat{\mathbf{X}}_b|_n^-$ and $\hat{\mathbf{Y}}_b|_n^-$ are respectively the estimated pre-impact state of the robot's base (e.g., trunk) along the x- and y-directions. We use the HT-LIP model and the current actual values of the base state at time t, i.e., $(\mathbf{Y}_b(t), \mathbf{X}_b(t))$, to compute the horizontal pre-impact state of the robot as: $\hat{\mathbf{X}}_b|_n^- = \overline{\mathbf{\Phi}}(\overline{f}_n; \tau_n - t, 0)\mathbf{X}_b(t)$ and $\hat{\mathbf{Y}}_b|_n^- = \overline{\mathbf{\Phi}}(\overline{f}_n; \tau_n - t, 0)\mathbf{Y}_b(t)$. Here $\overline{\mathbf{\Phi}}$ is given in (4.19).

Since $\mathbf{X}_r|_n^-$ is the exact solution to the hybrid model in (4.3), its value can be accurately computed. To reduce the computational load, we propose an approximate computation of $\mathbf{X}_r|_n^-$ by enforcing the positional symmetry of the HT-LIP model. The reference pre-impact position $x_r|_n^-$ is designed as $x_r|_n^- := 0.5 v_{x,d} \Delta\tau_n$ and reference pre-impact velocity $\dot{x}_r|_n^-$ is approximated as $\dot{x}_r|_n^- = v_{x,d}$. This is valid for real-world implementation because: (a) the step duration is small (typically 0.2 s to 0.3 s), resulting in minimal variation in $\dot{x}_r|_n^-$, and (b) any sufficiently small errors caused by the approximation of $\dot{x}_r|_n^-$ can be handled by the robust full-body controller.

The desired CoM position trajectories with respect to the CoP (i.e., $x_d(t)$ and $y_d(t)$) can then be computed based on the desired pre-impact state, the desired step length, and the HT-LIP model in (4.3).

4.3.3 Middle Layer: Full-Order Trajectory Generation

This section presents the full-order desired trajectory generation in real-time by combining the higher layer's footstep location data with the robot's kinematic model. This approach effectively connects the actual robot model with the HT-LIP model, which acts as a template for stabilizing the unactuated dynamics of the robot. Reliable tracking of desired trajectories generated by the middle layer ensures the stability of the underactuated robot dynamics.

The full-order desired trajectories $\mathbf{h}_d$ are chosen as the desired trajectories of the robot's base pose (i.e., position and orientation), denoted as $\mathbf{h}_{d,b}$, and swing feet position, denoted

as $\mathbf{h}_{d,sw}$; that is:

$$\mathbf{h}_d := [\mathbf{h}_{d,b}^T,\ \mathbf{h}_{d,sw}^T]^T. \tag{4.30}$$

The full-order current trajectories $\mathbf{h}_c$ of the robot are defined as:

$$\mathbf{h}_c := [\mathbf{h}_{c,b}^T,\ \mathbf{h}_{c,sw}^T]^T, \tag{4.31}$$

where $\mathbf{h}_{c,b}$ denotes the current base pose and $\mathbf{h}_{c,sw}$ denotes the current swing feet position trajectories.

Base trajectories

The vector of the robot's desired base trajectories is given as:

$$\mathbf{h}_{b,d} := [x_{d,b}, y_{d,b}, z_{d,b}, \phi_{d,b}, \theta_{d,b}, \psi_{d,b}]^T, \tag{4.32}$$

where $(x_{d,b},\ y_{d,b},\ z_{d,b})$ and $(\phi_{d,b},\ \theta_{d,b},\ \psi_{d,b})$ are the base position and orientation with respect to the world frame, respectively. In the middle layer, the base is represented by the CoM of the robot.

We express the desired horizontal base trajectories $x_{d,b}$ and $y_{d,b}$ as the sum of the nominal CoP location $(x_s,\ y_s)$ and the HT-LIP CoM desired trajectories $x_d(t)$ and $y_d(t)$:

$$x_{d,b} := x_s + x_d(t) \ \text{ and } \ y_{d,b} := y_s + y_d(t). \tag{4.33}$$

To respect assumption (A1), The base z-trajectory is designed to maintain a constant height relative to the support point of the HT-LIP. Further, the base yaw trajectory is planned based on the user-specified yaw rate $\omega_{b,d}$. Finally, for simplicity and without loss of generality, the base roll and pitch angles are set to zero to maintain a steady trunk posture. In summary, these desired trajectories are defined as:

$$z_{b,d} = z_0,\ \phi_{b,d} = 0,\ \theta_{b,d} = 0, \text{ and } \psi_{b,d} = \psi_0 + \omega_{b,d}\Delta t,$$

where ψ_0 is the base yaw at the beginning of a continuous phase and Δt is the time spent in

a continuous phase.

Swing feet trajectories

To generate the swing feet x- and y-trajectories, we utilize the higher-layer desired footstep length output in (4.29), post-switching swing foot initial position (i.e., $x_{f_i}|_n^+,\ y_{f_i}|_n^+$), and desired continuous-phase duration. The swing feet's z-height is set to be a user-specified kinematically feasible value for generating swing feet z-trajectories.

Without loss of generality, we use the quadrupedal trotting gait as an example to illustrate the design of the desired swing feet trajectories. Using $x_{f_i},\ y_{f_i}$, and z_{f_i} to denote the x-, y-, and z-trajectories of the swing foot i ($i \in \{1,\ 2\}$) with respect to S expressed in the world frame

$$\mathbf{h}_{d,sw} := [x_{d,f_1}, y_{d,f_1}, z_{d,f_1}, x_{d,f_2}, y_{d,f_2}, z_{d,f_2}]^T = [\mathbf{h}_{d,sw_1}^T, \mathbf{h}_{d,sw_2}^T]^T, \tag{4.34}$$

where. The swing foot trajectories are generated based on the real-time computation of desired step lengths in (4.29)) as:

$$\begin{aligned} x_{d,f_i}(s) &:= x_{f_i}|_n^+ + P_x(s)(u_x - x_{f_i}|_n^+), \\ y_{d,f_i}(s) &:= y_{f_i}|_n^+ + P_y(s)(u_y - y_{f_i}|_n^+), \\ z_{d,f_i}(s) &:= P_h(s) z_{f_i}|_n^+ + P_z(s) h_{sw_{max}}, \end{aligned} \tag{4.35}$$

where $x_{f_i}|_n^+,\ y_{f_i}|_n^+$, and $z_{f_i}|_n^+$ are the x-, y-, and z-coordinates of the initial position of the swing foot trajectory, $h_{sw_{max}}$ is the maximum swing foot height, and $P_x,\ P_y,\ P_z,$ and P_h are normalized Bézier interpolation functions of parameter $s := \frac{t}{\Delta\tau_n} \in [0, 1]$.

4.3.4 Lower Layer: Full-Body Torque Control

In quadrupedal trotting, underactuated dynamics exist when the number of degrees of freedom (DoF) exceeds the number of independently actuated joints. In our specific case, we have 13 DoF and 12 independently actuated joints, resulting in one degree of underactuation.

The objective of controller design is to reliably track desired full-order trajectories generated by the proposed underactuated HT-LIP based planner. Running the controller in real-time (500 Hz) requires a sufficiently accurate and relatively simple model.

For real-time control of quadrupeds, we utilize a single rigid body dynamic (SRBD) model, drawing inspiration from [51, 1]. Unlike the HT-LIP model, which considers the single point of the center of pressure, the SRBD model incorporates the influence of feet-ground interaction using the full-order kinematics model of the robot's legs on the evolution of the base pose dynamics. Despite being simpler than the full-order model, the SRBD model remains suitable for real-time control of quadruped with lightweight limbs [1, 51].

In the SRBD model-based control, we compute the optimal contact forces based on the actual contact state of the robot limbs and the desired base motion of the robot by solving a least square minimization problem. To ensure feasibility, we convert the least square minimization problem into a standard QP, where the feasibility of the contact forces is explicitly set as a constraint.

The computed desired contact forces are mapped to the actuated joints of the robot by utilizing the Jacobian of the contact points between the quadruped feet and walking surface based on the full-order kinematic model of the robot. In addition to controlling the contact forces of the stance foot for reliable base trajectory tracking, we compute the feedforward and feedback torque for swing foot trajectory tracking. In the following paragraph, we provide a brief description of the SRBD model-based control approach for quadruped locomotion on a dynamically moving surface.

The SRBD model approximating quadrupedal dynamics is expressed as [51]:

$$\mathbf{SF} = \mathbf{b}, \tag{4.36}$$

where

$$\mathbf{S} := \begin{bmatrix} \mathbf{I}_3 & \mathbf{I}_3 & \mathbf{I}_3 & \mathbf{I}_3 \\ \mathbf{P}_{1c_\times} & \mathbf{P}_{2c_\times} & \mathbf{P}_{3c_\times} & \mathbf{P}_{4c_\times} \end{bmatrix}, \ \mathbf{b} := \begin{bmatrix} m(\ddot{\mathbf{P}}_{wc} - \mathbf{g}) \\ \mathbf{I}_c \dot{\omega}_{wc} + \omega_{wc} \times \mathbf{I}_c \omega_{wc} \end{bmatrix}.$$

Here, $\mathbf{I}_3$ is the 3×3 identity matrix. The matrices $\mathbf{P}_{ic_\times}$ ($i \in \{1,2,3,4\}$) is the skew-symmetric matrix corresponding to the radius vector between the CoM of the robot and the foot i expressed in the world frame. The vector $\ddot{\mathbf{P}}_{wc}$ represents the acceleration of the CoM with respect to the world frame. The parameter m is the lumped scalar mass of the robot, $\mathbf{g} := [0,0,-g]^T$ represents the acceleration due to gravity in the world frame, $\mathbf{I}_c$ is

the centroidal inertia matrix for the lumped rigid body, ω_{wc} is the vector of the rigid body's angular velocity in the world frame, and $\mathbf{F} \in \mathbb{R}^{12\times 1}$ is the vector of contact forces expressed in the world frame.

Under the assumption (A2) of this dissertation, the nonlinear term $\omega_{wc} \times \mathbf{I}_c \omega_{wc}$ in the orientation dynamics can be ignored and the SRBD dynamics in (4.36) are linear.

To ensure reliable tracking of the base trajectory, we use PD feedback on base trajectory tracking error for computing the desired inertial wrench of the base $\mathbf{b}_d$ [51, 1] as:

$$\mathbf{b}_d := \begin{bmatrix} m\mathbf{I}_3 \\ \mathbf{I}_c \end{bmatrix} (\ddot{\mathbf{h}}_{d,b} + \begin{bmatrix} -\mathbf{g} \\ \mathbf{0}_3 \end{bmatrix}), \tag{4.37}$$

where $\mathbf{0}_3$ is a 3×1 zero vector and $\ddot{\mathbf{h}}_{d,b} := -\mathbf{K}_p(\mathbf{h}_{c,b} - \mathbf{h}_{d,b}) - \mathbf{K}_d(\dot{\mathbf{h}}_{c,b} - \dot{\mathbf{h}}_{d,b})$ is the desired base acceleration. Here, $\mathbf{h}_{c,b}$ is the current base trajectory and $\mathbf{K}_p$, $\mathbf{K}_d \in \mathbb{R}^{6\times 6}$ is a positive-definite tunable diagonal matrix representing the PD gain. The orientation error, computed as part of $(\mathbf{h}_{c,b} - \mathbf{h}_{d,b})$, is determined using the logarithmic map applied to the orientation error on $SO(3)$. Here, $SO(3)$ is the special orthogonal group in three dimensions representing the set of all rotation matrices in 3-D Euclidean space [187].

The SRBD model-based desired force is computed by solving the following QP problem.

$$\begin{aligned} \min_{\mathbf{F}} \quad & (\mathbf{SF} - \mathbf{b}_d)^T \mathbf{Q} (\mathbf{SF} - \mathbf{b}_d) \\ \text{s.t.} \quad & \mathbf{GF} < \mathbf{d}, \end{aligned} \tag{4.38}$$

where $\mathbf{Q} \in {}^{6\times 6}$ is a symmetric positive definite weight matrix and $\mathbf{GF} < \mathbf{d}$ encodes the force bounds, friction cone, and unilateral constraints on the contact forces.

The joint torque for tracking the desired base trajectory is computed from the optimized desired contact force as:

$$\boldsymbol{\tau}_b := -\mathbf{B}^T \mathbf{J}^T \mathbf{F}_d, \tag{4.39}$$

where $\mathbf{F}_d \in \mathbb{R}^{12\times 1}$ is the optimal contact force obtained by solving the QP in (4.38), the Jacobian matrix $\mathbf{J} \in \mathbb{R}^{12\times 18}$ relates the velocity of the quadruped's feet in the world frame to the generalized velocity of the quadruped, and $\mathbf{B}$ is the motor selection matrix. The computation of the feet Jacobian $\mathbf{J}$ is based on the robot's full-order model.

The control input for tracking swing foot trajectory is designed as a feedback controller with an approximate feedforward term. The desired swing foot acceleration for tracking the swing foot trajectory is given as:

$$\ddot{\mathbf{h}}_{d,sw_i} := -\mathbf{K}_{p_i}(\mathbf{h}_{c,sw_i} - \mathbf{h}_{d,sw_i}) - \mathbf{K}_{d_i}(\dot{\mathbf{h}}_{c,sw_i} - \dot{\mathbf{h}}_{d,sw_i}),$$

where $\mathbf{h}_{c,sw_i}$ ($i \in \{1,2,3,4\}$) is the current swing foot trajectory. $\mathbf{K}_{p_i}$ and $\mathbf{K}_{d_i} \in \mathbb{R}^{3\times 3}$ are the positive-definite tunable diagonal matrices representing the PD gains. The feedforward input torque term for swing foot i is computed as:

$$\boldsymbol{\tau}_{ff_i} = \mathbf{J}_{sw_i}^T \mathbf{M}_{os_i}(\ddot{\mathbf{h}}_{d,sw_i} - \dot{\mathbf{J}}_{sw_i}\dot{\mathbf{q}}_i) + \mathbf{C}_i, \tag{4.40}$$

where $\mathbf{M}_{os_i} := (\mathbf{J}_{sw_i}\mathbf{M}^{-1}\mathbf{J}_{sw_i}^T)^{-1}$ is the operation space inertia matrix for swing leg i, $\mathbf{J}_{sw_i}$ is Jacobian of the i^{th} swing foot relating the velocity of the quadruped's feet in the world frame to the generalized velocity of the quadruped, $\mathbf{C}_i$ is the vector of centripetal Coriolis and gravitational term for the i^{th} swing leg, and $\mathbf{q}_i \in \mathbb{R}^3$ is the generalized position of the i^{th} swing leg. In our implementation, we approximated $\mathbf{C}_i$ by only the gravitation term. This approximation is made based on the observation that the gravitational component accounts for a significant portion of $\mathbf{C}_i$, emphasizing its dominant influence at low and moderate walking speeds.

The feedback torque for the swing leg is given as:

$$\boldsymbol{\tau}_{fb_i} = \mathbf{J}_{sw_i}^T \mathbf{M}_{os_i}\ddot{\mathbf{h}}_{d,sw_i}. \tag{4.41}$$

The overall control law for the full-order robot is given as the sum of the torque required for base trajectory tracking τ_b and torque required for swing foot trajectory tracking; that is, $\boldsymbol{\tau} = \boldsymbol{\tau}_b + \boldsymbol{\tau}_{ff} + \boldsymbol{\tau}_{fb}$. This control input ensures reliable tracking with bounded errors by combining computed input torques that (a) minimize the tracking error of the base trajectory using a QP in (4.38), and (b) effectively track the swing foot trajectory through the feedforward term in (4.40) and PD feedback input in (4.41).

4.4 Full-Order Model Based Stability Analysis

This section presents the closed-loop stability analysis that validates the provable stability for the CoM dynamics of the hybrid, time-varying, full-order model under the proposed hierarchical control framework.

As explained in Section 4.3 the trotting quadruped of interest to this study is underactuated. Since its degree of underactuation is one, its underactuated dynamics are two-dimensional, which are represented by the dynamics of the forward CoM position and velocity (i.e., $\mathbf{X}$) associated with the full-order robot model. The actuated dynamics of the full-order model correspond to the base and swing foot trajectories that are directly driven by the lower-layer controller.

Since the actual CoM dynamics of the full-order model are not directly actuated, we need to explicitly analyze its stability. In this study, the actual dynamics of the CoM forward position and velocity $\mathbf{X}$ are approximated by the proposed HT-LIP model given in (4.3).Although the proposed HT-LIP footstep control law provably ensures the asymptotic stability of the closed-loop HT-LIP model under uncertain surface motions, the stability of the closed-loop dynamics of the CoM state $\mathbf{X}$ for the actual full-order model still needs to be analyzed. This is due to the discrepancy between the HT-LIP and the actual dynamics of $\mathbf{X}$.

4.4.1 S2S error system of actual CoM dynamics

Based on the closed-loop error system dynamics of the HT-LIP model given in (4.6), the S2S error system of the actual CoM dynamics can be expressed as:

$$\mathbf{e}|_{n+1}^{-} = \mathbf{A}_{d,n}\mathbf{e}|_{n}^{-} + \mathbf{d}_n, \tag{4.42}$$

where $n \in \mathbb{Z}^+$ and the vector $\mathbf{d}_n$ represents the lumped discrepancy between the actual S2S dynamics of the CoM and the reduced-order HT-LIP model, including the ignored nonlinear term in the S2S dynamics and the difference between the desired and actual footstep locations.

4.4.2 Stability analysis

Similar to [179], we consider the boundedness of the model discrepancy $\mathbf{d}_n$ as:

$$\|\mathbf{d}_n\| < d \quad \forall n \in \mathbb{N}, \tag{4.43}$$

where d is a positive constant. This boundedness assumption is reasonable for hardware implementation when the desired step duration is designed as finite (assumption (A12)) and the initial tracking error is relatively small. We denote the set of all possible values of $\mathbf{d}_n$ satisfying (4.43) as $\mathcal{D}$; that is, $\mathbf{d}_n \in \mathcal{D}$.

We use $\mathcal{E}$ to denote the minimum invariance set [188] such that for all $\mathbf{e}|_n^- \in \mathcal{E}$ and $\mathbf{d}_n \in \mathcal{D}$, we have $\mathbf{e}|_{n+1}^- \in \mathcal{E}$. Also, the asymptotic stability condition for the closed-loop error system of the HT-LIP model is established in Theorem 4. Consequently, the S2S dynamics in (4.42) are locally stable [188, 189] if the asymptotic stability condition for the HT-LIP model in Theorem 4 is met and if the uncertainty boundedness condition in (4.43) holds.

4.5 Simulations and Experiments

This section presents hardware experiment results to demonstrate the proposed control framework can stabilize quadrupedal trotting on a DRS with an aperiodic and unknown vertical motion even in the presence of various uncertainties. The experiment video is available at: https://youtu.be/BMPU0BJQC64.

4.5.1 Simulation Setup

This section briefly describes the setup utilized in MATLAB and PyBullet simulations.

(1) DRS motion

To simulate representative cases of DRS motion considered in the framework, we incorporated the following general vertical DRS motion as illustrated in Fig. 4-4.

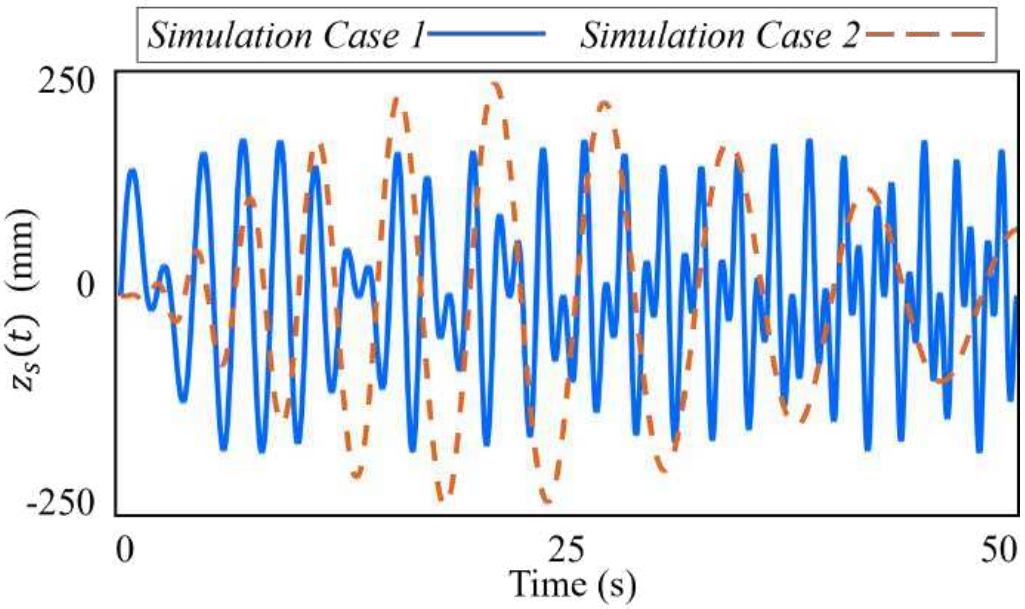

Figure 4-4: Vertical motion profile of the DRS for simulation cases.

(SC1) Simulation Case 1: general vertical motion of the DRS given as:

$$z(t) = 100\ (\sin 3t + \sin(t\sqrt{t/2+1}))\ \text{mm}. \tag{4.44}$$

(SC2) Simulation Case 2: aperiodic vertical motion of the DRS given as:

$$z_s(t) = 5t^2 \sin\left(\sqrt{100t+1}\right) \cdot e^{-t/10}\ \text{mm}. \tag{4.45}$$

(2) MATLAB setup

The robot dynamics are simulated using MATLAB's `ode45` function. The user specifies the approximate upper bound of the DRS vertical acceleration and desired velocity or velocity profile as high-level inputs. Foot-stepping control gains are computed by solving the QP designed in Section 4.3.2. We use a full-order model-based QP controller [178, 34]. The two QPs are solved using the MATLAB `quadprog` function.

(3) PyBullet setup

The PyBullet simulator is based on the Bullet Physics engine. We simulated a Go1 robot and a DRS model defined by the urdf files. The QP problems for foot-stepping control gain in Section 4.3.2 and joint torque computations in Section 4.3.4 are solved using the `OSQP` [190] solver.

(4) Controller setup

The controller setup for PyBullet simulations is given in Table 4.1. For MATLAB simulations, we use a full-order QP controller based on the formulation and setup provided in Secs. 4 and 5 of [34].

Table 4.1: Controller setup for PyBullet simulations

Controller parameter	Value
$\mathbf{Q}$	$\mathrm{diag}(1,1,2,2,2,1)$
$\mathbf{K}_p$	$\mathrm{diag}(1000,1000,500,1000,1000,1000)$
$\mathbf{K}_d$	$\mathrm{diag}(200,200,100,200,200,200)$
$\mathbf{K}_{p,i}$	$\mathrm{diag}(200,200,200)$
$\mathbf{K}_{d,i}$	$\mathrm{diag}(1,0.1,2)$

(5) User-specified HT-LIP parameters

Table 4.2 gives the user-specified range of the HT-LIP model parameters used as the input to the proposed framework (specifically, the higher and middle layers) in simulations and hardware experiments. To demonstrate the real-time computation of the foot-stepping control gain $\mathbf{K}$ and the versatility of the trotting gait features that the proposed framework can realize, these gait parameters are changed by the user and fed into the framework during simulations and experiments.

Table 4.2: Ranges of HT-LIP parameters used in simulations and experiments

Parameter	Range
CoM height above the surface z_0 (cm)	$[22,\ 26]$
Step duration $\Delta\tau_n$ (s)	$[0.15,\ 0.4]$
Walking speed (cm/s)	$[15,\ 25]$
Nominal step length $u_{x,r}$ (cm)	$[0,\ 15]$

4.5.2 Hardware Experiment Setup

(1) Treadmill

Our experiments use a Motek M-Gait treadmill to emulate a vertically moving DRS (Fig. 4-5). The treadmill can perform programmed pitching and sway movements. It weighs 750 kg, measures 2.3 m $\times$ 1.82 m $\times$ 0.5 m, and is equipped with two belts (each powered by a 4.5 kW servo motor). The robot is positioned approximately 0.8 m from the treadmill's pitching axis.

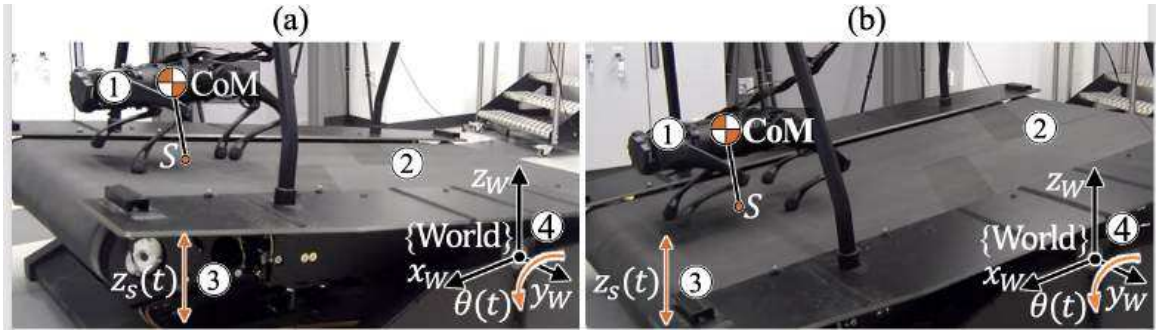

Figure 4-5: Illustration of the experimental setup. ①: Go1 quadrupedal (Unitree Robotics). ②: M-Gait treadmill (Motek Medical). ③: direction of the vertical DRS/treadmill motion $z_s(t)$ at point S. ④: world frame attached to the treadmill's axis of pitching. The treadmill's pitch angle at time t is $\theta(t)$. Subplots (a) and (b) show the treadmill at its pitch angle limits.

(2) Unknown vertical DRS motions

The experiments utilize the treadmill's pitch motions ($\theta_s(t)$) to generate aperiodic, vertical DRS motions at the robot's footholds (i.e., near the treadmill's far end). Table 4.3 summarizes the pitch motions (HC1)-(HC5), which are unknown to the proposed control framework during experiments.

Table 4.3: DRS motions under different hardware experiment cases.

Cases	DRS motion
(HC1)	$\theta_s(t) = 4^\circ(\sin 3t + \sin(t\sqrt{0.5t+1}))$.
(HC2)	$\theta_s(t) = 4^\circ(\sin 6t + \sin(0.1t^2))$.
(HC3)	$\theta_s(t) = 0.2^\circ t^2 \sin\left(\sqrt{100t+1}\right) \cdot e^{-t/10}$.
(HC4)	$\theta_s(t) = 4^\circ(\sin 3t + \sin(t\sqrt{t/2+1}))$ and $y_s(t) = 50\sin(\pi t)$ mm.
(HC5)	$\theta_s(t) = 2.5^\circ(\sin 3t + \sin(t\sqrt{0.5t+1}))$.

Although the pitch angle is small, it induces a significant maximum vertical acceleration at the robot's footholds (about 3.5 m/s^2) with a minimal horizontal surface motion. Figures 4-6-4-10 display the unknown general aperiodic DRS motions (HC1)-(HC5).

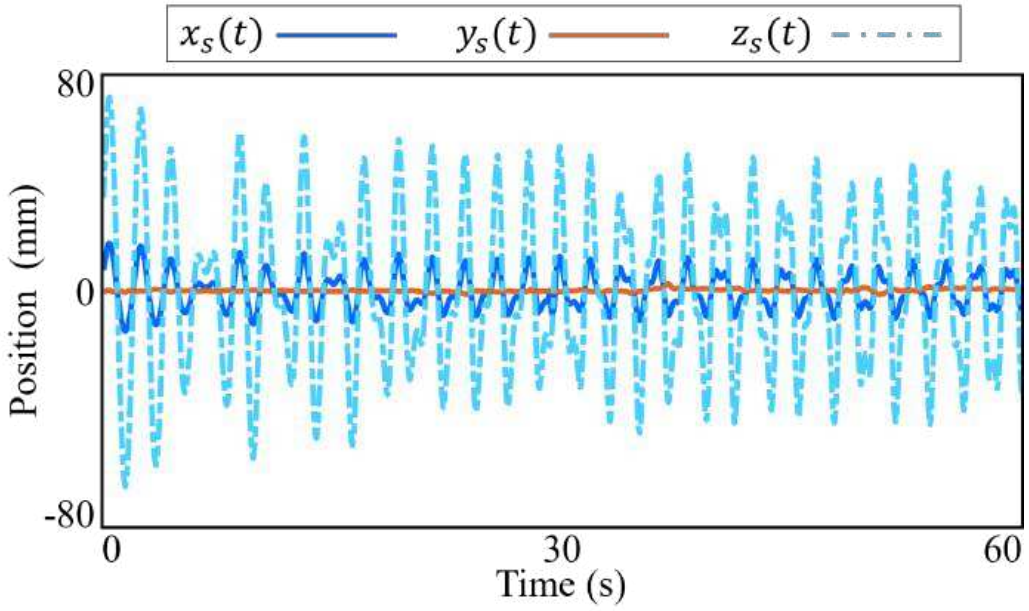

Figure 4-6: Ground-truth DRS motion in the world frame represented by a point where the robot performs the trotting gait during an experiment trial under DRS motion (HC1). The ground-truth data is collected by a mocap system.

(3) Additional uncertainties

To validate the robustness of the proposed approach beyond unknown vertical DRS motions, we test additional unmodeled uncertainties (Fig. 4-1). To assess the robustness against unknown sway surface motions, surface motion (HC4) contains a sway displacement $y_s(t)$

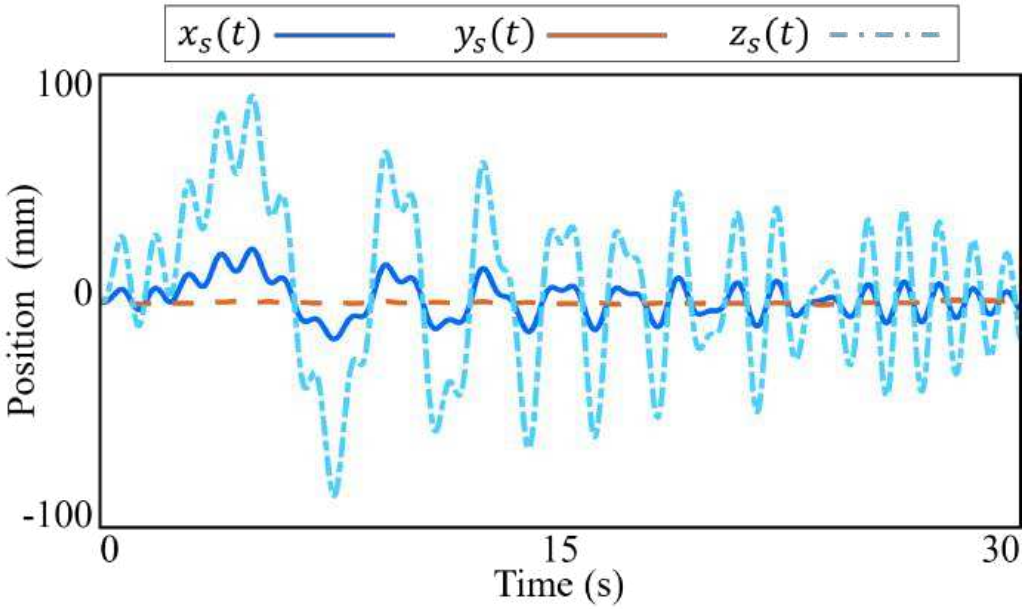

Figure 4-7: Ground-truth DRS motion in the world frame represented by a point where the robot performs the trotting gait during an experiment trial under DRS motion (HC2).

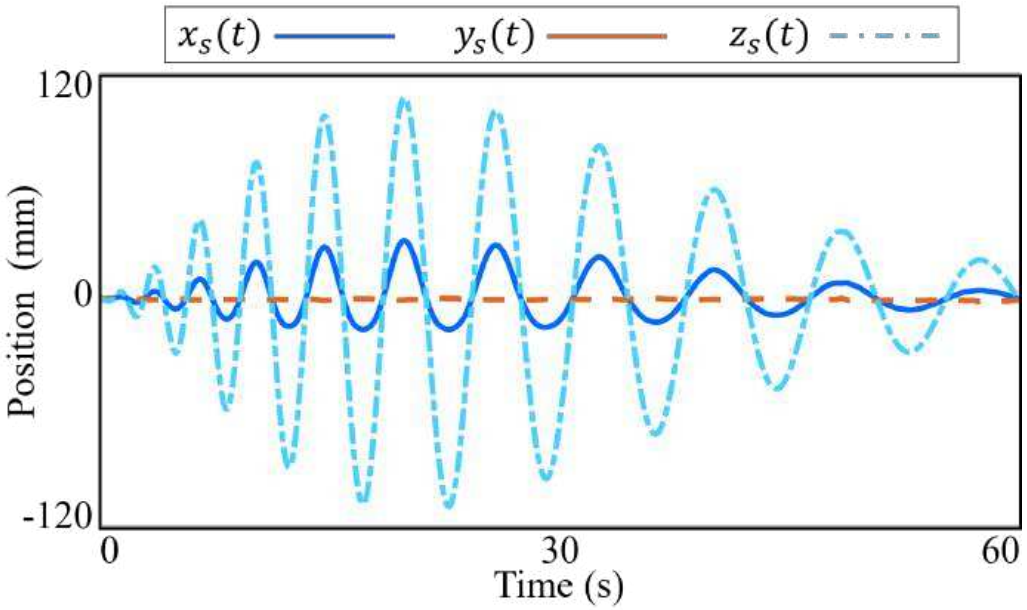

Figure 4-8: Ground-truth DRS motion in the world frame represented by a point where the robot performs the trotting gait during an experiment trial under DRS motion (HC3).

(see Table 4.3 and Fig. 4-9), causing a peak horizontal acceleration of 2.6 m/s^2 at the robot's footholds.

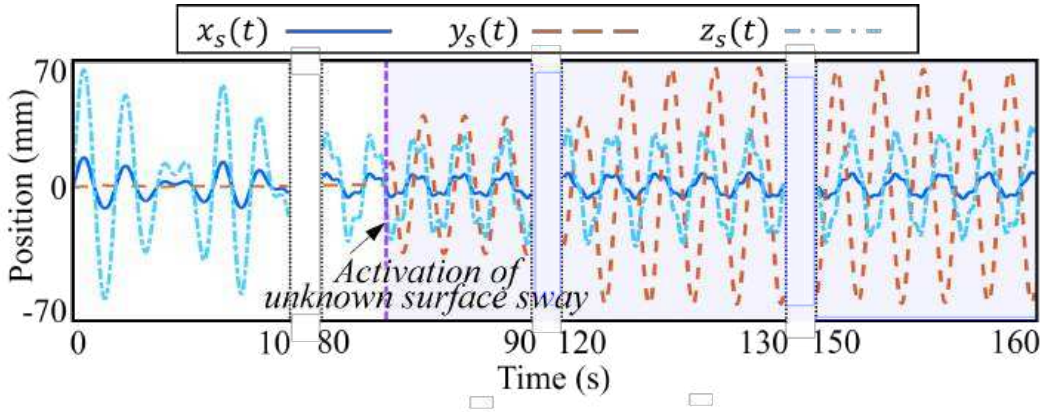

Figure 4-9: Ground-truth position trajectory of the point on the treadmill/DRS around which the robot performs the trotting gait during the unknown pitch and sway movement (HC4) of the DRS. The shaded area highlights the period during which the unknown DRS sway motion is active.

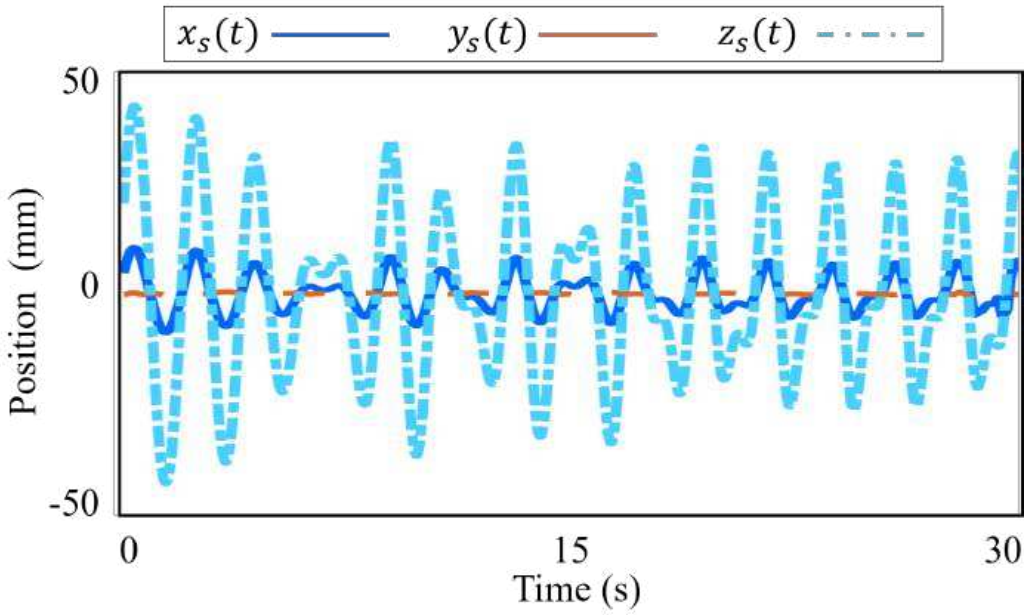

Figure 4-10: Ground-truth DRS motion in the world frame represented by a point where the robot performs the trotting gait during an experiment trial under DRS motion (HC5).

Besides surface sway, four other types of uncertainties are tested during (HC5) with maximum vertical and lateral accelerations respectively at 1.5 m/s^2 and 0.5 m/s^2. These uncertainties are: (i) uncertain friction coefficient (0.3-0.4) induced by a smooth glass surface while the framework considers a coefficient of 0.8; (ii) unknown rigid (10 lbs) and fluid (9 lbs) loads placed on the trunk, weighing respectively 36% and 32% of the robot's mass; (iii)

uneven (pebbled) surface with a maximum height of 6 cm; and (iv) sudden pushes lasting less than 0.2 s per push and inducing a robot heading error of 30° just after the push.

4.5.3 Control framework setup

The HT-LIP model parameters considered by the higher-layer footstep planner and middle-layer trajectory generator are given in Table 4.2. These parameters are changed during experiments so as to demonstrate the footstep control gain $\mathbf{K}$ can be computed in real-time and the control framework can address different trotting gait features (e.g., step duration). The estimation error of the vertical surface acceleration is up to 1 m/s^2. Note that the proposed framework only considers the estimated instead of the actual surface acceleration. The middle and lower layers approximate the robot's CoM at the base/trunk center. The parameters of the lower-layer controller for hardware experiments are specified in Table 4.4.

Table 4.4: Controller setup for hardware experiments

Controller parameters	Value
$\mathbf{Q}$	$\text{diag}(1,1,50,400,400,100)$
$\mathbf{K}_p$	$\text{diag}(80,60,120,120,120,40)$
$\mathbf{K}_d$	$\text{diag}(10,10,5,10,10,2)$
$\mathbf{K}_{p,i}$	$\text{diag}(80,120,40)$
$\mathbf{K}_{d,i}$	$\text{diag}(3,3,2)$

4.5.4 Experimental Results

This section reports experiment results under unknown DRS motions and various other types of uncertainties.

(1) Validation under unknown vertical surface motions

As shown in Figs. 4-11-4-13, the actual height and orientation of the robot's base (i.e., trunk) relatively closely track the desired base trajectories during the unknown and aperiodic vertical surface motions (HC1)-(HC3), indicating a stable trotting gait under the proposed control framework.

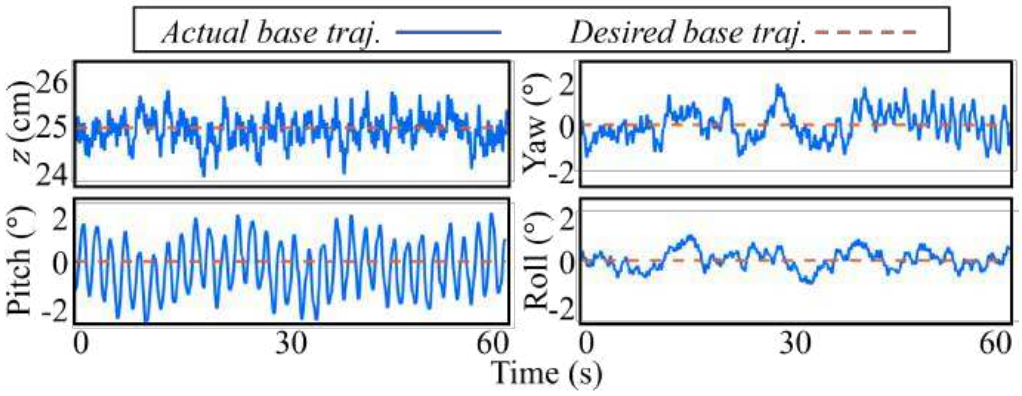

Figure 4-11: Desired and actual base trajectories under the hardware experiment case (HC1). The small tracking errors indicate stable robot trotting.

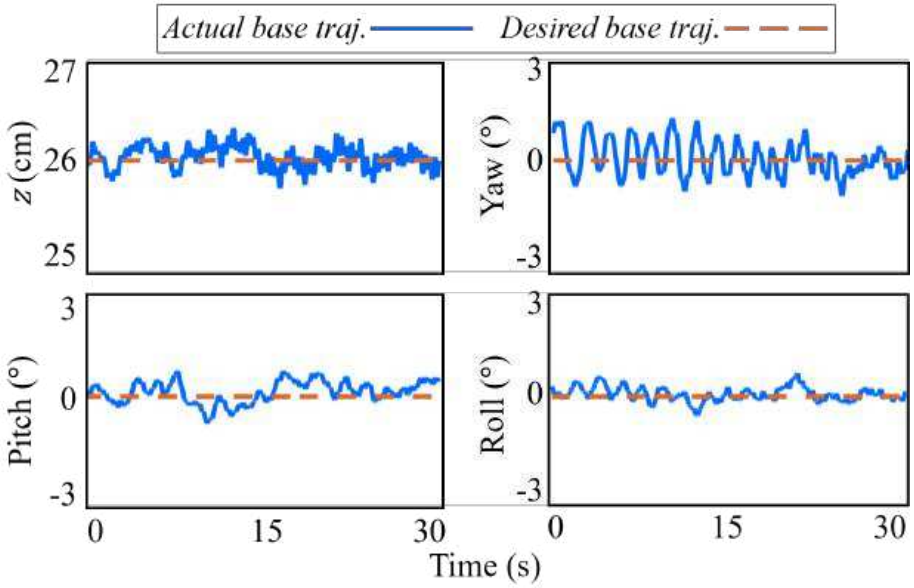

Figure 4-12: Desired and actual base trajectories under the hardware experiment case (HC2). The small tracking errors indicate stable robot trotting.

Further, the joint torque profile in Figs. 4-14-4-16 demonstrate a consistent torque pattern that respects the actuator limit of 22.5 N/m for all joints under surface motions (HC1)-(HC3).

The accurate base trajectory tracking and consistent torque profiles underline the effectiveness of the proposed framework in handling various general vertical surface motions.

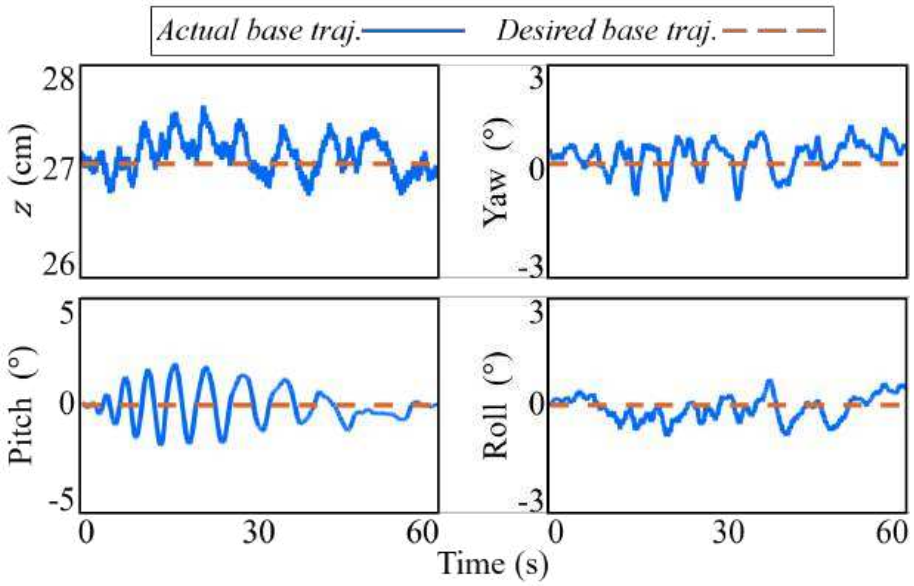

Figure 4-13: Desired and actual base trajectories under the hardware experiment case (HC3). The small tracking errors indicate stable robot trotting.

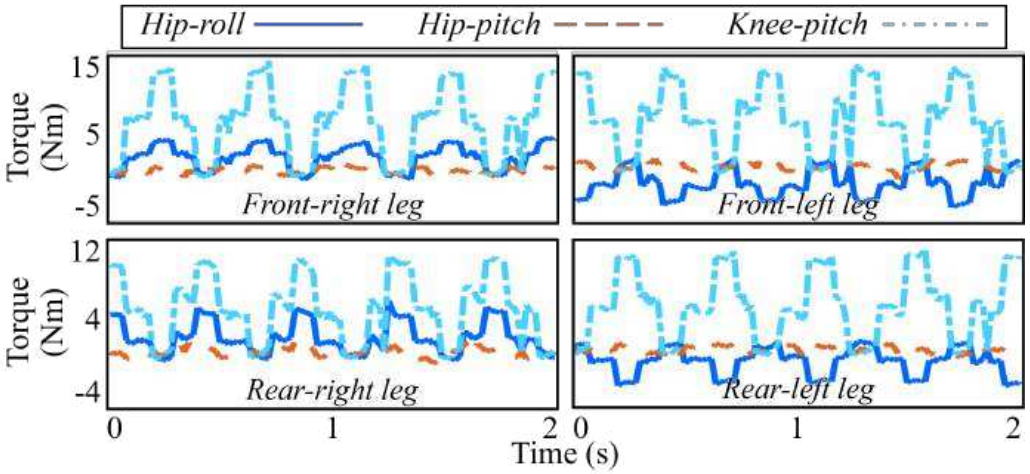

Figure 4-14: Torque profiles under the hardware experiment case (HC1), all of which respect the robot's actuator limit of 22.5 Nm.

(2) Validation under various additional uncertainties

To evaluate robustness, we conduct hardware validation experiments under uncertain cases described in Section 4.5.2.

The subplots (a) and (c) in Fig. 4-17 confirm that the robot's base height closely follows the desired value even under the unknown DRS sway motion and reduced surface friction.

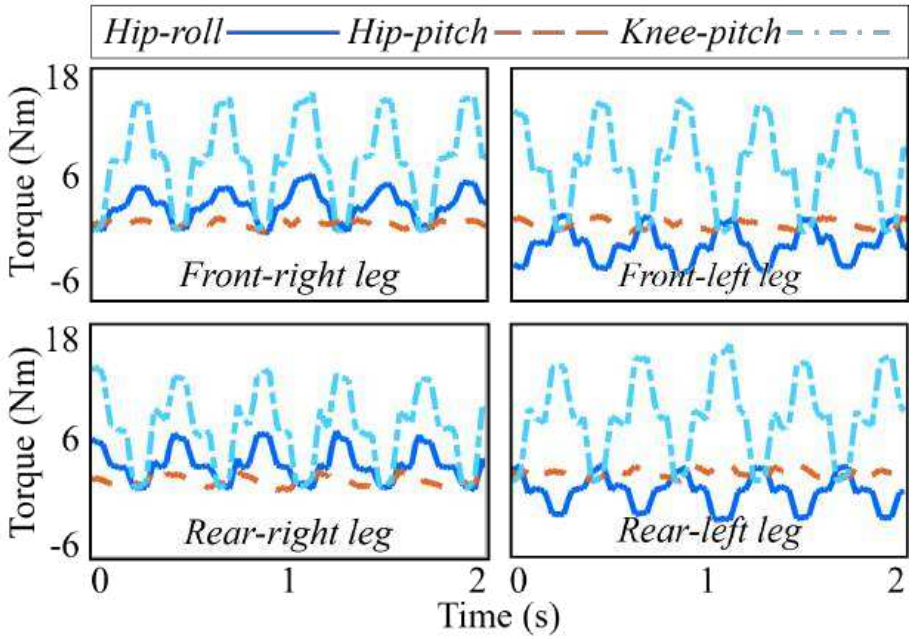

Figure 4-15: Torque profiles under the hardware experiment case (HC2), all of which respect the robot's actuator limit of 22.5 Nm.

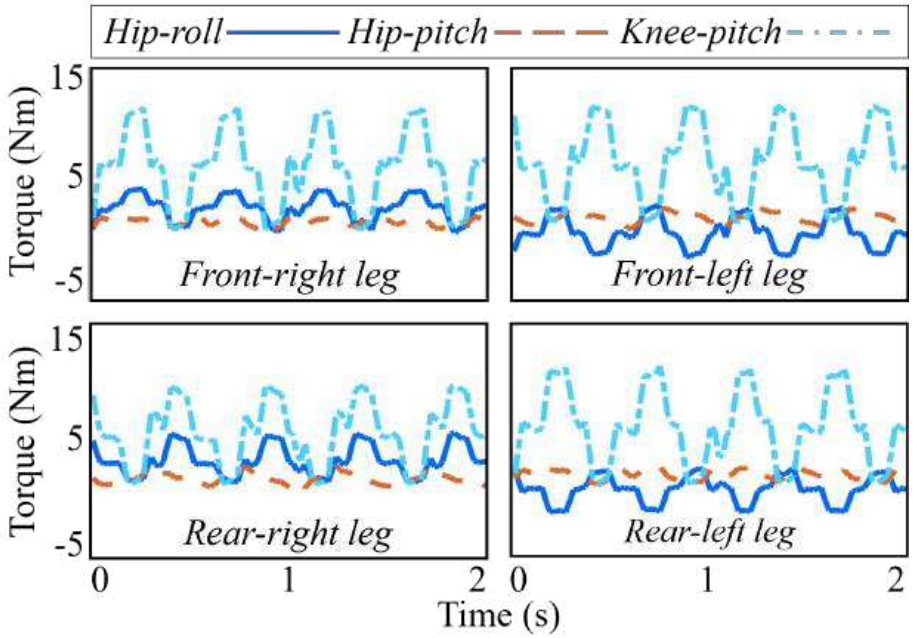

Figure 4-16: Torque profiles under the hardware experiment case (HC3), all of which respect the robot's actuator limit of 22.5 Nm.

The subplot (b) shows a notable oscillatory deviation of the actual base height from the desired value due to the unevenness of the pebbled surface, indicating a moderate level

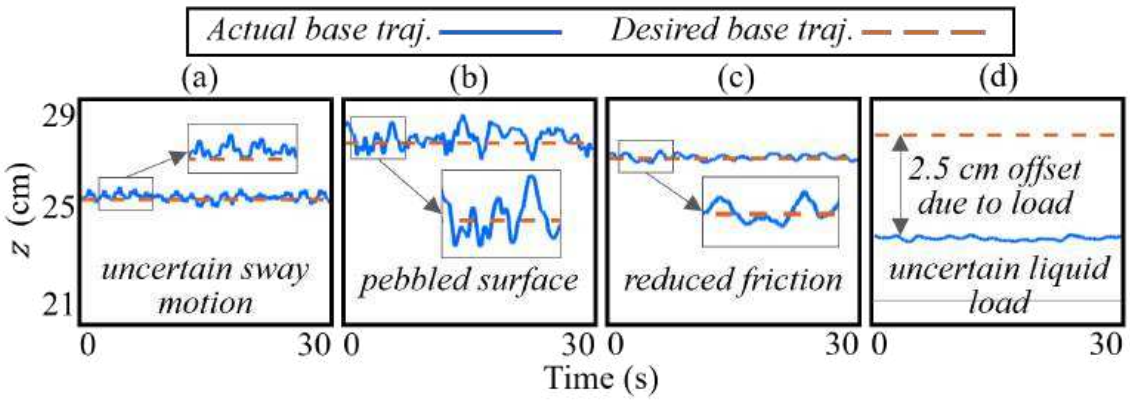

Figure 4-17: CoM height trajectories under various cases of uncertainties, all during the unknown vertical DRS motion (HC5). These cases include (a) unknown sway motion, (b) pebbled surface, (c) surface with reduced friction, and (d) uncertain load on the robot's trunk.

of violation of the constant base height assumption (i.e. assumption (A1)). The subplot (d) shows the significant uncertain liquid load applied to the robot's trunk causes a nearly constant base height tracking error of 2.5 cm. Still, both subplots (b) and (d) indicate stable locomotion despite uncertainties. The results under the unknown solid load are similar to subplot (d) and thus are omitted for brevity.

Figure 4-18 displays the push recovery results during the unknown vertical and lateral DRS motion (HC4).

The intermittent spikes in the robot's base height and orientation trajectories are induced by external pushes. As highlighted by the shaded areas in Fig. 4-18, the robot is able to recover within two seconds after each significant push, demonstrating the robustness of the proposed framework against external pushes during unknown DRS motions.

It is worth underlining that the robot sustained stable waking during robustness validation experiments while subject to the maximum possible DRS motion uncertainty in our lab setting, which uses a Motek M-Gait treadmill [191] as a DRS.

4.5.5 Comparative Experiments

To show the improved robustness of our proposed framework compared to existing controllers, we experimentally test the Go1 robot's proprietary controller and a state-of-the-art baseline controller [1] during unknown vertical surface motion (HC5). The baseline con-

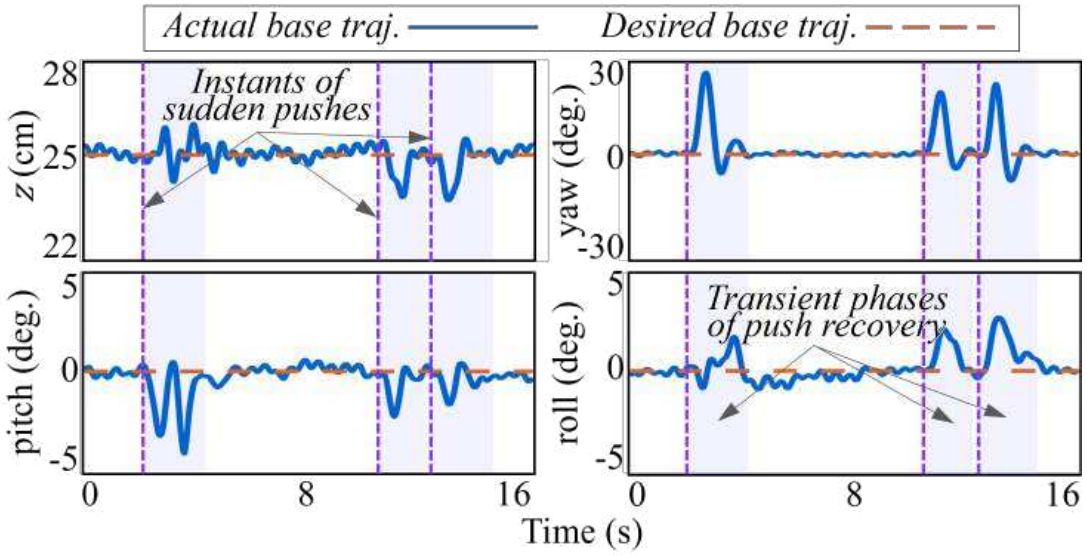

Figure 4-18: Robustness to sudden pushes under the uncertain DRS motion (HC4). The purple dashed lines highlight the push instants, while the shaded regions show the transient push recovery phases. The proposed control framework effectively stabilizes the perturbed trajectories to their desired values within 2 seconds.

troller shares the same low-layer torque controller as the proposed framework while its higher and lower layers assume a static ground.

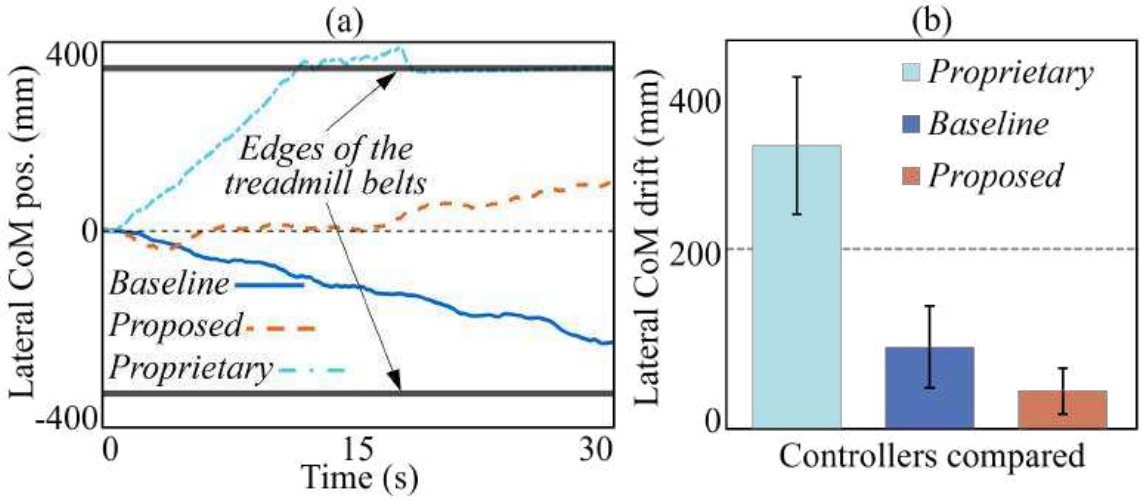

Figure 4-19: Lateral-position drift comparison with the robot's proprietary controller and a state-of-the-art controller [1] during the DRS motion (HC5): (a) lateral CoM position drift during a representative hardware experiment of 30 s and (b) average lateral drift (mean $\pm$ one standard deviation) during five experiment trials of 15 s. The proposed control approach achieves the least amount of lateral drift among the three approaches compared.

As illustrated by the lateral CoM position trajectories in Fig. 4-19, the robot exhibits a notably lower lateral drift under our proposed framework compared to the existing controllers. The relatively smaller drift of our proposed framework is partly due to the explicit treatment of the unknown DRS motion in the higher-layer planner, which is missing in the baseline controller. Also, both our framework and the baseline control approach correct the robot's heading direction based on the estimated absolute base position and yaw angle provided by the robot's default state estimator. In contrast, based on the fast lateral position drift under the proprietary controller, it is possible that the proprietary controller does not compensate for the position error.

Still, due to the drift in estimated absolute base position and yaw angle [148], the proposed approach exhibits a notable lateral position drift of approximately 10 cm between 15 s and 30 s. To improve path tracking accuracy, a more accurate state estimator is being developed and will be used in future work.

4.6 Discussion

One novel contribution of this study is the introduction of the HT-LIP model for locomotion during general (periodic or aperiodic) and vertical DRS motions. Similar to existing LIP models for static surfaces [179, 27, 21, 192], the HT-LIP model retains linearity. Yet, the model is also explicitly time-varying due to the surface motion, distinguishing it from the time invariance of existing models. Meanwhile, the model is homogeneous and is thus fundamentally different from the LIP model for horizontally moving surfaces [193].

Another key contribution is the construction of a discrete-time footstep control law that provably robustly stabilizes the HT-LIP system under variable footstep durations and unknown DRS accelerations. While the structure of the footstep controller is inspired by the previous H-LIP based framework [179], the proposed stability condition underlying the controller explicitly treats the the time dependence of the HT-LIP model. Further, the stability condition does not consider any accurate knowledge of the surface motion as in our previous work [20, 193]. Instead, the condition and the footstep controller only assume a finite bound of the surface acceleration at the robot-surface contact region (assumption

(A11)). Finally, the HT-LIP footstep controller is cast as a QP that not only exactly enforces the stability conditions but also enables real-time, reactive foot placement.

By incorporating the proposed HT-LIP footstep controller as a higher-layer planner for the full-order robot system, the proposed hierarchical control framework achieves robust locomotion under a broad range of unknown vertical surface accelerations and various additional types of uncertainties (Section 4.5.2). Moreover, the framework realizes notably less lateral global-position drift during comparative validation experiments under unknown vertical DRS motions and external pushes (Fig. 4-19).

Our future work will extend the proposed theoretical results and control framework design from linear DRS motions to simultaneous surface translation and rotation. Further, to improve the path tracking accuracy of the control framework, a more accurate state estimator that does not assume accurate knowledge of the DRS motion will be developed based on existing advanced estimation methods [194, 195, 196].

4.7 Conclusion

This chapter has introduced a hierarchical control framework for robust quadrupedal locomotion with variable footstep duration during unknown and general (periodic or aperiodic) vertical ground motions. A reduced-order model was derived by analytically extending the existing linear, time-invariant H-LIP model to explicitly consider the ground surface motion, resulting in a hybrid, time-varying LIP model (i.e., HT-LIP). Taking the HT-LIP model as a basis, a discrete-time footstep controller was constructed with a provable guarantee for robust stability, and was further cast as a quadratic program to enable real-time, reactive foot placement planning for the full-order robot system. The proposed control framework incorporated the HT-LIP footstep controller as a higher-layer planner, and its middle- and lower-layers were developed to plan and control the robot's full-body motions that agree with the desired robot behaviors supplied by the higher planner. Experiment results confirmed the robustness of the proposed framework in sustaining stable quadrupedal trotting under various unknown, aperiodic surface motions, external pushes and (solid and liquid) loads, and slippery and rocky surfaces.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

This dissertation research aims to overcome the challenges of legged locomotion on dynamic rigid surfaces (DRS) and facilitate the deployment of legged robots on real-world DRS such as ships. Towards this end, this dissertation has presented the following contributions:

- **Dynamic modeling of legged locomotion.** A sufficiently accurate and computationally efficient dynamic model can be used to enable real-time planning and control. A high-dimensional full-order model is more accurate but computationally expensive, whereas low-dimensional reduce-order models may provide a reasonable trade-off between model accuracy and computational efficiency. This dissertation has modeled the full-order robot dynamics associated with quadrupedal walking on DRS as a nonlinear, hybrid, time-varying system, which can be used for the stability analysis of the complete robot dynamical system. To facilitate real-time planning and control of underactuated legged locomotion on DRS, we have developed time-varying linear inverted pendulum models that are computationally efficient and adequately accurate.

- **Desired trajectory planning.** The desired trajectory plan provides a reference for a tracking controller. Controller performance depends on the underlying model accuracy, physical feasibility of the desired trajectory, and computational latency. With the limited onboard computational capacity, a trade-off is necessary to ensure

real-time planning capability in dynamically changing environments. This dissertation has proposed an analytical approximate solution of a time-varying LIP model on a periodically moving DRS for real-time trajectory planning. It has also developed a robustly stable footstep planning framework based on stability analysis of the hybrid time-varying LIP (HT-LIP) model for robust legged locomotion on dynamic surfaces with unknown vertical motion. The footstep planner in this dissertation was formulated as a quadratic programming problem that ensures the robust stability and physical feasibility of the footsteps.

- **Control framework for locomotion on DRS.** This dissertation has developed a full-order model-based, provably stable, continuous-phase control law that stabilizes the overall nonlinear, hybrid, time-varying model. Furthermore, it has also proposed an HT-LIP model-based robust control framework for quadrupedal locomotion on unknown vertical ground motion under various real-world uncertainties such as uneven surface, external load, and sudden pushes.

- **Stability analysis.** Stability analysis is essential for controller design and verification. Chapter 2 has presented the stability analysis of the nonlinear, hybrid, time-varying, full-order control system and derived conditions on controller gains that provably stabilize the closed-loop system under known periodic DRS motion. We have also derived stability conditions for robust foot placement under unknown vertical motion and analyzed the robust stability of the overall underactuated full-order model under the proposed control framework.

- **Validation in simulations and hardware experiments.** To assess the effectiveness of the proposed control frameworks, we have performed MATLAB and Pybullet simulations as well as hardware experiments. Results from thorough validations have been presented and analyzed using a Laikago and a Go1 quadruped developed by Unitree Robotics. The evaluation results in Chapter 2 have demonstrated the effectiveness of the proposed approach on a DRS with a known sinusoidal motion and under a moderate level of uncertainty. Validation in Chapter 3 has indicated the real-time replanning capability of the proposed analytical solution based planning

framework. Hardware validation in Chapter 4 has confirmed the robustness of the proposed framework in sustaining stable quadrupedal trotting under various unknown, aperiodic surface motions, external pushes and (solid and liquid) loads, and slippery and rocky surfaces.

5.2 Future Work

While this dissertation research has tackled several key challenges in DRS locomotion planning and control, there remain many open directions of future work for further improving the performance of legged locomotion on DRS.

- **Robust state estimation on DRS.** To enhance the path tracking performance of DRS locomotion, reliable state estimation is needed to accurately estimate the movement state of both the robot and the DRS. While existing advanced state estimation techniques work effectively on static surfaces [144, 197], they may not be effective for DRS locomotion because they typically assume the ground is static in the inertial frame. Recent research from our lab that focuses on state estimation for DRS locomotion [148, 198, 199] may potentially lead to a robust state estimator on DRS with unknown surface motion.

- **Stability analysis for locomotion under composite DRS motion.** This dissertation has established stability conditions for unknown, general, vertical DRS motion and experimentally verified the performance of the proposed control approach in the presence of lateral and vertical motion uncertainties. However, a comprehensive stability analysis of the locomotion model under general composite DRS motion is essential to inform the design of a control framework that can reliably ensure locomotion performance under such conditions. The recent work from our lab that addresses humanoid locomotion control under periodic ground sway [193] could potentially be integrated with the outcomes of this dissertation to derive the stability conditions for composite DRS motion.

- **Embedding learning in model-based methods.** Learning-based approaches have

achieved remarkably robust locomotion performance in the presence of a broad spectrum of real-world uncertainties when an adequate amount of reliable data is available for training and optimizing the learning algorithms [200]. Such a performance is due to their inherent stochastic optimization, which explores numerous locomotion scenarios that model-based methods may not handle well [200, 106]. However, pure learning-based methods lack provable performance guarantees [103]. It would be meaningful to integrate learning and model-based approaches to achieve both provability and enhanced performance.

- **Loco-manipulation on DRS** This dissertation has tackled some of the substantial challenges in DRS locomotion control to facilitate the deployment of legged robots in maritime applications, including ship inspection and maintenance. Since many of these real-world tasks naturally demand manipulation, a logical extension of this research is the exploration of loco-manipulation control on DRS.

References

[1] 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 *Proc. IEEE/RSJ Int. Conf. Intel. Rob. Syst.*, pp. 2245–2252, 2018.

[2] DARPA, "DARPA robotics challenge." https://www.darpa.mil/program/darpa-robotics-challenge, 2015. Accessed: 2020-09-10.

[3] M. Hutter, C. Gehring, A. Lauber, F. Gunther, C. D. Bellicoso, V. Tsounis, P. Fankhauser, R. Diethelm, S. Bachmann, M. Blösch, *et al.*, "Anymal-toward legged robots for harsh environments," *Advanced Robotics*, vol. 31, no. 17, pp. 918–931, 2017.

[4] D. Wooden, M. Malchano, K. Blankespoor, A. Howardy, A. A. Rizzi, and M. Raibert, "Autonomous navigation for bigdog," in *2010 IEEE international conference on robotics and automation*, pp. 4736–4741, Ieee, 2010.

[5] J. Li, H. Qin, J. Wang, and J. Li, "Openstreetmap-based autonomous navigation for the four wheel-legged robot via 3d-lidar and ccd camera," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 3, pp. 2708–2717, 2021.

[6] C. D. Bellicoso, M. Bjelonic, L. Wellhausen, K. Holtmann, F. Günther, M. Tranzatto, P. Fankhauser, and M. Hutter, "Advances in real-world applications for legged robots," *Journal of Field Robotics*, vol. 35, no. 8, pp. 1311–1326, 2018.

[7] H. Kolvenbach, D. Wisth, R. Buchanan, G. Valsecchi, R. Grandia, M. Fallon, and M. Hutter, "Towards autonomous inspection of concrete deterioration in sewers with legged robots," *Journal of field robotics*, vol. 37, no. 8, pp. 1314–1327, 2020.

[8] M. H. Raibert, *Legged robots that balance*. MIT press, 1986.

[9] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, "Hybrid zero dynamics of planar biped walkers," *IEEE Trans. Autom. Control*, vol. 48, no. 1, pp. 42–56, 2003.

[10] 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 *Proc. IEEE Int. Conf. Rob. Autom.*, vol. 2, pp. 1620–1626, 2003.

[11] P.-B. Wieber, R. Tedrake, and S. Kuindersma, "Modeling and control of legged robots," in *Springer handbook of robotics*, pp. 1203–1234, Springer, 2016.

[12] P. Liu, H. Yu, S. Cang, and L. Vladareanu, "Robot-assisted smart firefighting and interdisciplinary perspectives," in *2016 22nd International Conference on Automation and Computing (ICAC)*, pp. 395–401, IEEE, 2016.

[13] C. Peter Shadbolt, "U.s. navy unveils robotic firefighter."

[14] C. Wire, "Navy unveils firefighting robot."

[15] M. Ramezani, M. Brandao, B. Casseau, I. Havoutis, and M. Fallon, "Legged robots for autonomous inspection and monitoring of offshore assets," in *Offshore Technology Conference*, OnePetro, 2020.

[16] G. Quaglia, L. Bruzzone, G. Bozzini, R. Oderio, and R. P. Razzoli, "Epi. q-tg: mobile robot for surveillance," *Industrial Robot: An International Journal*, vol. 38, no. 3, pp. 282–291, 2011.

[17] S. Tadokoro, "Overview of the impact tough robotics challenge and strategy for disruptive innovation in safety and security," in *Disaster Robotics*, pp. 3–22, Springer, 2019.

[18] B. Morris and J. W. Grizzle, "Hybrid invariant manifolds in systems with impulse effects with application to periodic locomotion in bipedal robots," *IEEE Transactions on Automatic Control*, vol. 54, no. 8, pp. 1751–1764, 2009.

[19] M. S. Motahar, S. Veer, and I. Poulakakis, "Composing limit cycles for motion planning of 3d bipedal walkers," in *Proc. IEEE Conf. Dec. Contr.*, pp. 6368–6374, 2016.

[20] A. Iqbal, Y. Gao, and Y. Gu, "Provably stabilizing controllers for quadrupedal robot locomotion on dynamic rigid platforms," *IEEE/ASME Trans. Mechatron.*, vol. 25, no. 4, pp. 2035–2044, 2020.

[21] 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 *Proc. IEEE Int. Conf. Intel. Rob. Sys.*, vol. 1, pp. 239–246, 2001.

[22] S. Kajita, M. Morisawa, K. Miura, S. Nakaoka, K. Harada, K. Kaneko, F. Kanehiro, and K. Yokoi, "Biped walking stabilization based on linear inverted pendulum tracking," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4489–4496, 2010.

[23] 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)*, pp. 5148–5154, IEEE, 2015.

[24] C. Mastalli, I. Havoutis, M. Focchi, D. G. Caldwell, and C. Semini, "Motion planning for quadrupedal locomotion: Coupled planning, terrain mapping, and whole-body control," *IEEE Trans. Rob.*, vol. 36, no. 6, pp. 1635–1648, 2020.

[25] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *Proc. IEEE Int. Conf. Humanoid Rob.*, pp. 200–207, 2006.

[26] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback control of dynamic bipedal robot locomotion*, vol. 28. CRC press, 2007.

[27] Y. Gong and J. Grizzle, "Angular momentum about the contact point for control of bipedal locomotion: Validation in a LIP-based controller," *arXiv preprint arXiv:2008.10763*, 2020.

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

[29] J. Pratt, T. Koolen, T. De Boer, J. Rebula, S. Cotton, J. Carff, M. Johnson, and P. Neuhaus, "Capturability-based analysis and control of legged locomotion, part 2: Application to M2V2, a lower-body humanoid," *The international journal of robotics research*, vol. 31, no. 10, pp. 1117–1133, 2012.

[30] J. Englsberger, G. Mesesan, C. Ott, and A. Albu-Schäffer, "DCM-based gait generation for walking on moving support surfaces," in *Proc. of IEEE-RAS Int. Conf. Humanoid Rob.*, pp. 1–8, 2018.

[31] S. Caron, "Biped stabilization by linear feedback of the variable-height inverted pendulum model," in *Proc. IEEE Int. Conf. Rob. Autom.*, pp. 9782–9788, 2020.

[32] H. Zhao, A. Hereid, W.-l. Ma, and A. D. Ames, "Multi-contact bipedal robotic locomotion," *Robotica*, vol. 35, no. 5, pp. 1072–1106, 2017.

[33] X. Xiong, J. Reher, and A. D. Ames, "Global position control on underactuated bipedal robots: Step-to-step dynamics approximation for step planning," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2825–2831, IEEE, 2021.

[34] A. Iqbal and Y. Gu, "Extended capture point and optimization-based control for quadrupedal robot walking on dynamic rigid surfaces," in *Proc. IFAC Mod. Est. Contr. Conf.*, vol. 54, pp. 72–77, 2021.

[35] R. Blickhan, "The spring-mass model for running & hopping," *Journal of Biomechanics*, vol. 22, no. 11, pp. 1217–27, 1989.

[36] R. Blickhan and R. J. Full, "Similarity in multilegged locomotion: Bouncing like a monopode," *Journal of Comparative Physiology A*, vol. 173, pp. 509–17, Nov 1993.

[37] R. M. Alexander, "Three uses for springs in legged locomotion," *The International Journal of Robotics Research*, vol. 9, pp. 53–61, 1990.

[38] P. Holmes, R. J. Full, D. Koditschek, and J. Guckenheimer, "The dynamics of legged locomotion: Models, analyses, and challenges," *SIAM review*, vol. 48, no. 2, pp. 207–304, 2006.

[39] H. Geyer, A. Seyfarth, and R. Blickhan, "Spring-mass running: simple approximate solution and application to gait stability," *Journal of Theoretical Biology*, vol. 232, pp. 315–28, 2005.

[40] A. J. Spence, S. Revzen, J. Seipel, C. H. Mullens, and R. J. Full, "Insects running on elastic surfaces.," *Journal of Experimental Biology*, vol. 213 11, pp. 1907–20, 2010.

[41] J. Schmitt and P. Holmes, "Mechanical models for insect locomotion:dynamics and stability in horizontal plane –ii.," *Biological Cybernetics*, vol. 83, pp. 517–527, 2000.

[42] R. M. Ghigliazza, R. Altendorfer, P. Holmes, and D. E. Koditschek, "A simply stabilized running model," *SIAM Journal of Applied Dynamical Systems*, vol. 2, pp. 187–218, 2003.

[43] A. Iqbal, Z. Mao, and Y. Gu, "Modeling, analysis, and control of slip running on dynamic platforms," *ASME L. Dyn. Sys. Contr.*, vol. 1, no. 2, 2021.

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

[45] Y. Zhao, B. R. Fernandez, and L. Sentis, "Robust optimal planning and control of non-periodic bipedal locomotion with a centroidal momentum model," *Int. J. Rob. Res.*, vol. 36, no. 11, pp. 1211–1242, 2017.

[46] A. Goswami and V. Kallem, "Rate of change of angular momentum and balance maintenance of biped robots," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, vol. 4, pp. 3785–3790, IEEE, 2004.

[47] C. Li, Y. Ding, and H.-W. Park, "Centroidal-momentum-based trajectory generation for legged locomotion," *Mechatronics*, vol. 68, p. 102364, 2020.

[48] W. Du, M. Fnadi, and F. Benamar, "Rolling based locomotion on rough terrain for a wheeled quadruped using centroidal dynamics," *Mechanism and Machine Theory*, vol. 153, p. 103984, 2020.

[49] Z. Zhou, B. Wingo, N. Boyd, S. Hutchinson, and Y. Zhao, "Momentum-aware trajectory optimization and control for agile quadrupedal locomotion," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7755–7762, 2022.

[50] M. Bjelonic, R. Grandia, O. Harley, C. Galliard, S. Zimmermann, and M. Hutter, "Whole-body mpc and online gait sequence generation for wheeled-legged robots," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 8388–8395, IEEE, 2021.

[51] M. Focchi, A. Del Prete, I. Havoutis, R. Featherstone, D. G. Caldwell, and C. Semini, "High-slope terrain locomotion for torque-controlled quadruped robots," *Autonomous Robots*, vol. 41, no. 1, pp. 259–272, 2017.

[52] J. W. Grizzle, G. Abba, and F. Plestan, "Asymptotically stable walking for biped robots: Analysis via systems with impulse effects," *IEEE Transactions on Automatic Control*, vol. 46, no. 1, pp. 51–64, 2001.

[53] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, "Hybrid zero dynamics of planar biped walkers," *IEEE Trans. Autom. Control*, vol. 48, no. 1, pp. 42–56, 2003.

[54] Y. Gu, B. Yao, and C. S. G. Lee, "Exponential stabilization of fully actuated planar bipedal robotic walking with global position tracking capabilities," *ASME J. Dyn. Sys., Meas., Contr.*, vol. 140, no. 5, 2018.

[55] Y. Gu and C. Yuan, "Adaptive robust trajectory tracking control of fully actuated bipedal robotic walking," in *2020 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pp. 1310–1315, 2020.

[56] Y. Gu and C. Yuan, "Adaptive robust tracking control for hybrid models of three-dimensional bipedal robotic walking under uncertainties," *Journal of Dynamic Systems, Measurement, and Control*, vol. 143, no. 8, p. 081007, 2021.

[57] Y. Zhao and Y. Gu, "A non-periodic planning and control framework of dynamic legged locomotion," *International Journal of Intelligent Robotics and Applications*, vol. 4, pp. 95–108, 2020.

[58] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. John Wiley & Sons, 2020.

[59] M. W. Spong, "An historical perspective on the control of robotic manipulators," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 5, pp. 1–31, 2022.

[60] R. Ortega and M. W. Spong, "Adaptive motion control of rigid robots: A tutorial," *Automatica*, vol. 25, no. 6, pp. 877–888, 1989.

[61] J.-P. Laumond *et al.*, *Robot motion planning and control*, vol. 229. Springer, 1998.

[62] R. Tedrake, "Lqr-trees: Feedback motion planning on sparse randomized trees," 2009.

[63] J.-C. Latombe, *Robot motion planning*, vol. 124. Springer Science & Business Media, 2012.

[64] J. Kuffner, S. Kagami, K. Nishiwaki, M. Inaba, and H. Inoue, "Online footstep planning for humanoid robots," in *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, vol. 1, pp. 932–937, IEEE, 2003.

[65] J. Chestnutt, M. Lau, G. Cheung, J. Kuffner, J. Hodgins, and T. Kanade, "Footstep planning for the honda asimo humanoid," in *Proceedings of the 2005 IEEE international conference on robotics and automation*, pp. 629–634, IEEE, 2005.

[66] R. Deits and R. Tedrake, "Footstep planning on uneven terrain with mixed-integer convex optimization," in *2014 IEEE-RAS international conference on humanoid robots*, pp. 279–286, IEEE, 2014.

[67] S. Kajita, T. Nagasaki, K. Kaneko, and H. Hirukawa, "Zmp-based biped running control," *IEEE Robotics Automation Magazine*, vol. 14, pp. 63–72, June 2007.

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

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

[70] J. Englsberger, C. Ott, M. A. Roa, A. Albu-Schäffer, and G. Hirzinger, "Bipedal walking control based on capture point dynamics," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4420–4427, 2011.

[71] A. Hereid and A. D. Ames, "FROST: Fast robot optimization and simulation toolkit," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, pp. 719–726, 2017.

[72] J. T. Betts, "Survey of numerical methods for trajectory optimization," *Journal of guidance, control, and dynamics*, vol. 21, no. 2, pp. 193–207, 1998.

[73] A. W. Winkler, C. D. Bellicoso, M. Hutter, and J. Buchli, "Gait and trajectory optimization for legged systems through phase-based end-effector parameterization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1560–1567, 2018.

[74] O. Villarreal, V. Barasuol, P. M. Wensing, D. G. Caldwell, and C. Semini, "Mpc-based controller with terrain insight for dynamic legged locomotion," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2436–2442, IEEE, 2020.

[75] M. Farrokhsiar, G. Pavlik, and H. Najjaran, "An integrated robust probing motion planning and control scheme: A tube-based mpc approach," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1379–1391, 2013.

[76] S. Xin, R. Orsolino, and N. Tsagarakis, "Online relative footstep optimization for legged robots dynamic walking using discrete-time model predictive control," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 513–520, IEEE, 2019.

[77] P. Lin, S. Chen, and C. Liu, "Model predictive control-based trajectory planning for quadrotors with state and input constraints," in *2016 16th International Conference on Control, Automation and Systems (ICCAS)*, pp. 1618–1623, 2016.

[78] M. Nolte, M. Rose, T. Stolte, and M. Maurer, "Model predictive control based trajectory generation for autonomous vehicles—an architectural approach," in *2017 IEEE Intelligent Vehicles Symposium (IV)*, pp. 798–805, IEEE, 2017.

[79] S. Gros, M. Zanon, R. Quirynen, A. Bemporad, and M. Diehl, "From linear to nonlinear mpc: bridging the gap via the real-time iteration," *International Journal of Control*, vol. 93, no. 1, pp. 62–80, 2020.

[80] A. Zheng, "A computationally efficient nonlinear mpc algorithm," in *Proceedings of the 1997 American Control Conference (Cat. No. 97CH36041)*, vol. 3, pp. 1623–1627, IEEE, 1997.

[81] L. Drnach, J. Z. Zhang, and Y. Zhao, "Mediating between contact feasibility and robustness of trajectory optimization through chance complementarity constraints," *Frontiers in Robotics and AI*, vol. 8, 2021.

[82] S.-S. Leu, A.-T. Chen, and C.-H. Yang, "A ga-based fuzzy optimal model for construction time–cost trade-off," *International Journal of Project Management*, vol. 19, no. 1, pp. 47–58, 2001.

[83] R. K. Singh and S. Goswami, "Multi-objective optimization of distributed generation planning using impact indices and trade-off technique," *Electric Power Components and Systems*, vol. 39, no. 11, pp. 1175–1190, 2011.

[84] M. Vukobratović and B. Borovac, "Zero-Moment Point: thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, no. 01, pp. 157–173, 2004.

[85] K. Mitobe, G. Capi, and Y. Nasu, "Control of walking robots based on manipulation of the zero moment point," *Robotica*, vol. 18, no. 6, pp. 651–657, 2000.

[86] S. Kajita, M. Morisawa, K. Harada, K. Kaneko, F. Kanehiro, K. Fujiwara, and H. Hirukawa, "Biped walking pattern generator allowing auxiliary zmp control," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2993–2999, IEEE, 2006.

[87] S. Kajita, T. Nagasaki, K. Kaneko, and H. Hirukawa, "Zmp-based biped running control," *IEEE robotics & automation magazine*, vol. 14, no. 2, pp. 63–72, 2007.

[88] S. Nakaura, M. Sampei, *et al.*, "Balance control analysis of humanoid robot based on zmp feedback control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, pp. 2437–2442, IEEE, 2002.

[89] T. Sugihara, Y. Nakamura, and H. Inoue, "Real-time humanoid motion generation through zmp manipulation based on inverted pendulum control," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, vol. 2, pp. 1404–1409, IEEE, 2002.

[90] S. Caron, A. Escande, L. Lanari, and B. Mallein, "Capturability-based pattern generation for walking with variable height," *IEEE Transactions on Robotics*, vol. 36, no. 2, pp. 517–536, 2019.

[91] S. Caron, A. Escande, L. Lanari, and B. Mallein, "Capturability-based pattern generation for walking with variable height," *IEEE Trans. Rob.*, vol. 36, no. 2, pp. 517–536, 2019.

[92] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, "Hybrid zero dynamics of planar biped walkers," *IEEE trans. on autom. control*, vol. 48, no. 1, pp. 42–56, 2003.

[93] B. Morris and J. W. Grizzle, "Hybrid invariant manifolds in systems with impulse effects with application to periodic locomotion in bipedal robots," *IEEE Trans. Autom. Control*, vol. 54, no. 8, pp. 1751–1764, 2009.

[94] R. W. Sinnet, M. J. Powell, R. P. Shah, and A. D. Ames, "A human-inspired hybrid control approach to bipedal robotic walking," *Proc. of IFAC*, vol. 44, no. 1, pp. 6904–6911, 2011.

[95] A. Hereid, E. A. Cousineau, C. M. Hubicki, and A. D. Ames, "3D dynamic walking with underactuated humanoid robots: A direct collocation framework for optimizing hybrid zero dynamics," in *Proc. of IEEE International Conference on Robotics and Automation*, pp. 1447–1454, 2016.

[96] M. J. Powell, W.-L. Ma, E. R. Ambrose, and A. D. Ames, "Mechanics-based design of underactuated robotic walking gaits: Initial experimental realization," in *Proc. of IEEE-RAS International Conference on Humanoid Robots*, pp. 981–986, 2016.

[97] L. Sentis, *Synthesis and control of whole-body behaviors in humanoid systems*, vol. 68. 2007.

[98] D. Kim, J. Lee, O. Campbell, H. Hwang, and L. Sentis, "Computationally-robust and efficient prioritized whole-body controller with contact constraints," *arXiv preprint arXiv:1807.01222*, 2018.

[99] A. Sherikov, D. Dimitrov, and P.-B. Wieber, "Whole body motion controller with long-term balance constraints," in *2014 IEEE-RAS International Conference on Humanoid Robots*, pp. 444–450, IEEE, 2014.

[100] F. Farshidian, E. Jelavić, A. W. Winkler, and J. Buchli, "Robust whole-body motion control of legged robots," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4589–4596, IEEE, 2017.

[101] M. Hutter, C. Gehring, A. Lauber, F. Gunther, C. D. Bellicoso, V. Tsounis, P. Fankhauser, R. Diethelm, S. Bachmann, M. Blösch, *et al.*, "Anymal-toward legged robots for harsh environments," *Advanced Robotics*, vol. 31, no. 17, pp. 918–931, 2017.

[102] 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*, vol. 3, no. 3, pp. 1458–1465, 2018.

[103] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Science Robotics*, vol. 4, no. 26, p. eaau5872, 2019.

[104] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science robotics*, vol. 5, no. 47, p. eabc5986, 2020.

[105] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "Rma: Rapid motor adaptation for legged robots," *arXiv preprint arXiv:2107.04034*, 2021.

[106] X. Da, Z. Xie, D. Hoeller, B. Boots, A. Anandkumar, Y. Zhu, B. Babich, and A. Garg, "Learning a contact-adaptive controller for robust, efficient legged locomotion," in *Conference on Robot Learning*, pp. 883–894, PMLR, 2021.

[107] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning robust perceptive locomotion for quadrupedal robots in the wild," *Science Robotics*, vol. 7, no. 62, p. eabk2822, 2022.

[108] A. Iqbal, S. Veer, and Y. Gu, "Analytical solution to a time-varying lip model for quadrupedal walking on a vertically oscillating surface," *Mechatronics*, no. NA, 2023.

[109] A. Iqbal, S. Veer, and Y. Gu, "Asymptotic stabilization of aperiodic trajectories of a hybrid-linear inverted pendulum walking on a vertically moving surface," in *Proc. Amer. Contr. Conf.*, pp. 3030–3035, 2023.

[110] A. Iqbal, S. Veer, C. Niezrecki, and Y. Gu, "Ht-lip model based robust control of quadrupedal robot locomotion under unknown vertical ground motion." *In preparation*, 2023.

[111] Econofact, "The role of shipping in world trade." `https://econofact.org/the-role-of-shipping-in-world-trade`. Accessed on 03rd Nov 2023.

[112] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch, *et al.*, "Anymal-a highly mobile and dynamic quadrupedal robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, pp. 38–44, 2016.

[113] M. Raibert, K. Blankespoor, G. Nelson, and R. Playter, "Bigdog, the rough-terrain quadruped robot," *Proc. Int. Federation Autom. Control*, vol. 41, no. 2, pp. 10822–10825, 2008.

[114] S. Galeani, L. Menini, and A. Potini, "Robust trajectory tracking for a class of hybrid systems: An internal model principle approach," *IEEE Transaction on Automatic Control*, vol. 57, no. 2, pp. 344–359, 2011.

[115] J. B. Biemond, N. van de Wouw, W. M. H. Heemels, and H. Nijmeijer, "Tracking control for hybrid systems with state-triggered jumps," *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 876–890, 2012.

[116] F. Forni, A. R. Teel, and L. Zaccarian, "Follow the bouncing ball: Global results on tracking and state estimation with impacts," *IEEE Trans. Autom. Control*, vol. 58, no. 6, pp. 1470–1485, 2013.

[117] Y. Gu, B. Yao, and C. G. Lee, "Straight-line contouring control of fully actuated 3-D bipedal robotic walking," in *Proc. Amer. Contr. Conf.*, pp. 2108–2113, 2018.

[118] W. K. Chan, Y. Gu, and B. Yao, "Optimization of output functions with nonholonomic virtual constraints in underactuated bipedal walking control," in *2018 Annual American Control Conference (ACC)*, pp. 6743–6748, 2018.

[119] Y. Gu, B. Yao, and C. G. Lee, "Bipedal gait recharacterization and walking encoding generalization for stable dynamic walking," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1788–1793, 2016.

[120] Y. Gu, B. Yao, and C. G. Lee, "Time-dependent orbital stabilization of underactuated bipedal walking," in *2017 American Control Conference (ACC)*, pp. 4858–4863, 2017.

[121] Y. Gao, K. Barhydt, C. Niezrecki, and Y. Gu, "Provably stabilizing global-position tracking control for hybrid models of multi-domain bipedal walking via multiple lyapunov analysis," *arXiv preprint arXiv:2304.13943*, 2023.

[122] Y. Gao and Y. Gu, "Global-position tracking control of multi-domain planar bipedal robotic walking," in *Dynamic Systems and Control Conference*, vol. 59148, p. V001T03A009, 2019.

[123] Y. Gao, X. Da, and Y. Gu, "Impact-aware online motion planning for fully-actuated bipedal robot walking," in *2020 American Control Conference (ACC)*, pp. 2100–2105, 2020.

[124] M. A. Sharbafi, M. J. Yazdanpanah, M. N. Ahmadabadi, and A. Seyfarth, "Parallel compliance design for increasing robustness and efficiency in legged locomotion—proof of concept," *IEEE/ASME Transaction on Mechatron.*, vol. 24, no. 4, pp. 1541–1552, 2019.

[125] A. Mazumdar, S. J. Spencer, C. Hobart, J. Salton, M. Quigley, T. Wu, S. Bertrand, J. Pratt, and S. P. Buerger, "Parallel elastic elements improve energy efficiency on the steppr bipedal walking robot," *IEEE/ASME Trans. Mechatron.*, vol. 22, no. 2, pp. 898–908, 2016.

[126] F. Guenther, H. Q. Vu, and F. Iida, "Improving legged robot hopping by using coupling-based series elastic actuation," *IEEE/ASME Trans. Mechatron.*, vol. 24, no. 2, pp. 413–423, 2019.

[127] X. Liu, A. Rossi, and I. Poulakakis, "A switchable parallel elastic actuator and its application to leg design for running robots," *IEEE/ASME Trans. Mechatron.*, vol. 23, no. 6, pp. 2681–2692, 2018.

[128] B. Zhong, S. Zhang, M. Xu, Y. Zhou, T. Fang, and W. Li, "On a CPG-based hexapod robot: AmphiHex-II with variable stiffness legs," *IEEE/ASME Trans. Mechatron.*, vol. 23, no. 2, pp. 542–551, 2018.

[129] C. Li, T. Zhang, and D. I. Goldman, "A terradynamics of legged locomotion on granular media," *Sc.*, vol. 339, no. 6126, pp. 1408–1412, 2013.

[130] J. Aguilar and D. I. Goldman, "Robophysical study of jumping dynamics on granular media," *Nat. Phys.*, vol. 12, no. 3, p. 278, 2016.

[131] C. M. Hubicki, J. J. Aguilar, D. I. Goldman, and A. D. Ames, "Tractable terrain-aware motion planning on granular media: an impulsive jumping study," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3887–3892, 2016.

[132] X. Xiong, A. D. Ames, and D. I. Goldman, "A stability region criterion for flat-footed bipedal walking on deformable granular terrain," in *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4552–4559, 2017.

[133] B. Henze, R. Balachandran, M. Roa-Garzón, C. Ott, and A. Albu-Schäffer, "Passivity analysis and control of humanoid robots on movable ground," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3457–3464, 2018.

[134] Y. Gu, B. Yao, Y. Gao, and C. S. G. Lee, "Asymptotic global-position tracking control of 3-d fully actuated bipedal robotic walking on flat surfaces," *IEEE/ASME Trans. Mechatron. (under review)*.

[135] *Robot Dynamics Lecture Notes*, vol. https://rsl.ethz.ch/education-students/lectures/robotdynamics.html. ETH Zürich: Robotic Systems Lab, 2020.

[136] H. K. Khalil, *Nonlinear systems*. Prentice Hall, 1996.

[137] M. S. Branicky, "Multiple lyapunov functions and other analysis tools for switched and hybrid systems," *IEEE Transactions on Automatic Control*, vol. 43, no. 4, pp. 475–482, 1998.

[138] C. S. Chaney and K. I. Matveev, "Modeling of steady motion and vertical-plane dynamics of a tunnel hull," *Int. J. Naval Architecture Ocean Eng.*, vol. 6, no. 2, pp. 323–332, 2014.

[139] P. Simeonov, *Impulsive differential equations: periodic solutions and applications*. 2017.

[140] E. Coumans, "Bullet physics simulation," in *Proc. ACM SIGGRAPH*, 2015.

[141] "Motek m-gait." https://summitmedsci.co.uk/products/motek-mgait/. Accessed: 2019-11-28.

[142] R. Unitree, "Laikago." http://www.unitree.cc/product/, 2022. Accessed: 2022-07-08.

[143] M. Bloesch, M. Hutter, M. A. Hoepflinger, S. L., C. Gehring, C. D. Remy, and R. Siegwart, "State estimation for legged robots-consistent fusion of leg kinematics and imu," *Robot.*, vol. 17, pp. 17–24, 2013.

[144] R. Hartley, M. Ghaffari, R. M. Eustice, and J. W. Grizzle, "Contact-aided invariant extended kalman filtering for robot state estimation," *Int. J. Robot. Res.*, vol. 39, no. 4, pp. 402–430, 2020.

[145] C.-L. Shih, J. W. Grizzle, and C. Chevallereau, "From stable walking to steering of a 3D bipedal robot with passive point feet," *Rob.*, vol. 30, no. 7, pp. 1119–1130, 2012.

[146] M. S. Motahar, S. Veer, and I. Poulakakis, "Composing limit cycles for motion planning of 3d bipedal walkers," in *Proc. IEEE Conf. Dec. Contr.*, pp. 6368–6374, 2016.

[147] Y. Gao and Y. Gu, "Global-position tracking control of a fully actuated NAO bipedal walking robot," in *Proc. Amer. Contr. Conf.*, pp. 4596–4601, 2019.

[148] Y. Gao, C. Yuan, and Y. Gu, "Invariant filtering for legged humanoid locomotion on dynamic rigid surfaces," *IEEE/ASME Trans. Mechatron.*, 2022, in press.

[149] Y. Zheng and K. Yamane, "Ball walker: A case study of humanoid robot locomotion in non-stationary environments," in *Proc. IEEE Int. Conf. Rob. Autom.*, pp. 2021–2028, 2011.

[150] C. Yang, B. Zhang, J. Zeng, A. Agrawal, and K. Sreenath, "Dynamic legged manipulation of a ball through multi-contact optimization," in *Proc. IEEE Int. Conf. Intel. Rob. Syst.*, pp. 7513–7520, 2020.

[151] F. Asano, "Modeling and control of stable limit cycle walking on floating island," in *Proc. IEEE Int. Conf. Mechatron.*, pp. 1–6, 2021.

[152] P. Kapitza, "Dynamic stability of a pendulum with an oscillating point of support," *Zh. Eksp. Teor. Fiz*, vol. 21, p. 588, 1951.

[153] E. A. Tannuri, J. V. Sparano, A. N. Simos, and J. J. Da Cruz, "Estimating directional wave spectrum based on stationary ship motion measurements," *App. Ocean Res.*, vol. 25, no. 5, pp. 243–261, 2003.

[154] Y. Gong and J. W. Grizzle, "Zero dynamics, pendulum models, and angular momentum in feedback control of bipedal locomotion," *ASME J. Dyn. Sys., Meas., Contr.*, vol. 144, no. 12, p. 121006, 2022.

[155] V. C. Paredes and A. Hereid, "Resolved motion control for 3d underactuated bipedal walking using linear inverted pendulum dynamics and neural adaptation," in *Proc. IEEE Int. Conf. Int. Rob. Sys.*, pp. 6761–6767, 2022.

[156] X. Xiong and A. Ames, "3-D underactuated bipedal walking via H-LIP based gait synthesis and stepping stabilization," *IEEE Trans. Rob.*, vol. 38, no. 4, pp. 2405–2425, 2022.

[157] M. Dai, X. Xiong, and A. D. Ames, "Data-driven step-to-step dynamics based adaptive control for robust and versatile underactuated bipedal robotic walking," *arXiv preprint arXiv:2209.08458*, 2022.

[158] H.-K. Yoon, G.-J. Lee, and D.-K. Lee, "Development of the motion monitoring system of a ship," *J. Navig. Port Res.*, vol. 32, no. 1, pp. 15–22, 2008.

[159] P. M. Gahlinger, "Cabin location and the likelihood of motion sickness in cruise ship passengers," *J. Trav. Med.*, vol. 7, no. 3, pp. 120–124, 2000.

[160] T. B. Benjamin and F. J. Ursell, "The stability of the plane free surface of a liquid in vertical periodic motion," *Proc. Roy. Soc. London.*, vol. 225, no. 1163, pp. 505–515, 1954.

[161] M. Farkas, *Periodic motions*. Springer, 2013.

[162] F. Phelps III and J. Hunter Jr, "An analytical solution of the inverted pendulum," *Amer. J. Phys.*, vol. 33, no. 4, pp. 285–295, 1965.

[163] F. Werth, N. Gheorghe, F. Major, V. Gheorghe, G. Werth, S. Major, and G. Werth, *Charged Particle Traps: Physics and Techniques of Charged Particle Field Confinement*. Springer Ser. Atom., Opt., Plas. Phys., Springer, 2005.

[164] H. Bateman, *Higher transcendental functions [volumes i-iii]*, vol. 1. McGraw-Hill Book Company, 1953.

[165] J. Dougall, "The solution of mathteu's differential equation," *Proc. Edinburgh Math. Soc.*, vol. 34, pp. 176–196, 1915.

[166] G. Floquet, "Sur les équations différentielles linéaires à coefficients périodiques," in *Annales scientifiques de l'École normale supérieure*, vol. 12, pp. 47–88, 1883.

[167] Y. Gu, Y. Gao, B. Yao, and C. G. Lee, "Global-position tracking control for three-dimensional bipedal robots via virtual constraint design and multiple lyapunov analysis," *ASME J. Dyn. Syst., Meas., Contr.*, vol. 144, no. 11, p. 111001, 2022.

[168] A. W. Winkler, C. D. Bellicoso, M. Hutter, and J. Buchli, "Gait and trajectory optimization for legged systems through phase-based end-effector parameterization," *IEEE Rob. Autom. L.*, vol. 3, no. 3, pp. 1560–1567, 2018.

[169] S. Xin, R. Orsolino, and N. Tsagarakis, "Online relative footstep optimization for legged robots dynamic walking using discrete-time model predictive control," in *Proc. IEEE/RSJ Int. Conf. Intel. Rob. Syst.*, pp. 513–520, 2019.

[170] Y. Gu, B. Yao, and C. G. Lee, "Feasible center of mass dynamic manipulability of humanoid robots," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5082–5087, 2015.

[171] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Prog.*, vol. 106, no. 1, pp. 25–57, 2006.

[172] Y. Gao, V. Paredes, A. Hereid, and Y. Gu, "Exponential stabilization of periodic LIP walking on a horizontally moving surface," *Proc. Dyn. Walk. Conf.*, 2022.

[173] R. T. Fawcett, A. Pandala, A. D. Ames, and K. A. Hamed, "Robust stabilization of periodic gaits for quadrupedal locomotion via qp-based virtual constraint controllers," *IEEE Contr. Syst. L.*, vol. 6, pp. 1736–1741, 2021.

[174] A. Iqbal, S. Veer, and Y. Gu, "Drs-lip: Linear inverted pendulum model for legged locomotion on dynamic rigid surfaces," *arXiv preprint arXiv:2202.00151*, 2022.

[175] A. Iqbal and Y. Gu, "Extending capture point to dynamic rigid surfaces," *Dynamic Walking*, 2021.

[176] A. Iqbal, S. Veer, and Y. Gu, "Analytical approximate solution to Mathieu's equation enables real-time motion planning for legged robot walking on a vertically moving surface," in *Dynamic Walking Conference*, 2022.

[177] M. Hutter, C. Gehring, A. Lauber, F. Gunther, C. D. Bellicoso, V. Tsounis, P. Fankhauser, R. Diethelm, S. Bachmann, M. Blösch, *et al.*, "Anymal-toward legged robots for harsh environments," *Advanced Robotics*, vol. 31, no. 17, pp. 918–931, 2017.

[178] K. A. Hamed, J. Kim, and A. Pandala, "Quadrupedal locomotion via event-based predictive control and QP-based virtual constraints," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4463–4470, 2020.

[179] X. Xiong and A. Ames, "3d underactuated bipedal walking via h-lip based gait synthesis and stepping stabilization," *arXiv preprint arXiv:2101.09588*, 2021.

[180] M. Dai, X. Xiong, and A. Ames, "Bipedal walking on constrained footholds: Momentum regulation via vertical com control," in *Proc. IEEE Int. Conf. Rob. Autom*, pp. 10435–10441, 2022.

[181] A. Iqbal, S. Veer, and Y. Gu, "Real-time walking pattern generation of quadrupedal dynamic-surface locomotion based on a linear time-varying pendulum model," *arXiv preprint arXiv:2301.03097*, 2023.

[182] L. Bergdahl, "Wave-induced loads and ship motions," tech. rep., Chalmers University of Technology, 2009.

[183] J. Zhang, K. H. Johansson, J. Lygeros, and S. Sastry, "Zeno hybrid systems," *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 11, no. 5, pp. 435–451, 2001.

[184] X. Da and J. Grizzle, "Combining trajectory optimization, supervised machine learning, and model structure for mitigating the curse of dimensionality in the control of bipedal robots," *The International Journal of Robotics Research*, vol. 38, no. 9, pp. 1063–1097, 2019.

[185] J. Daafouz and J. Bernussou, "Parameter dependent lyapunov functions for discrete time systems with time varying parametric uncertainties," *Syst. Contr. Lett.*, vol. 43, no. 5, pp. 355–359, 2001.

[186] A. McNabb, "Comparison theorems for differential equations," *J. Math. Anal. Appl.*, vol. 119, no. 1-2, pp. 417–428, 1986.

[187] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.

[188] S. V. Rakovic, P. Grieder, M. Kvasnica, D. Q. Mayne, and M. Morari, "Computation of invariant sets for piecewise affine discrete time systems subject to bounded disturbances," in *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, vol. 2, pp. 1418–1423, IEEE, 2004.

[189] L. Magni, G. De Nicolao, R. Scattolini, and F. Allgöwer, "Robust model predictive control for nonlinear discrete-time systems," *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 13, no. 3-4, pp. 229–246, 2003.

[190] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, "Osqp: An operator splitting solver for quadratic programs," *Mathematical Programming Computation*, vol. 12, no. 4, pp. 637–672, 2020.

[191] "Motek, Inc." https://www.motekmedical.com/. Accessed: 2021-08-10.

[192] S. Caron, A. Escande, L. Lanari, and B. Mallein, "Capturability-based pattern generation for walking with variable height," *IEEE Trans. Robot.*, vol. 36, no. 2, pp. 517–536, 2019.

[193] Y. Gao, Y. Gong, V. Paredes, A. Hereid, and Y. Gu, "Time-varying ALIP model and robust foot-placement control for underactuated bipedal robotic walking on a swaying rigid surface," in *2023 American Control Conference (ACC)*, pp. 3282–3287, 2023.

[194] Y. Gao, C. Yuan, and Y. Gu, "Invariant filtering for legged humanoid locomotion on dynamic rigid surfaces," *arXiv e-prints*, pp. arXiv–2201, 2022.

[195] Z. Zhu, S. M. R. Sorkhabadi, Y. Gu, and W. Zhang, "Design and evaluation of an invariant extended Kalman filter for trunk motion estimation with sensor misalignment," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 4, pp. 2158–2167, 2022.

[196] Z. Zhu, S. M. R. Sorkhabadi, Y. Gu, and W. Zhang, "Invariant extended kalman filtering for human motion estimation with imperfect sensor placement," in *2022 American Control Conference (ACC)*, pp. 3012–3018, 2022.

[197] M. Bloesch, M. Hutter, M. A. Hoepflinger, S. Leutenegger, C. Gehring, C. D. Remy, and R. Siegwart, "State estimation for legged robots-consistent fusion of leg kinematics and imu," *Robotics*, vol. 17, pp. 17–24, 2013.

[198] Y. Gao, C. Yuan, and Y. Gu, "Invariant extended kalman filtering for hybrid models of bipedal robot walking," in *Proc. IFAC Mod. Est. Contr. Conf.*, vol. 54, pp. 290–297, 2021.

[199] Y. Gao and Y. Gu, "Hybrid invariant extended kalman filtering for legged locomotion on dynamic rigid surfaces," *Dynamic Walking*, 2021.

[200] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science robotics*, vol. 5, no. 47, 2020.

Appendix A

Appendix: Supplementary Materials of Chapter 2 "Full-Order Modeling and Provably Stabilizing Control Under Periodic Surface Motions"

A.1 Introduction

This document contains supplementary materials for Chapter 2.

A.2 Description of Figures

A.2.1 Additional Validation Results with a Periodically Pitching Rigid Platform

Figures A-1 and A-2 display the validation results obtained from MATLAB, Pybullet, and experiments with the reference gaits (G2) and (G3), respectively, for robot walking on a periodically pitching rigid platform (i.e., the platform (P1)). These figures demonstrate the effectiveness of the proposed control approach similar to Fig. 6 in Section VI-C of the main manuscript.

The PD gains for gait (G2) are chosen as $\mathbf{K}_p = \text{diag}(100, 36, 121, 100, 36, 121, 36, 64, 121)$ and $\mathbf{K}_d = \text{diag}(20, 12, 22, 20, 12, 22, 12, 16, 22)$. The PD gains for gait (G3) are chosen as $\mathbf{K}_p = \text{diag}(100, 49, 121, 100, 49, 121, 36, 64, 121)$ and $\mathbf{K}_d = \text{diag}(20, 14, 22, 20, 14, 22, 12, 16, 22)$.

Specifically, Figures A-1 (a) and A-2 (a) show the joint trajectory tracking results, which demonstrate that reliable trajectory tracking is achieved through both simulations and experiments. Figures A-1 (b) and A-2 (b) show the base roll and pitch trajectories, which both indicate that the robot keeps a relatively steady base pose while walking on the pitching platform. Figures A-1 (c) and A-2 (c) show that simulations and experiments share relatively consistent trends in the joint torque profiles.

A.2.2 Additional Validation Results on Robustness

Figure A-3 displays the controller validation results obtained from MATLAB, Pybullet, and experiments with the reference gait (G2) under the uncertainties (U2). The PD gains are chosen the same as the gait (G2).

Interpretation of the results is given in Chapter 2.6.4 of the main manuscript, which is quoted next:

"With the uncertainties (U2), up to ± 20 % uncertainties, which approximately correspond to a variation of 8 cm in the stance-foot height over 10 gait cycles, are added to the belt speed of the treadmill. The robot's motion was shaky during experiments, but the robot was able to sustain motion for over twenty steps, which indicates that the inherent robustness of the proposed control approach is able to tackle the implemented uncertainties in the treadmill belt speed."

A.2.3 Reference Trajectories for Gait (G1)-G(3)

Reference trajectories used for controller validation are shown in Figs. A-4-A-6. As explained in Section VI-A of the main manuscript, these trajectories are generated using the optimization-based planning method introduced in Section V of the main manuscript.

List of Figures

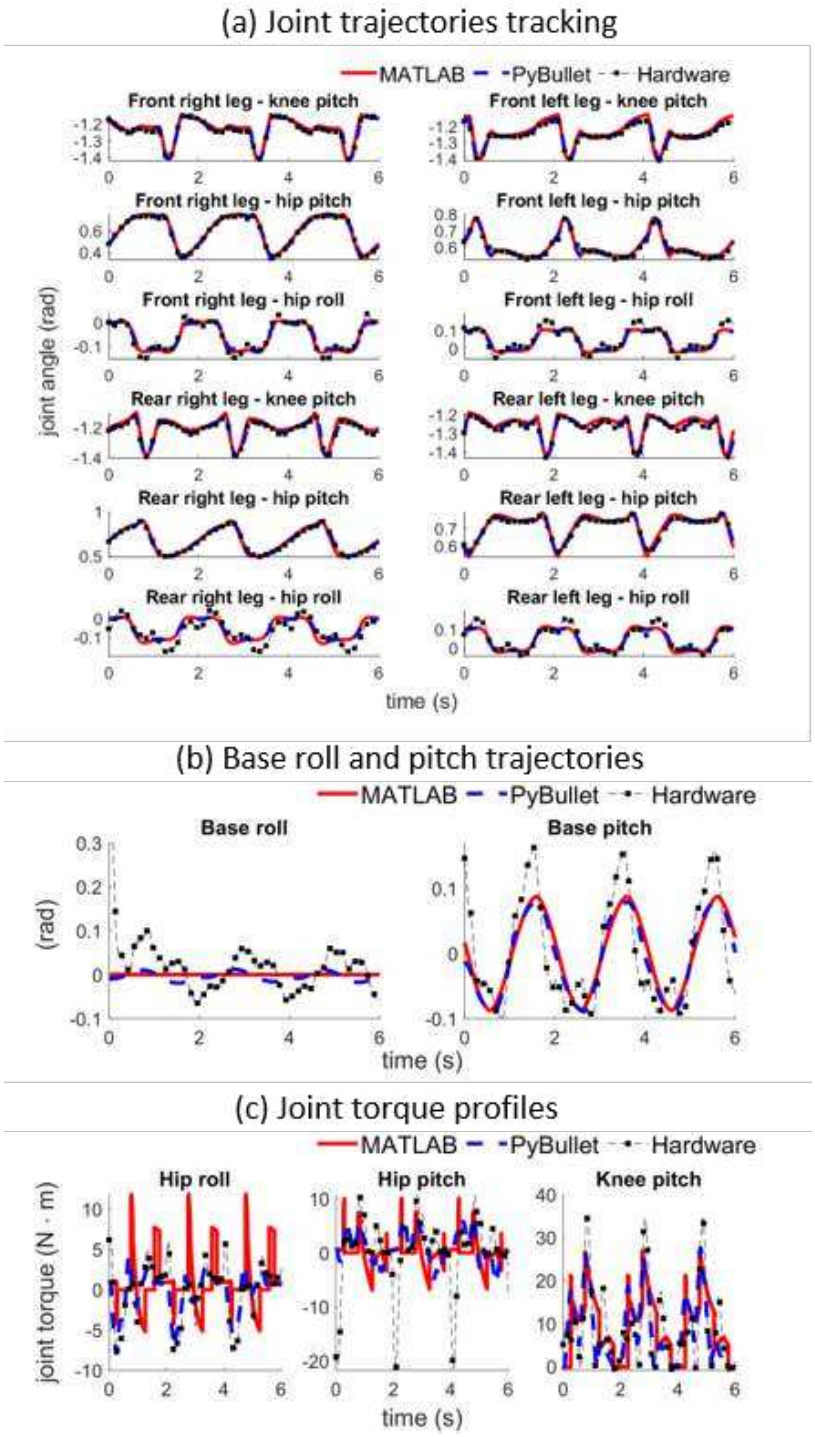

Figure A-1: Trajectory tracking results with gait (G2): (a) joint position trajectories, (b) base roll and pitch trajectories, and (c) joint torque profiles of the rear-left leg.

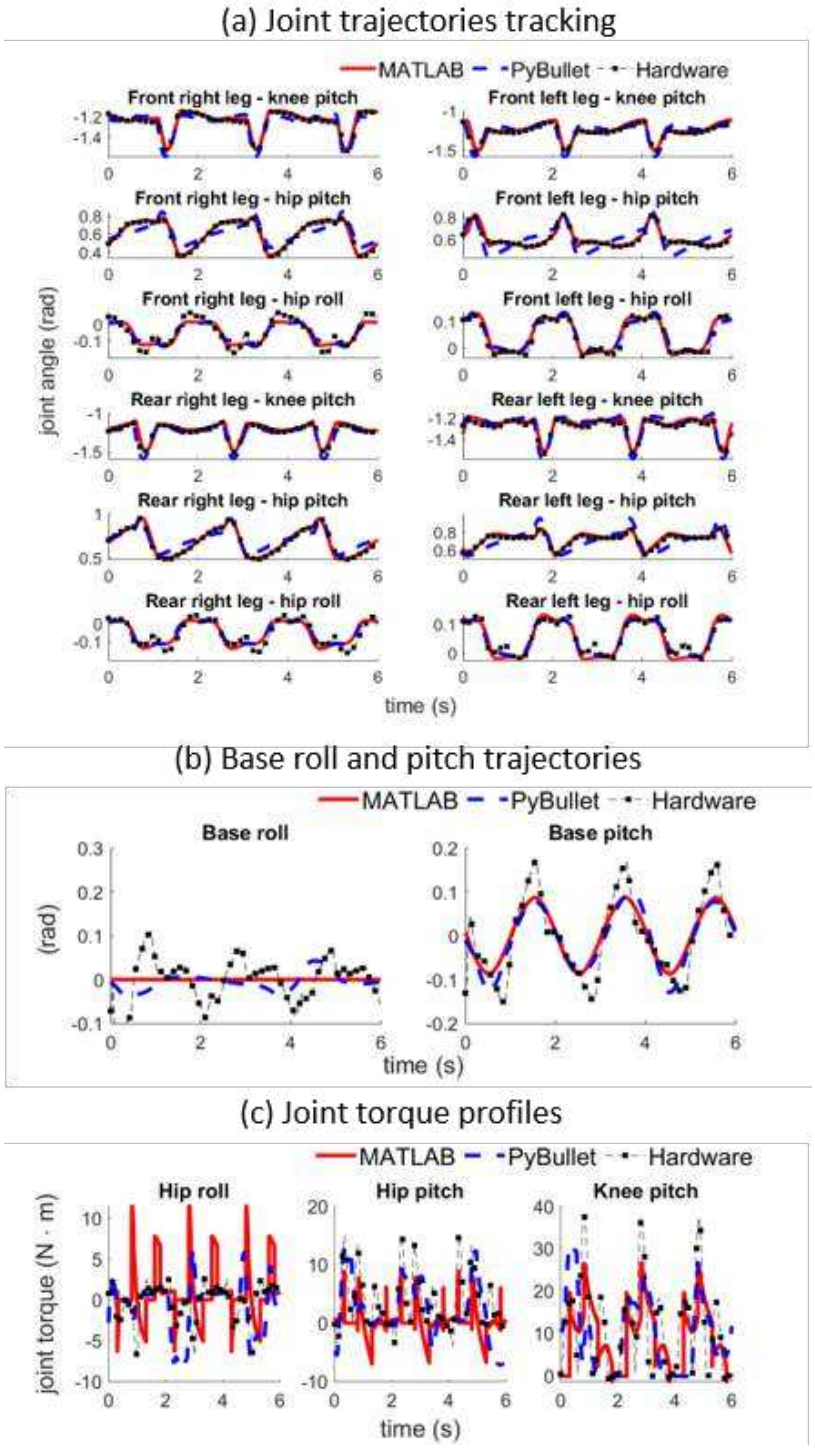

Figure A-2: Trajectory tracking results with gait (G3): (a) joint position trajectories, (b) base roll and pitch trajectories, and (c) joint torque profiles of the rear-left leg.

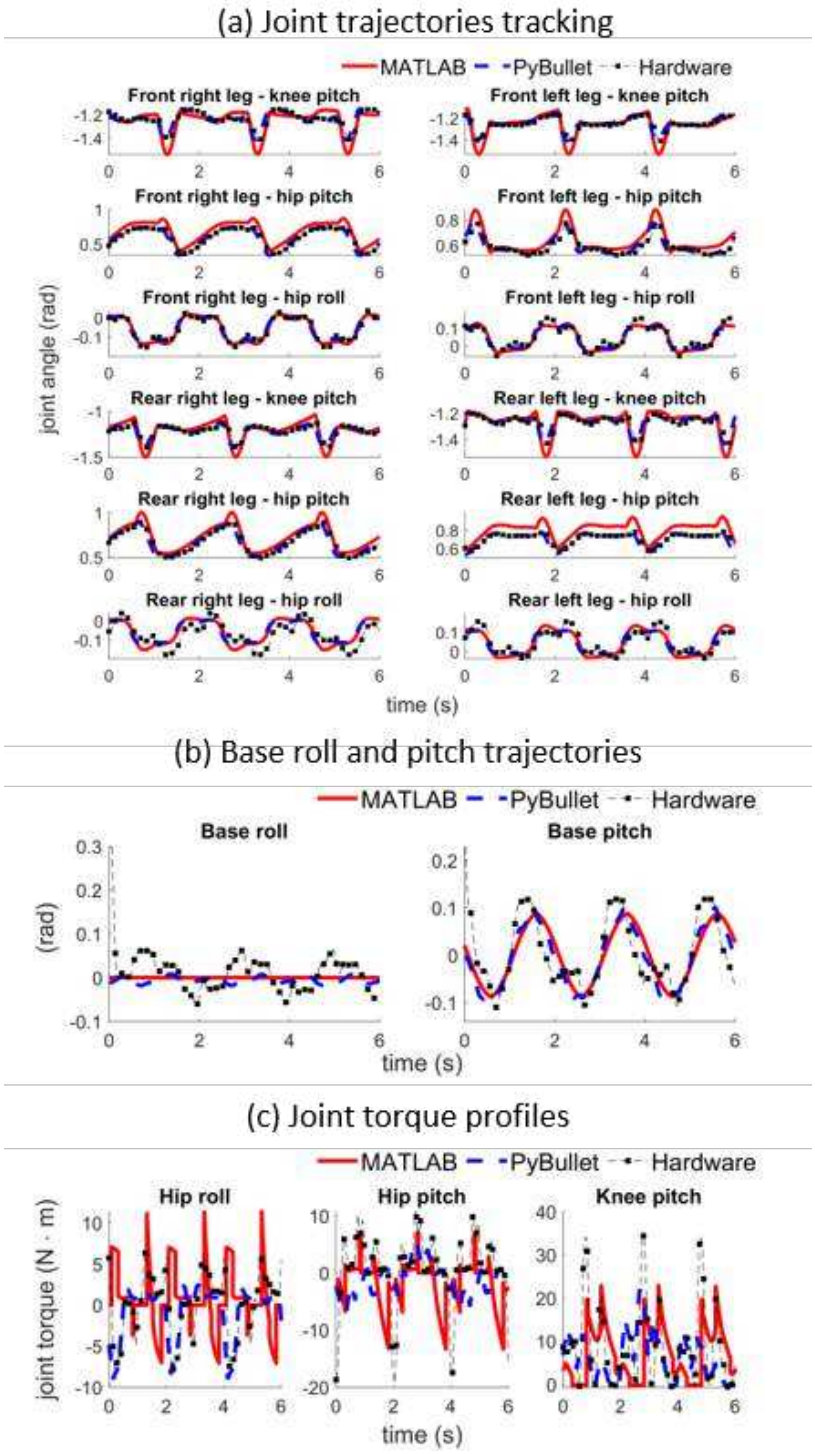

Figure A-3: Trajectory tracking results with gait (G2) under uncertainties (U2): (a) joint position trajectories, (b) base roll and pitch trajectories, and (c) joint torque profiles of the rear-left leg.

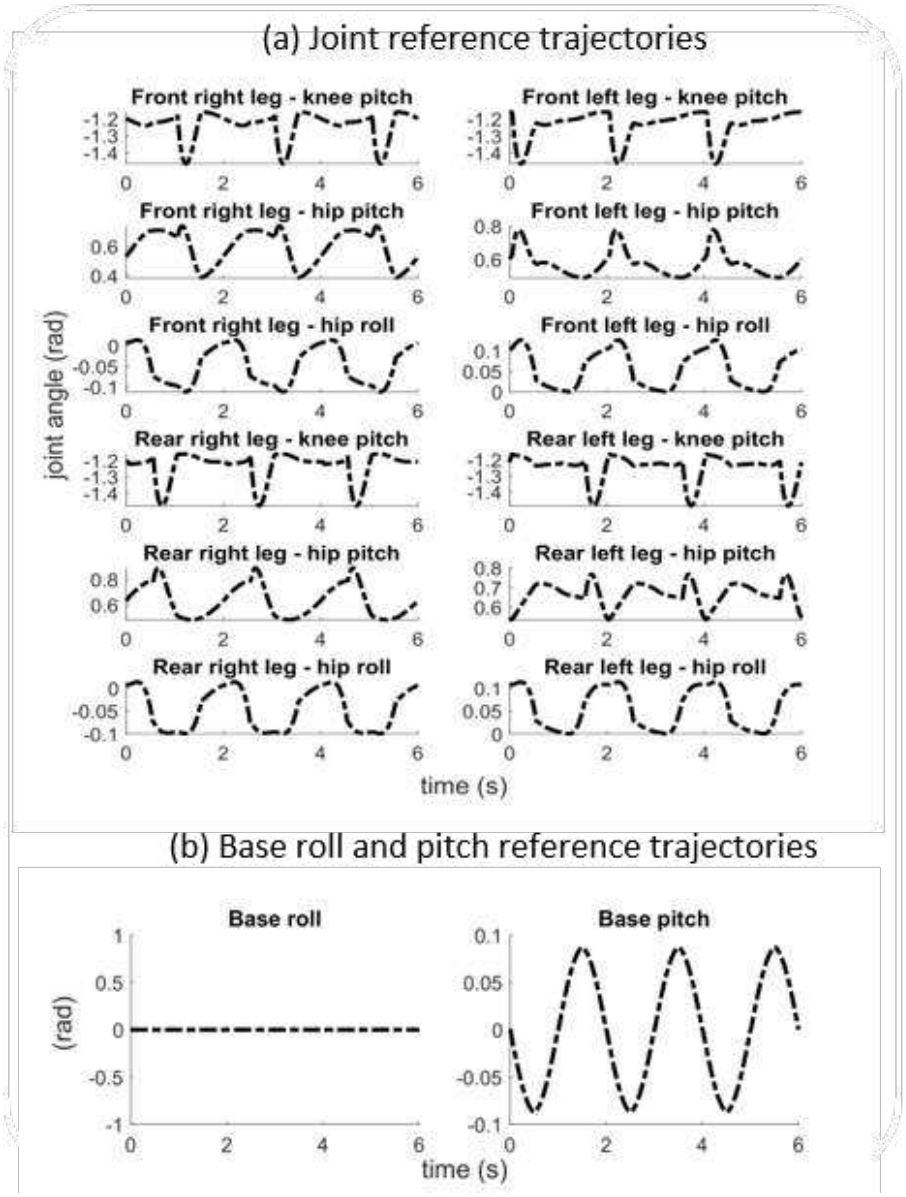

Figure A-4: Reference trajectory for the gait (G1): (a) joint reference trajectories and (b) base roll and pitch reference trajectories.

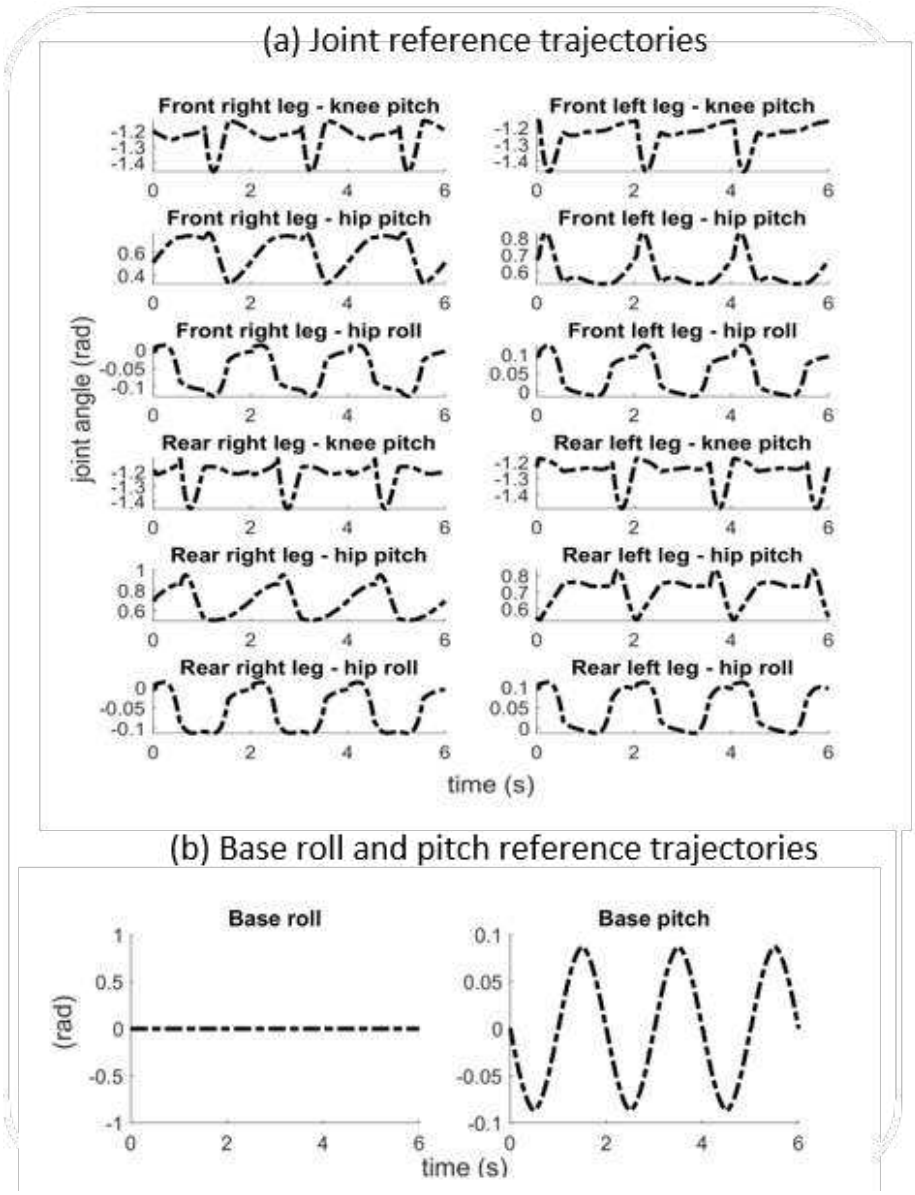

Figure A-5: Reference trajectory for the gait (G2): (a) joint reference trajectories and (b) base roll and pitch reference trajectories.

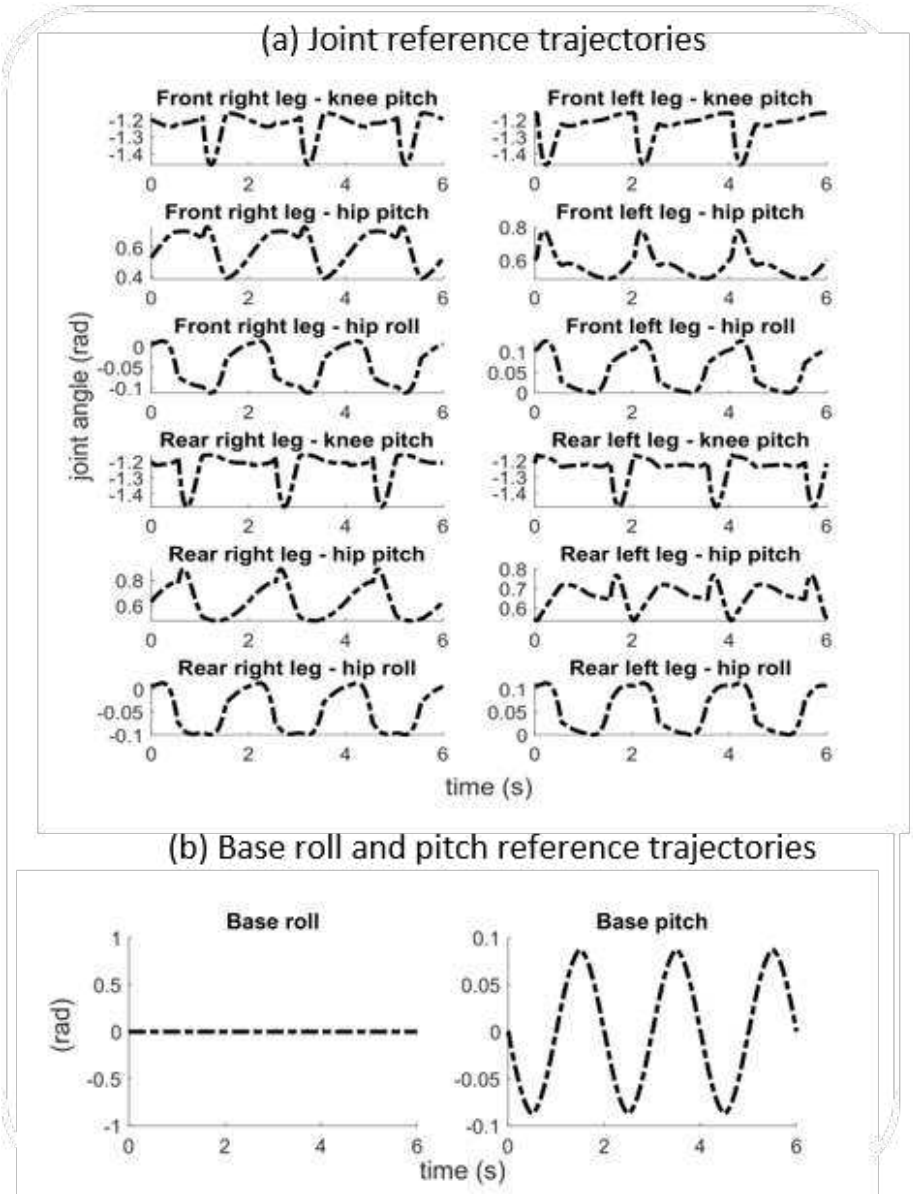

Figure A-6: Reference trajectory for the gait (G3): (a) joint reference trajectories and (b) base roll and pitch reference trajectories.

Appendix B

Author Biography

Amir Iqbal is a Ph.D. candidate in Mechanical Engineering at the University of Massachusetts Lowell and was a Research Intern at Purdue University. He received a B.Tech. degree in Aerospace Engineering from the Indian Institute of Space Science and Technology, Thiruvananthapuram, Kerala, India in 2012. He was a former Scientist/Engineer at the ISRO Satellite Center, Bengaluru, India. His research interests include legged locomotion, control theory, trajectory optimizations, machine learning, and exoskeleton control.