Reduced Order Model Inspired Robotic Bipedal Walking: A Step-to-step Dynamics Approximation based Approach

Thesis by
Xiaobin Xiong

In Partial Fulfillment of the Requirements for the
Degree of
Doctor of Philosophy

CALIFORNIA INSTITUTE OF TECHNOLOGY
Pasadena, California

2021
Defended [May 25th]

ACKNOWLEDGEMENTS

My PhD journey has been quite uncommon. As it draws to a close, I would like to thank many individuals and organizations. First, I would like to thank my advisor Dr. Aaron Ames for his kind support and guidance throughout my PhD study. Working with him has been very productive and enjoyable over the past years. I have had almost all the freedom to explore whatever excites me and interests me in my research since I joined his lab at Caltech in the fall of 2017. I cannot express my thanks enough. I would also like to thank all my thesis committee members: Dr. Joel Burdick in MCE, Dr. Richard Murray in CDS, and Dr. Koushil Screenath from UC Berkeley. They have always been very supportive, friendly, and inspiring.

I would also like to thank a couple of organizations that have supported my PhD. I want to thank the IST of Caltech and Amazon for awarding me the Amazon Fellowship in AI, and thank the IST of Caltech for awarding me the Simoudis Discovery Fund. I also want to thank all the staff and employees of Caltech for making it such a great place to live and work in. I thank Agility Robotics for building the bipedal robot Cassie which became the primary robotic platform I used for my thesis.

I further want to thank lots of labmates and collaborators I met during my PhD. I want to thank Dr. Daniel Goldman, Dr. Patricio Vela, Dr. Alex Chang, Dr. Christian Hubicki, Dr. Yasemin Özkan Aydın, and other folks in the Crab lab at Georgia Tech. I want to thank Tyler Westenbroek, Yuxiao Chen, Reza Ahmadi, Min Dai, Terry Suh, Matthew Powell, Ayonga Hereid, Ugo Rosolia, Thomas Gurriet, Shishir Kolathaya, Andrew Singletary, Maegan Tucker, Andrew Taylor, Jenna Reher, Eric Ambrose, and all the others in the AMBER lab for being great collaborators and labmates. I also want to thank all my friends I met in ATL and LOS for making my journey easier than it would have otherwise been.

Last but not least, I want to thank my family. My parents and my sisters have given me unlimited care and endless support that assisted me to pursue my goal. This thesis is dedicated to them.

ABSTRACT

Controlling bipedal robotic walking is a challenging task. The dynamics is hybrid, nonlinear, high-dimensional, and typically underactuated. Complex physical constraints have to be satisfied in the walking generation. The stability in terms of not-falling is also hard to be encoded in the walking synthesis. Canonical approaches for enabling robotic walking typically rely on large-scale trajectory optimizations for generating optimal periodic behaviors on the full-dimensional model of the system; then the stabilities of the controlled behaviors are analyzed through the numerically derived Poincaré maps. This full-dimensional periodic behavior based synthesis, despite being theoretically rigorous, suffers from several disadvantages. The trajectory optimization problem is computationally challenging to solve. Non-trivial expert-tuning is required on the cost, constraints, and initial conditions to avoid infeasibilities and local optimality. It is cumbersome for realizing non-periodical behaviors, and the synthesized walking can be sensitive to model uncertainties.

In this thesis, we propose an alternative approach of walking synthesis that is based on reduced order modeling and dynamics approximation. We formulate a discrete step-to-step (S2S) dynamics of walking, where the step size is treated as the control input to stabilize the pre-impact horizontal center of mass (COM) state of the robot. Stepping planning thus is converted into a feedback control problem. To effectively and efficiently solve this feedback stepping planning problem, an underactuated Hybrid Linear Inverted Pendulum (H-LIP) model is proposed to approximate the dynamics of underactuated bipedal walking; the linear S2S dynamics of the H-LIP then approximates the robot S2S dynamics. The H-LIP based stepping controller is hence utilized to plan the desired step sizes on the robot to control its pre-impact horizontal COM state. Stable walking behaviors are consequently generating by realizing the desired step size in the output construction and stabilizing the output via optimization-based controllers. We evaluate this approach successfully on several bipedal walking systems with an increase in the system complexity: a planar five-linkage walker AMBER, an actuated version of the Spring Loaded Inverted Pendulum (aSLIP) in both 2D and 3D, and finally the 3D underactuated robot Cassie. The generated dynamic walking behaviors on these systems are shown to be highly versatile and robust. Furthermore, we show that this approach can be effectively extended to realizing more complex walking tasks such as global trajectory tracking and walking on rough terrain.

PUBLISHED CONTENT AND CONTRIBUTIONS

[1] Mohamadreza Ahmadi, **Xiong, Xiaobin**, and Aaron D Ames. Risk-sensitive path planning via cvar barrier functions: Application to bipedal locomotion. In *Arxiv Preprint: https://arxiv.org/abs/2011.01578*, 2020.
X.X. participated in the problem formulation, led the control design and implementation, and participated in the writing of the manuscript.

[2] Min Dai, **Xiong, Xiaobin**, and Aaron Ames. Bipedal walking on constrained footholds: Momentum regulation via vertical com control. In *ArXiv preprint: https://arxiv.org/abs/2104.10367*, 2021.
X.X. participated the problem formulation, the control design, and the writing of the manuscript.

[3] Jonathan R Gosyne, Christian M Hubicki, **Xiong, Xiaobin**, Aaron D Ames, and Daniel I Goldman. Bipedial locomotion up sandy slopes: Systematic experiments using zero moment point methods. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, pages 994–1001. IEEE, 2018. URL `https://ieeexplore.ieee.org/document/8624959`.
X.X. designed and built the experiment apparatus.

[4] HJ Suh, **Xiong, Xiaobin**, Andrew Singletary, Aaron D Ames, and Joel W Burdick. Energy-efficient motion planning for multi-modal hybrid locomotion. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 7027–7033, 2020. doi: 10.1109/IROS45743.2020.9340761. URL `https://ieeexplore.ieee.org/document/9340761`.
X.X. participated the problem formulation, the control design, and the writing of the manuscript.

[5] **Xiong, Xiaobin** and Aaron Ames. Sequential motion planning for bipedal somersault via flywheel slip and momentum transmission with task space control. pages 3510–3517, 2020. doi: 10.1109/IROS45743.2020.9341467. URL `https://ieeexplore.ieee.org/document/9341467`.
X.X. proposed the problem formulation, conducted the control design and implementation, and participated in the writing of the manuscript.

[6] **Xiong, Xiaobin** and Aaron D Ames. Bipedal hopping: Reduced-order model embedding via optimization-based control. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3821–3828. IEEE, 2018. URL `https://ieeexplore.ieee.org/document/8593547`.
X.X. proposed the problem formulation, conducted the control design and implementation, and participated in the writing of the manuscript.

[7] **Xiong, Xiaobin** and Aaron D Ames. Coupling reduced order models via feedback control for 3d underactuated bipedal robotic walking. In *2018 IEEE-RAS*

18th International Conference on Humanoid Robots (Humanoids), pages 1–9. IEEE, 2018. URL `https://ieeexplore.ieee.org/document/8625066`.
X.X. proposed the problem formulation, conducted the control design and implementation, and participated in the writing of the manuscript.

[8] **Xiong, Xiaobin** and Aaron D Ames. Motion decoupling and composition via reduced order model optimization for dynamic humanoid walking with clf-qp based active force control. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1018–1024, 2019. URL `https://ieeexplore.ieee.org/document/8968215`.
X.X. proposed the problem formulation, conducted the control design and implementation, and participated in the writing of the manuscript.

[9] **Xiong, Xiaobin** and Aaron D Ames. Orbit characterization, stabilization and composition on 3d underactuated bipedal walking via hybrid passive linear inverted pendulum model. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4644–4651. IEEE, 2019. URL `https://ieeexplore.ieee.org/document/8968162`.
X.X. proposed the problem formulation, conducted the control design and implementation, and participated in the writing of the manuscript.

[10] **Xiong, Xiaobin** and Aaron D Ames. Dynamic and versatile humanoid walking via embedding 3d actuated slip model with hybrid lip based stepping. *IEEE Rob. and Automation Letters (R-AL)*, 5(4):6286–6293, 2020. URL `https://ieeexplore.ieee.org/document/9158385`.
X.X. proposed the problem formulation, conducted the control design and implementation, and participated in the writing of the manuscript.

[11] **Xiong, Xiaobin** and Aaron D Ames. 3d underactuated bipedal walking via h-lip based gait synthesis and stepping stabilization. In *Submitted to IEEE Trans. On Robotics (T-RO), Arxiv Preprint: https://arxiv.org/abs/2101.09588*, 2021.
X.X. proposed the problem formulation, conducted the control design and implementation, and participated in the writing of the manuscript.

[12] **Xiong, Xiaobin** and Aaron D Ames. Slip walking over rough terrain via h-lip stepping and backstepping-barrier function inspired quadratic program. *IEEE Robotics and Automation Letters (R-AL)*, 6(2):2122–2129, 2021. doi: 10.1109/LRA.2021.3061385. URL `https://ieeexplore.ieee.org/document/9361288`.
X.X. proposed the problem formulation, conducted the control design and implementation, and participated in the writing of the manuscript.

[13] **Xiong, Xiaobin**, Aaron D Ames, and Daniel I Goldman. A stability region criterion for flat-footed bipedal walking on deformable granular terrain. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4552–4559. IEEE, 2017. URL `https://ieeexplore.ieee.`

org/document/8206323.
X.X. proposed the problem formulation, conducted the experiment, the control design and implementation, and participated in the writing of the manuscript.

[14] **Xiong, Xiaobin**, Yuxiao Chen, and Aaron D Ames. Robust disturbance rejection for robotic bipedal walking: System-level-synthesis with step-to-step dynamics approximation. In *submitted to IEEE Conference on Decision and Control (CDC)*, 2021.
X.X. proposed the problem formulation, participated in the control design, realized the implementation, and participated in the writing of the manuscript.

[15] **Xiong, Xiaobin**, Jenna Reher, and Aaron D Ames. Global position control on underactuated bipedal robots: Step-to-step dynamics approximation for step planning. In *International Conference on Robotics and Automation (ICRA), Preprint: https://arxiv.org/abs/2011.06050*, 2021.
X.X. proposed the problem formulation, conducted the control design, and participated in the hardware implementation and the writing of the manuscript.

[16] Tyler Westenbroek, **Xiong, Xiaobin**, Aaron D Ames, and S Shankar Sastry. Optimal control of piecewise-smooth control systems via singular perturbations. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 3046–3053, 2019. URL https://ieeexplore.ieee.org/document/9029232.
X.X. participated in the problem formulation and the control design, led the implementation, and participated in the writing of the manuscript.

[17] Tyler Westenbroek, **Xiong, Xiaobin**, S Shankar Sastry, and Aaron D Ames. Smooth approximations for hybrid optimal control problems with application to robotic walking. In *IFAC Conference on Analysis and Design of Hybrid Systems (ADHS)*, 2021.
X.X. participated in the problem formulation and the control design, led the implementation, and participated in the writing of the manuscript.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

NOMENCLATURE

Control System

$\mathcal{L}_f\mathcal{Y}$	Lie derivative of $\mathcal{Y}$ w.r.t. f.
$\mathcal{Y}$	Output vector.
$\square^*$	Superscript to indicate states/inputs on the periodic solution.
$\square^+$	Superscript to indicate states after the impact event.
$\square^-$	Superscript to indicate states before the impact event.
$f(x)$	Drift vector field.
$g(x)$	Control vector field.
V	Lyapunov function.

Dynamics and Models

β	Leg angle of the aSLIP.
λ	Constant in the dynamics the H-LIP in the SSP.
$\mathbf{E}_o$	Orbital energy.
ϕ	Euler angle of the pelvis.
ψ	Euler angle of the swing foot.
σ	Orbital slope.
$\square^{\text{des}}$	Superscript to indicate desired output quantity.
$\square_{\text{sw}}$	Subscript to indicate the pivot point at foot-ground contact on the swing leg.
$\square_{\text{L,R}}$	Subscript to indicate quantity of the left or right leg.
τ	Motor torques, inputs.
$B_\tau(q)$	Actuation matrix in the EOM.
$C(q,\dot{q})$	Coriolis vector in the EOM.
d	Orbital offset for periodic orbits of the H-LIP.
F	Forces.

$G(q)$	Gravitational vector in the EOM.
$h(q)$	Holonomic constraints.
$J(q)$	Jacobain matrix.
L	Leg length.
$M(q)$	Mass matrix in the EOM.
$q, \dot{q}, \ddot{q}$	Robot coordinates: positions, velocities, accelerations.
T	Time duration of the walking cycle.
T_{DSP}	Time duration of the DSP.
T_{SSP}	Time duration of the SSP.
v^d	Desired net walking velocity.
z_0	Constant height of the H-LIP.

General Constants

α_i	Coefficient in the Bézier polynomial.
$\mathbf{1}_N$	A row vector of size N with all elements being 1.
$\mathbf{g}$	Gravitation vector.
$\mathcal{B}$	Bézier polynomial.
g	Gravitation constant.
K, D	Stiffness and damping of the leg spring.
k, d	Stiffness and damping of the joint spring.
I	Identity matrix.
K_p, K_d	Proportional and derivative gains.
m	Mass.

Step-to-step Dynamics

$\mathbf{e}$	Error state between the horizontal COM state of the robot and the horizontal state of the H-LIP.
$\mathbf{x}$	Pre-impact state of the horizontal COM; discrete state in the S2S dynamics. $\mathbf{y}$ is added to denote the state in $y - z$ plane for 3D walking.

$\mathcal{P}$	Poincaré return map.
A	Linear matrix in the S2S dynamics.
B	Actuation matrix in the S2S dynamics.
E	Error invariant set.
K	Feedback gain vector in the H-LIP based stepping.
p	Position of the horizontal COM w.r.t. the stance foot.
u	Step size; control input in the S2S dynamics.
v	Velocity of the horizontal COM.
w	Dynamics error between the H-LIP and the robot.

LIST OF ACRONYMS

EOM	Equation of Motion
DoF(s)	Degree(s) of Freedom
ROM(s)	Reduced Order Model(s)
ODE	Ordinary Differential Equation
IK	Inverse Kinematics
COM	Center of Mass
COP	Center of Pressure
ZMP	Zero Moment Point
LIP	Linear Inverted Pendulum
H-LIP	Hybrid Linear Inverted Pendulum
P1	Period 1
P2	Period 2
SLIP	Spring Loaded Inverted Pendulum
aSLIP	actuated Spring Loaded Inverted Pendulum
3D-aSLIP	3 Dimensional actuated Spring Loaded Inverted Pendulum
SSP	Single Support Phase
DSP	Double Support Phase
S2S	Step-to-step
SE(2), SE(3)	Special Euclidean Group
w.r.t.	with respective to
RPY	Roll, Pitch, Yaw
IMU	Intertia Measurement Unit
GPC	Global Position Control
HZD	Hybrid Zero Dynamics
MPC	Model Predictive Control

CLF	Control Lyapunov Function
LQR	Linear Quadratic Regulator
CBF	Control Barrier Function
QP	Quadratic Program
CLF-QP	Control Lyapunov Function based Quadratic Program
RES-CLF-QP	Rapidly Exponentially Stabilizing Control Lyapunov Function based Quadratic Program
BBF-QP	Backstepping Barrier Function based Quadratic Program
TSC-QP	Task Space Control based Quadratic Program
PD/PID	Proportional-Derivative/Proportional–Integral–Derivative

Chapter 1

INTRODUCTION

Bipedal robots have existed in the long history of robotics research. The development of bipedal robots is spread over multiple disciplines such as mechanical, electrical, and computer engineering. The overall picture of the research in bipedal robots can be learned in several books such as [47, 101, 120]. The scope of this thesis is on the model-based motion synthesis for bipedal robotic walking. Thus, in this chapter of the introduction, we will briefly introduce bipedal robots with an emphasis on the characteristics that are important for motion synthesis. Then, we will briefly describe the state-of-the-art approaches for solving the motion synthesis problem on bipedal robots. Last, we outline the structure of the following chapters.

1.1 Bipedal Robots

Bipedal robots are built to move under the alternation of two feet that make contact with the ground. The feet are extended by internal actuation via linkages/mechanisms, which are called legs. The two legs are connected by a pelvis or an upper body that may contain a pair of arms. The most commonly seen bipedal robots are humanoids, which are designed to mimic the morphology of a human. Other types of bipedal robots can be simply designed to have two legs. Fig. 1.1 shows some representative examples of the bipedal robots up to date. Their morphology in the design is very diverse.

The robots are designed to move in a general 3D environment. For simplification, some of the early studies had been focused on planarized robots that are constrained on a circular boom so that it moves approximately in a plane. For instance, the two-legged hopper in MIT legged lab [101], MABEL [51], and AMBER [148] are planar (2D) bipedal robots with a fewer number of degrees of freedom (DoFs) compared to 3D bipedal robots. The control approaches that are developed on 2D robots can be extended to general 3D robots. The development of the 2D robots sometimes started at an early stage of design evaluation to facilitate the design of 3D robots.

Another important characteristic is on the foot design. Both the point feet and planar feet are common. The planar feet can be mostly seen on humanoid robots such as

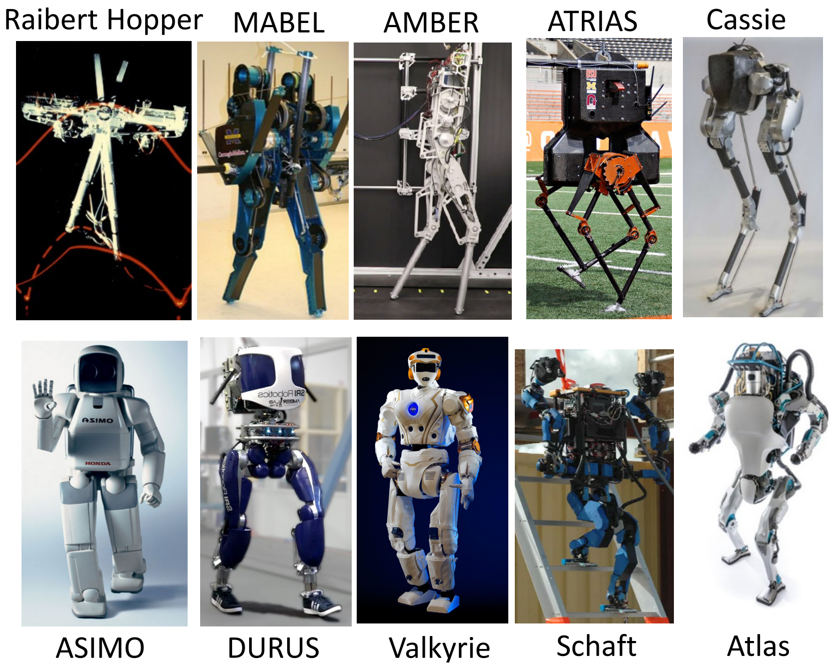

Figure 1.1: Representative examples of bipedal robots.

ASIMO [111], DURUS [104], and ATLAS [49]. Although in some literature [50] humanoids are typically referred to as fully-actuated robots, they are underactuated when the feet are not flatly contacting the ground. For instance, the robot ATLAS is underactuated when it is performing a backflip; the centroidal angular momentum [93] in the flight phase conserves, and the changing rate of the angular momentum is not affected by the internal actuation of the robot. For walking with flat feet (i.e. the feet remain flatly on the contact surface), theoretically speaking, the robots are fully-actuated. However, the robot cannot be effectively controlled like a manipulator since the reaction moment and force at the ground contact are bounded. In other words, controlling the ankle angle to follow certain arbitrary trajectories can break the contact between the foot surface and the ground, which makes the robot no longer a pinned manipulator.

The point (trivial) feet can be seen on the robots such as AMBER and MABEL, which have point contact with the ground during walking and thus have underactuation at contact. The creation of point feet may be confusing as underactuation always challenges the control synthesis. The consideration of using point-foot can be on several folds. Without physical feet, the robot no longer needs to have actuation located at the ankle since there is no ankle, which can significantly decrease the leg inertia and thus easily generate dynamic behaviors. The point-foot nature can also simplify the contact descriptions for walking on rough and uncertain terrain. Last, realizing walking does not require ankle actuation from the perspective of dynamic balancing. It is also important to note that some 3D robots such as ATRIAS have passive nontrivial feet that are used to create contact yaw moment to prevent passive tuning. Robots with point feet cannot stand statically. Thus, some robots such as Cassie have partially actuated feet so that standing behavior can be realized.

Besides the obvious differences in the morphology and structure of the bipedal robots, the components of the internal actuation are also important but beyond the scope of this thesis. By and large, the bipedal robots are electricity-powered, and the individual actuation is realized by an electrical motor with a gear reduction mechanism. A few exceptions are the hydraulic or pneumatic actuated robots such Raibert hopper [101], Petman and Atlas [2] by Boston Dynamics. With design optimization and advanced manufacturing, Atlas has a significant improvement in its power density, range of motion, and inertia distribution compared to the initial version that was deployed in the 2015 DARPA robotics challenge. As in the year 2021, Atlas has been able to perform jumping, backflip, and versatile dancing maneuvers. This level of agility greatly results from the powerful hydraulic actuation. A simple analogy is that a control problem becomes hard as the control input becomes bounded. For electrically actuated robots, the available torques at individual joints are more limited compared to hydraulic actuators. With these being said, the motion synthesis problem remains the same mathematically regardless of the type of actuation on the robot.

Scope of this thesis: The research in bipedal robots is very broad. In this thesis, we will mainly focus on the motion synthesis problem for realizing walking on underactuated bipedal robots. The motion synthesis in canonical terminologies includes motion/trajectory planning on the internal DoFs and feedback control for trajectory tracking. Unlike the planning and control problems on robotic arms or vehicles, the planning on bipedal robotics has to be cognitive about the problem of balancing. In other words, assuming the feedback controller can provide good tracking performance, the motion planning should make sure that the robot does not fall over. This thesis develops an approximation-based stepping stabilization to solve the step planning problem to achieve desired walking behaviors. Moreover, the proposed approach is realized on various walking systems and extended to challenging walking tasks such as global trajectory tracking and walking on rough terrains.

1.2 State-of-the-art Approaches

In this section, we will briefly present some of the state-of-the-art approaches to motion synthesis of bipedal robots. The literature on this topic is very rich, and the development of individual approaches is either mathematically or heuristically deep. Here, we summarize the commonly seen approaches in the following categories. However, it is important to note that lots of approaches sit in the middle or are combinations of some. In-depth explanations should be found in the references themselves.

Zero Moment Point

The Zero Moment Point (ZMP) [64] based approaches [63, 111, 124] develops motion synthesis for humanoid robots with the Linear Inverted Pendulum (LIP) model for planning the desired center of mass (COM) dynamics so that the ZMP remains inside the support convex region of the contact foot or feet in case of double support. The desired COM dynamics are typically realized by inverse kinematics; later developments in [35, 36] apply inverse dynamics based optimization controllers for realization. The condition that the ZMP remains inside the support region is to prevent the foot rotation on the ground so that the humanoid is effectively like a fully actuated robotic manipulation. In other words, the ZMP based approach cares about "foot stability" [64, 127, 145]. There are other conditions in the motion to make sure the ZMP dynamics of the robot is close to the LIP model: the vertical COM position is typically constrained parallel to the ground surface with a constant distance, and the swing foot velocity at impact is designed to be small.

These conditions simplify the motion synthesis problem as the ZMP-LIP model has a linear dynamics, and thus continuous planning can be realized online via Model Predictive Control [29, 136]. The realized walking behaviors however are often times very conservative due to the enforced conditions of the vertical COM and swing foot dynamics. The later development of the ZMP-based approach has heavily focused on the mitigation of the conservative behaviors by the integration of angular momentum based control [65, 89, 114, 122, 136] and foot-placement integration [37]. Despite being widely developed and applied, by and large, the ZMP-based approaches lack a strong mathematical description of the condition that renders the desired walking behaviors on the complex bipedal robots. Additionally, it can not be rigorously applied on underactuated bipedal robots with point feet where the COP and the horizontal COM cannot be continuously controlled.

Reduced Order Models

The LIP model is a reduced order model (ROM) [41, 84]. One important application is to use the passive LIP (no ankle actuation) to calculate the capture point [70, 73, 99, 100, 121] for humanoid robots. The ankle actuation is still required in this framework so that the horizontal COM state of the robot behaves close to that of the LIP in the continuous domain. There are other ROMs developed in the literature to understand and realize locomotion behaviors on high dimensional bipedal robots. The ROM-based approaches attempt to either provide a *template dynamics* behavior [76, 140] for dynamics embedding or an *approximation* to the robot dynamics [98, 138] for motion synthesis. The original intention behind using ROM in the literature can be obscure but typically falls into these two categories or inherently is a combination of the two.

One popular ROM is the spring-loaded inverted pendulum (SLIP) [17, 39, 95, 101, 149] model and its variants. In the pioneering studies in [101], the SLIP was proposed and used to approximate and control the dynamics of several legged robots that could hop and run. It almost simultaneously raised attention in the bio-mechanics research community as it was found to be able to approximate the dynamic walking and running behaviors of animals and humans [44, 116]. The latter research on the SLIP focused on designing and building robots [4, 6, 58] that are like the SLIP, controlling the SLIP and its variants [94, 97, 112, 117, 126], and dynamics embedding of the SLIP (with its controllers) on bipedal robots [42, 76, 106, 118, 131].

The ROM-based approaches are also widely applied as one of the main streams for realizing dynamic behaviors on legged robots. One main evident promise of the ROM is to facilitate fast online planning or reactive/feedback planning on the low-dimensional dynamics model, which can be used on the high dimensional robots. Additionally, the ROMs can provide fundamental understandings about the treatment of underactuation. The thesis builds upon the ideas of ROMs for dynamics approximation. In the latter chapters, details about the principles of the approximation will be revealed in a more rigorous fashion.

Hybrid Zero Dynamics

The hybrid zero dynamics (HZD) [9, 50, 134] based approach builds on a hybrid dynamical model of robotic walking. The hybrid model is a general mathematical description of robotic locomotion behaviors. This thesis also uses this description;

see more details in the Chapter 2. The HZD approach seeks a stable periodic solution to this hybrid dynamic model of the behavior, e.g., walking, and the periodic solution then denotes a periodic motion on the robot. The early development of the HZD [134] focuses on underactuated planar walking, for which a monotonically increasing phase variable (e.g. the stance leg angle) is defined as the internal clock. The desired trajectories of the controllable states (outputs) are parameterized based on the phase variable. The parameters of the desired trajectories thus determine a low-dimensional manifold, and the periodic solution (i.e. gait) can be found as long as it is hybrid-invariant. For planar walking, the stability condition of the gait can be derived; an optimization problem is thus formulated to solve for the parameters to get a stable periodic solution of walking.

The later development of HZD extends beyond underactuated planar bipedal walking. In cases where the robot has non-trivial foot and ankle actuation, partial hybrid zero dynamics (PHZD) [9] is developed to incorporate the ankle actuation for forward velocity tracking. Multi-domain walking [104] has also been realized. For 3D bipedal walking, the monotonically increasing phase variable cannot be defined easily, and stability constraints cannot be derived explicitly. Thus, time-based trajectory parameterization [56, 68] becomes popular in the implementation. The stability of an orbit is checked afterward in simulation by examining the eigenvalues of the linearized Poincaré return map [113, 134]. Modern numerical optimization techniques such as direct collocation [54] and other numerical methods [108] are also developed to efficiently solve the trajectory optimization problem of the HZD.

Another development that is rooted in HZD is on the trajectory tracking, which solves the problem of enforcing the system to evolve on the optimized zero dynamics manifold. The tracking is originally realized via partial feedback linearization [134], which cannot deal with torque bounds and ground contact constraints. To address this issue, control Lyapunov function based quadratic programs (CLF-QP) [10, 138] have been formulated and realized for trajectory tracking. This optimization-based feedback control is very useful for solving the tracking problem subject to practical constraints. More details can be seen in Chapter 2.

The practical implementation of the HZD approaches however challenges its theoretical soundness. The drawbacks include the lack of robustness, intuitive perception of underactuation, and exhaustive processes of numerical optimization. The approach relies on a perfect model of robotic walking, which in real life does not exist. The physical parameters and foot-ground contact cannot be modeled precisely. Although there exist arguments such as input- or parameter-to-state stability [68] around the optimized orbit, such robustness typically is hard to be included in the synthesis process. Additionally, since the stabilization of the underactuation is implicitly checked post-planning (after the trajectory optimization), the underactuation is not directly addressed in terms of active stabilization via re-planning. The numerical trajectory optimization process is also difficult to solve since it is a large non-convex optimization problem. Iterative tuning on the cost function and constraints is required to obtain reasonable solutions. Those drawbacks are exactly the motivation of the work in this thesis. Instead of building upon the HZD framework, we investigate directly on the underactuation, which will be explained in the latter chapters of Approach.

1.3 Thesis Outline

This thesis addresses the problem of motion synthesis for realizing bipedal walking. The developed approach builds on the concept of ROM-based approximations and hybrid dynamics model of robotic bipedal walking. The core component is a formulation of stepping controllers in the trajectory planning. The walking behaviors based on the stepping controllers are generated with high versatility and robustness. Furthermore, the approach is extended for realizing various walking tasks.

The thesis is outlined as follows. Chapter 2 presents the preliminaries on robotic bipedal walking. The latter of the thesis is then followed by three major components: the proposed approach, the applications to several walking systems, and the extensions to various walking tasks. For the proposed approach, Chapter 3 presents the walking generation that is based on the approximation of the step-to-step (S2S) dynamics via the Hybrid Linear Inverted Pendulum (H-LIP), and Chapter 4 then derives the orbit characterization and stabilization on the H-LIP. The approach is then applied to the walking of AMBER in Chapter 5, actuated Spring Loaded Inverted Pendulum (aSLIP) in Chapter 6, and finally the 3D underactuated bipedal robot Cassie in simulation in Chapter 7 and on the hardware in Chapter 8. Moreover, for the extensions, we show that the approach is extended to the global position control in Chapter 9, walking on rough terrain in Chapter 10, and dynamics embedding on fully-actuated humanoids in Chapter 11.

Chapter 2

PRELIMINARIES

In this chapter, we briefly introduce the mathematical modeling of robotic walking and the motion synthesis problem for realizing dynamic walking behaviors on the robot. In the modeling part, we will also introduce the underactuation on bipedal walking and the step-to-step (S2S) dynamics of walking, both of which are important concepts for realizing robotic bipedal walking. The motion synthesis problem includes the description of walking behaviors via output trajectories and the feedback controllers that stabilize the output trajectories.

2.1 Hybrid Dynamic Model of Walking

To describe a mathematical model of the walking of a bipedal robot, we first need a model of the robot. The bipedal robots are typically designed with rigid linkages connected by rotational or prismatic joints. Thus, similar to the model of robotic manipulators [80, 88, 120], a rigid-body multi-linkage system is used to describe the dynamics of the robot. Each rigid body represents each linkage on the robot. Each joint constrains the motion between two linkages and yields one DoF. In general, we use q to describe the configuration of a bipedal robot, then $q \in SE(3) \times \mathbb{R}^n$, where n is the number of the robot joints.

The robot is actuated by forces/torques from the actuation placed at the joints. Then, a second order dynamics is derived to describe the evolution of the robot:

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) = B_\tau \tau + J_h(q)^T F_h, \tag{2.1}$$

where $M(q)$ is the mass matrix, $C(q, \dot{q})$ contains the Coriolis and centrifugal forces, $G(q)$ is the gravitational vector, B_τ is the actuation matrix, τ is the actuation vector, $J_h(q)$ represents the Jacobian of the holonomic constraints, and F_h contains the holonomic forces. For conciseness, the dependencies on $q, \dot{q}$ may be dropped in the latter.

For robotic walking, the holonomic constraints $h(q) = 0$ are typically on the foot-ground contact, which yields a second order dynamic constraint on the system:

$$J_h \ddot{q} + \dot{J}_h \dot{q} = 0, \tag{2.2}$$

where $J_h = \frac{\partial h(q)}{\partial q}$. Solving $\ddot{q}$ from Eq. (2.1) and plugging it into Eq. (2.2) yields

$$F_h = A_h \tau + b_h, \tag{2.3}$$

$$A_h := -J_M^{-1} J_h^{-1} B_\tau, \tag{2.4}$$

$$b_h := J_M^{-1}(J_h M^{-1}(G + C) - \dot{J}_h \dot{q}), \tag{2.5}$$

where $J_M = J_h M^{-1} J_h^T$. Plugging Eq. (2.3) into Eq. (2.1) yields

$$\ddot{q} = M^{-1}(B_\tau + J_h^T A_h)\tau + M^{-1}(J_h^T b_h - C - G). \tag{2.6}$$

Taking $x = [q^T, \dot{q}^T]^T$ as the state of the robot, its dynamics becomes

$$\dot{x} = f(x) + g(x)\tau, \tag{2.7}$$

where

$$f(x) = \begin{bmatrix} \dot{q} \\ M^{-1}(J_h^T b_h - C - G) \end{bmatrix}, \quad g(x) = \begin{bmatrix} 0 \\ M^{-1}(B_\tau + J_h^T A_h) \end{bmatrix}. \tag{2.8}$$

Eq. (2.7) represents the continuous dynamics of the robot. Based on the number of feet that contact the ground and how they contact the ground, the number of the holonomic constraints and thus the continuous dynamics change. In this thesis, we mainly deal with flat foot-ground contact and point foot-ground contact. When two feet both contact the ground, the robot is in a double support phase (DSP). When only one foot contacts the ground, the robot is in a single support phase (SSP). The transition from the SSP to the DSP happens when the swing foot strikes the ground. The impact between the foot and the ground is assumed to be plastic [50, 134], where the velocity of the foot instantaneously becomes zero. The discrete event on the system velocity is denoted by

$$\dot{q}^+ = (I - M^{-1}J_h^T(J_h M J_h^T)^{-1} J_h)\dot{q}^-, \tag{2.9}$$

where I is the identity matrix, and the superscripts $^-$ and $^+$ represent the instants before and after the transition, respectively. The transition from the DSP to the SSP happens when one of the stance feet lifts off, for which the state does not have any jumps. In general, we denote the transitions between domains (phases) as

$$x^+ = \Delta_{k \to k+1}(x^-), \tag{2.10}$$

where k is the index of the domain. As a result, the walking of the robot is modeled by a hybrid dynamical system [9, 50]. The hybrid system is composed of continuous

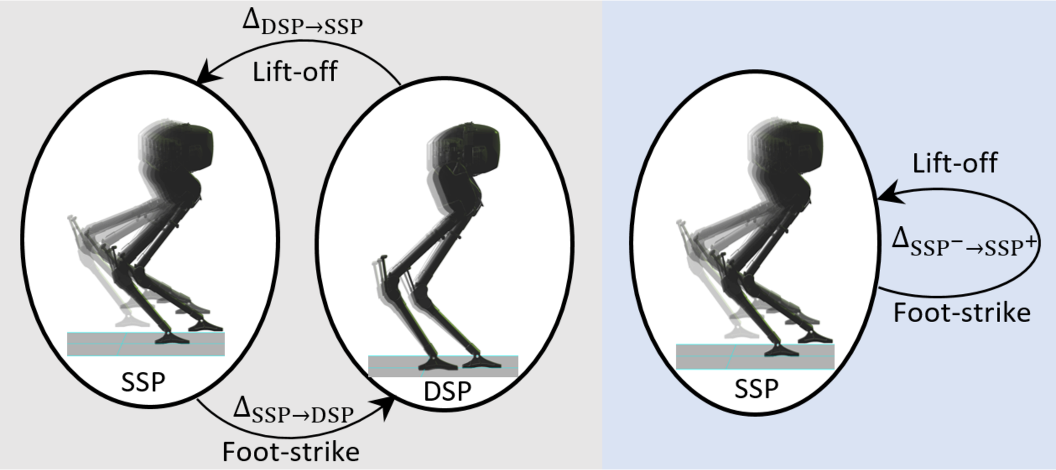
$\Delta_{\mathrm{DSP}\to\mathrm{SSP}}$
Lift-off
SSP
DSP
$\Delta_{\mathrm{SSP}\to\mathrm{DSP}}$
Foot-strike
Lift-off
$\Delta_{\mathrm{SSP}^-\to\mathrm{SSP}^+}$
Foot-strike
SSP

Figure 2.1: The hybrid graphs of one-domain walking and two-domain walking.

dynamics in different domains and discrete transitions between the domains. The typical structure of the hybrid model of bipedal walking is shown in Fig. 2.1. For walking without the DSP, which typically happens on robots without evident compliance in the leg, the system directly goes into another SSP after the impact. We refer to this walking as one-domain walking. The walking with the DSP is referred to as two-domain walking. The one-domain walking can be viewed as a special case of a two-domain walking with an instantaneous DSP. Thus, the two-domain walking will be chosen as the general model that is studied for walking in Chapter 3. The developed approach then can be applied to both the one-domain walking such as the walking of AMBER in Chapter 5 and the two-domain walking such as the walking of Cassie in Chapter 7.

Foot Underactuation

Underactuation [123] on bipedal robots can happen in the internal DoFs of the robot or its global DoFs (SE(3)) in different locomotion behaviors. The internal underactuation typically happens when the robot has compliance elements. For instance, the robot Cassie has springs in its legs. By using a DoF to approximately model each spring deflection, the leg becomes underactuated at the spring joint. Details can be seen in Chapter 7. The SLIP model also has internal underactuation, since the spring creates additional DoF that cannot be directly controlled.

The most commonly seen underactuation on the bipedal robots is on the global DoFs. For instance, during running when the robot is in the flight phase having no

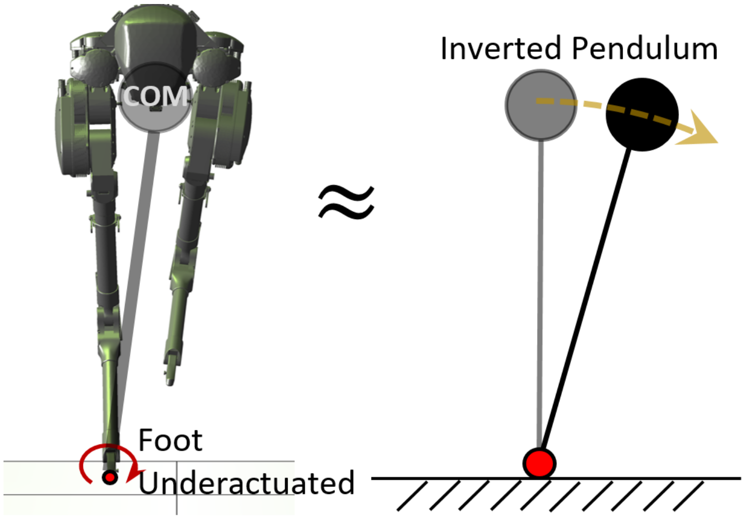

Figure 2.2: The foot-underactuation of bipedal walking with the illustration by an inverted pendulum.

contact with the ground, the centroidal dynamics [93] are underactuated, i.e., the centroidal dynamics is not affected by the internal joint actuation. The centroidal dynamics include the translational COM dynamics and centroidal angular momentum dynamics. For bipedal walking behaviors, the global underactuation typically happens at the foot-ground contact. As being discussed in Chapter 1, the foot may be designed without actuation to rotate (for reducing the leg inertia), or the foot contacts the ground with its edges. Both can be viewed as the actuation is missing at the contact location. This is in contrast to fully-actuated humanoid walking, where the foot rotation typically remains actuated from the control synthesis, e.g., the ZMP based approach [64].

The foot underactuation prevents the direct and continuous control on the underactuated DoFs at the contact, which in practice is effectively the center of mass (COM) of the robot in the horizontal plane. A simple illustration is that an inverted pendulum would roll passively without any actuation at the contact with the ground (see Fig. 2.2). Theoretically speaking, the robot has rotational linkages, which can generate rotational momentum and thus indirectly affect the COM [65, 100, 144], e.g., the angle of a flywheel-inverted pendulum can be controlled via the continuous rotation of the flywheel [92]. However, the joints on robots typically have limited ranges of motion, limited control bandwidth, and limited torques in practice. Thus, it is not possible to purely depend on the angular momentum to continuously control the COM. Therefore, the horizontal COM state of the underactuated bipedal robot is equivalently the underactuated ("weakly actuated" [25]) states in practice. The underactuation creates challenges in stabilizing the horizontal COM dynamics in the continuous domain, especially in the SSP. As a consequence, to create stable walking behaviors with this underactuation, the hybrid dynamics structure has to be utilized, which yields the following step-to-step dynamics formulation for motion synthesis.

Step-to-step Dynamics

The hybrid dynamical system is a comprehensive mathematical description of bipedal walking behaviors. It is however complex and unintuitive to work with for motion synthesis to address the underactuation. As we discussed in Chapter 1, HZD based framework utilizes trajectory optimization to synthesize individual periodic orbit on the complex hybrid dynamical system, and the orbit stability is typically checked via the Poincaré return map numerically. This is an indirect approach to passively stabilize the underactuation when generating the desired walking behaviors. In this thesis, we formulate a discrete step-to-step (S2S) dynamics of walking, equivalently the Poincaré return map at the pre-impact instant. The trajectory will be directly synthesized based on the S2S dynamics to directly stabilize the underactuation on the horizontal COM state. Here, we first introduce the S2S dynamics mathematically.

Assuming the robot is controlled to periodically lift-off and touch-down the foot, the hybrid dynamics of the robot repeats a walking cycle. This can be realized by creating a periodic trajectory of the swing foot and then realizing the trajectory tracking via feedback control. The motion does not need to be strictly periodical w.r.t. time. Let $\{q^-, \dot{q}^-\}$ be the pre-impact state of the robot. The evolution of the pre-impact state at the step level, i.e., the S2S dynamics of the robot, can be represented by

$$\{q^-_{k+1}, \dot{q}^-_{k+1}\} = \mathcal{P}(q^-_k, \dot{q}^-_k, \tau(t)), \tag{2.11}$$

where k is the index of the step, and $\tau(t)$ represents the torques that are applied during the step k. Each step starts with a DSP (if exists) and a following SSP.

Let $\mathbf{x}^R = [p^R, v^R]^T$ be the horizontal COM state of the pre-impact state of the robot in the sagittal plane. p^R, v^R are the horizontal position and velocity of the COM of the robot, which are functions of the pre-impact state $\{q^-, \dot{q}^-\}$. Thus, the S2S dynamics of the horizontal COM state can be represented by

$$\mathbf{x}^R_{k+1} = \mathcal{P}_{\mathbf{x}}(q^-_k, \dot{q}^-_k, \tau(t)). \tag{2.12}$$

In the latter part of this thesis, Eq. (2.12) will be directly referred to as the S2S dynamics of the bipedal walking robot. It is important to note that due to the nonlinear and hybrid dynamics of the robot, the exact expression of the S2S dynamics can not be computed in closed-form. Thus, synthesizing the controller based on the S2S dynamics is difficult in general. As a consequence, we propose a S2S dynamics approximation that is based on a simple model in this thesis to facilitate online and robust planning, which will be elaborated in the next chapter.

Remark. *The notation of the S2S dynamics in legged locomotion is an adaptation of the Poincaré return map [66] in nonlinear dynamics, which has been widely used to check the stability of a periodic solution [50]. The S2S dynamics [15] has appeared frequently in controlling SLIP running [16, 43]. By investigating the evolution of the apex states, the S2S dynamics/return map of running can be easily obtained on the SLIP. Feedback controllers thus can be synthesized based on the S2S dynamics to stabilize the running of the SLIP. However, the S2S dynamics of the walking of the bipedal robotic systems has not been shown to be obtained easily, which is one of the motivations of the proposed approach for approximating the S2S dynamics.*

2.2 Optimization-based Feedback Controller

To realize walking behaviors, one typically considers the problem of motion synthesis as two components: motion planning described by the desired trajectories and trajectory tracking via feedback control. As being discussed previously, the underactuation has to be addressed at the planning layer, which is the main focus of this thesis. In this section, we will present the optimized-based feedback controllers which can solve the trajectory tracking problem on bipedal walking robots. Before jumping into that, we first elaborate on the motion planning part.

Assuming the walking is defined by trajectories of its DoFs, one has to define the desired trajectories for all the DoFs with actuation in each domain, which are defined as the outputs in nonlinear control terminology [66, 113]. Taking the five-linkage walker [134] as an example, the robot has four actuators and five DoFs when one of the feet contacts the ground. Thus, we need to define four trajectories on the four actuated DoFs or four independent functions of the actuated DoFs. In the HZD approach [134], the desired trajectories of the four actuated DoFs are defined via Bézier polynomial functions with the underactuated DoF being the phase variable. One can also select a combination of features that are functions of the actuated DoFs. A straight-forward kinematic combination is on the vertical height of the

COM, torso angle, and the vertical and horizontal position of the swing foot. Here, the generation of the desired trajectories of the outputs is called motion planning (trajectory generation). Then, the problem of feedback control (trajectory tracking) is to drive the actual outputs to follow the desired output trajectories. The trajectory tracking is a canonical nonlinear control problem. The challenges on bipedal walking are on the satisfaction of the stringent physical constraints, which can be solved via the following optimization-based controllers.

Control Lyapunov Function based Quadratic Programs (CLF-QP)

We first present the control Lyapunov function based quadratic programs (CLF-QP) [10, 86] approach for output stabilization. For general purposes, we assume that the selected outputs include outputs with both relative degree 1 ($\mathcal{RD}_1$) and degree 2 ($\mathcal{RD}_2$) [66]. Let $\mathcal{RD}_1$ outputs be represented by $\mathcal{Y}_1 \in \mathbb{R}^{o_1}$ and $\mathcal{RD}_2$ outputs represented by $\mathcal{Y}_2 \in \mathbb{R}^{o_2}$ with o_1 and o_2 being their dimensions. Without loss of generality, we assume that the desired motion is of time-based trajectories; thus the outputs can be defined as follows [7] [66]:

$$\mathcal{Y}_1(q, \dot{q}, t) = \dot{\mathcal{Y}}_1^a(q, \dot{q}) - \mathcal{Y}_1^d(t), \tag{2.13}$$

$$\mathcal{Y}_2(q, t) = \mathcal{Y}_2^a(q) - \mathcal{Y}_2^d(t), \tag{2.14}$$

where the superscript a denotes the actual and d denotes the desired. The objective of the control is to drive $\mathcal{Y}_1 \to 0$ and $\mathcal{Y}_2 \to 0$. Differentiating $\mathcal{Y}_1$ once and $\mathcal{Y}_2$ twice yields the affine control system on the output dynamics:

$$\begin{bmatrix} \dot{\mathcal{Y}}_1 \\ \ddot{\mathcal{Y}}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} \mathcal{L}_f \mathcal{Y}_1(q, \dot{q}) - \dot{\mathcal{Y}}_1^d \\ \mathcal{L}_f^2 \mathcal{Y}_2(q, \dot{q}) - \ddot{\mathcal{Y}}_2^d \end{bmatrix}}_{\mathcal{L}_\mathbf{f}} + \underbrace{\begin{bmatrix} \mathcal{L}_g \mathcal{Y}_1(q, \dot{q}) \\ \mathcal{L}_g \mathcal{L}_f \mathcal{Y}_2(q, \dot{q}) \end{bmatrix}}_{\mathcal{A}} \tau, \tag{2.15}$$

where $\mathcal{A}$ is the decoupling matrix, $\mathcal{L}$ denotes the Lie derivative and $\tau \in \mathbb{R}^m$ is the control input in the EOM in Eq. (2.1). The dependency on t is dropped from here to simplify the notation. In case when τ can be found to satisfy the following equality:

$$\mathcal{A}\tau = -\mathcal{L}_\mathbf{f} + \mu, \tag{2.16}$$

the output dynamics becomes this linear control system:

$$\dot{\eta} = \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & I_2 \\ 0 & 0 & 0 \end{bmatrix}}_{F} \eta + \underbrace{\begin{bmatrix} I_1 & 0 \\ 0 & 0 \\ 0 & I_2 \end{bmatrix}}_{G} \mu, \tag{2.17}$$

where $\eta = \left[\mathcal{Y}_1^T, \mathcal{Y}_2^T, \dot{\mathcal{Y}}_2^T\right]^T$, I_1 and I_2 are identity matrices with dimension o_1 and o_2 respectively, and μ is the auxiliary control input [66]. One can choose μ to exponentially stabilize the linear system. For example, choosing [8]

$$\mu = \begin{bmatrix} -\epsilon \mathcal{Y}_1 \\ -2\epsilon\dot{\mathcal{Y}}_2 - \epsilon^2 \mathcal{Y}_2 \end{bmatrix}, \tag{2.18}$$

results in the linear output dynamics:

$$\begin{bmatrix} \dot{\mathcal{Y}}_1 \\ \dot{\mathcal{Y}}_2 \\ \ddot{\mathcal{Y}}_2 \end{bmatrix} = \begin{bmatrix} -\epsilon I_1 & 0 & 0 \\ 0 & 0 & I_2 \\ 0 & -\epsilon^2 I_2 & -2\epsilon I_2 \end{bmatrix} \begin{bmatrix} \mathcal{Y}_1 \\ \mathcal{Y}_2 \\ \dot{\mathcal{Y}}_2 \end{bmatrix}, \tag{2.19}$$

which is exponentially stable when $\epsilon > 0$. However, such μ does not utilize the natural dynamics of the system and oftentimes may not be realizable on the robotic system if there are stringent physical constraints (e.g., torque bounds) that must be enforced.

The above construction motivates constructing rapidly exponentially stabilizing control Lyapunov functions (RES-CLF) [10] from continuous time algebraic Riccati equations (CARE)[1] or continuous time Lyapunov equations (CTLE)[2] to stabilize the output dynamics exponentially at a chosen rate ε. Given a solution $P = P^T > 0$ to CTLE or CARE with $Q = Q^T > 0$, the Lyapunov function is constructed as:

$$V_\varepsilon(\eta) = \eta^T \underbrace{I_\varepsilon P I_\varepsilon}_{P_\varepsilon} \eta, \tag{2.20}$$

where $I_\varepsilon = \mathrm{diag}(I_1, \frac{1}{\varepsilon} I_2, I_2)$. The goal of exponential stabilizing $\eta \longrightarrow 0$ is encoded by the condition:

$$\dot{V}_\varepsilon(\eta) \leq -\frac{\gamma}{\varepsilon} V_\varepsilon(\eta), \tag{2.21}$$

with some $\gamma > 0$, where

$$\dot{V}_\varepsilon(\eta) = \mathcal{L}_F V_\varepsilon(\eta) + \mathcal{L}_G V_\varepsilon(\eta)\mu, \tag{2.22}$$

$$\mathcal{L}_F V_\varepsilon(\eta) = \eta^T (F^T P_\varepsilon + P_\varepsilon F)\eta, \tag{2.23}$$

$$\mathcal{L}_G V_\varepsilon(\eta) = 2\eta^T P_\varepsilon G. \tag{2.24}$$

Eq. (2.21) and (2.22) indicate an inequality constraint on μ to achieve exponential stability. This naturally leads to the formulation of quadratic program (QP) to find

[1] $F^T P + PF - PGG^T P + Q = 0$.

[2] $F^T P + PF + Q = 0$.

μ to minimize the quadratic cost $\mu^T \mu$. With (2.16), the cost and constraint of the QP can be transformed back onto the original control input τ by noting that:

$$\mu^T \mu = \tau^T \mathcal{A}^T \mathcal{A} \tau + 2 \mathcal{L}_\mathbf{f}{}^T \mathcal{A} \tau + \mathcal{L}_\mathbf{f}{}^T \mathcal{L}_\mathbf{f}, \tag{2.25}$$

and the inequality from Eq. (2.21) and (2.22) becomes

$$\mathcal{L}_F V_\varepsilon(\eta) + \mathcal{L}_G V_\varepsilon(\eta) \mathcal{L}_\mathbf{f} + \mathcal{L}_G V_\varepsilon(\eta) \mathcal{A} \tau \leq -\frac{\gamma}{\varepsilon} V_\varepsilon(\eta). \tag{2.26}$$

Now the QP can be formulated in terms of solving for τ at a current state $(q, \dot{q})$ as follows:

$$\begin{aligned} \tau^* = \underset{\tau \in \mathbb{R}^m}{\operatorname{argmin}} \ & \tau^T \mathcal{A}^T \mathcal{A} \tau + 2 \mathcal{L}_\mathbf{f}{}^T \mathcal{A} \tau && (2.27) \\ \text{s.t. } & A^{\text{CLF}}(q, \dot{q}) \tau \leq b^{\text{CLF}}(q, \dot{q}), && (\text{CLF}) \end{aligned}$$

where

$$\begin{aligned} A^{\text{CLF}}(q, \dot{q}) &:= \mathcal{L}_G V_\varepsilon(q, \dot{q}) \mathcal{A}(q, \dot{q}), && (2.28) \\ b^{\text{CLF}}(q, \dot{q}) &:= -\frac{\gamma}{\varepsilon} V_\varepsilon(q, \dot{q}) - \mathcal{L}_F V_\varepsilon(q, \dot{q}) - \mathcal{L}_G V_\varepsilon(q, \dot{q}) \mathcal{L}_\mathbf{f}(q, \dot{q}). && (2.29) \end{aligned}$$

The result of solving the CLF-QP is a feedback optimal control law to drive the outputs $[\dot{\mathcal{Y}}_1^a(q, \dot{q}); \mathcal{Y}_2^a(q)]$ to follow the desired time based trajectories $[\mathcal{Y}_1^d(t); \mathcal{Y}_2^d(t)]$ with exponentially convergence. This formulation also applies when there are only relative degree 2 outputs to be tracked [55, 139].

For applications of using CLF-QP on robotic systems, torque bounds and additional nontrivial constraints can must be included in the QP [8]. When the feet contact the ground, the ground reaction forces (GRFs) have to satisfy the physics constraints such as nonnegative normal forces and non-slipping, formulated by $RF_{\text{GRF}} \leq 0$, where R is a constant matrix. As GRF is affine w.r.t. the input torque in Eq. (2.3), the GRF constraint can be directly enforced on the input:

$$A^{\text{GRF}} \tau \leq b^{\text{GRF}}, \tag{2.30}$$

where $A^{\text{GRF}} = RA_h$, $b^{\text{GRF}} = -Rb_h$. Additionally, the motor torque must be within the feasible limits of the robot hardware. For electrical motors, the available motor torque depends on the rotational speed. Then, the torque limit is denoted by

$$\tau_{lb}(\dot{q}) \leq \tau \leq \tau_{ub}(\dot{q}). \tag{2.31}$$

Therefore, with the additional physical constraints, the resulting QP controller for each domain is given as follows:

$$
\begin{aligned}
\tau^* = \underset{\tau\in\mathbb{R}^m,\delta\in\mathbb{R}}{\operatorname{argmin}} \ & \tau^T \mathcal{A}^T \mathcal{A}\tau + 2\mathcal{L}_{\mathbf{f}}{}^T \mathcal{A}\tau + p\delta^2, && (2.32)\\
\text{s.t.} \quad & A^{\text{CLF}}(q,\dot{q})\tau \leq b^{\text{CLF}}(q,\dot{q}) + \delta, && \text{(CLF)}\\
& A^{\text{GRF}}\tau \leq b^{\text{GRF}}, && \text{(GRF)}\\
& \tau_{lb}(\dot{q}) \leq \tau \leq \tau_{ub}(\dot{q}). && \text{(Torque Limit)}
\end{aligned}
$$

To increase the feasibility of the QP, the CLF constraint is relaxed [10] by introducing δ and penalizing the relaxation by adding $p\delta^2$ in the cost with some large positive constant p. In practice, the relaxation avoids instantaneous infeasibility of the QP and can still stabilize the output as long as the desired motion is well-defined.

The CLF-QP is a clever formulation that treats the Lyapunov stability condition as an inequality condition and solves for the control input via a QP. There are several arising issues beyond the focus of this thesis. Theoretically, this QP should be solved continuously for stabilizing the outputs. The real-world implementation can only be a discrete approximation of the QP controller, depending on the control frequency of the hardware, which can cause discontinuity issues [87] that fail the implementation on the hardware. Alternative formulations [87, 105] of the CLF-QP have been proposed to mitigate this problem and increase the robustness. From our experiences of numerical implementations in [138–141], the performance heavily depends on the feasibility of the QP and appropriate scaling of the outputs. This is because the defined outputs are typically different physical quantities of different units, and using the same CLF condition for all outputs can easily create conflicts with the GRF constraints and torque limits. Additionally, in the case of infeasibility, one also needs to check if the desired trajectories are fundamentally feasible to track at the chosen convergence rate.

Task Space Controller based Quadratic Program (TSC-QP)

One can also apply task space controller [12, 33, 110, 130, 143, 144] based QP for trajectory tracking. There are some variations in the literature [130] with different terminologies: whole-body control [115] and operational space control [12]. Here, for simplicity, we present a minimalistic formulation with only the motor torques as the optimization variables. We consider the outputs include both $\mathcal{Y}_2$ and $\mathcal{Y}_1$ for generality. First, from Eq. (2.6), we have an affine relation $\ddot{q} = \bar{A}\tau + \bar{b}$ with

$$\bar{A} = M^{-1}(B_\tau + J_h^T A_h), \tag{2.33}$$

$$\bar{b} = M^{-1}(J_h^T b_h - C - G). \tag{2.34}$$

As the derivatives of outputs are affine w.r.t. $\ddot{q}$, they are also affine w.r.t. to τ:

$$\dot{\mathcal{Y}}_1 = A_1\ddot{q} + b_1 = \mathbf{A}_1\tau + \mathbf{B}_1, \tag{2.35}$$

$$\ddot{\mathcal{Y}}_2 = A_2\ddot{q} + b_2 = \mathbf{A}_2\tau + \mathbf{B}_2, \tag{2.36}$$

where the explicit expression of each term is omitted. Compactly,

$$\dot{\mathbf{Y}} = \mathbf{A}\tau + \mathbf{B}, \tag{2.37}$$

with $\mathbf{Y} = [\mathcal{Y}_1; \dot{\mathcal{Y}}_2]$. The desired derivatives (accelerations) for stabilization are

$$\dot{\mathcal{Y}}_1^{\text{des}} = -K_{p_1}\mathcal{Y}_1, \tag{2.38}$$

$$\ddot{\mathcal{Y}}_2^{\text{des}} = -K_{p_2}\mathcal{Y}_2 - K_{d2}\dot{\mathcal{Y}}_2, \tag{2.39}$$

where $K_{p_1} > 0, K_{p_2} > 0, K_{d2} > 0$ are the feedback gain matrices. This motivates a quadratic cost to minimize the difference between the actual and desired accelerations:

$$J_{\text{TSC}}(\tau) = |\mathbf{A}\tau + \mathbf{B} - \dot{\mathcal{Y}}^{des}|^2, \tag{2.40}$$

with $\dot{\mathcal{Y}}^{des} = [\dot{\mathcal{Y}}_1^{\text{des}}; \ddot{\mathcal{Y}}_2^{\text{des}}]$. Additionally, two physical constraints, the GRF constraint and the torque limit, are included to formulate the quadratic program based controller:

$$\tau^* = \underset{\tau\in\mathbb{R}^m}{\text{argmin}} \quad J_{\text{TSC}}(\tau), \tag{2.41}$$

$$\text{s.t.} \quad A^{\text{GRF}}\tau \leq b^{\text{GRF}}, \tag{GRF}$$

$$\tau_{lb}(\dot{q}) \leq \tau \leq \tau_{ub}(\dot{q}). \tag{Torque Limit}$$

Therefore, we can see the TSC controller is formulated in a straightforward way. Despite the loss of theoretical soundness, the TSC controller has been shown to work in practice. In our previous publication, [138–140] used CLF-QP for stabilization, and [143, 144] used the TSC-QP. Based on our experience, both controllers can yield equivalent tracking performance on high dimensional systems with proper gain-tuning and setup. In this thesis, the in-depth comparisons and evaluations between the two or their variations will not be investigated or compared. They will be used merely as low-level feedback controllers to stabilize the outputs for evaluating the proposed approach on the walking synthesis. The optimization-based controllers will be formulated and solved at 1kHz for several systems using qpOASES [38] in this thesis.

Chapter 3

APPROACH ⊃ MODEL APPROXIMATION

This chapter and the next chapter present our main approach to solving the motion synthesis problem. The following chapters will be on the applications and extensions of the proposed approach. In this chapter, we propose a low-dimensional model of walking, based on which we develop a walking synthesis approach. The low-dimensional walking model is an underactuated version of the canonical LIP [64] model with a hybrid domain structure. Thus, we name it as Hybrid-Linear Inverted Pendulum (H-LIP). The H-LIP will be used to approximate underactuated robotic bipedal walking. The robot will be simultaneously enforced to behave like the H-LIP.

We first present the dynamics model of the H-LIP in Section 3.1, and then derive the step-to-step (S2S) dynamics of the walking of the H-LIP in Section 3.2, which takes the step size as the input to control the pre-impact state discretely. Lastly, we present the H-LIP based walking synthesis in Section 3.3 and its stepping stabilization for underactuated walking generation in Section 3.4.

3.1 Hybrid Dynamics of Walking

The canonical Linear Inverted Pendulum (LIP) model is composed of a constant-height point mass attached on telescopic legs with actuated support pivots [64]. Recall that it has been widely used for fully-actuated humanoids in ZMP based approach [35, 64], as being discussed previously. It has been used as a target walking system to provide the desired COM dynamics of walking for the full robot to embed. The passive version of LIP (without actuation at the pivots) has also appeared in the literature [34, 37, 71, 100] for footstep planning to mitigate the overuse of ankle torques on humanoid robots.

With the aim to approximate the hybrid walking nature of underactuated bipedal walking, we formally present a hybrid walking model that extends the LIP model. The actuation at the feet is removed to resemble the underactuation on bipedal robots with point feet. We add Double Support Phase (DSP) in walking to generalize the walking. It then has both Single Support Phase (SSP) and DSP in its walking (see Fig. 3.1). Therefore, we name the model the Hybrid Linear Inverted Pendulum

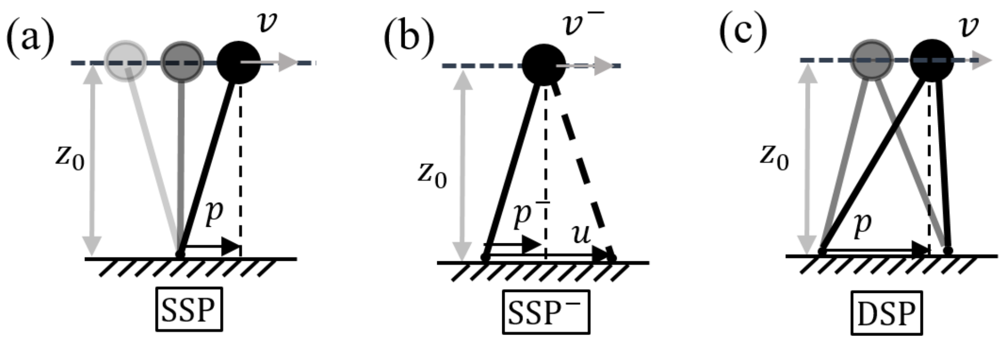

Figure 3.1: The walking of the Hybrid-LIP model in the SSP (a), at the pre-impact state (b), and during the DSP (c).

(H-LIP). For the SSP, it is a passive LIP; for the DSP, we assume that its velocity is constant. The hybrid dynamics can be written as

$$\ddot{p} = \lambda^2 p, \qquad \text{(SSP)}$$

$$\ddot{p} = 0, \qquad \text{(DSP)}$$

where $\lambda = \sqrt{\frac{g}{z_0}}$, z_0 is the nominal height of the point mass, p is the forward position of the mass relative to the stance foot, and $v = \dot{p}$ is the forward velocity of the mass. The stance foot is defined as the foot contacting the ground in the SSP, and the stance foot remains unchanged when the H-LIP transits from the SSP into the DSP. The stance foot alters when the H-LIP transits from the DSP into the next SSP. The transitions between the domains are assumed to be smooth, and thus the impact maps are defined as

$$\Delta_{\text{SSP}\to\text{DSP}} : \begin{cases} v^+ = v^- \\ p^+ = p^- \end{cases} \tag{3.1}$$

$$\Delta_{\text{DSP}\to\text{SSP}} : \begin{cases} v^+ = v^- \\ p^+ = p^- - u \end{cases} \tag{3.2}$$

where u is the step size from the stance foot position to the landing foot. The second map of the position is to alternate the support leg. The transitions are assumed to be time-based; in other words, the durations of the two domains, $\{T_{\text{SSP}}, T_{\text{DSP}}\}$, are fixed. The closed-form solution can be derived:

$$\text{SSP} \quad : \quad \begin{cases} p(t) = c_1 e^{\lambda t} + c_2 e^{-\lambda t} \\ v(t) = \lambda(c_1 e^{\lambda t} - c_2 e^{-\lambda t}) \end{cases} \tag{3.3}$$

$$\text{DSP} \quad : \quad \begin{cases} p(t) = p^-_{\text{SSP}} + v^-_{\text{SSP}} t \\ v(t) = v^-_{\text{SSP}} \end{cases} \tag{3.4}$$

where

$$c_1 = \frac{1}{2}(p^+_{\text{SSP}} + \frac{1}{\lambda}v^+_{\text{SSP}}), \tag{3.5}$$

$$c_2 = \frac{1}{2}(p^+_{\text{SSP}} - \frac{1}{\lambda}v^+_{\text{SSP}}), \tag{3.6}$$

and $[p^+_{\text{SSP}}, v^+_{\text{SSP}}]$, $[p^-_{\text{SSP}}, v^-_{\text{SSP}}]$ are the states in the beginning and at the end of the SSP, respectively.

3D H-LIP: The H-LIP is a planar model. Similar to the LIP, the H-LIP can be presented in the 3-dimensional space. Since its dynamics are completely decoupled in each plane, a H-LIP in 3D is equivalent to two orthogonally coupled planar H-LIPs. The dynamics in each plane will be identical. We omit the description here.

Equivalence to a One-Domain System: The hybrid dynamics of the H-LIP with two domains can be equivalently simplified to a one-domain hybrid system. This will simplify the descriptions of periodic orbits. Since the closed-form solution of the DSP is known, we virtually treat the DSP and its associated transitions as a single transition from the final state of the SSP to the initial state of the next SSP. Thus, the transition is defined as

$$\Delta_{\text{SSP}^- \to \text{SSP}^+} \quad : \quad \begin{cases} v^+ &= v^-, \\ p^+ &= p^- + v^- T_{\text{DSP}} - u. \end{cases} \tag{3.7}$$

As a result, we have a hybrid dynamics consisted of a continuous SSP dynamics and a virtual discrete transition. When $T_{\text{DSP}} = 0$, the dynamics becomes an actual one-domain system with only SSP, which is the passive LIP (LIP with point foot) model in the literature [45, 70, 79, 103].

Correspondence to robotic walking: The assumptions on the H-LIP are designed to approximate the horizontal COM dynamics on the underactuated bipedal robot. The contact is unactuated to match the foot underactuation. We include the DSP in the model to make it general to represent both one-domain walking and two-domain walking on the robot. The assumption of the constant COM height is to simplify the dynamics, which will be enforced on the robot.

3.2 Step-to-step Dynamics

In this section, we derive the step-to-step (S2S) dynamics of the H-LIP. As being discussed in the previous chapter, the S2S dynamics maps the hybrid dynamical system into a discrete dynamical system at the step level. Unlike the S2S dynamics

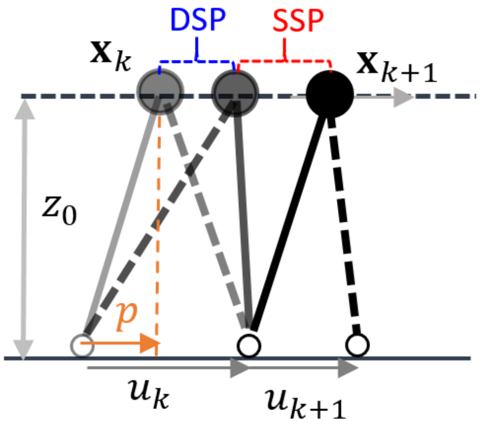

Figure 3.2: Illustration of the S2S dynamics of the H-LIP

of the robot, the S2S dynamics of the H-LIP can be derived in closed-form. Note that the dynamics of the H-LIP are piecewise linear. As the durations are fixed, the states at the discrete events of touch-down (before the transition from the SSP to the DSP) can be related from the closed-form solution.

The dynamics in the SSP in the state-space representation is

$$\underbrace{\frac{d}{dt}\begin{bmatrix}p\\v\end{bmatrix}}_{\dot{\mathbf{x}}_{\text{H-LIP}}} = \underbrace{\begin{bmatrix}0&1\\\lambda^2&0\end{bmatrix}}_{A_{\text{SSP}}}\underbrace{\begin{bmatrix}p\\v\end{bmatrix}}_{\mathbf{x}_{\text{H-LIP}}}. \tag{3.8}$$

Thus the final state of the SSP is calculated from the initial state of the SSP:

$$\mathbf{x}^-_{\text{SSP}_{\text{H-LIP}}} = e^{A_{\text{SSP}}T_{\text{SSP}}}\mathbf{x}^+_{\text{SSP}_{\text{H-LIP}}}. \tag{3.9}$$

The transition in Eq. (3.7) can be written as

$$\mathbf{x}^+_{\text{SSP}k+1} = \begin{bmatrix}1&T_{\text{DSP}}\\0&1\end{bmatrix}\mathbf{x}^-_{\text{SSP}k} + \begin{bmatrix}-1\\0\end{bmatrix}u_k. \tag{3.10}$$

Plugging Eq. (3.10) into Eq. (3.9) yields

$$\mathbf{x}^-_{\text{SSP}k+1} = \underbrace{e^{A_{\text{SSP}}T_{\text{SSP}}}\begin{bmatrix}1&T_{\text{DSP}}\\0&1\end{bmatrix}}_{A}\mathbf{x}^-_{\text{SSP}k} + \underbrace{e^{A_{\text{LIP}}T_{\text{SSP}}}\begin{bmatrix}-1\\0\end{bmatrix}}_{B}u_k, \tag{3.11}$$

where

$$\begin{aligned}A &= \begin{bmatrix}\cosh(T_{\text{SSP}}\lambda) & T_{\text{DSP}}\cosh(T_{\text{SSP}}\lambda)+\frac{1}{\lambda}\sinh(T_{\text{SSP}}\lambda)\\ \lambda\sinh(T_{\text{SSP}}\lambda) & \cosh(T_{\text{SSP}}\lambda)+T_{\text{DSP}}\sinh(T_{\text{SSP}}\lambda)\end{bmatrix}\\ B &= \begin{bmatrix}-\cosh(T_{\text{SSP}}\lambda)\\-\lambda\sinh(T_{\text{SSP}}\lambda)\end{bmatrix}.\end{aligned}$$

From now on, we treat the final state of the SSP as the discrete state of the hybrid dynamics of the H-LIP. We then drop some subscripts and superscripts and compactly rewrite the above equation as

$$\boxed{\mathbf{x}_{k+1} = A\mathbf{x}_k + Bu,} \tag{3.12}$$

which is referred as the ***step-to-step (S2S)*** **dynamics** of the H-LIP. The S2S is a discrete linear time-invariant system with the step size being the input.

3.3 Walking Synthesis

Now we present the H-LIP based walking synthesis for 3D underactuated bipedal walking (Fig. 3.3). The H-LIP is set to approximate the underactuated walking, and the walking itself should be specified to best match the H-LIP to reduce the approximation error. Although the bipedal robot may have different morphology, the outputs can be chosen as a combination of swing foot configuration, vertical COM, and upper body orientation. The desired output trajectories thus can be specified as follows.

Vertical COM Height: The vertical height of the COM z_{COM} of the walking should remain *approximately* constant during walking. When it is possible, one can strictly enforce z_{COM} to be constant. For instance, we will constrain the robot AMBER to have a constant z_{COM}. For underactuated robots with passive compliance in the leg (e.g., Cassie), strictly enforcing this condition will be challenging; hence, we will only make sure that z_{COM} is approximately constant on the robot.

Vertical Swing Foot Trajectory: The second component is on the vertical trajectory of the swing foot z_{sw}. As the step frequency on the H-LIP is assumed to be constant, the swing foot on the robot is expected to periodically lift-off and strike the ground with the same frequency. This creates continuing hybrid execution on the dynamical system and makes sure that the S2S dynamics of the robot exists. The duration of the foot in the swing phase (i.e., SSP) should match the duration of the SSP of the H-LIP. As a result, the vertical swing foot trajectory should evolve on a desired time-based trajectory $z_{\text{sw}}^{\text{des}}(t)$, which creates the lift-off and touch-down behaviors based on time. The construction of the time-based trajectory is not unique, and thus we will show several different constructions in the latter chapters of application.

Horizontal Swing Foot Trajectory: As the step size is the control input on the H-LIP, the horizontal trajectory of the swing foot should be constructed to achieve certain desired step sizes on the robot. Since the impact is time-based, the horizontal

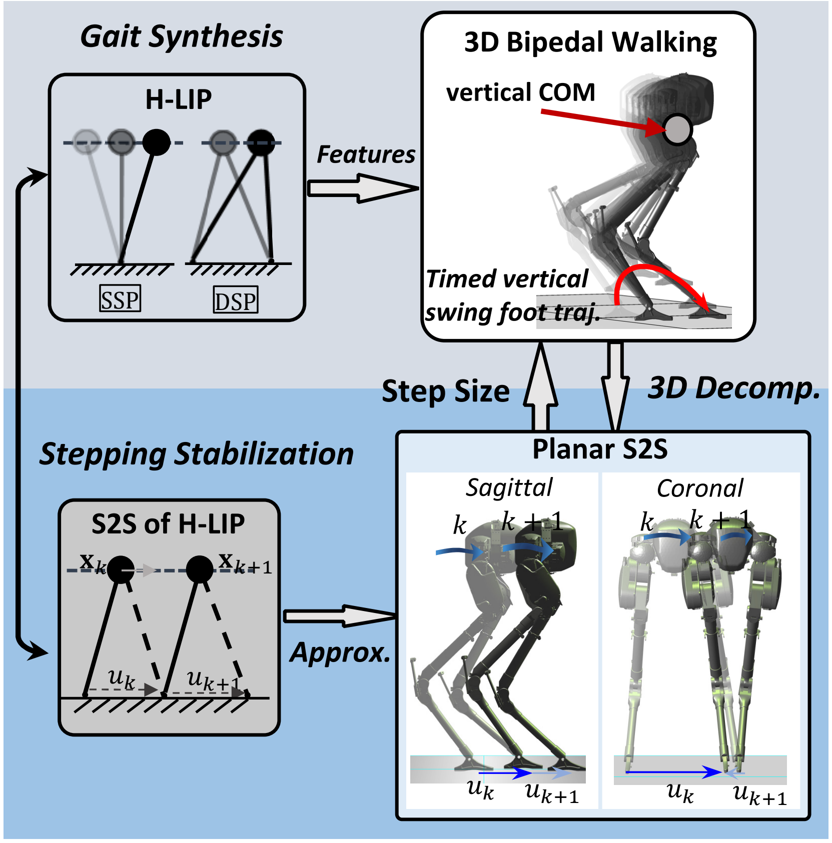

Figure 3.3: Illustration of the H-LIP based walking synthesis on the robot.

trajectory of the swing foot is constructed to swing to the desired step location at the time of impact. The desired step size is continuously calculated by the H-LIP based stepping controller, and the desired horizontal trajectory is continuously constructed to achieve the desired step size, which will be explained in Section 3.4.

3D Walking Decomposition: For 3D walking, the horizontal trajectory of the swing foot is two-dimensional, and the 3D motion requires an orthogonal composition of two H-LIPs. We select the sagittal plane and coronal plane of the robot as the decomposition of the robotic walking. The horizontal COM state, swing foot position, and step sizes of the robot are decoupled into those in the sagittal and coronal plane, respectively. Consequently, the desired step sizes are individually calculated in each plane from the H-LIP based stepping controller to stabilize the robotic walking in each plane.

3.4 S2S Dynamics based Stepping Stabilization

We next present the H-LIP based stepping controller to generate the desired step size on the robot for achieving the desired walking. The stepping in the sagittal plane is used as an example; it is applied similarly to the coronal plane. As the robot is controlled to periodically lift-off and touch-down the foot, the hybrid dynamics of walking repeats a walking cycle. Recall the definitions in Chapter 2: q is the robot configuration, the superscript $^-$ represents the pre-impact instant, τ represents the motor torques, $\mathbf{x}^R = [p^R, v^R]^T$ is the horizontal COM at the pre-impact instant, and the S2S dynamics of the horizontal COM state is:

$$\mathbf{x}^R_{k+1} = \mathcal{P}_{\mathbf{x}}(q^-_k, \dot{q}^-_k, \tau(t)). \tag{3.13}$$

As we have elaborated, the exact expression of the S2S dynamics can not be computed in closed-form. Since we design the gait of the robot based on the H-LIP, the S2S dynamics of the robot should be close to the S2S dynamics of the H-LIP. Therefore, we use the S2S dynamics of the H-LIP in Eq. (3.12) to approximate the S2S dynamics of the robot. Eq. (3.13) can be rewritten as

$$\mathbf{x}^R_{k+1} = A\mathbf{x}^R_k + Bu^R_k + w, \tag{3.14}$$

$$w := \mathcal{P}_{\mathbf{x}}(q^-_k, \dot{q}^-_k, \tau(t)) - A\mathbf{x}^R_k - Bu^R_k, \tag{3.15}$$

where u^R_k is the step size of the robot, A and B come from the S2S dynamics of the H-LIP in Eq. (3.12), and w is the difference of the S2S dynamics between the robot and the H-LIP. w is also the integration of the difference of the continuous dynamics between the two systems over one step. As the gait of the robot is designed to match the walking of the H-LIP, the dynamics error should be small. Each step also happens in a finite time (determined by the vertical trajectory of the swing foot), thus the realizable walking velocity is bounded. Another way to look at this fact is that the robot has bounded step size ($u \leq u_{\max}$) and can only realize one step in a non-trivial time period ($T \geq T_{\min}$); hence, the walking velocity equals to $\frac{u}{T} \leq \frac{u_{\max}}{T_{\min}}$, which is clearly bounded. As one can imagine, w is state and input dependent and may not be bounded generally. In practice, since the robot has bounded states and step sizes, w then can be bounded over the set of walking behaviors of interest. Thus, we treat w as a bounded disturbance to the discrete S2S dynamics; w belongs to a bounded set: $w \in W$. In the latter chapters of applications, we will evaluate this assumption numerically.

The S2S approximation in Eq. (3.14) provides a convenient way to design a S2S dynamics based stepping controllers for u to control the pre-impact horizontal COM state $\mathbf{x}^R$. Assume that the desired walking behavior of the H-LIP is described by $\mathbf{x}^{\text{H-LIP}}$ and $u^{\text{H-LIP}}$, which will be synthesized in the next chapter. Let $u^R = u^{\text{H-LIP}} + u^{\mathbf{e}}$, and let $\mathbf{e} = \mathbf{x}^R - \mathbf{x}^{\text{H-LIP}}$ be the error state. The error dynamics becomes

$$\mathbf{e}^R_{k+1} = A\mathbf{e}^R_k + Bu^{\mathbf{e}}_k + w. \tag{3.16}$$

Thus, we need to design a S2S dynamics based stepping controller to stabilize the error dynamics subject to the bounded disturbance. In our previous publications [141–143], we apply the ***H-LIP based stepping***:

$$u^{\text{des}} = u^{\text{H-LIP}} + K(\mathbf{x}^R_k - \mathbf{x}^{\text{H-LIP}}_k), \tag{3.17}$$

where K is the feedback gain to make $A + BK$ stable, i.e., $\texttt{eig}(A + BK) < 1$. Applying the H-LIP based stepping yields the error dynamics:

$$\mathbf{e}_{k+1} = (A + BK)\mathbf{e}_k + w. \tag{3.18}$$

Since $A + BK$ is stable, the error dynamics has a minimum disturbance invariant set E. By definition,

$$(A + BK)E \oplus W \in E, \tag{3.19}$$

where $\oplus$ is the Minkowski sum. We call E the error invariant set, i.e., if $\mathbf{e}_k \in E$ then $\mathbf{e}_{k+1} \in E$. If W is small, then E is small. Thus the desired walking behavior (of the horizontal COM state) can be first realized on the H-LIP, and then applying the H-LIP based stepping yields the behavior to be approximately realized on the robot, with the error being bounded by E. Note that the feasible step size on the robot is bounded, i.e., $u^R \in U$. Thus, the desired behavior of the H-LIP should satisfy $u^{\text{H-LIP}} \in U \ominus KE$. In the latter when it is possible, the superscripts ^{R} and $^{\text{H-LIP}}$ are be omitted for conciseness. The gain K can be simply selected from the following two controllers.

<u>*Deadbeat Controller:*</u> The deadbeat controller [137] is calculated to stabilize the error dynamics within minimum steps. Since there is one input and two states, the deadbeat gain satisfies

$$(A + BK_{\text{deadbeat}})^2 = \mathbf{0}, \tag{3.20}$$

solving which yields the deadbeat gain that can be used in the H-LIP based stepping in Eq. (3.17).

<u>*Linear Quadratic Regulator:*</u> We can apply the linear quadratic regulator (LQR) in the context that it minimizes a custom-tuned cost function:

$$J_{\text{LQR}} = \sum_{k=1}^{\infty} (\mathbf{e}_k^T Q \mathbf{e}_k + u_k^T R u_k + 2\mathbf{e}_k^T N u_k). \tag{3.21}$$

where Q, R, N are cost parameters. The resultant optimal state feedback gain is

$$K_{\text{LQR}} = -(R + B^T P B)^{-1}(B^T P A + N^T), \tag{3.22}$$

where P is solved from the discrete-time algebraic Riccati equation (DARE) [19]. The LQR gain thus can be used in Eq. (3.17) for stepping stabilization.

Remark. *The H-LIP based stepping is a S2S dynamics based stepping controller with a closed-form solution, which is convenient to implement in practice. There are other linear controllers [19] one can synthesize to stabilize the error dynamics in Eq.* (3.16)*. Additionally, different gains K can be applied as long as $A + BK$ is stable. In this thesis, we do not explore all the possible formulations of the S2S based stepping controllers or different gain K. Instead, for consistency, we apply the deadbeat gain $K_{deadbeat}$ to evaluate the approach on different robots in the chapters of applications, which also demonstrates that gain tuning is not necessary.*

Implementation and Evaluation

To summarize, the H-LIP model and the S2S based stepping controller are presented to propose a walking/gait synthesis approach for underactuated bipedal robots. The walking synthesis includes the construction of the vertical COM trajectory, the vertical swing foot trajectory, and the horizontal swing foot trajectory to realize the H-LIP based stepping controller. The desired walking behavior will be described on the H-LIP. Then, a S2S based stepping controller such as the H-LIP based stepping in Eq. (3.17) calculates the desired step size u^{des} for the robot in the SSP to stabilize its walking closely to the H-LIP walking.

The desired step size u^{des} is a function of the horizontal COM state of the robot at pre-impact as shown in Eq. (3.17). However, the pre-impact COM state cannot be known in the continuous SSP. We thus use the current COM state to replace the pre-impact COM in Eq. (3.17), which creates the same desired step size u^{des} at the pre-impact instant. Thus, u^{des} is continuously changing w.r.t. the horizontal COM state in the SSP. u^{des} is then continuously used for constructing the desired horizontal swing foot position in the output synthesis. Specifically, the desired horizontal trajectories of the swing foot $x_{\text{sw}}^{\text{des}}$ (w.r.t. the stance foot) can be designed to be

$$x_{\text{sw}}^{\text{des}} = (1 - \mathcal{B}_h(t))x_{\text{sw}}^{+} + \mathcal{B}_h(t)u^{\text{des}}, \tag{3.23}$$

where x_{sw}^{+} is the horizontal position of the swing foot in the beginning of the current SSP w.r.t. the stance foot. $\mathcal{B}_h(t)$ is a smooth curve that transits from 0 ($t = 0$) to

1 ($t = T_{\text{SSP}}$). We use Bézier polynomials for the construction; see Appendix D for details. Note that t is the clock of the gait, which resets to 0 after each step. After constructing the desired output trajectories, we then apply the optimization-based controllers for trajectory tracking as been described in Chapter 2. We sketch the process in the following Algorithm. In the latter chapters, we will first present the generation of the desired behaviors on the H-LIP, and then validate and extend the proposed walking synthesis in detail on various bipedal walking systems for realizing different walking behaviors in simulation and experiment.

Algorithm 1 H-LIP based Walking Synthesis

```
Initialization: Desired Walking Behavior
while Every Control loop do
    Generate Desired H-LIP Walking
    Get Desired Step Size ⟵ H-LIP based Stepping Controller
    Construct Desired Output Trajectories
    Output Stabilization via Optimization-based Controllers
end while
```

After the implementation, we numerically evaluate the error state and the disturbance invariant set. Since w cannot be calculated analytically, we numerically calculated the data of w at each step during walking and then identify a polytope that contains all w, which is used as an approximation of W. When K_{deadbeat} is used, $A + BK$ is nilpotent ($\exists n$, s.t. $(A + BK)^n = \mathbf{0}$). Thus $E = E_n := \oplus_{i=0}^{n-1}(A + BK)^i W$ can be calculated exactly [102]. When K_{LQR} is used, E can be inner-approximated with $E_{n\in\mathbb{N}}$ (i.e. $E_n \subset E$) or outer-approximated using the techniques in [69, 102] and the reference therein. The set operations will be calculated using MPT [53].

Comparison

The H-LIP is used as a model to provide an approximation to the bipedal robotic walking behaviors so that the S2S dynamics based stepping controllers can be synthesized for walking realization. It is important and necessary to compare this approach with existing approaches at a high level.

Comparison with ZMP-LIP: The H-LIP is a variant of the canonical LIP model [61] with foot underactuation and hybrid dynamics structure. As been discussed in Chapter 1, the LIP model has been extensively applied in the ZMP approaches [37, 64] for realizing humanoid walking. The LIP is continuously actuated, and the center of pressure (COP) is the output of the continuous linear control system of the

LIP; the H-LIP is only discretely actuated by swapping support legs, and the COP is not continuously controllable. One can also view the ZMP-LIP approaches as using the ZMP of the LIP to approximate the ZMP of the robot, and the LIP dynamics is embedded on the humanoid; the ZMP condition thus is approximately realized on the humanoid. In this thesis, we use the dynamics of the H-LIP to approximate the horizontal COM dynamics of the underactuated bipedal robots, which do not have foot actuation and thus can not strictly embed the pendulum dynamics. Additionally, compared to the passive LIP in [100] [24], the H-LIP has the DSP and fixed domain durations. The addition of the DSP allows its application to walking with non-trivial DSP.

Comparison with HZD: Compared to the HZD [46, 50] approaches, the complete walking motion of the full-dimensional state is not pre-optimized or pre-determined. The desired walking behavior will be specified by the H-LIP model in terms of COM height, domain period, and the desired pre-impact horizontal COM state. The S2S dynamics based stepping controller then stabilizes the walking of the robot closely to the desired behaviors. Thus, the resultant walking behavior on the robot is directly stabilized during walking. Moreover, there is no need to check Poincaré return map for stability; the S2S dynamics based stepping controller directly deals with the underactuation stabilization by continuously changing the step sizes in the planning loop. From the computation perspective, the HZD approaches require offline optimizations for motion planning while our approach solves the planning online with trivial computation (e.g., the H-LIP based stepping is in closed-form).

Comparison with existing S2S based approach: As been discussed in Chapter 2, the S2S based approach has been appeared in controlling SLIP running behaviors [16, 43] since its S2S dynamics can be obtained easily. As for bipedal walking systems, by and large, the S2S dynamics based control has been focused on the linearization at the fix-point [13, 14, 75, 131] of a periodic solution on the return map (very few exceptions [16, 85] learned the S2S). This thesis, instead, approximates the S2S dynamics of the robotic walking over a large region of the state-space at the Poincaré section. Additionally, the S2S approximation is linear and readily facilitates feedback controllers to be synthesized.

Chapter 4

APPROACH ⊃ ORBIT CHARACTERIZATION AND STABILIZATION

In this chapter, we present the identifications of the periodic walking orbits of the H-LIP, which will provide the desired walking behavior for the S2S based stepping controller. The linear dynamics of the H-LIP promote the periodic orbits to be geometrically characterized in the phase space. Based on the number of steps that one orbit contains, we classify the orbits into Period-N orbits, where N is the number of steps. As the time duration (period) of one step is fixed, a Period-N orbit has a duration of period×N. Here we only study Period-1 (P1) and Period-2 (P2) orbits of the H-LIP. P1 orbits are the one-step orbits, and P2 orbits are the two-step orbits.

We begin by introducing the phase portraits of the H-LIP, and then in Section 4.1 and 4.2 we present the characterizations of P1 orbits and P2 orbits, respectively. In Section 4.3 we show the equivalent characterizations of the orbits. Lastly, in Section 4.4, we derive the stepping stabilization to control the walking of the H-LIP to generate desired walking behaviors, which are used as the nominal walking behaviors in the H-LIP based stepping in Eq. (3.17).

Phase Portraits: The state space of the H-LIP is a two-dimensional, thus we can present the periodic orbits explicitly in its phase portraits. For the H-LIP in the SSP, its phase portrait is identical to that of the passive LIP (see Fig. 4.1 (a)); the phase portrait in the DSP is simple, as shown in Fig. 4.1 (b). The phase portrait in the

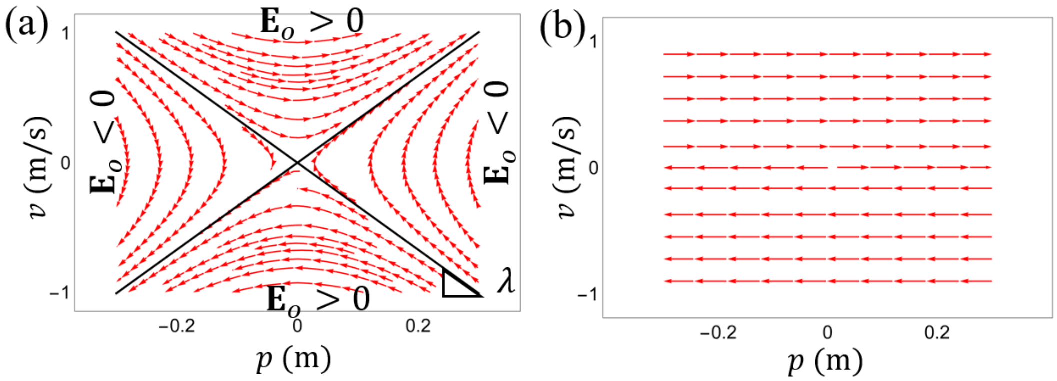

Figure 4.1: Phase portraits of the H-LIP walking in its (a) SSP and (b) DSP.

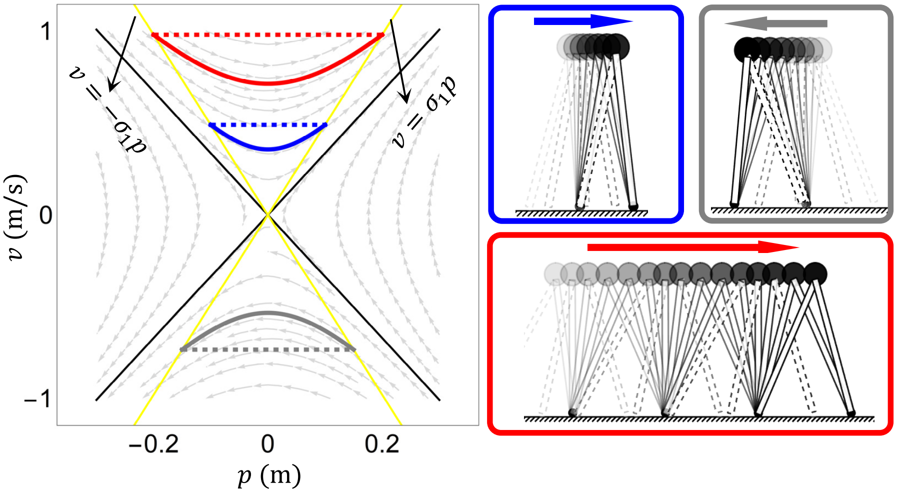

Figure 4.2: Illustration of the P1 orbits of the H-LIP (left) and the corresponding time-lapse figures of the orbits (right). The red, blue, and grey lines are the orbits, the yellow lines are the orbital lines of characteristics, and the black lines are the asymptotes where $\mathbf{E}_o = 0$. One the orbit, the solid lines represent the states in the SSP and the dashed lines represent the transitions between the SSP.

SSP can be divided into four regions by the asymptotes (the cross lines defined by $v = \pm\lambda p$), based on the *orbital energy*:

$$\mathbf{E}_o(p, v) = v^2 - \lambda^2 p^2. \tag{4.1}$$

On the asymptotes, $\mathbf{E}_o = 0$. The orbital energy is either positive or negative in each region. More importantly, the orbital energy is conserved in the SSP. The physical meaning of $\mathbf{E}_o > 0$ is that the H-LIP rotates over the stance foot, i.e., the system passes through the states where $p = 0$.

For conciseness, we use the equivalent one-domain system in Section 3.1 of Chapter 3. Thus, the orbits can be represented by a continuous trajectory in the SSP with a discrete transition. In the following, we present the geometric characterization of P1 and P2 orbits in the phase portrait of the SSP. The subscripts of $_{\text{SSP}}$ on the states are omitted. Additionally, the pre-impact states and the step sizes of the orbits are presented explicitly from the desired net (average) velocity of walking.

4.1 Period-1 Orbit

For ease of understanding, we first characterize the P1 orbits.

Theorem 4.1.1. *For all the Period-1 orbits, the initial states* $[p^+, v^+]$ *and the final states* $[p^-, v^-]$ *of the SSP are on the line* $v = -\sigma_1 p$ *and the line* $v = \sigma_1 p$*, respectively,* *where the orbital slope is defined as*

$$\sigma_1 := \lambda \coth\left(\frac{T_{\text{SSP}}}{2}\lambda\right). \tag{4.2}$$

The lines $v = \pm\sigma_1 p$ are called the orbital lines. Each state on the orbital line $v = -\sigma_1 p$ is the initial state of the SSP of a unique Period-1 orbit with the step length being

$$u_1 = 2p^- + v^- T_{\text{DSP}}. \tag{4.3}$$

Here, we define the orbital lines with the orbital slopes to identify the boundary states of the SSP of the P1 orbits. The proof is as follows.

Proof. Since $v^+ = v^-$ and $\mathbf{E}_o^+ = \mathbf{E}_o^-$, we have $p^+ = -p^-$. Substituting these conditions into Eq. (3.3) with simple algebra manipulations yields

$$\frac{v^+}{p^+} = -\lambda \coth\left(\frac{T_{\text{SSP}}}{2}\lambda\right), \tag{4.4}$$

which indicates that the initial states are on the line $v = -\sigma_1 p$. The rest of the theorem follows immediately. □

Fig. 4.2 illustrates different P1 orbits of the system with $T_{\text{SSP}} = 0.5$s, each of which has a different net velocity. A different duration of T_{SSP} would produce a different set of orbital lines (the cross yellow lines). As $T_{\text{SSP}} \longrightarrow \infty$, the orbital lines converges to the black lines. Thus, it is easy to verify that all the P1 orbits only exist in the $\mathbf{E}_o > 0$ regions as $\sigma_1 \geq \lambda$.

Without further illustrations, one can find that there exist infinite P1 orbits, and they are all connected. Given a desired net velocity v^d, there is a unique P1 orbit for realization. It is obvious that the step size of the P1 orbit is

$$u_1^* = v^d (T_{\text{DSP}} + T_{\text{SSP}}). \tag{4.5}$$

The final states of SSP of the P1 orbit can then be calculated from Eq. (4.2) and Eq. (4.3), which yields

$$[p^*, v^*] = [1, \sigma_1] \frac{u_1^*}{2 + T_{\text{DSP}}\sigma_1}. \tag{4.6}$$

Note that we use the superscript * to represent the states and inputs on the periodic solution given by v^d.

4.2 Period-2 Orbit

P2 orbits take two steps to complete a periodic walking. We differentiate the consecutive two steps by its stance foot, indexed by L/R. Similar to the P1 orbits, we identify the orbital slope and orbital lines of P2 orbits, and therefore the P2 orbits are geometrically characterized:

Theorem 4.2.1. *For all the Period-2 orbits, the orbital lines are $v = \pm\sigma_2 p + d_2$, where the orbital slope σ_2 is defined as*

$$\sigma_2 := \lambda \tanh\left(\frac{T_{\text{SSP}}}{2}\lambda\right), \tag{4.7}$$

and d_2 is a constant determined by the desired net velocity. Each state on the line $v = -\sigma_2 p + d_2$ represents an initial state of a Period-2 orbit, with the step size being

$$u_{\text{L/R}} = 2p^-_{\text{L/R}} + T_{\text{DSP}} v^-_{\text{L/R}}. \tag{4.8}$$

Proof. We can first show that any initial states on the line of $v = -\sigma_2 p + d_2$ flow to the line $v = \sigma_2 p + d_2$ after $t = T_{\text{SSP}}$. This is evident by applying the closed-form solution from Eq. (3.3) with $t = T_{\text{SSP}}$. Given $v^+ = -\sigma_2 p^+ + d_2$, one can verify that $v^- - \sigma_2 p^- - d_2 \equiv 0$. Thus, the orbital lines of characteristics are $v = \pm\sigma_2 p + d_2$.

Then we select a state on the line $v = -\sigma_2 p + d_2$ as the initial state of the SSP with the left leg being the support leg of the H-LIP, i.e. $v^+_{\text{L}} = -\sigma_2 p^+_{\text{L}} + d_2$. The final state satisfies $v^-_{\text{L}} = \sigma_2 x^-_{\text{L}} + d_2$. Applying the impact map in Eq. (3.7) with the step size in Eq. (4.8) yields

$$v^+_{\text{R}} = v^-_{\text{L}},\ p^+_{\text{R}} = -\frac{v^-_{\text{L}} - d_2}{\sigma_2} = -\frac{v^+_{\text{R}} - d_2}{\sigma_2}, \tag{4.9}$$

which indicates that the initial state of SSP with right leg as the support is on the line $v = -\sigma_2 p + d_2$ again. Taking another step, the system goes back to its original state when left leg becomes the support. Therefore, the initial state on the line creates a P2 orbit with the step length in Eq. (4.8). □

Unlike the P1 orbits, for which only one orbit can be found to achieve a certain desired velocity, there are infinite P2 orbits to achieve a certain desired velocity. The result is stated in the following proposition.

Proposition 4.2.2. *There exist infinite Period-2 orbits to achieve a desired net velocity v^d. The initial states of the SSP of all the periodic orbits for v^d lie on the*

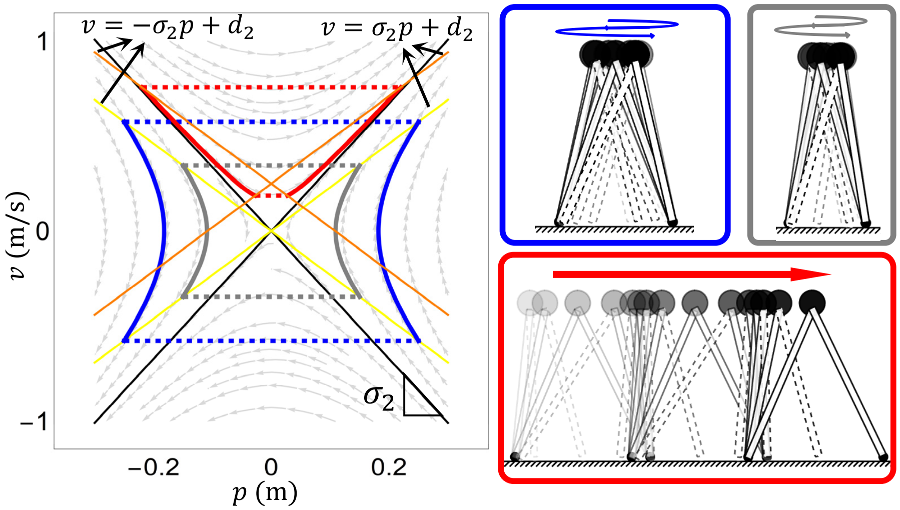

Figure 4.3: Illustration of the P2 orbits of the H-LIP (left) and the corresponding time-lapse figures of the orbits (right). Individual color indications are the same to the case for P1 orbits.

line $v = -\sigma_2 p + d_2$ *with* d_2 *being*

$$d_2 = \frac{\lambda^2 \mathrm{sech}^2(\frac{\lambda}{2}T_{\mathrm{SSP}})(T_{\mathrm{SSP}} + T_{\mathrm{DSP}})v^d}{\lambda^2 T_{\mathrm{DSP}} + 2\sigma_2}. \tag{4.10}$$

Proof. Based on Theorem 4.2.1, selecting any initial state $[p^+_{\mathrm{L}}, v^+_{\mathrm{L}}]$ on the line $v = -\sigma_2 p + d_2$ yields a P2 orbit. The traveled distance over the P2 orbit is $2(p^-_{\mathrm{L}} - p^+_{\mathrm{L}}) + T_{\mathrm{DSP}}(v^+_{\mathrm{L}} + v^-_{\mathrm{L}})$. It equals $2(T_{\mathrm{SSP}} + T_{\mathrm{DSP}})v^d$. □

Geometrically, d_2 shifts the set of cross orbital lines up or down. The magnitude of d_2 determines the net velocity of the P2 orbit. The infinite number of realization for one desired velocity can also be seen through the fact that the step sizes are determined by v^d:

$$u^*_{\mathrm{L}} + u^*_{\mathrm{R}} = 2(T_{\mathrm{DSP}} + T_{\mathrm{SSP}})v^d. \tag{4.11}$$

There are infinite combination of $u^*_{\mathrm{L}}, u^*_{\mathrm{R}}$ to satisfy Eq. (4.11) and therefore infinite P2 orbits to realize the desired net velocity. Selecting one step size determines the other one and thus determines the P2 orbit. The final states $[p^*_{\mathrm{L/R}}, v^*_{\mathrm{L/R}}]$ are on the orbital lines:

$$v^*_{\mathrm{L/R}} = \sigma_2 p^*_{\mathrm{L/R}} + d_2. \tag{4.12}$$

The selected step size thus determines the boundary positions from Eq. (4.8):

$$p^*_{\mathrm{L/R}} = \frac{u^*_{\mathrm{L/R}} - T_{\mathrm{DSP}} d_2}{2 + T_{\mathrm{DSP}}\sigma_2}. \tag{4.13}$$

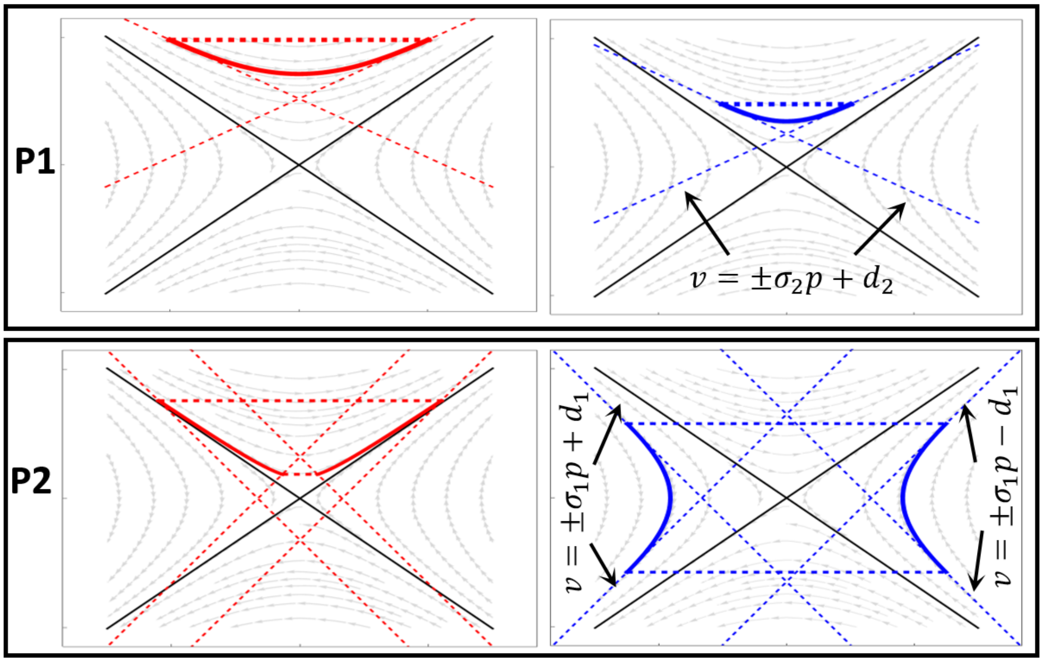
P1
$v = \pm\sigma_2 p + d_2$
P2
$v = \pm\sigma_1 p + d_1$
$v = \pm\sigma_1 p - d_1$

Figure 4.4: Equivalent characterization of the periodic orbits. The dashed cross lines are the equivalent orbital lines.

4.3 Equivalent Characterization

The P1 and P2 orbits are characterized by their orbital lines, respectively. We also find that under certain conditions, the orbital lines of P1 orbits can also characterize P2 orbits and vice versa. It is clear that when $u_{\mathrm{L}} = u_{\mathrm{R}}$, a P2 orbit becomes an equivalent P1 orbit, which can be stated as the following proposition.

Proposition 1.3 *The orbital lines $v = \pm\sigma_2 p + d_2$ characterize the P1 orbits when $u_{\mathrm{L}} = u_{\mathrm{R}}$, which yields the final state of the SSP as*

$$p^* = \frac{d_2 \sinh(T_{\mathrm{SSP}}\lambda)}{2\lambda}, \ v^* = \sigma_2 p^* + d_2. \tag{4.14}$$

Similarly, P1 orbital lines can characterize P2 orbits:

Proposition 1.4 *The extended P1 orbital lines $v = \pm\sigma_1(p \pm d_1)$ characterize the P2 orbits: the initial states are on $v = -\sigma_1(p \pm d_1)$ and the final states are on $v = \sigma_1(p \pm d_1)$. The corresponding step sizes are as stated in Eq.* (4.8).

The non-uniqueness of P2 orbits to realize the desired velocity comes from the non-uniqueness of d_1. Given a d_1, the final states of the P2 orbits can thus be determined. Fig. 4.4 illustrates the equivalent characterizations of the orbits in Fig.

4.2 and 4.3. In the latter, we only use the results from Theorem 4.1.1 and 4.2.1 to find the desired walking orbits.

4.4 Orbit Stabilization

The characterized obits are used as a description of the desired walking behaviors. Given a desired walking velocity and a step frequency, we can use the characterization to identify the corresponding pre-impact states. In this section, we derive the stabilization on the H-LIP to its desired walking behaviors. In particular, we first present a heuristic synthesis that is based on velocity feedback, the gain of which can be derived from the hybrid dynamics based on the contraction principle [141]. Then, we show canonical linear control stabilizations that are based on the S2S dynamics of walking. Finally, in the last part, we show the application of H-LIP based stepping for stabilization, which then renders a wide class of linear controllers to function. For the practical purposes of implementation, one can directly refer to the last subsection.

A Heuristic Stabilization

In this part, we present the stabilization of the orbits of the H-LIP via heuristically synthesized controllers in [139, 141]. Since the dynamics of the H-LIP are piecewise linear, we can formally prove this controller and find the valid range of the feedback gain. We present the following two theorems for stabilization.

Theorem 4.4.1. *Given a desired Period-1 orbit with σ_1 in Eq.* (4.2) *and v^* in Eq.* (4.6)*, the following step size*

$$u_1^{cl} = p^- + v^- T_{\text{DSP}} + \frac{v^-}{\sigma_1} + K_v(v^- - v^*), \tag{4.15}$$

can globally stabilize the H-LIP with the K_v in

$$0 < K_v < \frac{2}{\lambda}\text{csch}(T_{\text{SSP}}\lambda), \tag{4.16}$$

and the optimal gain

$$K_v^* = \frac{1}{\lambda}\text{csch}(T_{\text{SSP}}\lambda), \tag{4.17}$$

globally stabilizes the velocity by one step and the position by two steps.

Proof. The desired pre-impact state of the SSP of the H-LIP is $[p^* = \frac{v^*}{\sigma_1}, v^*]$. To prove this stabilization, we first show that the step-to-step velocity is contracting to the desired velocity v^*; in other words,

$$\left|v_{k+1}^- - v^*\right| = c\left|v_k^- - v^*\right|, \tag{4.18}$$

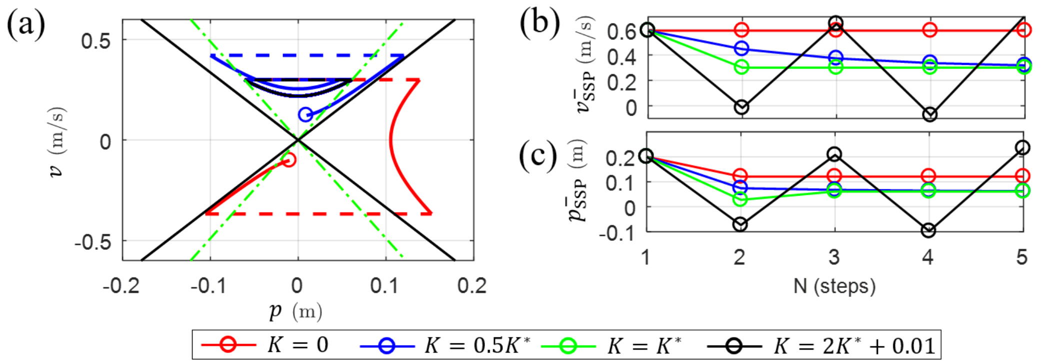

Figure 4.5: Illustration of the stabilization on P1 orbit: (a) optimal stabilization from random initial states (the black is the desired orbit, and the blue and red are the simulated trajectories from the states marked by the circles), and (b,c) the simulated trajectory using the same controller from the same initial state but with different gains.

with $0 < c < 1$, where k and $k+1$ index the current step and the next step, respectively. Suppose that we have an arbitrary pre-impact state $[p^-_k, v^-_k]$. By the impact maps in Eq. (3.1) and (3.2) with u^{cl}_1 and closed-form solutions in Eq. (3.3) and (3.4), the pre-impact velocity of next step satisfies

$$v^-_{k+1} - v^* = (1 - K_v\lambda\sinh(T_{\rm SSP}\lambda))(v^-_k - v^*). \tag{4.19}$$

Obviously, K_v must be in the range in Eq. (4.16) so that the contracting mapping in Eq. (4.18) is satisfied. Otherwise, the velocity error stays the same or increases. Additionally, for the position,

$$p^-_{k+1} - p^* = (\frac{1}{\sigma_1} - K_v\cosh(T_{\rm SSP}\lambda))(v^-_k - v^*). \tag{4.20}$$

Thus $p^- \longrightarrow p^*$ as $v^- \longrightarrow v^*$, which proves the stabilization of the position subsequently after the velocity.

Lastly, plugging the optimal gain of Eq. (4.17) into Eq. (4.19) and (4.20) indicates that $v^-_{k+1} = v^*$ and $p^-_{k+2} = p^*$ for arbitrary states $[p^-_k, v^-_k]$, which means that, the velocity is stabilized by one step and the position is stabilized by two steps. □

Theorem 4.4.2. *Given a desired Period-2 orbit with σ_2 in Eq.* (4.7) *and desired boundary velocities v^*_L, v^*_R in Eq.* (4.12)*, the following step size*

$$u^{cl}_2 = p^- + T_{\rm DSP}v^- + \frac{v^- - d_2}{\sigma_2} - K_v(v^- - v^*_{\rm L/R}), \tag{4.21}$$

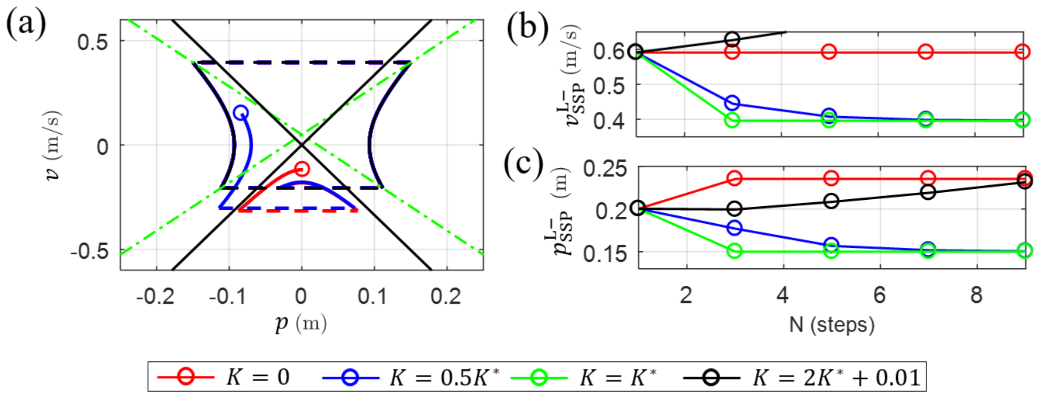

Figure 4.6: Illustration of the stabilization on P2 orbits.

can stabilize the H-LIP globally with the K_v in Eq. (4.16)*. And the optimal gain in Eq.* (4.17) *stabilizes the velocity by one step and the position by two steps globally.*

The proof is similar to the previous one and thus is omitted. In [139], the feedback stepping control law was used to stabilize the lateral balance of the underactuated walking of Cassie with zero lateral velocity, i.e. $d_2 = 0$. Here the stabilization law in Eq. (4.21) is stated for the general case for all P2 orbits.

Fig. 4.5 (a) and Fig. 4.6 (a) illustrate the stabilization of this stepping with the optimal gain to a P1 and a P2 orbit from random initial states, respectively. Additionally, different gains are compared for the stabilization in Fig. 4.5 (b, c) and Fig. 4.6 (b, c), which numerically validate the theorems in simulation.

Remark. *One may think that this stepping controller is a variant of the Raibert style stepping controller [101] [106]. However, the Raibert style controllers are oftentimes implemented as PID/PD terms. Heuristic tuning is required. Here the optimal gain eliminates the tuning process completely. Moreover, we present rigorous derivations on the stabilizations. In fact, we will see from the next two parts that the optimal stepping controllers are actually deadbeat controllers.*

Stabilization via S2S Dynamics

In this section, we show the stabilizations that are based on the S2S dynamics. The orbit stabilization can be viewed as generating a controller on u such that the state of the linear S2S dynamics is controlled to the desired final state of the orbit. It then becomes a canonical linear control problem. The following presents this approach.

Stabilization for P1 Orbits: The desired final state of the SSP fully determines the P1 orbits. Suppose the desired state is $\mathbf{x}^* = [p^*, v^*]^T$. The error state is defined as $\mathbf{e} = \mathbf{x} - \mathbf{x}^*$, and the error dynamics is

$$\mathbf{e}_{k+1} = A\mathbf{e}_k + Bu_k + (A - I)\mathbf{x}^*. \tag{4.22}$$

The goal is to stabilize $\mathbf{e}_k \to 0$ as $k \to \infty$. As there is only one input, $(A - I)\mathbf{x}^*$ must be in the column space of B for $\mathbf{x}^*$ to be stabilized, i.e.,

$$(A - I)\mathbf{x}^* = cB, \tag{4.23}$$

where $c \in \mathbb{R}$ is a constant. Then the selection of $\mathbf{x}^*$ must satisfy the condition that $\mathbf{x}^* = c(A - I)^{-1}B$, which exactly represents the orbital line of the P1 orbits of the H-LIP ($v^* = \sigma_1 p^*$). Thus the following conclusion can be made:

Proposition 4.4.3. *The error state in Eq.* (4.22) *can be stabilized to 0 if and only if* $\mathbf{x}^*$ *are selected on the orbital line of characteristics of P1 orbits.*

Therefore, for P1 orbit stabilization, we can rewrite the error dynamics as

$$\mathbf{e}_{k+1} = A\mathbf{e}_k + B\underbrace{(u_k + c)}_{\bar{u}_k}, \tag{4.24}$$

where

$$c = -T_{\text{DSP}}v^* - \frac{2v^*}{\lambda}\tanh(\frac{\lambda}{2}T_{\text{SSP}}). \tag{4.25}$$

It is easy to design a state feedback controller with $\bar{u}_k = K\mathbf{e}_k$, and the closed-loop system then is

$$\mathbf{e}_{k+1} = (A + BK)\mathbf{e}_k, \tag{4.26}$$

which becomes the error dynamics in Eq. (3.16) with $w = 0$. Recall that for this system with a single input and two states, two steps is required to stabilize the system to 0 from any initial stable globally. It is easy to find the deadbeat gain K such that $(A + BK)^2 = \mathbf{0}$, which is

$$K = [1, T_{\text{DSP}} + \frac{1}{\lambda}\coth(T_{\text{SSP}}\lambda)]. \tag{4.27}$$

This deadbeat controller yields the step size:

$$u_k = K\mathbf{e}_k - c = p + p^* + T_{\text{DSP}}v + \frac{1}{\lambda}\coth(T_{\text{SSP}}\lambda)(v - v^*), \tag{4.28}$$

which is verified to be equal to the optimal stepping controller in Theorem 4.4.1. The deadbeat controller globally stabilizes the system to the P1 orbit with two steps.

Stabilization for P2 Orbits: The P2 orbits have two alternating final states of the SSP. Let $p^*_\mathrm{L}, p^*_\mathrm{R}, v^*_\mathrm{L}, v^*_\mathrm{R}, d^*_2$ be the elements of the desired orbit. To represent the P2 orbits as a control objective, we augment the state dimension, and the linear dynamics becomes

$$\underbrace{\begin{bmatrix} p_{k+1} \\ v_{k+1} \\ 1 \\ -1 \end{bmatrix}}_{\bar{\mathbf{x}}_{k+1}} = \underbrace{\begin{bmatrix} A_{1,1} & A_{1,2} & 0 & 0 \\ A_{2,1} & A_{2,2} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}}_{\bar{A}_4} \underbrace{\begin{bmatrix} p_k \\ v_k \\ 1 \\ 1 \end{bmatrix}}_{\bar{\mathbf{x}}_k} + \underbrace{\begin{bmatrix} B_1 \\ B_2 \\ 0 \\ 0 \end{bmatrix}}_{\bar{B}_4} u_k,$$

with the output being

$$\bar{\mathcal{Y}}_k = \underbrace{\begin{bmatrix} 1 & 0 & -a_1 & -a_2 \\ 0 & 1 & -b_1 & -b_2 \end{bmatrix}}_{\bar{C}_4} \bar{\mathbf{x}}_k, \tag{4.29}$$

where $a_1 = \frac{p^*_\mathrm{L}+p^*_\mathrm{R}}{2}, a_2 = \frac{p^*_\mathrm{L}-p^*_\mathrm{R}}{2}, b_1 = \frac{v^*_\mathrm{L}+v^*_\mathrm{R}}{2}, b_2 = \frac{v^*_\mathrm{L}-v^*_\mathrm{R}}{2}$, and the subscripts of A, B indicate the individual element in the matrix A, B. The alternating of the left and right leg is represented by the alternating of signs of the last element in $\bar{\mathbf{x}}_k$. The stabilization of the P2 orbit is represented by stabilizing

$$\bar{\mathcal{Y}}_k \rightarrow 0. \tag{4.30}$$

Similar to the deadbeat controller for P1 orbit, we can design a state feedback controller:

$$u_k = K_4 \bar{\mathbf{x}}_k. \tag{4.31}$$

Then the deadbeat controller yields $\bar{\mathcal{Y}}_{k+2} = \bar{C}_4(\bar{A}_4 + \bar{B}_4\bar{K}_4)^2\bar{\mathbf{x}}_k \rightarrow 0$ for all $\bar{\mathbf{x}}$. To make all the elements in the matrix $\bar{C}_4(\bar{A}_4 + \bar{B}_4\bar{K}_4)^2$ zero, the feedback gain is

$$K_4 = [1, T_\mathrm{DSP} + \frac{\coth(T_\mathrm{SSP}\lambda)}{\lambda}, -\frac{d^*_2}{2\lambda}\coth(\frac{\lambda}{2}T_\mathrm{SSP}), \frac{d_2 \sinh(T_\mathrm{SSP}\lambda) - 2p^*_\mathrm{L}\lambda}{2\lambda + 2\lambda\cosh(T_\mathrm{SSP}\lambda)}]. \tag{4.32}$$

Similarly, this deadbeat controller is also verified to be equal to the optimal controller in Theorem 4.4.2.

The Simplest Formulation via H-LIP based Stepping

The control via the S2S dynamics has provided more than the deadbeat controller for orbit stabilization. Here we present the simplest derivation that uses the H-LIP based stepping in Eq. (3.17), which renders a wide class of controllers to stabilize the desired orbits of the H-LIP.

Note that on the P1 or P2 orbit, the *nominal system* is

$$\text{P1:} \quad \mathbf{x}^* = A\mathbf{x}^* + Bu^*, \tag{4.33}$$

$$\text{P2:} \quad \mathbf{x}^*_{\text{L/R}} = A\mathbf{x}^*_{\text{R/L}} + Bu^*_{\text{R/L}}. \tag{4.34}$$

Let the step controllers be the H-LIP based stepping controller:

$$\text{P1:} \quad u_{\text{P1}} = K(\mathbf{x} - \mathbf{x}^*) + u^*, \tag{4.35}$$

$$\text{P2:} \quad u_{\text{P2}} = K(\mathbf{x} - \mathbf{x}^*_{\text{R/L}}) + u^*_{\text{R/L}}. \tag{4.36}$$

The *closed-loop systems* are thus

$$\text{P1:} \quad \mathbf{x}_{k+1} = A\mathbf{x}_k + B(K(\mathbf{x} - \mathbf{x}^*) + u^*), \tag{4.37}$$

$$\text{P2:} \quad \mathbf{x}_{k+1} = A\mathbf{x}_k + B(K(\mathbf{x} - \mathbf{x}^*_{\text{R/L}}) + u^*_{\text{R/L}}). \tag{4.38}$$

Subtracting the *nominal systems* from the *closed-loop systems* yields the error dynamics with

$$\text{P1:} \quad \mathbf{e}^{\text{P1}}_{k+1} = A\mathbf{x}_k + Bu_{\text{P1}} - (A\mathbf{x}^* + Bu^*) = (A + BK)\mathbf{e}^{\text{P1}}_k, \tag{4.39}$$

$$\text{P2:} \quad \mathbf{e}^{\text{P2}}_{\text{L/R}_{k+1}} = A\mathbf{x}_{\text{R/L}_k} + Bu_{\text{P2}} - (A\mathbf{x}^*_{\text{R/L}} + Bu^*_{\text{R/L}}) = (A + BK)\mathbf{e}^{\text{P2}}_{\text{R/L}_k}, \tag{4.40}$$

with the error being defined as

$$\text{P1:} \quad \mathbf{e}^{\text{P1}}_k = \mathbf{x}_k - \mathbf{x}^*, \tag{4.41}$$

$$\text{P2:} \quad \mathbf{e}^{\text{P2}}_{\text{R/L}_k} = \mathbf{x}_{\text{R/L}_k} - \mathbf{x}^*_{\text{R/L}}. \tag{4.42}$$

Thus same error dynamics means that: as long as the closed-loop matrix $A + BK$ is stable, the system will be stabilized to the desired orbit. One can view the orbit stabilization of the H-LIP as stabilizing the H-LIP that is not on the orbit yet to another H-LIP that is evolving on the desired orbit. Since the dynamics difference w is zero, the error will be driven to zero.

Remark. *When K is the deadbeat gain, the step controllers are deadbeat controllers. So far, we have described three ways to derive the deadbeat controller. The last one is the most elegant one. Additionally, the feedback controllers in Eq.* (4.35) *and* (4.36) *also render various linear control methods as follows to function for orbit stabilization besides the LQR controller.*

Lyapunov-based Controller: One can also design a Lyapunov function on the error dynamics $\mathbf{e}_{k+1} = A\mathbf{e}_k + Bu^{\mathbf{e}}_k$ as

$$V_k = \mathbf{e}^T_k Q\mathbf{e}_k, \tag{4.43}$$

with Q being a positive definite matrix. The condition of making sure V decreases can be

$$V_{k+1} < \gamma V_k, \tag{4.44}$$

where $0 < \gamma < 1$ is a free chosen parameter for convergence. The closed-form solution can be derived:

$$u_k^{\mathbf{e}} = \frac{1}{B^T B}(-\mathbf{e}_k^T A^T Q B + \sqrt{(\mathbf{e}_k^T A^T Q B)^2 - B^T B(\mathbf{e}_k^T A^T Q A \mathbf{e}_k - \gamma \mathbf{e}_k^T \mathbf{e}_k)}), \tag{4.45}$$

which yields the step size controller to be $u = u_k^{\mathbf{e}} + u^*$.

Model Predictive Controller: The linear error dynamics also allows the Model Predictive Controller (MPC) [21, 78] to function for orbit stabilization. The use of MPC to stabilize the system to a specific point is a relatively standard technique. We thus omit the details here. A similar but more complex version of the MPC can be in Chapter 9 for global position tracking.

Comparison to Capture Point: The deadbeat stepping controller on the H-LIP is similar to the capture point controller [70, 100]. In the capture point controller, the step location is determined by the passive LIP model so that the robot can come to a stop, i.e., $v \to 0$ as $t \to \infty$. The walking with zero velocity on the H-LIP is the P1 orbit with $v^* = 0$. Additionally, if we assume $T_{\text{SSP}} \to \infty$ and $T_{\text{DSP}} \to 0$, the step size controller in Eq. (4.28) becomes identical to the instantaneous capture point controller:

$$u = p + \cancelto{0}{p^*} + \cancelto{0}{T_{\text{DSP}}} v + \frac{1}{\lambda}\cancelto{1}{\coth(T_{\text{SSP}}\lambda)}(v - \cancelto{0}{v^*}) = p + \frac{v}{\lambda}.$$

Thus, the capture point controller on the H-LIP is a special case of the deadbeat stepping controller on this model. More importantly, the capture point controller based on the LIP model is typically directly applied to the robot, whereas the stepping controller on the H-LIP is not directly applied on the robot but is used at the nominal step size $u^{\text{H-LIP}}$ in Eq. (3.17). The desired step size is then continuously calculated by Eq. (3.17) to construct the desired horizontal trajectory of the swing foot.

Chapter 5

APPLICATION ⊃ FIVE-LINKAGE WALKER AMBER

Starting from this chapter, we begin to present the application of the proposed motion synthesis approach on various bipedal walking systems. We start by applying it on a canonical bipedal robot model: the five linkage walker [134, 135]. To be specific, the model of the custom-built robot AMBER is used. Despite its simplicity, it has been treated as the fundamental rigid robot model for underactuated bipedal walking. It should be emphasized that the walking of AMBER has non-trivial impact between the swing foot and the ground. Despite the fact that the H-LIP has trivial impact, we show that the H-LIP based stepping can still be applied to generate walking on AMBER.

5.1 Robot Model

The robot AMBER has a torso, two thighs, and two shins. It is actuated by two hip joints and two knee joints. The contact with the ground is point-contact, and thus the walking is underactuated. The specifications of AMBER is shown in Fig. 5.1. The walking is described by a single SSP; in other words, the DSP is instantaneous. As we described in Chapter 2, the continuous dynamics is described

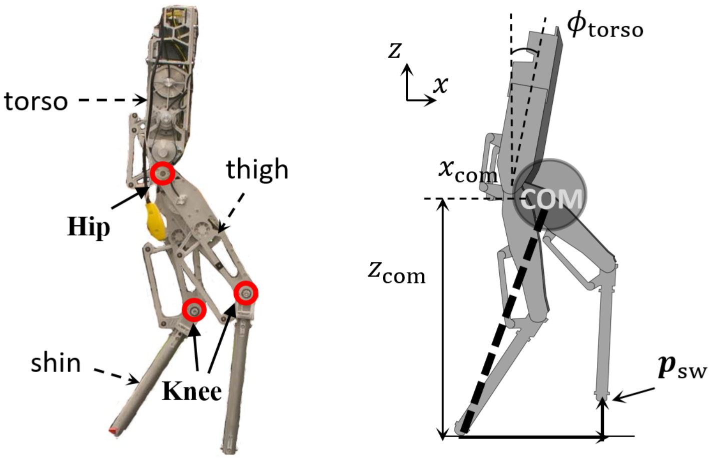

Figure 5.1: The five linkage walker AMBER and its output definition for walking

by the Euler-Lagrange equation with the floating-base coordinates:

$$M\ddot{q} + C + G = B_v\tau + J_h^T F_h, \tag{5.1}$$

$$J_h\ddot{q} + \dot{J}_h\dot{q} = 0, \tag{5.2}$$

where $q \in SE(2) \times \mathbb{R}^4$, $\tau \in \mathbb{R}^4$, $F_h \in \mathbb{R}^2$, and $J_h \in \mathbb{R}^{2\times 7}$. Eq. (5.2) describes the holonomic constraints that fix the stance toe position.

Impact Model: Recall that the impact happens when the swing toe strikes the ground. We assume that the impact is plastic: the swing toe stays on the ground, and the previous stance toe lifts off the ground. Thus, the stance toe and the swing toe swap at the impact. The post-impact velocity can be calculated from Eq. (2.9).

Both AMBER and another five linkage walker Rabbit [134] have been well-studied in the HZD [50, 134] framework. Recall that in the HZD, walking is described by periodic orbits, where the actuated trajectories are parameterized by the underactuated phase coordinate and then optimized via non-convex parameter optimization. The stability of individual periodic orbit is encoded via inequality constraints. Here, we show the first application of the proposed H-LIP based synthesis in Chapter 3 to generate walking on the five linkage walker AMBER.

5.2 Walking Synthesis

As there are four motors on AMBER, we construct four outputs to describe the walking. Based on the H-LIP based walking synthesis, we first consider the height of the center of mass (COM) z_{COM} as one output. Additionally, the swing toe trajectory $\mathbf{p}_{\text{sw}}$ should be constructed to create the periodicity of the walking; the swing toe periodically lifts off and strikes the ground with a fixed frequency, and it should point towards achieving the desired step size. The last output is the torso pitch angle ϕ_{torso}. Compactly, the outputs are constructed as

$$\mathcal{Y}_{\text{SSP}}(q,t) = \begin{bmatrix} z_{\text{COM}}(q) \\ \phi_{\text{torso}}(q) \\ \mathbf{p}_{\text{sw}}(q) \end{bmatrix} - \begin{bmatrix} z_{\text{COM}}^{\text{des}} \\ \phi_{\text{torso}}^{\text{des}} \\ \mathbf{p}_{\text{sw}}^{\text{des}}(q,t) \end{bmatrix}. \tag{5.3}$$

Here we set the desired COM height $z_{\text{COM}}^{\text{des}}$ and the desired torso angle $\phi_{\text{torso}}^{\text{des}}$ as constants. The desired horizontal trajectory of the swing toe is composed by two orthogonal components: $\mathbf{p}_{\text{sw}}^{\text{des}}(q,t) = [x_{\text{sw}}^{\text{des}}(q,t), z_{\text{sw}}^{\text{des}}(t)]^T$, which are constructed as follows.

Vertical Trajectory of the Swing Toe: The desired vertical trajectory is constructed to achieve a certain apex height (foot clearance) and a certain impact velocity at a certain instant. The non-trivial apex height is for preventing scuffing during walking. The non-trivial impact velocity ensures the executing of the impact event. This can be constructed via polynomial functions such as Bézier polynomials; see Chapter 7 for an example. Here we show a simple construction via sinusoidal and piece-wise linear functions. Let $z_{\text{sw}}^{\text{des}}(t) = z_1(t)+z_2(t)+z_3(t)$. The apex height and non-scuffing is realized by $z_1(t) = \frac{1}{2}(1-\cos(\frac{t}{2\pi}T))(z_{\text{sw}}^{\max} + \frac{T}{4}v_{\text{imp}})$, where $z_{\text{sw}}^{\max}$ is the apex height and T is the fixed duration. $T = T_{\text{SSP}}$ since the DSP is instantaneous. Note that $\dot{z}_1(t = T) = 0$. $z_2(t)$ is shown in Fig. 5.2 to realize the impact velocity $v_{\text{imp}} < 0$. Lastly, $z_3(t) = \frac{t}{T}z_{\text{sw}}^{\text{neg}}$ with $z_{\text{sw}}^{\text{neg}} < 0$ being a small value to ensure the actual toe strikes the ground under the feedback control.

Forward Trajectory of the Swing Toe: The construction of the desired forward position of the swing toe is to realize the H-LIP based stepping controller, which calculates the desired step size u^{des} given the current horizontal COM state in the SSP. Recall that the desired forward trajectory of the swing toe (w.r.t. the stance toe) is constructed from the initial lift-off position to the final position to have the desired step size:

$$x_{\text{sw}}^{\text{des}}(q,t) = (1-\mathcal{B}_h(t))x_{\text{sw}}^{+} + \mathcal{B}_h(t)u^{\text{des}}, \tag{5.4}$$

where x_{sw}^{+} is the initial position of the swing toe at the current step. Recall that $\mathcal{B}_h(t)$ is a Bézier polynomial function increasing from 0 to 1 within the period T.

Output Stabilization: Given the output construction, we apply the feedback controller in Chapter 2 for stabilizing the outputs $\mathcal{Y} \to 0$.

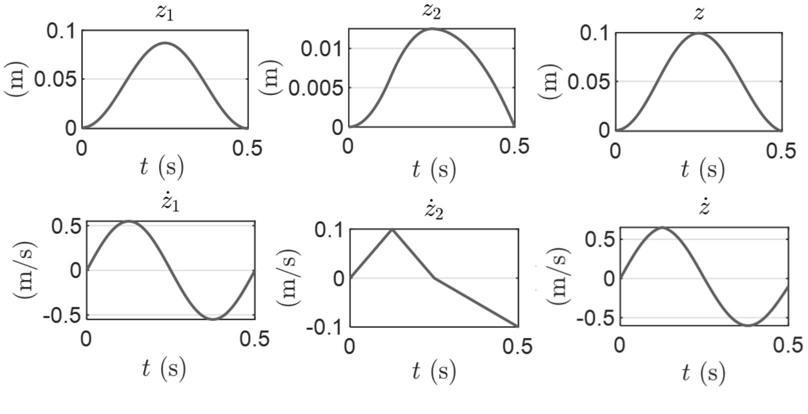

Figure 5.2: Trajectory construction on the swing toe of AMBER.

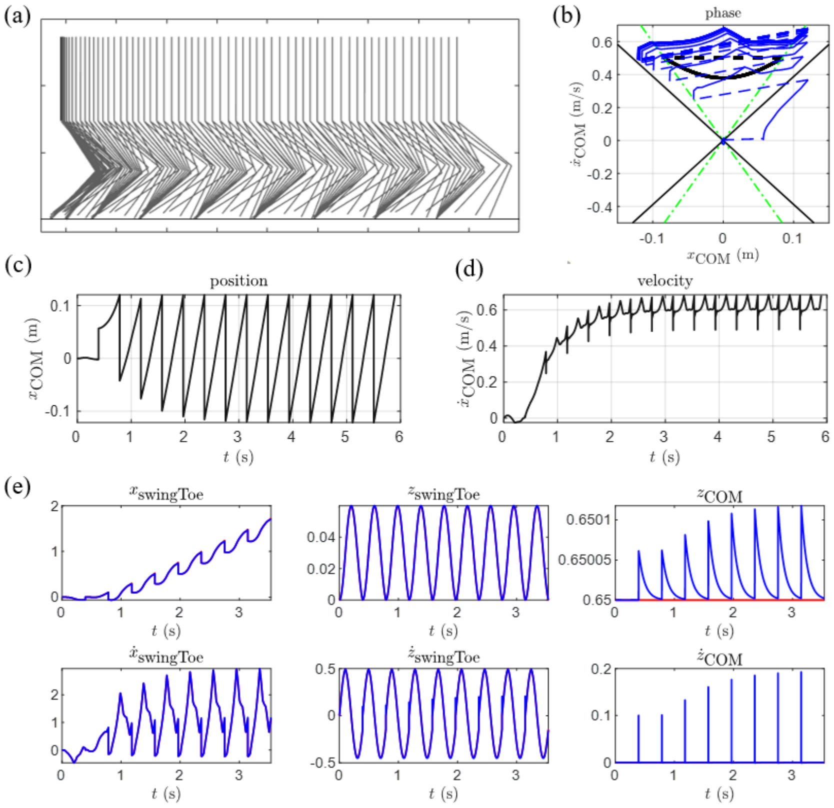

Figure 5.3: An example of a P1 realization to $v^d = 0.5$ (m/s): (a) the stick figures of the simulated walking, (b) the convergence of the orbits (blue) in terms of the forward COM state compared with the orbit (black) of the H-LIP, (c, d) the forward COM state trajectories vs time, and (e) the output tracking with the red indicating the desired trajectories and the blue indicating the actual trajectories.

5.3 Results and Conclusion

Based on the output construction, the parameters which determine the walking behavior are the height of the COM $z^{\rm des}_{\rm COM}$, the torso angle $\phi^{\rm des}_{\rm torso}$, the duration of walking T, the apex height of the swing toe $z^{\rm max}_{\rm sw}$, and the impact velocity of the swing toe $v_{\rm imp}$. We evaluate the approach in simulation. The dynamics is integrated via MATLAB `ODE 45` function with event-based triggering for contact detection. The optimization-based controller here is formulated via TSC-QP and solved via QPOASES [38] at 1kHZ. The robot starts from a static configuration at the beginning of a SSP where $x_{\rm COM} = 0$. The simulation results can be seen in `https://youtu.be/Dwz5iGDr0_w`.

The walking task is considered as achieving a certain periodic walking behavior on the horizontal COM states of the walker, which is realized by the H-LIP based stepping controller. We first specify the desired walking speed v^d and select the types of orbits of the H-LIP for realization. The H-LIP is first stabilized to the desired behavior via its stabilization. Then the H-LIP based stepping controller is applied to determine the desired step size. The output of the forward position of the swing toe is then constructed in Eq. (5.4). The rest of the outputs are constructed identically regardless of the desired behaviors.

P1 Orbit Realization: We first evaluate the H-LIP based walking synthesis to generate P1 walking on the walker to achieve different desired walking velocities v^d from -1 (m/s) to 1 (m/s) with a 0.2 (m/s) increment. The free chosen parameters are $T = 0.4$s, $z_{\text{COM}}^{\text{des}} = 0.65$m, $v_{\text{imp}} = -0.1$ (m/s), and $\phi_{\text{torso}} = 0°$. The maximum walking speed is limited by the maximum step size (size of the walker and the chosen COM height) and the step frequency $\frac{1}{T}$. All walking behaviors with different desired velocities are realized and stabilized from the static configuration. Fig. 5.3 shows an example of the realization. The walking converges to a periodic orbit. The shape of the converged orbit does not align exactly with the orbit of the H-LIP since the actual dynamics between the two are not the same, which is expected. The converged walking velocity is close to the desired one, shown in Fig. 5.4.

P2 Orbit Realization: The realization of P2 orbit is also evaluated. We select the desired velocity to change from -0.5 (m/s) to 0.5 (m/s) with a 0.2 (m/s) increment. The free chosen parameters are the same as those in the P1 realization. One step size is chosen as 0.1 m to determine the desired P2 orbit. The obvious characteristic of the P2 orbit is that it has two different alternating step sizes. Fig. 5.5 shows one simulated walking. The walking also converges to an orbit quickly. Even though the shape of the orbit is still not identical to the orbit of the H-LIP, it converged to its own P2 orbit. Various walking behaviors are thus realized.

Conclusion: In this chapter, we demonstrated the H-LIP based walking synthesis

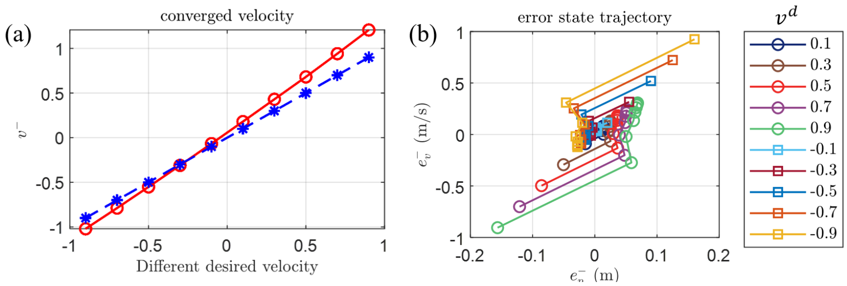

Figure 5.4: (a) The converged boundary velocities (red) of AMBER compared with that of the H-LIP (blue). (b) The convergence of the state error between the translational state of AMBER and that of the H-LIP.

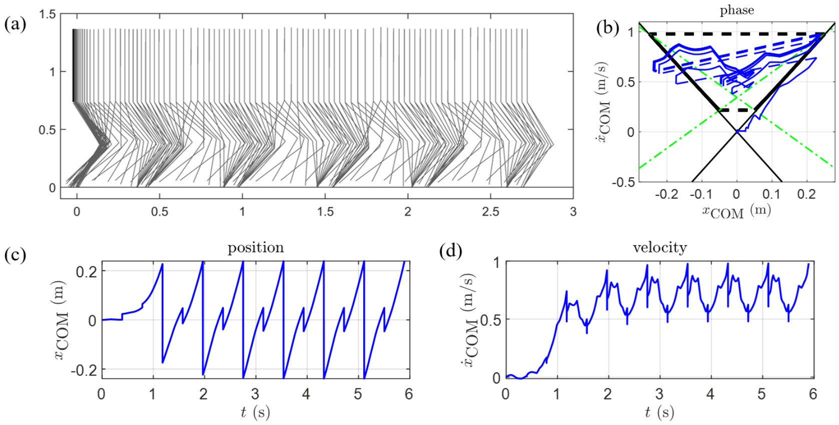

Figure 5.5: An example of a P2 realization to $v^d = 0.5$ (m/s): (a) the stick figures of the simulated walking, (b) the convergence of the orbits (blue) in terms of the forward COM state comparing with the orbit (black) of the H-LIP, and (c, d) the forward COM state trajectories vs time.

on the five linkage walker AMBER, where the underactuation is on the point foot contact. The horizontal dynamics of the COM in the continuous phase obviously is not the same as that of the H-LIP. Additionally, the impact is not trivial, which does not match the smooth transition assumption on the H-LIP. Thus the horizontal S2S dynamics of the walker is not the same as the S2S dynamics of H-LIP. Since the stepping makes sure that the state error is bounded in the disturbance invariant set, the walking on the walker can be realized to approximately follow the desired walking of the H-LIP. Even though the realized orbit is not the same, the walking is stabilized to the desired walking velocity approximately. No trajectory optimization is performed and all the walking behaviors are generated continuously. More importantly, the walking parameters such as the vertical COM and the step frequency can be changed without any modifications of the approach. In the latter chapters, we will evaluate this approach on more complex systems in more depth.

Chapter 6

APPLICATION $\supset$ ACTUATED SLIP

In this chapter, we present the application of the H-LIP based synthesis for generating walking on an Spring Loaded Inverted Pendulum (SLIP) model. Recall in Chapter 1 that the SLIP model has been widely studied in the biomechanics community for understanding the dynamics of human locomotion [44]. In robotic locomotion, it has also inspired robot designs [4, 58], [101] and control methodologies [42], [101], [55] for generating dynamic behaviors on bipedal robots.

Here, we add actuation and damping on the canonical SLIP model [44, 109] so that energy conservation can be avoided in order to easily create versatile walking behaviors. Our motivation of this modification comes from controlling jumping on the robot Cassie [138, 144]. It has also been shown to effectively approximate this class of SLIP-like robots such as ATRIAS [106] and Cassie [138] in the literature [26, 107, 125]. Similar modifications of the SLIP by adding leg length actuation can also be found in [48, 77, 94, 106, 119]. For the spring, we can let the stiffness and damping be constants or be nonlinear functions of the length of the leg, which comes from the specific leg spring on Cassie; see Appendix B. We name this SLIP in this thesis as the actuated Spring Loaded Inverted Pendulum (aSLIP) and the robots that it approximate as the aSLIP-like robots. Both the planar and 3D versions of aSLIP are studied.

We start by describing the mechanics model of the aSLIP and its hybrid dynamics model of walking in Section 6.1. Then, we apply the H-LIP based walking synthesis to generate walking on the aSLIP model. Unlike the robot AMBER where the vertical COM can be directly controlled, the vertical position of the mass of the aSLIP cannot be easily controlled as an output via the optimization-based controllers in Chapter 1. To solve this problem, we present an indirect approach in Section 6.2 and a direct approach in Section 6.2 to realize the H-LIP based synthesis on the aSLIP.

6.1 Model

The canonical SLIP model [39] is consisted by a point mass and two spring-loaded legs. The spring has no damping, and hence the system is energy-conservative. The

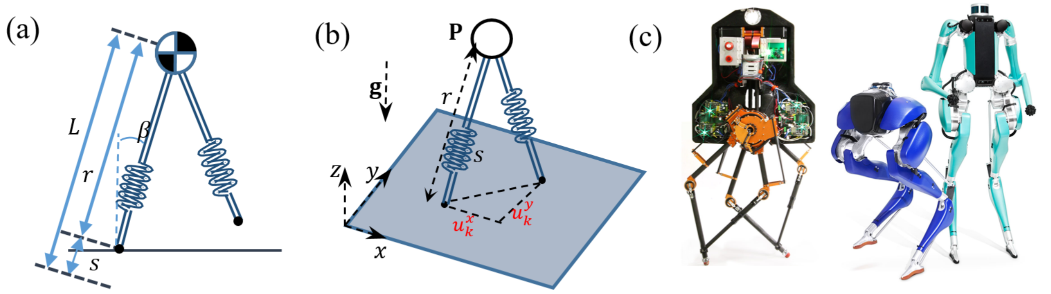

Figure 6.1: (a, b) The aSLIP model in 2D and in 3D, and (c) the aSLIP-like robots ATRIAS, Cassie, and Digit (Photo by Dan Saelinger).

aSLIP [138, 139] is an actuated SLIP with leg length actuation and damping. The dynamics of its point mass is

$$m\ddot{\mathbf{P}} = \sum \mathbf{F} + m\mathbf{g}, \tag{6.1}$$

where m is the mass of the aSLIP, $\mathbf{P}$ is the position vector of the point mass, $\mathbf{F} = \mathbf{F}_{\mathrm{L/R}}$ are the leg forces on the left and right legs, and $\mathbf{g}$ is the gravitational vector. The leg force is assumed to be zero when the leg is not in contact with the ground. The magnitude of the leg force is

$$|\mathbf{F}_{\mathrm{L/R}}| = \mathrm{K}s + \mathrm{D}\dot{s}, \tag{6.2}$$

where s is the spring deformation, and K and D are the stiffness and damping of the leg spring, respectively. The dynamics of the whole system can also be written in polar coordinates. For instance, the dynamics of the aSLIP in 2D in the SSP are

$$\ddot{r} = \frac{F}{m} - g\cos(\beta) + r\dot{\beta}^2, \tag{6.3}$$

$$\ddot{\beta} = \frac{1}{r}(-2\dot{\beta}\dot{r} + g\sin(\beta)), \tag{6.4}$$

$$\ddot{s} = \ddot{L} - \ddot{r}, \tag{6.5}$$

where β is the leg angle, L is the (uncompressed) leg length, and r is the distance between the point mass and the point of contact, i.e. the real leg length. Eq. (6.5) comes from the holonomic constraint that $s = L - r$. Fig. 6.1 shows the aSLIP model with its leg parameters. The complete expressions of the dynamics in the polar coordinates can be found in the Appendix A.

Actuation: There are a wide variety of actuated versions of the SLIP [76, 77, 94, 97] proposed in the literature for controlling robotic hopping, running, and walking. The forms of actuation are added in the leg to make better correspondences to the robotic systems. For this aSLIP, the actuation on the leg is in the form of changing the leg length. This comes from the physical design of the bipedal robot Cassie [139], which will be carefully explained in Chapter 7 and Appendix B. Mathematically, as a result of simplification on the aSLIP, we define the actuation as $\tau = \ddot{L}$. Additionally, the step size $u = [u^x, u^y]^T$ is another input of the system. Since the legs have no inertia, the step size can be set directly. In the literature, this has also been equivalently referred to touch-down angle of the swing leg [101, 126]. Lastly, it is obvious that the number of control inputs is much less than the DoFs of the model, which means that the aSLIP is highly underactuated.

Comparison to Canonical SLIP: The main differences of the aSLIP compared to the canonical SLIP are two folds. First, the aSLIP has damping for energy dissipation. Second, the actuation is enabled via the change of the leg length $\ddot{L}$. The aSLIP that comes from Cassie has nonlinear leg springs, where the stiffness and damping are functions of the leg length L, i.e., $\mathrm{K}(L), \mathrm{D}(L)$; we can also set K, D to be constants for generality.

Hybrid Model of Walking: The aSLIP model can generate both hopping/running or walking behaviors. In terms of walking, the behavior can be modeled as a hybrid dynamical system with two domains, i.e. the SSP and the DSP, depending on the number of legs contacting the ground. The transition from the DSP to the SSP happens when the ground reaction force on one leg reaches 0; the transition from the SSP to the DSP happens when the swing foot strikes the ground, i.e., the state of the system transverses the switching surface, which is defined by

$$\mathbf{S} = \{x^q \in \mathbb{R}^n | z_{\mathrm{sw}}(x^q) = 0, \dot{z}_{\mathrm{sw}}(x^q) < 0\}, \tag{6.6}$$

where x^q describes all the states of the aSLIP, and $z_{\mathrm{sw}}(x^q), \dot{z}_{\mathrm{sw}}(x^q)$ are the vertical position and velocity of the swing toe, respectively. The impact map and its assumptions can be found in the Appendix A.

6.2 Indirect Approach via Trajectory Optimization

In this section, we present the indirect approach via trajectory optimization to assist the realization of the H-LIP based walking synthesis on the aSLIP model. This indirect approach is primarily inspired by [106], where a hand-designed periodic leg length actuation is used to realize 3D walking with Raibert style stepping controller [101]. Here, we demonstrate that by optimizing a stepping-in-place motion and perturbing it by changing the step size based on the H-LIP based stepping, desired walking motion can be continuously generated. Compared to the application in Chapter 5 on AMBER, this indirect approach does not directly control the vertical COM or the vertical swing foot position. The realization of the H-LIP based synthesis is indirectly implemented by the repetition of the periodic leg length actuation.

Trajectory Optimization for Stepping-in-place

The trajectory optimization problem is formulated using direct collocation [56, 59] to find an optimal periodic solution for the aSLIP walking. The method is similar for 2D and 3D-aSLIP. In general, the trajectory optimization problem can be written as

$$a^* = \underset{a}{\operatorname{argmin}} \sum_{i=1}^{N} \frac{\Delta t}{2} J_{\text{walking}} \tag{6.7}$$

$$\text{s.t.} \quad a_{\min} \leq a \leq a_{\max}, \tag{6.8}$$

$$\mathbf{c}_{\min} \leq \mathbf{c}(a) \leq \mathbf{c}_{\max}, \tag{6.9}$$

where J is the cost function, i is the indexing for the discretized optimization variables, N is the number of nodes, Δt is the time discretization, a represents all the optimization variables, and $\mathbf{c}$ includes all the constraints. The discretization and integration methods are identical to these in [59, 138]: an even nodal spacing is used for discretizing the trajectory in time for each domain, and the defect constraints are described algebraically by implicit trapezoidal method. As we are interested in generating a periodic stepping-in-place motion (walking with zero velocity), continuity of states between the consecutive domains are enforced. As for the cost function, we find that it is desirable to minimize the virtually consumed energy on the aSLIP by defining the cost as

$$J_{\text{walking}} = \ddot{L}_{\text{L}}^{i2} + \ddot{L}_{\text{R}}^{i2}, \tag{6.10}$$

where i is the index of the nodes, and the subscript L and R again stand for the left and right leg, respectively. Additionally, the following constraints are included in Eq. (6.9).

<u>*Dynamic Constraints*</u>: The dynamics of the aSLIP are enforced via equality constraints via the trapezoidal integration method.

<u>*Task Constraints*</u>: The walking velocity is specified via the step size u by

$$u = v^d T, \tag{6.11}$$

where v^d is the desired net velocity and $T = T_{\mathrm{SSP}} + T_{\mathrm{DSP}}$ is the domain duration. For stepping-in-place motion, $v^d = 0$. The domain durations are also be constrained by

$$T_{\max} \geq T_{\mathrm{DSP/SSP}} \geq T_{\min}. \tag{6.12}$$

Physical Constraints: The leg length should be in a physically feasible range:

$$L_{\max} \geq L^i \geq L_{\min}. \tag{6.13}$$

The spring deformations are positive and limited by the maximum deformation, thus

$$s_{\max} \geq s^i \geq 0. \tag{6.14}$$

Additionally, the leg forces at contact should be non-negative:

$$F^i \geq 0. \tag{6.15}$$

Domain Constraints: The aSLIP transits from the DSP to the SSP when the previous stance foot is about to lift off. This is encoded by

$$F^{N_{\mathrm{DSP}}} = 0, \tag{6.16}$$

where N_{DSP} is the last index on the optimization variables in the DSP. It transits from the SSP to the DSP when the swing foot strikes the ground:

$$z_{\mathrm{sw}}^{N_{\mathrm{SSP}}} = 0, \;\; \dot{z}_{\mathrm{sw}}^{N_{\mathrm{SSP}}} < 0, \tag{6.17}$$

where N_{SSP} is the last index on the optimization variables in the SSP. Additionally, to ensure the motion is periodic, the continuity of the states between the domains should be enforced:

$$m^{N_{\mathrm{DSP}}} = m^{1_{\mathrm{SSP}}}, m^{1_{\mathrm{DSP}}} = \Delta(m^{N_{\mathrm{SSP}}}), \tag{6.18}$$

where $1_{\mathrm{SSP/DSP}}$ refers to the first index on the optimization variables in the DSP/SSP, and Δ represents the impact map from the SSP to the DSP; again see details in Appendix A.

The optimization problem is solved using IPOPT [128]. Fig. 6.2 shows an optimization result for a walking gait with $T_{\mathrm{SSP}} = 0.5s, T_{\mathrm{DSP}} = 0.15s$. Note that the optimization for generating a periodic walking is performed only once.

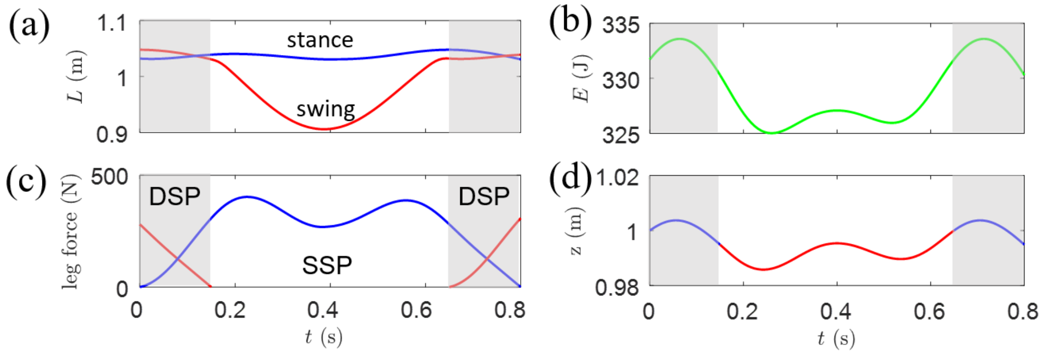

Figure 6.2: An example of the optimized walking on the 2D aSLIP in terms of the trajectories of the leg length (a), system energy (b), leg forces (c) and mass height (d).

Walking Generation via H-LIP based Stepping

The optimized trajectory can be viewed as a nominal walking behavior. Due to the high degrees of underactuation, it is not possible to control all states of the aSLIP to completely follow the nominal ones. To realize the optimized walking behavior and also consider the application of the H-LIP based walking synthesis, we choose the leg length as the output and control the step location to make sure that the horizontal state of the aSLIP evolves to the desired behavior. As the leg length is controlled to extend and contract periodically, the periodic lift-off and touch-down of the aSLIP can be preserved as long as the step location is reasonably selected.

Leg Length: The optimized trajectory of the leg length $L(t)^{\text{des}}$ is used as the desired output trajectory. As we assume the $\ddot{L}$ are the inputs, trajectory tracking of the leg length is realized via the linear feedback controller:

$$\ddot{L} = \ddot{L}^{\text{des}}(t) - K_p^L(L - L^{\text{des}}(t)) - K_d^L(\dot{L} - \dot{L}^{\text{des}}(t)), \tag{6.19}$$

where K_p^L, K_d^L are the PD feedback gains on the leg length controller.

Step Location: Since the swing leg of the aSLIP in the SSP has no dynamics, the step size can be directly set to the desired value. Similar to Eq. (5.4) in Chapter 5, the following smoothing is used to transit the swing leg to the desired step size so that the swing leg angle changes smoothly:

$$x_{\text{sw}}^{\text{des}}(t) = (1 - \mathcal{B}_h(t))x_{\text{sw}}^{+} + \mathcal{B}_h(t)u^{\text{des}}, \tag{6.20}$$

where $x_{\text{sw}}^{\text{des}}(t)$ is the desired swing foot position relative to the stance foot in the SSP, u^{des} is the desired step size from the H-LIP based stepping controller in Eq. (3.17),

$\mathcal{B}_h(t)$ again is the Bézier polynomial from 0 to 1 with appropriate timing, and x_{sw}^{+} is the initial swing toe position at the current step.

Realization: Before realizing the proposed approach, we outline the procedures here. The aSLIP is simulated from the initial state of the optimized stepping-in-place motion. The leg length is controlled to follow the optimized periodic trajectory. Given a desired walking behavior, the desired pre-impact state of the H-LIP is calculated. Then, the H-LIP based stepping is continuously calculated to provide the desired step size in the SSP. The leg angle is set to realize the step size.

Now we apply the proposed approach for continuously generating walking motion on the aSLIP model. We first present the results on the 2D-aSLIP model with both P1 and P2 orbits as the target orbits. The performance of the H-LIP stepping controller is also evaluated. We then compare the H-LIP based stepping with the canonical Raibert style stepping controller [101, 106]. Lastly, we realize the approach to generate walking on the 3D-aSLIP via two decoupled H-LIP stepping controllers in the $x - z$ and $y - z$ planes.

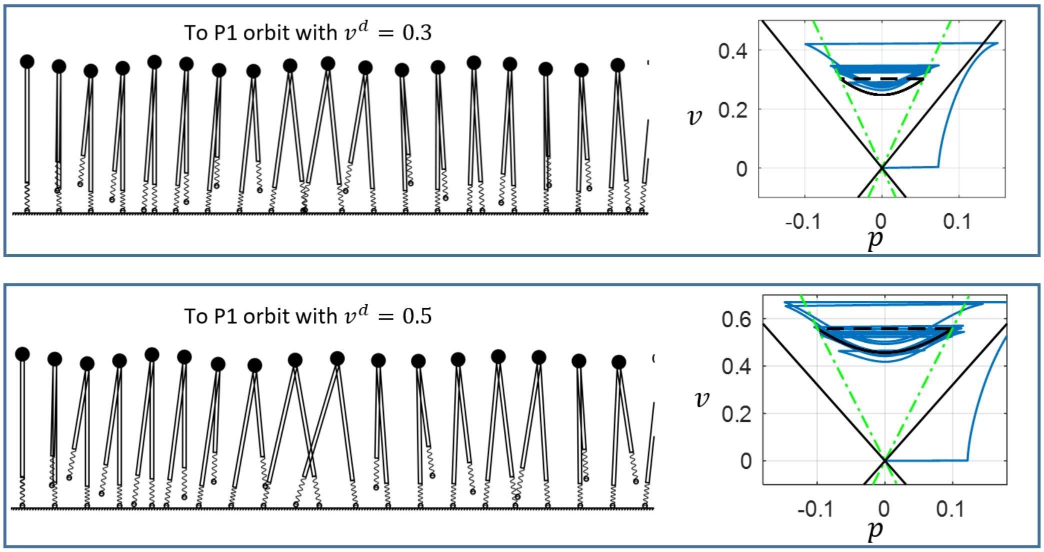

Figure 6.3: P1 orbit stabilization: (left) the time-lapse figures of walking from stepping-in-place to various speeds, and (right) the phase plots of the mass state $[p, v]$ of the walking. The black orbits indicate the desired orbits of the H-LIP for the same velocities, and the blue curves are the state trajectories of the aSLIP.

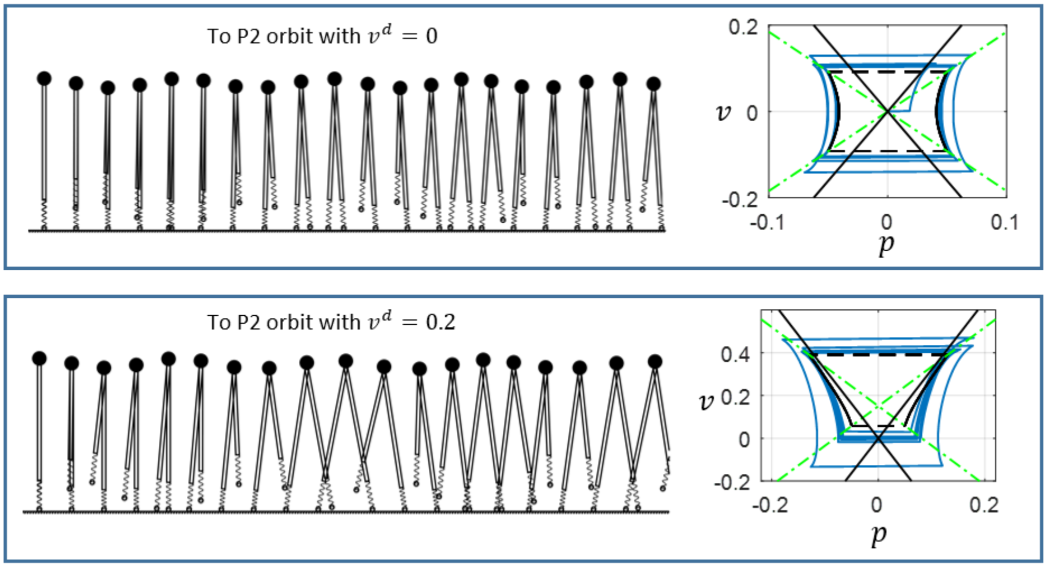

Figure 6.4: P2 orbit stabilization: (left) the time-lapse figures of walking from stepping-in-place to various speeds, and (right) the phase plots of the mass state $[p, v]$ of the walking. The black orbits indicate the desired orbits of the H-LIP for the same velocities, and the blue curves are the state trajectories of the aSLIP.

Results on 2D-aSLIP Walking Generation

Here, we demonstrate that both P1 and P2 orbits can be stabilized with negligible velocity errors. The stepping-in-place motion represents the origin of the H-LIP in its state space. The H-LIP based stepping stabilizes the aSLIP walking smoothly from the origin to different orbits with different speeds. Fig. 6.3 and 6.4 show the stabilization of the aSLIP walking from 0 m/s to P1 and P2 orbits with different velocities. The aSLIP walks to the desired orbit of the H-LIP closely. Fig. 6.5 illustrates the convergence of the stabilization for two walking simulations in terms of the leg forces, velocities, domain durations, and step sizes.

Since the leg length repeats the same trajectory from the stepping-in-place motion, the sum of the durations of the SSP and the DSP is guaranteed to be consistent across any orbits that are generated from the stepping controller. It is expected that the leg internal behavior varies smoothly w.r.t. the variation of the forward velocity, so is the whole system behavior. Fig. 6.6 shows the stabilization to P1 orbits with velocities from 0.1 – 0.9 m/s. The converged $T_{\text{SSP}} \searrow$ smoothly and the step size $u \nearrow$, as $v^d \nearrow$ smoothly. All converged velocity errors are within 10%, indicating the success of gait generation and stabilization via H-LIP base approach.

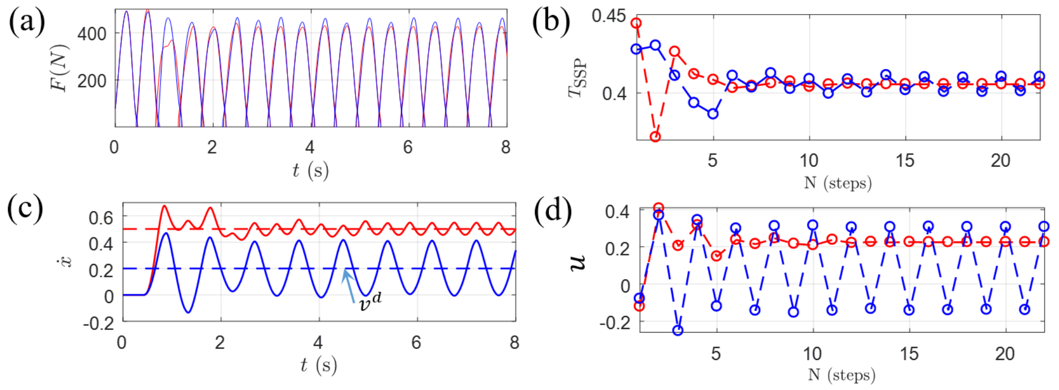

Figure 6.5: (a) The leg forces F v.s. time of the P1 walking in Fig. 6.3 with $v^d = 0.3$m/s. (b) The duration of the SSP T_{SSP}, (c) the mass velocity $\dot{x}$, and (c) the step size u v.s. time of the walking of the aSLIP; the red indicates the walking of P1 orbit with $v^d = 0.5$ (m/s), and the blue indicates the walking of P2 orbit with $v^d = 0.2$ (m/s).

Comparison with Raibert controller: We also compare the H-LIP stepping stabilization with canonical Raibert style stepping controller [97, 101], in the form of $u^d = u_0 + K_p(v_k - v^d) + K_d(v_k - v_{k-1})$, where u_0 is the nominal optimized step size, K_p, K_d are the tunable proportional and derivative gains and $k, k-1$ index the current step and previous step. The optimal $K_p^* = 0.1832$ is chosen from the H-LIP stepping in Eq. (4.15), (4.21); the K_d is set within $[-0.04, 0.04]$. The applied step size is used as the criterion on the stabilization. As the results in the figure indicate, it is not evident if the *derivative* term can improve the stabilization. The observation

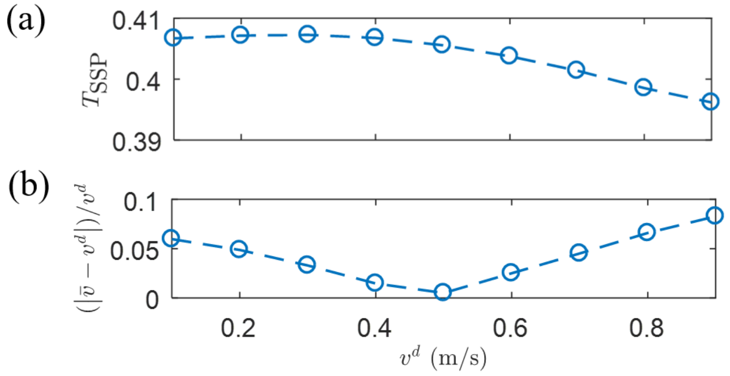

Figure 6.6: (a) The time-lapse figures of walking from stepping-in-place to various speeds. (b) The phase plots of the mass state $[x, \dot{x}]$ of the walking. The black orbits indicate the desired orbits of the H-LIP for the same velocities.

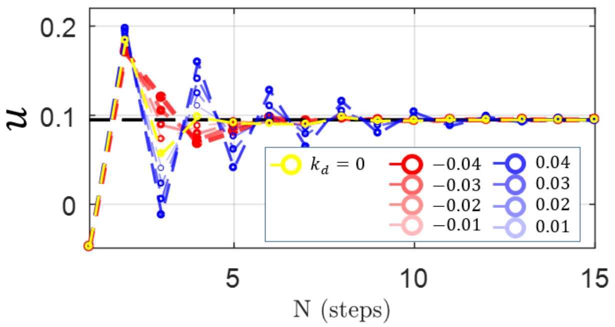

Figure 6.7: The step size u under the Raibert controller with different gains.

is that: the H-LIP can already stabilize the system; secondly, inappropriate selection of K_d can destabilize the system. Thus, the additional *derivative* term may be a *placebo* that has existed in previous literature. It may be useful in practice to deal with the imperfection of the noises on the hardware; however, in our realization of the approach on the hardware of Cassie in Chapter 8, we do not see the necessity of adding the derivative term.

Results on 3D-aSLIP Walking Generation

To realize 3D walking on aSLIP, we first select and compose P1 and P2 orbits of the H-LIP into 3D. Two types of orbits in two planes ($x - z$ and $y - z$) provide three possible combinations, i.e., two P1 orbits in 3D (P1-P1), two P2 orbits (P2-P2), and one P1 and one P2 orbit (P1-P2). In order to avoid kinematic collisions between the feet, the nominal step size should be non-trivial. Thus P1-P1 is not possible to realize; step sizes are 0 for zero speed walking, which creates violations. P2-P2 and P1-P2 can be realized. More about the orbit composition can be seen in Chapter 7 for the application on the 3D robot Cassie.

To illustrate the process, we use the P1-P2 composition as an example. Suppose the desired 3D walking behavior is to walk forward with 0.3m/s speed. A P1 orbit is selected in the $x - z$ plane with $v^d = 0.3$m/s. A P2 orbit is selected in the $y - z$ plane with $v^d = 0$m/s. One of the nominal step sizes for the P2 orbit is selected to determine the P2 orbit. Given the desired behaviors of the H-LIPs, we apply the stepping controller in Eq. (3.17) to generate desired step sizes in $x - z$ and $y - z$ planes separately for the aSLIP.

Similarly, we let the 3D-aSLIP start from the stepping-in-place walking, which is obtained via another trajectory optimization problem on the 3D-aSLIP. The actuation of the leg length remains periodically without modification. Executing the step sizes

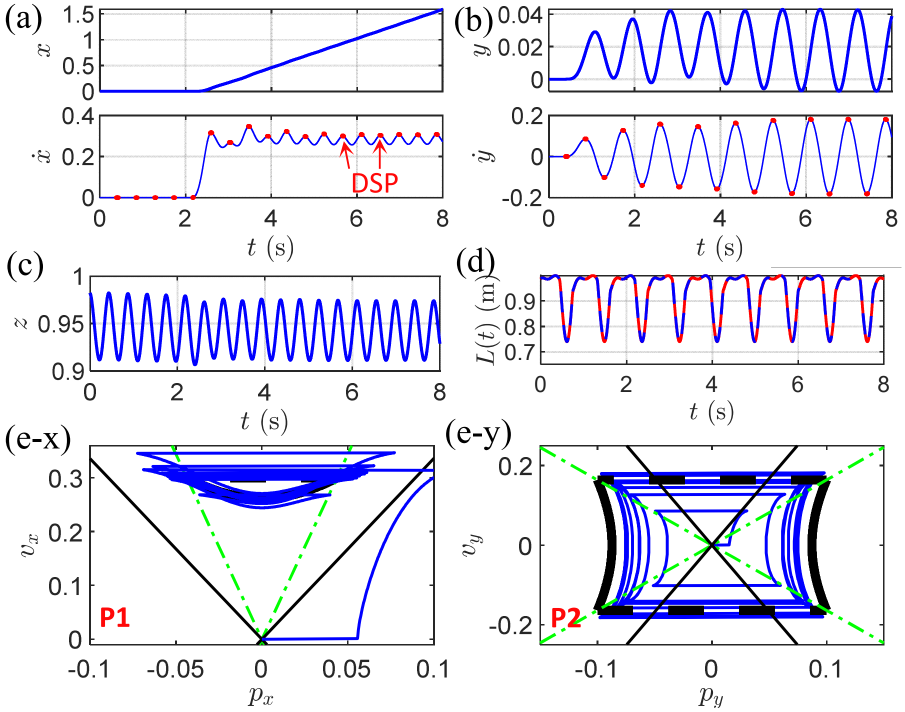

Figure 6.8: An example of the generated 3D periodic walking: (a-b) horizontal trajectories of the mass vs time (the red indicates the velocities in the DSP), (c) the vertical trajectory of the mas, (d) the leg length tracking (the red and blue are the desired and actual L, respectively), and (e) the phase portraits of the mass (blue) in the $x - z$ plane (e-x) and $y - z$ plane (e-y) comparing to the desired orbits of the H-LIP (black).

from the stepping controller perturbs the stepping-in-place walking into desired 3D walking. Fig. 6.8 shows the simulation result. The aSLIP starts to execute the forward walking at $t = 3$s. As indicated in the phase plot in Fig. 6.8 (d1-d2), the translational states of the aSLIP converge closely to the desired P1 and P2 orbits of the H-LIP in each plane. 3D walking is thus generated on the aSLIP.

6.3 Direct Approach via Backstepping Barrier Functions

The indirect approach was first realized in our previous publication [141, 143]. Later, we developed a direct approach to directly control the vertical COM of the aSLIP so that the H-LIP based walking synthesis can be easily realized on the highly underactuated system of the aSLIP. Moreover, in Chapter 10, we will show that the direct approach can be applied with the H-LIP to realize walking of the aSLIP on rough terrain. In the following, we will present the Backstepping Barrier function based Quadratic Program (BBF-QP) for directly controlling the vertical

COM state of the aSLIP, and then present the results to realize the aSLIP walking via the H-LIP based synthesis. The periodic vertical trajectory of the swing foot $z_{\text{sw}} = z - L_{\text{sw}}\cos(\beta_{\text{sw}})$ is controlled via a feedback linearizing controller. Similar to that on AMBER in Chapter 5, its desired trajectory is simply constructed to be $z^{\text{des}}_{\text{sw}} = z^{\max}_{\text{sw}}\cos(\frac{t}{T_{\text{SSP}}}\pi - \frac{1}{2}\pi)$, where $z^{\max}_{\text{sw}} > 0$ is chosen to avoid scuffing.

Backstepping-Control Lyapunov Function for Vertical Stabilization

In this part, we present the dynamics structure of the vertical mass state in each domain. Then, we show the canonical Lyapunov backstepping that guarantees to stabilize this class of control problem. Lastly, we formulate a control Lyapunov backstepping, which yields an inequality condition for the control and opens opportunities to include extra constraints on the input in an optimization.

Strict-feedback Form of the Vertical State: The objective is to drive the vertical position of the mass to follow a desired trajectory. We define the output as

$$\eta := \begin{bmatrix} z - z^{\text{des}}(t) \\ \dot{z} - \dot{z}^{\text{des}}(t) \end{bmatrix}, \tag{6.21}$$

where $[z^{\text{des}}(t), \dot{z}^{\text{des}}(t)]$ is the desired vertical trajectory for the point mass of the aSLIP to follow. For walking on flat ground, z^{des} is a constant. Differentiating the output yields the output dynamics:

$$\dot{\eta} = f_\eta + g_\eta F^{\text{SSP/DSP}}_z, \tag{6.22}$$

$$f_\eta := \begin{bmatrix} \dot{z} - \dot{z}^{\text{des}} \\ -\text{g} - \ddot{z}^{\text{des}} \end{bmatrix}, \quad g_\eta := \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix}. \tag{6.23}$$

$F^{\text{SSP/DSP}}_z$ is the net vertical force in each domain:

$$\text{SSP:} \quad F^{\text{SSP}}_z = F_s\cos(\beta_s), \tag{6.24}$$

$$\text{DSP:} \quad F^{\text{DSP}}_z = F_{s_1}\cos(\beta_{s_1}) + F_{s_2}\cos(\beta_{s_2}), \tag{6.25}$$

where the stance leg is denoted by the subscript s. Recall β is the leg angle. We view $F^{\text{SSP/DSP}}_z$ as the fictitious input to this system in Eq. (6.22). Differentiating $F^{\text{SSP/DSP}}_z$ yields the affine dynamics with the actual actuation of the leg length as the input. The dynamics are different in the SSP and DSP:

$$\dot{F}^{\text{SSP}}_z = f^{\text{SSP}}_z + g_z\tau^{\text{SSP}}_z, \tag{6.26}$$

$$\dot{F}^{\text{DSP}}_z = f^{\text{DSP}}_z + g_z\tau^{\text{DSP}}_z, \tag{6.27}$$

where $g_z = \mathrm{D} \neq 0$, and,

$$\tau_z^{\mathrm{SSP}} := \cos(\beta_s)\tau_s, \tag{6.28}$$

$$\tau_z^{\mathrm{DSP}} := \cos(\beta_{s_1})\tau_{s_1} + \cos(\beta_{s_2})\tau_{s_2}. \tag{6.29}$$

The expressions of f_z^{SSP} and f_z^{DSP} are omitted. $\tau_z^{\mathrm{SSP/DSP}}$ represents the vertical component of the leg length actuation. The leg angles $\beta \in (-\frac{\pi}{2}, \frac{\pi}{2})$, thus given a desired $\tau_z^{\mathrm{SSP/DSP}}$, there always exists leg length actuation for realization. Note that the desired τ_z^{DSP} is not uniquely realized by the leg length actuation τ_{s_1} and τ_{s_2}. Note that here we use τ_s represent the leg length actuation on the stance leg in the SSP, and thus τ_{s_1} and τ_{s_2} represent the leg length actuation on the two stance legs in the DSP.

As a result, for walking in both domains, we can synthesize $\tau_z \in \mathbb{R}$ to stabilize the vertical trajectory of the mass based on the dynamics in the *strict-feedback form* [66]:

$$\boxed{\begin{aligned} \dot{\eta} &= f_\eta + g_\eta F_z, \\ \dot{F}_z &= f_z + g_z \tau_z. \end{aligned}} \qquad \begin{aligned} &(6.30) \\ &(6.31) \end{aligned}$$

Lyapunov Backstepping: Lyapunov Backstepping [67] can be applied to stabilize the dynamics in Eq. (6.30) and (6.31). For the dynamics in Eq. (6.30) with F_z being the input, a feedback linearization controller can be synthesized:

$$\bar{F}_z = \frac{1}{g_{\eta 2}}(-f_{\eta 2} + K_{\mathrm{IO}}\eta), \tag{6.32}$$

where $K_{\mathrm{IO}} = [K_p, K_d]$ is the linear feedback gain and the subscript $_2$ indicates the second element of the vector. This yields the linear closed-loop dynamics:

$$\dot{\eta} = f_\eta + g_\eta \bar{F}_z = \begin{bmatrix} 0 & 1 \\ K_p & K_d \end{bmatrix} \eta := A_{cl}\eta. \tag{6.33}$$

Thus K_{IO} is chosen with $K_p < 0, K_d < 0$ so that A_{cl} is stable (with negative eigenvalues). On the closed-loop dynamics, a Lyapunov function can be found: $V_\eta = \eta^T P \eta$, with $P > 0$ (being positive definite). P satisfies the continuous-time Lyapunov function $PA_{cl} + A_{cl}^T P = -Q$, where $Q > 0$ is selected. It is easy to verify that:

$$\dot{V}_\eta = -\eta^T Q \eta \leq -\lambda_{\min}(Q)||\eta||^2, \tag{6.34}$$

where $\lambda_{\min}(Q)$ is the smallest eigenvalue of the matrix Q.

To synthesize the actual control from τ_z to stabilize $\eta \rightarrow 0$, we define

$$F_\delta := F_z - \bar{F}_z, \tag{6.35}$$

and the Lyapunov function to be

$$V(\eta, F_z) = V_\eta + \frac{1}{2}F_\delta^2. \tag{6.36}$$

Differentiating this yields

$$\begin{aligned}
\dot{V}(\eta, F_z) &= \frac{\partial V_\eta}{\partial \eta}(f_\eta + g_\eta F_z) + F_\delta \dot{F}_\delta \\
&= \frac{\partial V_\eta}{\partial \eta}(f_\eta + g_\eta \bar{F}_z) + \frac{\partial V_\eta}{\partial \eta} g_\eta F_\delta + F_\delta \dot{F}_\delta \\
&= \dot{V}_\eta + \frac{\partial V_\eta}{\partial \eta} g_\eta F_\delta + F_\delta \dot{F}_\delta \qquad (6.37) \\
&\leq -\lambda_{\min}(Q)||\eta||^2 + \frac{\partial V_\eta}{\partial \eta} g_\eta F_\delta + F_\delta \dot{F}_\delta. \qquad (6.38)
\end{aligned}$$

If we choose

$$\dot{F}_\delta = -\frac{\partial V_\eta}{\partial \eta} g_\eta - kF_\delta, \tag{6.39}$$

with $k > 0$, then

$$\begin{aligned}
\dot{V}(\eta, F_z) &\leq -\lambda_{\min}(Q)||\eta||^2 - kF_\delta^2 \\
&\leq -\min(\lambda_{\min}(Q), k)||[\eta^T, F_\delta]^T||^2. \qquad (6.40)
\end{aligned}$$

By Lyapunov's method, the system with (η, F_δ) as states is exponentially stable to the origin $(\eta, F_\delta) = (\mathbf{0}, 0)$. This provides a closed-form *Backstepping controller* on τ_z from Eq. (6.39). Since $\dot{F}_\delta = f_z + g_z\tau_z - \dot{\bar{F}}_z$, the controller is

$$\tau_z = \frac{1}{g_z}\left(-\frac{\partial V_\eta}{\partial \eta} g_\eta - kF_\delta + \dot{\bar{F}}_z - f_z\right). \tag{6.41}$$

Backstepping-CLF: The closed-form controller in Eq. (6.41) appears to be able to stabilize $\eta \rightarrow 0$. However, the resultant leg force F_z can be negative, which is not valid for walking. Here we develop the control Lyapunov function (CLF) of the backstepping to provide an *inequality condition* on the input τ_z for stabilizing η. The inequality motivates an optimization-based controller, in which the condition of non-negative leg forcing can be enforced via additional constraints, e.g., the control barrier function (CBF) in the next part.

Note that $\dot{V}$ is an affine function w.r.t. the input τ_z:

$$\begin{aligned}\dot{V}(\eta, F_z) &= \dot{V}_\eta + \frac{\partial V_\eta}{\partial \eta} g_\eta F_\delta + F_\delta (f_z + g_z \tau_z - \dot{\bar{F}}_z)\\ &= \dot{V}_\eta + \frac{\partial V_\eta}{\partial \eta} g_\eta F_\delta + F_\delta (f_z - \dot{\bar{F}}_z) + F_\delta g_z \tau_z.\end{aligned}$$

The exponential stability can be established via enforcing

$$\dot{V}(\eta, F_z) \leq -\gamma V(\eta, F_z), \tag{6.42}$$

with some chosen $\gamma > 0$. This yields a *Backstepping-CLF inequality*:

$$\boxed{A_{\text{CLF}}^{\text{Backstepping}} \tau_z \leq b_{\text{CLF}}^{\text{Backstepping}},} \tag{6.43}$$

where $A_{\text{CLF}}^{\text{Backstepping}} := F_\delta g_z$, $b_{\text{CLF}}^{\text{Backstepping}} := -\dot{V}_\eta - \frac{\partial V_\eta}{\partial \eta} g_\eta F_\delta - F_\delta (f_z - \dot{\bar{F}}_z) - \gamma V$. When $F_\delta \neq 0$, Eq. (6.43) is a constraint on τ_z. When $F_\delta = 0$, the inequality becomes $\dot{V}_\eta \leq -\gamma V = -\gamma V_\eta$, which is automatically satisfied as long as $0 \leq \gamma \leq \frac{\lambda_{\min}(Q)}{\lambda_{\max}(P)}$. As a result, as long as τ_z satisfies the backstepping-CLF inequality, η exponentially converges to 0. Note that this inequality is an affine condition on τ_s in SSP or τ_{s1} and τ_{s2} in DSP as indicated by Eq. (6.28). Thus, in the next part, we will formulate quadratic program (QP) based controllers that include the inequality in Eq. (6.43) with the incorporation of the control barrier functions.

Control Barrier Functions for Walking

In the application of walking, the leg forces should be positive during contact. Moreover, in the DSP, one leg force should gracefully cross 0 to initiate lift-off. These conditions can be described via sets and thus be enforced via control barrier function (CBF) with an inequality condition that guarantees set invariance on the dynamics. We start by introducing the CBF, show the application for the walking of the aSLIP, and finally integrate it with the Backstepping-CLF to formulate the final backstepping-barrier function based quadratic program (BBF-QP) controllers for walking.

Control Barrier Functions: The control barrier function [11] describes a condition for control input that guarantees for set invariance. We consider a super level set C of a continuously differentiable scalar function $h : \mathbb{R}^n \rightarrow \mathbb{R}$. By definition: $C = \{x \in \mathbb{R}^n | h(x) \geq 0\}$. Here we use x for a general state representation, instead of the horizontal position of the aSLIP. h is a *control barrier function* for the affine control system $\dot{x} = f(x) + g(x)u$ if

$$\sup_{u \in U} [\mathcal{L}_f h(x) + \mathcal{L}_g h(x) u + \alpha(h(x))] \geq 0, \tag{6.44}$$

where $\mathcal{L}_f h(x) = \frac{\partial h}{\partial x} f(x)$ and $\mathcal{L}_g h(x) = \frac{\partial h}{\partial x} g(x)$ are the Lie derivatives; $\mathcal{L}_f h(x) + \mathcal{L}_g h(x) = \dot{h}(x)$. U is the set where the input u is in, and $\alpha(\cdot)$ is an extended class $\mathcal{K}_\infty$ function[1]. This condition indicates that there exists an input to stabilize the set C, i.e., making sure $h(x) \geq 0$. If the state is in C, it will stay in the set forever if the *CBF inequality* is satisfied:

$$\boxed{\mathcal{L}_f h(x) + \mathcal{L}_g h(x) u \geq -\alpha(h(x)).} \tag{6.45}$$

This makes sure that the lower bound of the derivative $\dot{h}$ is increasing with the decrease of h. It can be proven that the set C is exponentially stable under this condition [11].

CBF for aSLIP Walking: Eq. (6.45) represents an inequality constraint on the input to make sure that $h \geq 0$, for which h is defined differently for the walking in the SSP and the DSP.

<u>*SSP:*</u> The stance leg force F_s should be non-negative, so is its vertical component F_z^{SSP}. Thus, we let

$$h_s = F_z^{\text{SSP}}, \quad \dot{h}_s \geq -\alpha(h_s), \tag{6.46}$$

which provides an inequality on the input τ_s:

$$A^s_{\text{CBF}} \tau_s \leq b^s_{\text{CBF}}, \tag{6.47}$$

$$A^s_{\text{CBF}} := -g_z \cos(\beta_s), \tag{6.48}$$

$$b^s_{\text{CBF}} := f_z^{\text{SSP}} + \alpha(h_s), \tag{6.49}$$

where we simply select $\alpha(\cdot)$ to be a linear function, i.e., $\alpha(h_s) = \alpha h_s$ This inequality naturally fits with the Backstepping-CLF inequality in Eq. (6.43) to formulate a backstepping-barrier function based quadratic program (BBF-QP) controller:

$$\boxed{\begin{aligned} (\tau_s, \delta) = &\underset{(\tau_s,\delta)\in\mathbb{R}^2}{\text{argmin}} \ \tau_s^2 + \delta^2 \\ \text{s.t.} \quad &A^{\text{Backstepping}}_{\text{CLF}} \tau_z^{\text{SSP}} \leq b^{\text{Backstepping}}_{\text{CLF}} + \delta, \\ &A^s_{\text{CBF}} \tau_s \leq b^s_{\text{CBF}}, \end{aligned}} \tag{6.50}$$

where δ is a relaxation variable to avoid infeasibility, similar to that in the CLF-QP in Chapter 2. In the case when the CBF constraint violates the Backstepping-CLF constraint, the Backstepping-CLF constraint is relaxed and the CBF constraint is still enforced.

[1] $\alpha : \mathbb{R} \rightarrow \mathbb{R}$, $\alpha(0) = 0$ and α is strictly monotonically increasing.

DSP: Both leg forces should remain non-negative. The leg force on s_2 should remain non-negative through out the DSP. Thus $h_{s_2} = F_{s_2}$ and the CBF inequality is

$$A^{s_2}_{\text{CBF}}\tau_{s_2} \leq b^{s_2}_{\text{CBF}}. \tag{6.51}$$

The leg force on s_1 should gradually decrease and reach to zero to trigger the transition into the next SSP. Let $F^0_{s_1}$ be the leg force on s_1 in the beginning of the DSP. A desired leg force trajectory can be designed: $F^d_{s_1}(t) = F^0_{s_1}(1 - \frac{t}{T_{\text{DSP}}})$. One may consider to design a feedback controller to drive $F_{s_1} \to F^d_{s_1}$. However, this creates high restriction on τ_{s_1} and can lead to conflict between the Backstepping-CLF inequality in Eq. (6.43) and the CBF inequality in Eq. (6.51).

To decrease F_{s_1} in a relaxed fashion, we create the inequality condition: $(1-c)F^d_{s_1} \leq F_{s_1} \leq (1+c)F^d_{s_1}$, where $c \in (0,1)$ is a relaxation coefficient. As shown in Fig. 6.9, this generates an admissible force region (indicated by the blue region), which decreases as the desired force $F^d_{s_1}$ decreases with time. This two inequalities can be converted into a single inequality: $h_{s_1} = (cF^d_{s_1})^2 - (F_{s_1} - F^d_{s_1})^2 \geq 0$. Note that this barrier function is ill-defined as $F^d_{s_1}$ approaches to 0 (the set C becomes trivial). Thus we increase the relaxation by adding a positive value ΔF in the inequality:

$$(1-c)F^d_{s_1} - \Delta F \leq F_{s_1} \leq (1+c)F^d_{s_1} + \Delta F, \tag{6.52}$$

which generates the red admissible region. By defining

$$h_{s_1} = (cF^d_{s_1} + \Delta F)^2 - (F_{s_1} - F^d_{s_1})^2 \geq 0, \tag{6.53}$$

the set C is always non-trivial before lift-off. Thus we have another CBF inequality: $A^{s_1}_{\text{CBF}}\tau_{s_1} \leq b^{s_1}_{\text{CBF}}$.

Similarly, the two CBF inequalities are incorporated with the Backstepping-CLF inequality to formulate the final BBF-QP controller for the walking in DSP:

$$\boxed{\begin{aligned}(\tau_{s_1}, \tau_{s_2}, \delta) = &\underset{(\tau_{s_1},\tau_{s_2},\delta)\in\mathbb{R}^3}{\text{argmin}} \quad \tau^2_{s_1} + \tau^2_{s_2} + \delta^2 \\ &\text{s.t.} \quad A^{\text{Backstepping}}_{\text{CLF}}\tau^{\text{DSP}}_z \leq b^{\text{Backstepping}}_{\text{CLF}} + \delta \\ &\qquad\; A^{s_1}_{\text{CBF}}\tau_{s_1} \leq b^{s_1}_{\text{CBF}}, A^{s_2}_{\text{CBF}}\tau_{s_2} \leq b^{s_2}_{\text{CBF}}.\end{aligned}} \tag{6.54}$$

Despite the complexity in the derivation, the BBF-QPs are designed simply to stabilize the vertical position of the mass to the desired trajectory and simultaneously satisfy the conditions on the leg forces during walking.

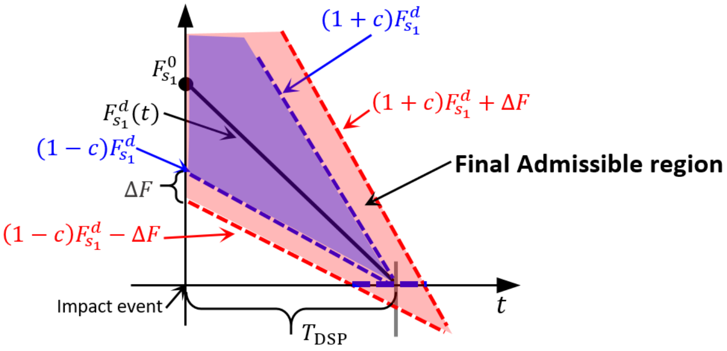

Figure 6.9: Contact force condition for lift-off in the DSP (the red region represents the admissible region for the leg force).

Results

The control procedure with the chosen parameters is presented in Algorithm 2. The stepping gain K in Eq. (3.17) is chosen to be the deadbeat gain. The BBF-QP controller is solved at 1kHz. The aSLIP starts from an initial static configuration and walks to a desired pre-impact velocity v^*, which is chosen based on the orbits of the H-LIP. The aSLIP parameters are chosen to match with the robot Cassie[1]. A video of the results can be seen in `https://youtu.be/fUZu6y-Gu4g`.

Algorithm 2 BBF-QP with H-LIP based Walking Synthesis

```
Initialization: Behavior: $z_0$ = 1m, $T_{\text{SSP}}$ = 0.4s, $T_{\text{DSP}}$ = 0.1s. Control: $\alpha = 500$,
  $\gamma = 10$, $k = 10$, $c = 0.5$, $\Delta F = 20$.
  while Simulation/Control loop do
    if SSP then
      Desired step size ← H-LIP stepping in Eq. (3.17).
      Desired swing foot position ← Eq. (6.20).
      $\tau$ ← BBF-QP in Eq. (6.50)
    else
      $\tau$ ← BBF-QP in Eq. (6.54)
    end if
  end while
```

We evaluate the approach for walking on flat ground to different desired velocities. Fig. 6.10 shows the results. The aSLIP converges closely to the desired walking of the H-LIP. The leg forces behave as expected. Fig. 6.10 (c) shows the trajectories of the horizontal velocity, which are not constant in the DSP. This contributes to the error w. We numerically calculate all w for different walking simulations and

[1] m = 33kg, and K = 8000N/m, D = 100Ns/m, which are the nonlinear leg spring parameters of Cassie at $L \approx$ 1m; see Appendix B. The spring is chosen to be linear for generality.

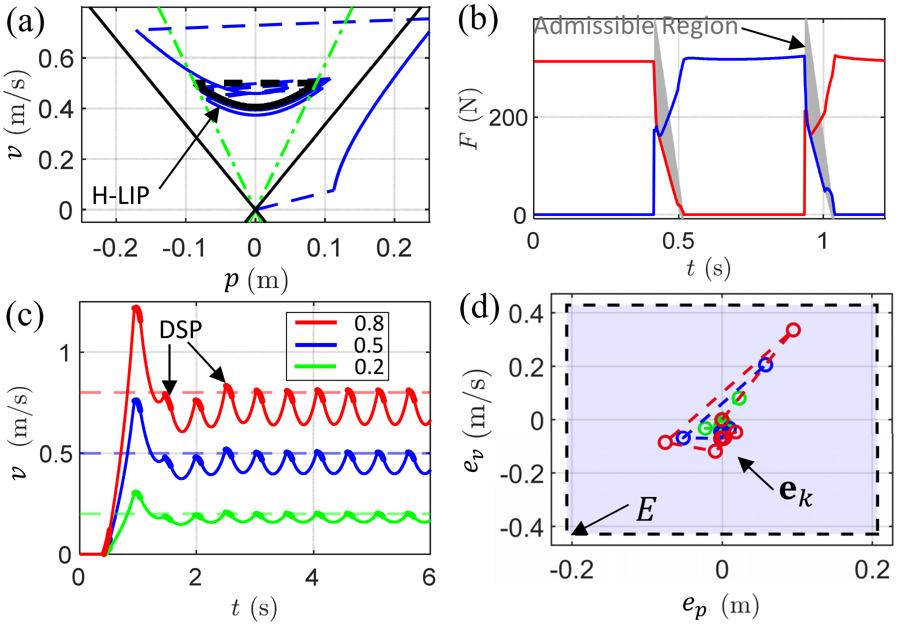

Figure 6.10: Walking using direct synthesis via BBF-QP: (a) the phase portrait of the horizontal state trajectory (blue) in the SSP for walking with $v^* = 0.5$m/s, and the black is the corresponding orbit of the H-LIP (green lines are the orbital lines, (b) the leg forces during walking with the gray region being the admissible region in the DSP; (c) the walking velocity and (d) the trajectories of the discrete error state $\mathbf{e}_k = [e_p, e_v]^T$ of walking with different $v^* = 0.2, 0.5, 0.8$m/s (indicated by dashed lines in (c)).

inner approximate W by a square. Since $(A + BK)^2 = 0$, $E = W \oplus (A + BK)W$; we get an inner approximation of E (shown in Fig. 6.10 (d)), and all the error states $\mathbf{e}$ are inside E, which validates the proposed approach of the H-LIP based stepping in Chapter 3. It is also important to note that, similar to the indirect synthesis on 3D-aSLIP, the direct synthesis via BBF-QP can be readily applied to 3D-aSLIP with two orthogonal planar stepping stabilizations via the H-LIP.

6.4 Conclusion

In this chapter, we demonstrate the application of the H-LIP based walking synthesis on the aSLIP model. Unlike the five-linkage walker AMBER, the realization on the aSLIP involves a higher level of complexity that comes from the underactuation of the springs in the leg. Both an indirect and a direct approach are proposed to address this spring underactuation. In the next chapter, we will see a similar but increasing complexity on the robot Cassie, and the proposed approaches here then become useful.

Chapter 7

APPLICATION ⊃ 3D UNDERACTUATED BIPEDAL ROBOT CASSIE

In this chapter, we present the application of the H-LIP based walking synthesis on the 3D underactuated bipedal robot Cassie (see Fig. 7.1). We first present the robot, its dynamics model, and the hybrid model of walking. Then, we describe an indirect walking synthesis that utilizes the aSLIP model to generate a desired periodic leg length trajectory to realize the H-LIP based approach. This indirect approach was originally proposed in our previous publication [141]. Lastly, we show another direct walking synthesis for realization, which was then proposed in our paper [142].

7.1 Robot Model

The Cassie robot is designed and built by Agility Robotics [1] as a full-scale 3D bipedal robot. It and its predecessor ATRIAS [58, 106] are designed to resemble the SLIP [109] with concentrated mass at its pelvis and lightweight legs [4], which are composed of closed kinematic chains with large leaf springs. The leg design

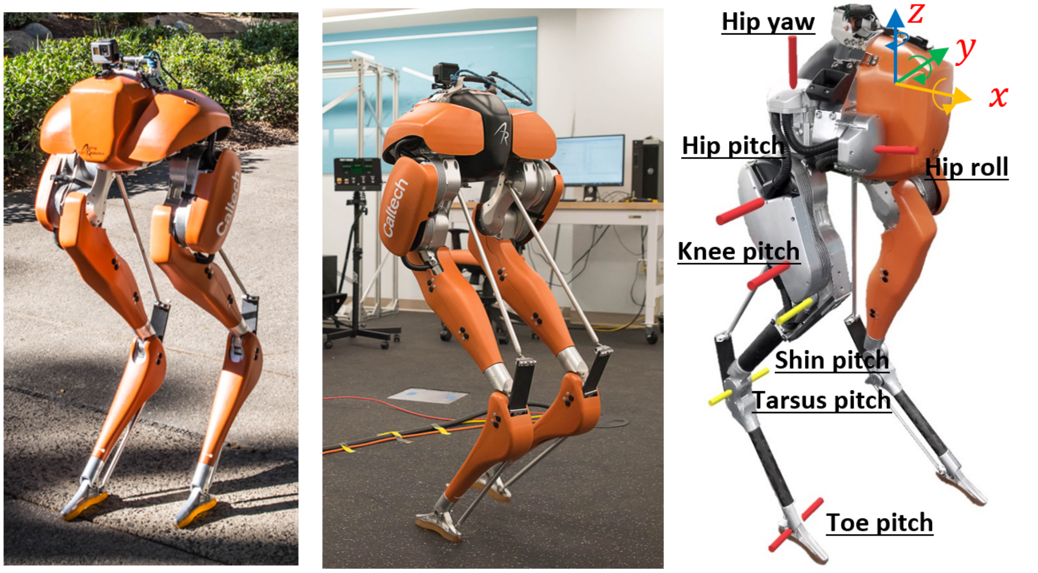

Figure 7.1: The underactuated bipedal robot Cassie.

is thus simple in philosophy but complex in details. The controllers that plan and render walking have been either on the rigid models [46, 55] or on the SLIP model [58, 106]. Here we present a full-dimensional model that best captures the dynamics of the robot including the complaint springs.

Robot Model: As shown in Fig. 7.1, Cassie has five motor joints (with the axis of rotation shown in red) on each leg, three of which locate at the hip and the other two are the knee and toe pitch. Fig. 7.2 (a) and (b) provide a close look at the leg kinematics and the abstract model, respectively. We model the shin and heel springs as torsion springs at the corresponding deflection axes. Therefore the spring torques are

$$\tau_{\text{shin}} = \mathrm{k}_{\text{shin}} q_{\text{shin}} + \mathrm{d}_{\text{shin}} \dot{q}_{\text{shin}}, \tau_{\text{heel}} = \mathrm{k}_{\text{heel}} q_{\text{heel}} + \mathrm{d}_{\text{heel}} \dot{q}_{\text{heel}},$$

where $\mathrm{k}_{\text{shin/heel}}, \mathrm{d}_{\text{shin/heel}}$ are the stiffness and damping, the parameters of which are provided by the manufacturer [1]. Since the achilles rod is very lightweight, we ignore the achilles rod and replace it by setting a holonomic constraint h_{rod} on the distance between the connectors (one locates on the inner side of the hip joint, the other locates at the end of the heel spring). The plantar rod is also removed and the actuation is applied to the toe pitch directly thanks to the parallel linkage design. These two simplifications remove unnecessary passive joints and associated configuration variables. As a consequence, the configuration of the leg can be described only by five motor joints, two spring joints and a passive tarsus joint. The total number of DoFs of the floating base model is then

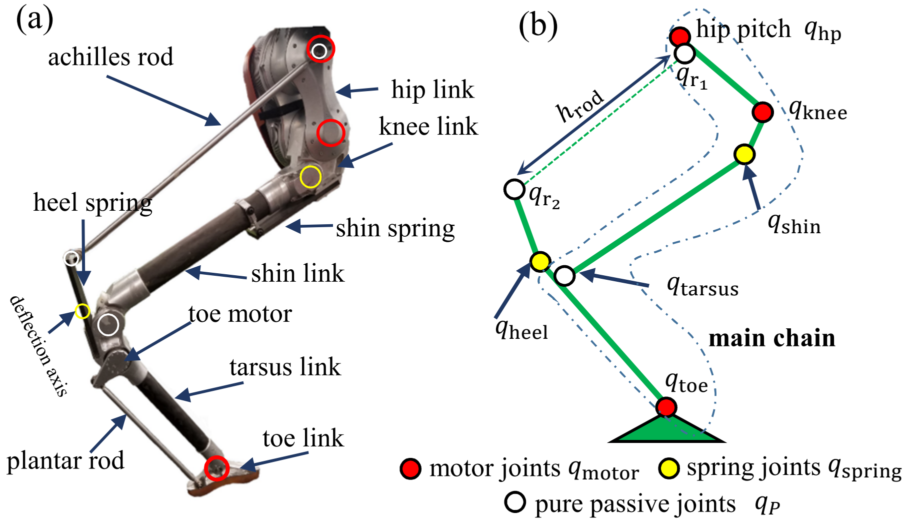

Figure 7.2: (a,b) Cassie's leg and its model.

$n = 8 \times 2 + 6 = 22$. The configuration of the robot thus is $q = [q_{\text{pelvis}}, q^{\text{L}}_{\text{leg}}, q^{\text{R}}_{\text{leg}}]^T$, where $q_{\text{pelvis}} = [q^{x,y,z}_{\text{pelvis}}, \phi_{\text{roll}}, \phi_{\text{pitch}}, \phi_{\text{yaw}}] \in SE(3)$ is the pelvis configuration, and $q^{\text{L}}_{\text{leg}}, q^{\text{R}}_{\text{leg}}$ are the configuration of the left and right leg, respectively. $q^{\text{L/R}}_{\text{leg}} = [q^{\text{roll}}_{\text{hip}}, q^{\text{yaw}}_{\text{hip}}, q^{\text{pitch}}_{\text{hip}}, q_{\text{knee}}, q_{\text{shin}}, q_{\text{tarsus}}, q_{\text{heel}}, q_{\text{toe}}]$, where the individual element is the joint angle. The motor joints are $q_{\text{motor}} = [q^{\text{roll}}_{\text{hip}}, q^{\text{yaw}}_{\text{hip}}, q^{\text{pitch}}_{\text{hip}}, q_{\text{knee}}, q_{\text{toe}}]^T$, and the spring joints are $q_{\text{spring}} = [q_{\text{shin}}, q_{\text{heel}}]^T$. Recall that the Euler-Lagrange equation with holonomic constraints is

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) = B_\tau \tau + J_h^T F_h, \tag{7.1}$$

$$J_h \ddot{q} + \dot{J}_h \dot{q} = 0, \tag{7.2}$$

where the meaning of individual element can be found in Chapter 2. Note that $\tau = [\tau_m^T, \tau_s^T]^T$ is the actuation vector; τ_m represents the motor torque vector, and τ_s is the vector of the torsional forces of the spring joints. Thus, $B_\tau = [B_m, B_s]$. $J_h = [J^T_{\text{rod}}, J^T_{\text{Foot}}]^T$ and $F_h = [F^T_{\text{rod}}, F^T_{\text{GRF}}]^T$, which represent the honolomic constraints on the push-rods and foot-ground contact.

Hybrid Model of Walking: The walking behavior of Cassie is modeled as a hybrid dynamical system with two domains: the SSP and the DSP, which depends on the number of feet contacting the ground. For simplification, we assume that the toe link (foot) in Fig. 7.1 always contacts the ground flatly. The compliance in the leg renders the DSP to be non-trivial. Similar to the hybrid walking model of the aSLIP, the walking goes from the SSP to the DSP when the swing toe strikes the ground, and it transits into the SSP from the DSP when one of the toes lifts off.

7.2 Indirect Walking Synthesis

In this section, we present the indirect walking synthesis via the H-LIP based stepping on the robot Cassie. We select the outputs and perturb the step sizes based on the H-LIP stepping to generate different desired walking behaviors. Although the philosophy is similar to the stepping on aSLIP in Chapter 6, the details require additional specifications on the full-dimensional robot. We first present output definitions that can deal with the compliant elements in the leg.

Leg Length: Similarly to the indirect synthesis on the aSLIP, we control the leg length to extend and retract periodically to create the basic periodic touch-down and lift-off behavior of the walking. It is important to make sure that the leg length can be controlled directly. Thus we select the virtual leg length as one of the outputs as being inspired by the jumping on Cassie [138]; see Appendix C. This eliminates the springs in the outputs and makes the springs underactuated under the control of the leg length. We select the parameters of the aSLIP to match the robot Cassie, and thus the aSLIP has a nonlinear leg spring. The desired periodic trajectory of the leg length $L^{\text{des}}(t)$ is the optimized leg length trajectory on the aSLIP from the trajectory optimization in Chapter 6. The springs on the full robot are expected to behave similarly to that of the aSLIP when leg length is actuated accordingly from the aSLIP model.

Step Size: Similar to previous applications, the H-LIP stepping is realized in the SSP by the continuous construction of the desired step length in the sagittal plane and the desired step width in the coronal plane. Recall that, in the SSP, we control the swing toe position to land to the desired location to achieve the desired step sizes. For 3D walking, we apply the 3D decomposition as been described in Chapter 3 that the stepping controller is realized individually in each plane. Recall that the desired step length u_x^{des} in the sagittal and step width u_y^{des} in the coronal plane are respectively calculated via the H-LIP based stepping:

$$u_x^{\text{des}} = u_x^{\text{H-LIP}} + K(\mathbf{x} - \mathbf{x}^{\text{H-LIP}}), \tag{7.3}$$

$$u_y^{\text{des}} = u_y^{\text{H-LIP}} + K(\mathbf{y} - \mathbf{y}^{\text{H-LIP}}), \tag{7.4}$$

where $\mathbf{x}$ and $\mathbf{y}$ are the horizontal COM states of the robot in its sagittal place and lateral plane, respectively. Similarly, the desired swing toe position relative to the stance foot in the horizontal plane is smoothly constructed to the desired step location:

$$x_{\text{sw}}^{\text{des}} = (1 - \mathcal{B}_h(t))x_{\text{sw}}^{+} + \mathcal{B}_h(t)u_x^{\text{des}}, \tag{7.5}$$

$$y_{\text{sw}}^{\text{des}} = (1 - \mathcal{B}_h(t))y_{\text{sw}}^{+} + \mathcal{B}_h(t)u_y^{\text{des}}, \tag{7.6}$$

where $[x_{\text{sw}}^{+},\ y_{\text{sw}}^{+}]$ is the initial swing foot horizontal location relative to the stance foot. Due to the fact that the compliant springs oscillate on the swing leg, we can only control the rigid representation of $\tilde{x}_{\text{sw}}^{\text{des}}(q)$, $\tilde{y}_{\text{sw}}^{\text{des}}(q)$ by setting $q_{\text{spring}}^{\text{sw}} = 0$, where the deflections of the shin and heel spring on the swing leg are set to zero; see Appendix C for the rigid representations.

Toe Actuation: It is interesting to notice that Cassie has toe-pitch actuation, which was designed to enable static standing. Without the toe-roll actuation, Cassie is still underactuated at the foot. To generalize the underactuation, we remove the toe-pitch actuation on the stance foot, which equivalently makes Cassie walk with point-feet underactuation. One can certainly use the toe actuation to contribute to the control of the horizontal dynamics. We will show this kind of application on the fully-actuated bipedal robots in Chapter 11.

Outputs Definition and Stabilization

DSP: We now mathematically define the outputs in the DSP. There are 8 motor joints excluding the toe-pitch joints. With two feet contacting the ground which adds 3 holonomic constraints on the distance between two feet, we define 5 outputs on the robot. As been noted above, the left and right leg length are used. It is also desirable to control the roll, pitch, and yaw angles of the pelvis so that the upper body does not oscillate during walking. As a result, we define the outputs for DSP as

$$\mathcal{Y}_{\text{DSP}}(q,t) = \begin{bmatrix} L_{\text{L}}(q) \\ L_{\text{R}}(q) \\ \phi_{\text{roll}}(q) \\ \phi_{\text{pitch}}(q) \\ \phi_{\text{yaw}}(q) \end{bmatrix} - \begin{bmatrix} L_{\text{L}}^{\text{des}}(t) \\ L_{\text{R}}^{\text{des}}(t) \\ \phi_{\text{roll}}^{\text{des}} \\ \phi_{\text{pitch}}^{\text{des}} \\ \phi_{\text{yaw}}^{\text{des}} \end{bmatrix}, \tag{7.7}$$

where $L_{\text{L/R}}^{\text{des}}(t)$ is the desired leg length trajectory from the trajectory optimization on the aSLIP, and $\phi_{\text{roll/pitch/yaw}}^{\text{des}}$ is the desired roll/pitch/yaw angle of the pelvis. The

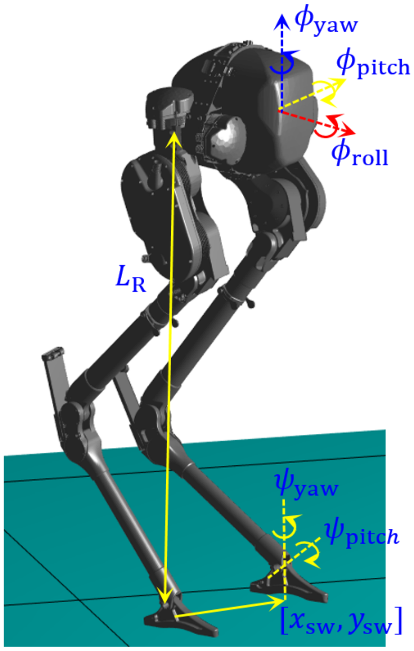

Figure 7.3: Output definition of the indirect synthesis for walking on Cassie.

desired pelvis angles are set to 0, and the desired yaw angle changes when turning is enabled (e.g. in Chapter 9 for tracking a global path).

SSP: In the SSP, we define 9 outputs since the robot has 9 actuators without the toe-pitch actuation on the stance foot. The 5 outputs defined in DSP are continually selected. Additional 4 outputs are on the swing leg, which are the swing toe pitch and yaw angles and the swing foot position in its sagittal and lateral planes. Therefore, the outputs for SSP are defined as

$$\mathcal{Y}_{\text{SSP}}(q,t) = \begin{bmatrix} L_{\text{L}}(q) \\ L_{\text{R}}(q) \\ \tilde{x}_{\text{sw}}(q) \\ \tilde{y}_{\text{sw}}(q) \\ \phi_{\text{roll}}(q) \\ \phi_{\text{pitch}}(q) \\ \phi_{\text{yaw}}(q) \\ \psi_{\text{pitch}}(q) \\ \psi_{\text{yaw}}(q) \end{bmatrix} - \begin{bmatrix} L^{\text{des}}_{\text{L}}(t) \\ L^{\text{des}}_{\text{R}}(t) \\ x^{\text{des}}_{\text{sw}} \\ y^{\text{des}}_{\text{sw}} \\ \phi^{\text{des}}_{\text{roll}} \\ \phi^{\text{des}}_{\text{pitch}} \\ \phi^{\text{des}}_{\text{yaw}} \\ \psi^{\text{des}}_{\text{pitch}} \\ \psi^{\text{des}}_{\text{yaw}} \end{bmatrix}, \tag{7.8}$$

where $\psi^{\text{des}}_{\text{pitch,yaw}}$ are the desired pitch and yaw angles on the swing toe linkage. The desired pitch angle is 0 so that the robot can produce flat contact with the ground. Similar to that in the DSP, the desired yaw angles are zeros unless turning is enabled.

Output Stabilization: To stabilize the outputs, i.e., driving $\mathcal{Y}_{\text{DSP/SSP}} \to 0$, we again apply the optimization-based controllers as been described in Chapter 2.

Simulation Results

In this part, we evaluate the indirect approach with H-LIP based synthesis to generate walking on Cassie. Similar to the walking on aSLIP, the robot Cassie should start from a periodic behavior. Differently, the robot is not a 3D point-mass model, thus purely repeating the leg length actuation is not going to produce a stepping-in-place behavior. Instead, the H-LIP based stepping is needed to stabilize a stepping-in-place behavior first, and then different walking is generated via the H-LIP based stepping for achieving different desired behaviors.

In [141] and this thesis, to generate the stepping-in-place motion from a static standing configuration, we first optimize the aSLIP with a torso model to generate a motion from a static standing configuration to the periodic stepping-in-place trajectory. This trajectory optimization in essence is similar to that for generating

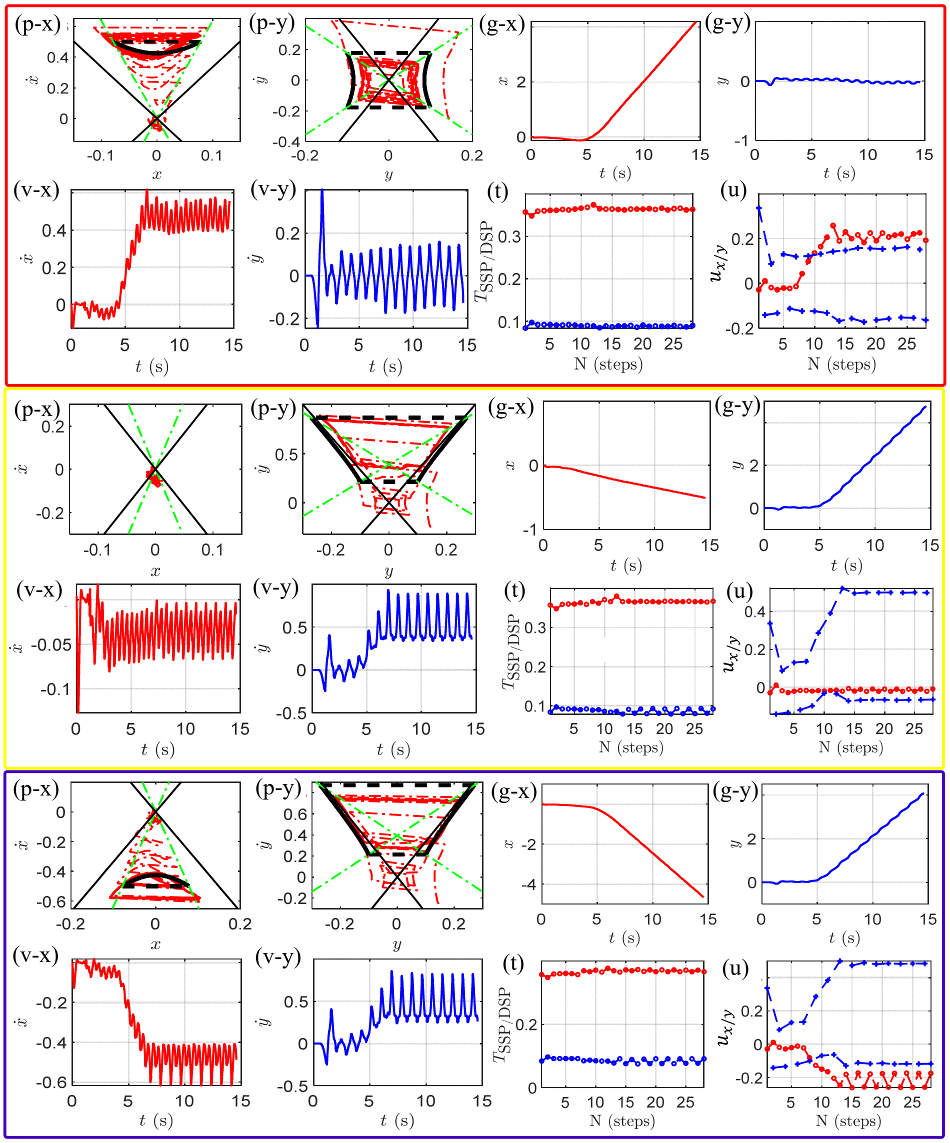

Figure 7.4: Walking via the indirect synthesis to realize different desired net velocities v_x^d, v_y^d with P1 orbit in its sagittal plane and P2 orbit in its lateral plane: $[v_x^d, v_y^d] = [0.5, 0]$m/s (red boxed plots), $[v_x^d, v_y^d] = [0, 0.5]$m/s (yellow boxed plots), and $[v_x^d, v_y^d] = [-0.5, 0.5]$m/s (blue boxed plots). (p-x, p-y) The phase plots of Cassie's translational states (red) in its sagittal and lateral planes comparing with the H-LIP orbits (black) with same desired velocities. (g-x, g-y) The global positions of the pelvis v.s. time t. (v-x, v-y) The forward and lateral velocities v_x, v_y v.s. time. (t) The convergence of the domain durations T_{DSP} (blue), T_{SSP} (red) over steps. (u) The convergence of the step length (red) in the sagittal plane and step width (blue) in the lateral plane over steps, where step width for achieving the P2 orbit is plotted separately based on the stance leg.

the stepping-in-place in Chapter 6. In other words, a segment of the leg length trajectory is added before the periodic leg length trajectory of the stepping-in-place motion so that the robot can start walking from standing.

We realize the method in Matlab simulation. The procedure is briefly explained in Algorithm 3. The dynamics is integrated using Matlab `ODE 113` function with event-based triggering for contact and domain switching. The optimization-based feedback control is solved at 1 kHz.

Algorithm 3 Indirect Synthesis on Cassie

```
$L_{\mathrm{L}}^{\mathrm{des}}(t), L_{\mathrm{R}}^{\mathrm{des}}(t), T_{\mathrm{SSP}}, T_{\mathrm{DSP}}, z_0 \leftarrow$ Stepping-in-place optimization on the aSLIP
$q_0$: Initial static standing configuration $\leftarrow$ IK.
Standing to stepping-in-place optimization on the aSLIP
while Simulation/Control loop do
  if SSP then
    Desired step sizes $u_x^{\mathrm{des}}$ and $u_y^{\mathrm{des}}$ from Eq. (7.3) and (7.4).
    Desired horizontal swing foot position $\{x, y\}_{\mathrm{sw}}^{\mathrm{des}}$ from Eq. (7.5).
  end if
  Update Outputs $\mathcal{Y}_{\mathrm{SSP/DSP}}(q, t)$ on Eq. (7.7) and (7.8)
  $\tau \leftarrow$ Optimization-based Controller
end while
```

We select P1 orbits in the sagittal plane and P2 orbits in the lateral plane for realizing different periodic walking with the desired forward velocity v_x^d and lateral velocity v_y^d. The robot is first stabilized to the stepping-in-place motion with zero velocities. After 3s, the desired velocities are set to different values for achieving directional walking. Fig. 7.4 show three different directional walking. From the phase plots of all the walking, we can see that the horizontal states of Cassie converge to the final orbits of the H-LIP with reasonable errors. The desired walking behaviors are realized and stabilized. The durations of the domains change slightly and converge as the periodic behaviors converge. The step sizes also converge. We conclude that the desired walking behaviors with different directional velocities are realized on Cassie via the H-LIP based indirect synthesis successfully. A video of the simulation results can be seen in `https://youtu.be/7Ix7yA5c19U`.

7.3 Direct Walking Synthesis

Now we present a direct approach to realize the H-LIP based walking synthesis on Cassie. Unlike the direct approach via BBF-QP on the aSLIP model, we find a set of output definitions that can approximately direct-control the vertical height of the COM and the swing foot. Since the S2S dynamics of the H-LIP is used to

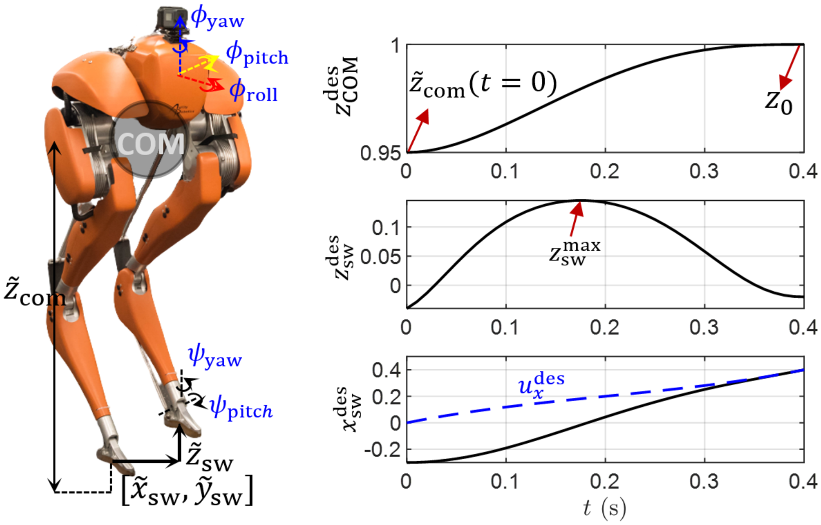

Figure 7.5: Illustrations of the definition and desired trajectories of the output.

approximate the S2S dynamics of the robot walking, the condition of the constant vertical height can be relaxed. In this section, we will also evaluate the stepping controller by numerically calculating the robust invariant set E in Chapter 1.

We consider the output in the SSP is defined to contain the vertical COM, swing foot position, and the orientation of the pelvis and the swing foot:

$$\mathcal{Y} = \begin{bmatrix} \tilde{z}_{\text{COM}}(q) \\ \tilde{x}_{\text{sw}}(q) \\ \tilde{y}_{\text{sw}}(q) \\ \tilde{z}_{\text{sw}}(q) \\ \phi_{\text{roll}}(q) \\ \phi_{\text{pitch}}(q) \\ \phi_{\text{yaw}}(q) \\ \psi_{\text{pitch}}(q) \\ \psi_{\text{yaw}}(q) \end{bmatrix} - \begin{bmatrix} z^{\text{des}}_{\text{COM}} \\ x^{\text{des}}_{\text{sw}} \\ y^{\text{des}}_{\text{sw}} \\ z^{\text{des}}_{\text{sw}} \\ \phi^{\text{des}}_{\text{roll}} \\ \phi^{\text{des}}_{\text{pitch}} \\ \phi^{\text{des}}_{\text{yaw}} \\ \psi^{\text{des}}_{\text{pitch}} \\ \psi^{\text{des}}_{\text{yaw}} \end{bmatrix}. \tag{7.9}$$

Due to the existence of the compliant springs in the legs, we select the rigid representations of the vertical COM $\tilde{z}_{\text{COM}}$ and the swing foot position $\{\tilde{x}, \tilde{y}, \tilde{z}\}_{\text{sw}}$ in the output; again, see Appendix C for details of the rigid output definitions. The desired orientations of the pelvis and the swing foot are chosen to be constant. The desired step sizes and thus the desired horizontal swing foot trajectories are designed identically to these in the indirect synthesis. The vertical COM position should be controlled to z_0, which is also the constant height of the H-LIP. At swapping support legs, $\tilde{z}_{\text{COM}}$ has a small discrete jump. The desired trajectory of the vertical COM position is then constructed as

$$z^{\text{des}}_{\text{COM}} = (1 - \mathcal{B}_h(t))\tilde{z}^{+}_{\text{COM}} + \mathcal{B}_h(t)z_0, \tag{7.10}$$

where $\tilde{z}^{+}_{\text{COM}}$ is the uncompressed COM height in the beginning of the SSP. Lastly, the vertical position of the swing foot $z^{\text{des}}_{\text{swing}}(t)$ is constructed as

$$z^{\text{des}}_{\text{sw}}(t) = \mathcal{B}_v(t, z^{\max}_{\text{sw}}, z^{\text{neg}}_{\text{sw}}), \tag{7.11}$$

where $\mathcal{B}_v$ is another Bézier polynomial to create lift-off and touch-down behaviors. It is designed to transit from 0 ($t = 0$) to $z^{\max}_{\text{sw}}$ (e.g., $t = \frac{T_{\text{SSP}}}{2}$) and back to $z^{\text{neg}}_{\text{sw}}$ ($t = T$). $z^{\max}_{\text{sw}}$ is a constant to determine the foot-ground clearance, and $z^{\text{neg}}_{\text{sw}}$ is a small negative value to ensure foot-strike at the end. Table 7.1 lists the coefficients used in the Bézier polynomials.

Table 7.1: Bézier Coefficients

On	Notation	Bézier Coefficients α
Vertical swing foot	$\mathcal{B}_v$	$[0, z^{\max}_{\text{sw}}\mathbf{1}_4, 0, z^{\text{neg}}_{\text{sw}}]$
Vertical COM height	$\mathcal{B}_h$	$[0, 0, \mathbf{1}_3]$

In the DSP, instead of re-formulating a different set of DSP outputs, we directly use the SSP outputs and set the desired values of the outputs on the swing foot to be the actual ones (including the horizontal positions and orientation), which preserves the holonomic constraints in the DSP and also simplifies the gait design. The uncompressed swing foot position continues to virtually intrude the ground. In this gait synthesis, the walking period is directly specified but individual domain duration is not. The swing foot intrusion indirectly determines the DSP duration.

This direct output construction provides the COM height z_0, step frequency (inverse of the walking period $\frac{1}{T}$), and the swing foot clearance $z^{\max}_{\text{sw}}$ to be individually chosen. The different combinations of the parameters render different walking behaviors. The desired walking velocity in each plane is individually stabilized via the H-LIP stepping, which works independently from the chosen gait parameters. Additionally, for P2 orbits on the robot, there are infinite orbits for realizing the same desired walking velocity. The combination of the gait parameters and orbit selections renders versatile walking behaviors on the robot.

Simulation Results

We now present the evaluation of the approach on Cassie in simulation. Assume the target final velocities v^t_x and v^t_y are given, and the goal is to control the robot to

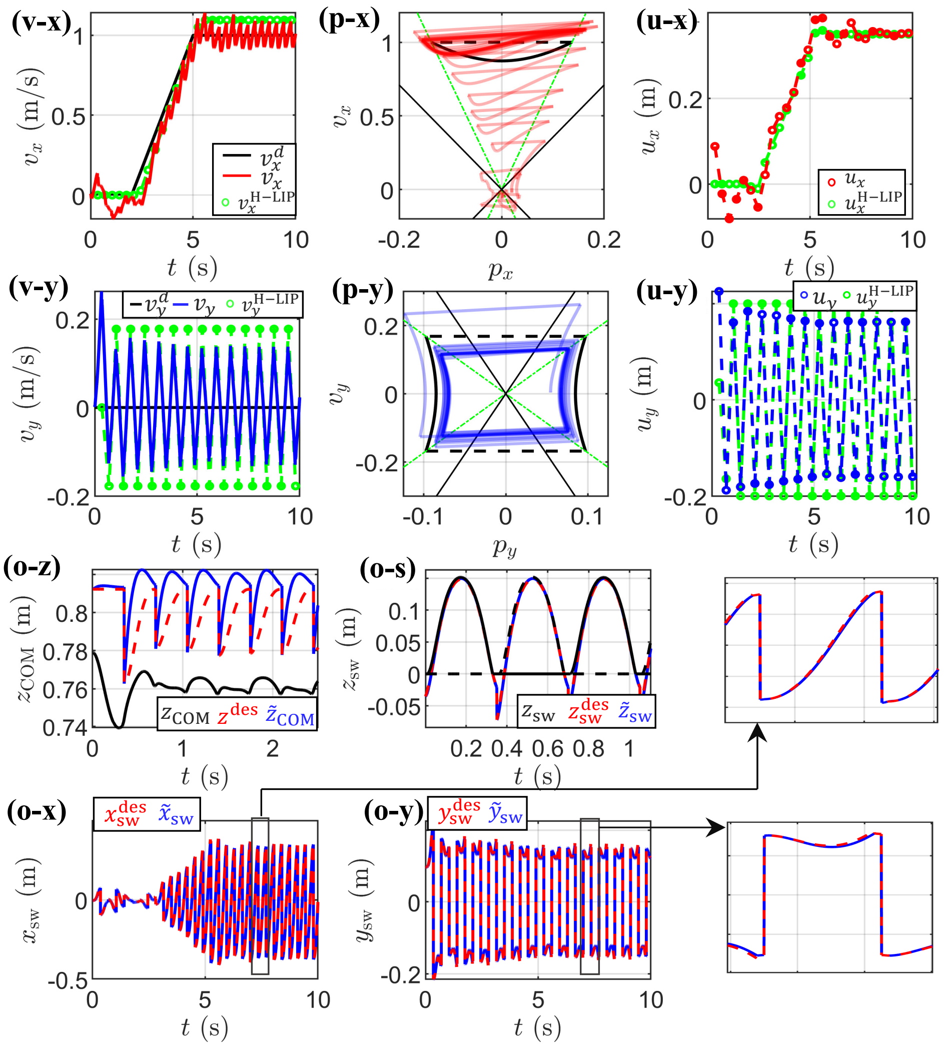

Figure 7.6: Simulation results on a forward walking with $v_x^t = 1$ m/s, $v_y^t = 0$ m/s, $u_{\text{L}}^{y*} = -0.2$m: the horizontal velocities of the COM (red and blue lines) in the sagittal plane (v-x) and the coronal plane (v-y) compared with the desired velocities $v_{x,y}^d(t)$ (black lines) and the corresponding velocities of the H-LIP (green circles); the phase plots of the horizontal COM states in the sagittal plane (p-x) and the coronal plane (p-y) compared with the H-LIP orbits (black) at the target velocities; comparisons of the step sizes (u-x, u-y) between the robot (red circles in the sagittal plane and blue circles in the coronal plane) and the H-LIP (green circles). (o) Output tracking with the red dashed lines indicating the desired trajectories and the blue continuous lines indicating the actual ones: (o-z) $\tilde{z}_{\text{COM}}$ (the black line is the actual vertical COM position of the robot z_{COM}); (o-s) the vertical swing foot trajectory $\tilde{z}_{\text{sw}}$ (the black lines are z_{sw}); the horizontal swing foot trajectories in the sagittal plane (o-x) and the coronal plane (o-y).

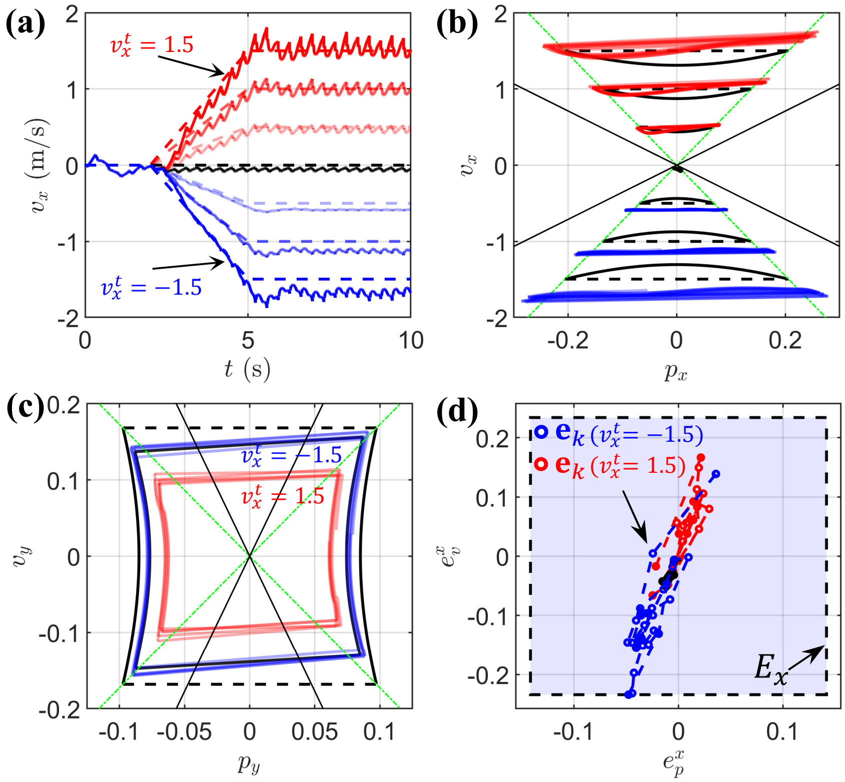

Figure 7.7: Comparison on forward walking with different target velocities v_x^t: (a) forward velocity of the COM (continuous lines) compared with the desired velocity profiles $v_x^d(t)$ (dashed lines); (b) the converged orbits (red and blue lines) of the sagittal COM states compared with the desired target orbits of the H-LIP (black lines). (c) the converged orbits (the red is with $v_x^t = 1.5$m/s and the blue is with $v_x^t = -1.5$m/s) of the coronal COM states compared with the target orbit of the H-LIP (black); (d) the error state trajectories (the blue and red circles) and the error invariant set E_x (the blue transparent square) in the sagittal plane.

realize the target velocities. We first select an orbit composition with one periodic orbit in each plane. Then continuous desired velocity profiles $v_{x/y}^d(t)$ are constructed to reach the target velocities. For simplicity, we use piecewise linear trajectories to design $v_{x/y}^d(t)$. For P2 orbits, the orbit-determining step size should be specified. The desired output trajectories are then constructed using the presented direct H-LIP based synthesis. The gait parameters such as the swing foot clearance and step frequency are specified in the beginning. In the following results, $z_0 = 1$m, $T = 0.4$s, $z_{\text{sw}}^{\text{max}} = 0.16$m, and $z_{\text{sw}}^{\text{neg}} = -0.03$m. The video of the simulation results can be seen in `https://youtu.be/-_QmNNBPfdg`.

Directional Walking: We first present a forward walking on Cassie as the basic

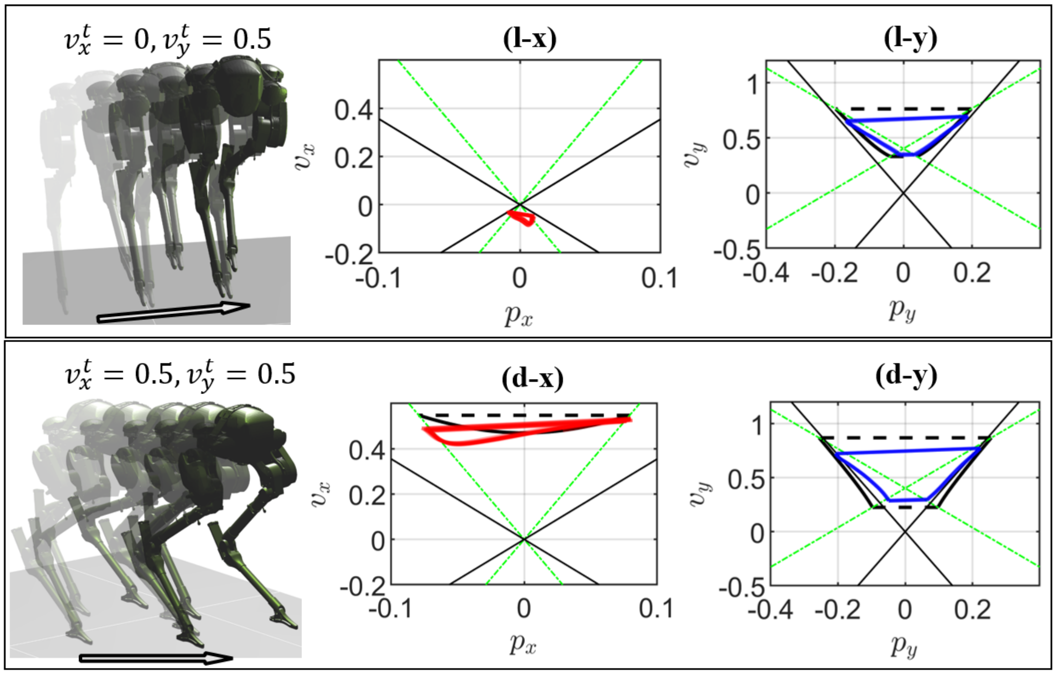

Figure 7.8: Lateral walking ($v^t_{x,y} = [0, 0.5]$m/s, $u^{y*}_{\mathrm{L}} = -0.08$m) and diagonal walking ($v^t_{x,y} = [0.5, 0.5]$m/s, $u^{y*}_{\mathrm{L}} = -0.3$m) and their converged orbits in the sagittal plane (l-x, d-x) and the coronal plane (l-y, d-y).

realization of the proposed H-LIP based direct synthesis. The orbit composition is chosen as having a P1 orbit in its sagittal plane and a P2 orbit in its coronal plane. $v^t_x = 1$m/s, and $v^t_y = 0$m/s, thus the robot only progresses in its sagittal plane. The orbit-determining step width of the P2 orbit is chosen to be $u^{y*}_{\mathrm{L}} = -0.2$m. The desired walking velocity $v^d_x(t)$ is chosen from 0 to ramp up to 1m/s within 3s. The velocity of the robot is controlled successfully to the target velocity via the H-LIP based approach. Fig. 7.6 shows the plots of the forward walking. The output trajectories are well-tracked via the optimized-based controller. The actual velocities converge to the desired ones with negligible errors, and the horizontal states of the robot converge closely to the desired orbits of the H-LIP with the target velocities in each plane.

Then we change the target velocity in the sagittal plane from -1.5m/s to 1.5m/s with a 0.5m/s increment. Fig. 7.7 shows the results. For clarity, in the phase portraits, we only plot the steady walking behaviors where the desired walking velocities become constant. We also demonstrate that the evolution of the error states is inside the error invariant set. The error states in each plane can be directly calculated from

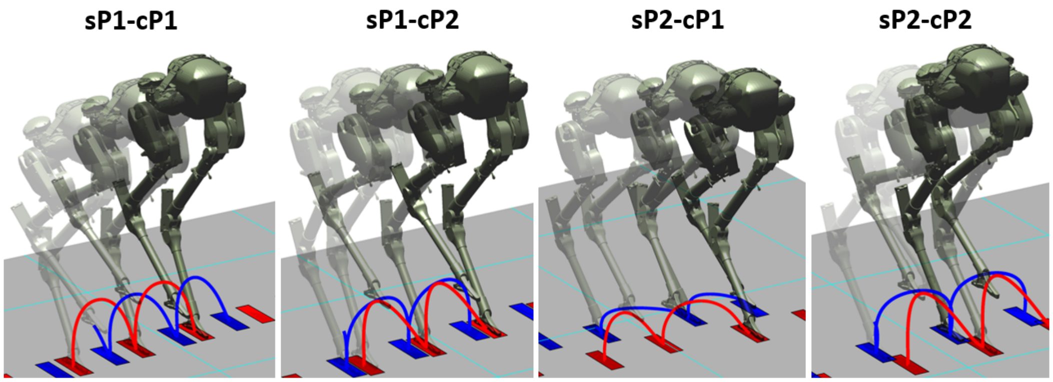

Figure 7.9: Simulated walking via different orbit compositions with the same target velocity ($v^t_{x,y} = [0, 0.5]$m/s). The trajectories of the swing foot are indicated by the red (left foot) and the blue (right foot) lines. The red and blue rectangles on the ground indicate the contact locations of the left and right feet, respectively.

the pre-impact states of the robot and the desired state of the H-LIPs. To calculate the error invariant set, we first numerically calculate the dynamics error w from the evolution of the error states in each realized walking behavior. Again we use all w to construct a polytope to numerically approximate W in each plane. As K is chosen from the deadbeat controller, the invariant set $E = (A + BK)W \oplus W$. Fig. 7.7 (d) shows that the error states are indeed inside the error invariant set.

The approach can also realize walking in different directions by selecting different desired velocities in each plane of the robot. Here we present walking in the lateral and diagonal direction. Fig. 7.8 illustrates the converged walking behaviors with different choices of the target velocities. The gait parameters are identical to the previous examples. By tracking the desired velocity in each plane, the robot walks in the desired direction. The converged orbits of the horizontal COM states are also relatively close to the desired orbits of the H-LIP in both cases. Moreover, by selecting different step width u^{y*}_L, different P2 orbits are realized in the coronal plane with the same desired velocity v^d_y (Fig. 7.8 (l-y) and (d-y)).

Variable Orbit Compositions: The two types of orbits of the H-LIP provide four kinds of orbit compositions in 3D. If the kinematic constraints are neglected, all four types of orbit compositions can be realized to achieve the same desired walking velocity. Fig. 7.9 illustrate the four realizations to achieve the lateral walking with $v^t_y = 0.5$m/s. Each realization is abbreviated by the type of orbit in each plane, e.g., sP1-cP1 indicates both P1 orbits are selected in the sagittal and coronal plane. For certain compositions, kinematic collisions can happen between the legs. E.g., the sP1-cP1 gait with $v^t_{x,y} = 0$ clearly creates foot overlaps. The complex leg design on Cassie further increases the possibilities of kinematic collisions between the legs. Since the kinematic constraints can be neglected in the simulation, we demonstrate all four types of orbit compositions that realize the same desired walking velocity.

7.4 Conclusion

Both the indirect and the direct approaches can be used to continuously generate walking behaviors with different velocities and orbit compositions. The direct approach has several advantages over the indirect one. Trajectory optimization on the aSLIP model is no longer needed. The walking behaviors are also stabilized to the desired ones with a few steps. For walking with dramatically changing desired velocities, the indirect approach may also produce temporary unplanned flight phases. Thus, we will realize the direct approach on the hardware of Cassie in the next chapter.

Chapter 8

APPLICATION ⊃ HARDWARE REALIZATION ON CASSIE

In this chapter, we realize the H-LIP based synthesis on the physical hardware of Cassie. Unlike that in simulation, the robot state is no longer completely and exactly known on the hardware. Some states can be measured, and the rest has to be estimated. Moreover, the computation capacity of the on-board computer has to be taken into consideration for realization. Therefore, we first present the control realization for practical considerations on the hardware, and then present the results of the experiment. In the experiment, we realize the directional walking, which is similar to what has been evaluated in the simulation. More importantly, we show that the gait parameters such as the step frequency, vertical COM height, swing foot clearance, and step width can be changed on the fly in the hardware realization, which demonstrates the walking versatility. Additionally, the robustness of the walking synthesis is evaluated by realizing stable walking behaviors under external pushes and on uncertain grassy terrain.

8.1 Hardware Implementation

The robot has joint encoders and IMU on the hardware to sense the configuration of the robot itself. We have not yet added any sensors such as Lidar or cameras on the robot to sense the external environment. Additionally, some of the joints in the closed-loop kinematic chains do not have encoders to measure the joint rotation angles. Thus, the challenges are on utilizing the partial information to extract the states that are needed to realize the H-LIP based synthesis. We elaborate the process in the following subsections.

Heel Spring Deflection

The shin spring is directly connected at the shin joint, and thus the encoder on the shin joint directly measures the deflection of the shin spring. There is no encoder to directly measure the deflection of the heel spring q_{heel}. We calculate q_{heel} based on the kinematics of the closed-loop chain:

$$q_{\text{heel}} = \texttt{Root}(h_{\text{rod}}(q_{\text{knee}}, q_{\text{shin}}, q_{\text{heel}}, q_{\text{tarsus}}) = 0), \tag{8.1}$$

where $q_{\text{knee}}, q_{\text{tarsus}}$ are directly measured by their joint encoders. Similarly, this is solved via Newton-Raphson method. For further simplifications on the calculation on the hardware, we used a parametric regression to approximate the solution.

Contact Detection

The robot Cassie is not equipped with contact sensors to detect foot-ground contact. It is possible to measure the deflection of the spring joints and set a threshold for contact detection. However, the springs can still have non-trivial deflections in the swing phase due to the inertia forces in the leg. Therefore, the threshold has to be set large enough to avoid false detection of contact. However, this can cause significant late-detection of impacts and early-detection of lift-offs. Instead, we use the measured torque from the input current (similar to the proprioceptive sensing [129]) along with the spring deflections to approximately calculate the contact forces at the feet. A threshold is then set on the magnitude of the forces to detect contact.

Since the vertical COM of the robot will be controlled approximately constant, we neglect the dynamics contribution to the contact forces. The EOM in Eq. (7.1) becomes

$$G(q) = B_\tau \tau + J_h^T F_h, \tag{8.2}$$

where $F_h = [F_{\text{rod}}^T, F_{\text{GRF}}^{\text{L}^T}, F_{\text{GRF}}^{\text{R}^T}]^T$ and $\tau = [\tau_m^T, \tau_s^T]^T$. τ_m is measured from the motor current and τ_s is calculated from the spring deflections. This equation is invariant w.r.t. the pelvis position $q_{\text{pelvis}}^{x,y,z}$, which is not known. Thus, we set $q_{\text{pelvis}}^{x,y,z}$ to 0. The rest of q are measured via the IMU, joint encoders and leg kinematics. F_h can be directly solved via the pseudo-inverse of J_h^T: $F_h = \texttt{pinv}(J_h^T)(G(q) - B_\tau \tau)$. The calculated $F_{\text{GRF}}^{\text{L/R}}$ are then low-pass filtered with a cutoff frequency of 100Hz. A

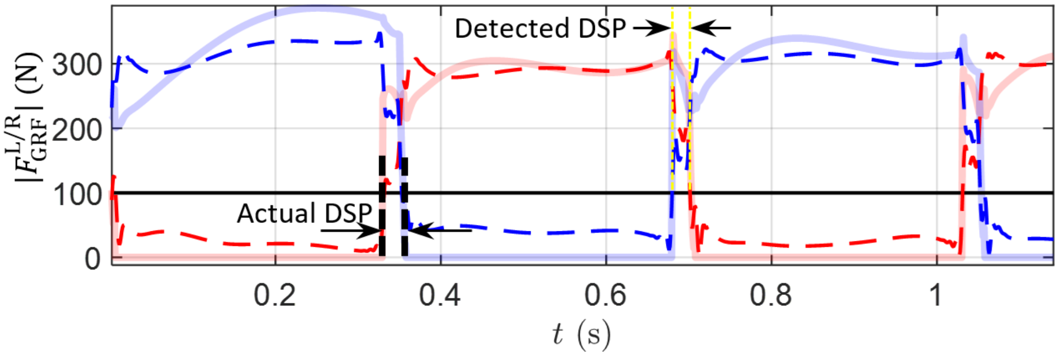

Figure 8.1: Contact detection via the GRF calculation: The transparent red and blue lines are the norm of the actual GRF on the robot in simulation, and the dashed red and blue lines are the calculated GRF for contact detection. The detected DSP is close to the actual DSP in simulation.

threshold is then set on the norm of $F_{\text{GRF}}^{\text{L/R}}$ to determine if the foot is in contact with the ground. Fig. 8.1 shows the contact detection via the GRF compared with the actual GRF in a simulated walking. The threshold is set to 100N, which provides precise contact detections.

COM Velocity Approximation

The transitional position and velocity of the floating-base $q_{\text{pelvis}}^{x,y,z}$ and $\dot{q}_{\text{pelvis}}^{x,y,z}$ cannot be directly measured. $\dot{q}_{\text{pelvis}}^{x,y,z}$ is required for calculating the COM velocity for realizing the walking. We implemented the extended Kalman filter in [18] for state estimation by utilizing the IMU. This state estimation required significant computation, e.g., a similar estimation scheme in [52] has to be implemented on a secondary computer on the robot. Additionally, the magnetometer drift inside the IMU also creates errors on the estimated velocities under certain circumstances. Due to those concerns, we instead approximate the COM velocity based on the H-LIP dynamics in the SSP.

We use the walking in the sagittal plane to illustrate the approximation. Let p_0 and v_0 be the horizontal position and velocity of the COM in the beginning of the SSP. The dynamics of the horizontal COM in the SSP can be approximated by the SSP dynamics of the H-LIP. Thus the current COM state of the robot $[p_t, v_t]$ in the SSP can be approximated by

$$\begin{bmatrix} p_t \\ v_t \end{bmatrix} \approx e^{A_{\text{SSP}}t} \begin{bmatrix} p_0 \\ v_0 \end{bmatrix}, \tag{8.3}$$

where A_{SSP} is defined in Eq. (3.8). Let $A_t := e^{A_{\text{SSP}}t}$. Given the measured positions p_0 and p_t and the current time t from the beginning of the SSP, the velocity approximations are

$$\begin{bmatrix} \tilde{v}_0 \\ \tilde{v}_t \end{bmatrix} = \begin{bmatrix} -A_t^{(1,1)}/A_t^{(1,2)} & 1/A_t^{(1,2)} \\ A_t^{(2,1)} - A_t^{(1,1)}A_t^{(2,2)}/A_t^{(1,2)} & A_t^{(2,2)}/A_t^{(1,2)} \end{bmatrix} \begin{bmatrix} p_0 \\ p_t \end{bmatrix}, \tag{8.4}$$

where the superscripts indicate the elements of the matrix A_t. Thus the continuous velocity approximation $\tilde{v}_t$ is obtained. The prediction of the pre-impact velocity $\tilde{v}_{t=T_{\text{SSP}}}$ can also be continuously approximated by the H-LIP dynamics in the SSP based on the current state $[p_t, \tilde{v}_t]^T$ and the time-to-impact $T_{\text{SSP}} - t$. The velocity approximation is solely based on the position of the COM w.r.t. the stance foot, which only uses joint encoders and orientation readings of the IMU and thus is robust to sensor noises. Moreover, we show that this approximation is valid for applying the H-LIP based stepping, only with generating a different error invariant set.

Let $\tilde{v}^-$ be the approximated velocity of the COM of the robot at the pre-impact. $\tilde{v}^- = \tilde{v}_{t=T_{\text{SSP}}}$ is calculated from Eq. (8.4). Let $\tilde{\mathbf{x}} = [p^-, \tilde{v}^-]^T$ represent the approximated COM state at the pre-impact. Assuming the COM position of the robot is measured with a negligible error, $\tilde{\mathbf{x}} - \mathbf{x} = [0, \tilde{v}^- - v^-]^T := \delta\mathbf{x}$, where v^- is the actual COM velocity of the robot at pre-impact event. Note that $\delta\mathbf{x}$ is bounded: the velocity error $\tilde{v}^- - v^-$ is the integration of the dynamics difference between the H-LIP and the robot in the SSP. The approximated state is used in the H-LIP based stepping, i.e., $u = u^{\text{H-LIP}} + K(\tilde{\mathbf{x}} - \mathbf{x}^{\text{H-LIP}})$. Therefore, the error dynamics becomes

$$\begin{aligned}\mathbf{e}_{k+1} &= A\mathbf{x}_k + Bu_k + w - A\mathbf{x}_k^{\text{H-LIP}} - Bu_k^{\text{H-LIP}}\\ &= (A + BK)\mathbf{e}_k + \underbrace{w + BK\delta\mathbf{x}_k}_{\tilde{w}}.\end{aligned}$$

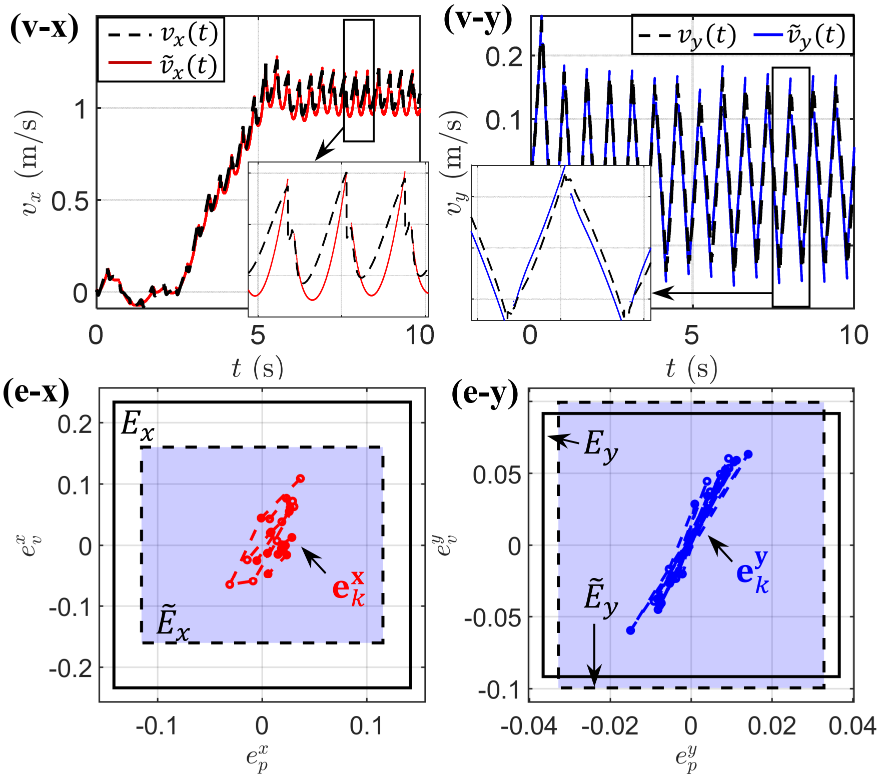

Figure 8.2: Validation on the H-LIP based velocity approximation on a simulated walking with $v_x^t = 1, v_y^t = 0$m/s, $u_{\text{L}}^{y*} = -0.2$m: (v-x, v-y) the approximated horizontal velocities of the COM $\tilde{v}_x(t)$ (red line) and $\tilde{v}_y(t)$ (blue line) in the SSP compared with the actual velocities (dashed black lines); (e-x, e-y) the error state trajectories $\mathbf{e}^{\mathbf{x},\mathbf{y}}$ and the new error invariant sets $\tilde{E}_{x,y}$ (blue transparent polytopes) in each plane compared with the error invariant sets $E_{x,y}$ (white polytopes) from the controller using the true COM velocity (Fig. 7.6).

$\tilde{w} \in \tilde{W}$ is bounded since w and $\delta\mathbf{x}$ are both bounded. This consequently creates a new error invariant set $\tilde{E}$.

To validate this, we use the H-LIP based velocity approximations to replace the actual horizontal velocities of the COM in the controller in simulation. The performance is comparable with the implementation that uses the true COM velocity in the previous chapter. Fig. 8.2 shows the results on a simulated forward walking as a proof. As the horizontal COM dynamics of the robot is close to the H-LIP dynamics, the velocity approximation works well (Fig. 8.2 (v-x) and (v-y)). Although the new disturbance $\tilde{w}$ is w plus another term, it does not necessarily mean that the size of $\tilde{W}$ and the resultant $\tilde{E}$ are larger. Here we get a smaller set in the sagittal plane (Fig. 8.2 (e-x)), and the sets in the coronal plane are of similar sizes (Fig. 8.2 (e-y)).

Joint-level Controller

The optimization-based controller in Chapter 2 can be potentially implemented on the hardware by utilizing the secondary computer on the robot. Here we apply a *PD + Gravitation Compensation* (PD+G) controller, which in practice provides an equivalent tracking performance and a much-lower computational effort. The PD+G controller is directly implemented on the main computer on the robot, which is written as $\tau_m = \tau_{\mathrm{PD}} + \tau_{\mathrm{G}}$, where τ_{PD} represents the PD component and τ_{G} represents the gravitation compensation part.

For the PD component, we directly map the desired acceleration of the output to the joint torques. The desired accelerations $\ddot{\mathcal{Y}}^d$ is identically chosen as that in Eq. (2.39). $\mathcal{Y}$ and $\dot{\mathcal{Y}}$ are measured on the hardware. Note that the selection of the outputs (e.g., Eq. (7.9)) are mainly functions of the motor joints. The actual acceleration of the output is assumed to be: $\ddot{\mathcal{Y}} = J_{\mathcal{Y}}\ddot{q}_m + \dot{J}_{\mathcal{Y}}\dot{q}_m$, where $J_{\mathcal{Y}} = \frac{\partial \mathcal{Y}}{\partial q_m}$. The desired accelerations of the motor joints are applied as the motor torques:

$$\tau_{\mathrm{PD}} = \ddot{q}_m^d = J_{\mathcal{Y}}^{-1}(\ddot{\mathcal{Y}}^d - \dot{J}_{\mathcal{Y}}\dot{q}). \tag{8.5}$$

For the gravity compensation, we need to find joint torques to cancel the gravitational terms in Eq. (8.2) based on the current configuration and contact. The problem is inverse to the contact detection. Given q, we find τ_{G} to minimize

$$||B_m\tau_{\mathrm{G}} + B_s\tau_s + J_h^T F_h - G(q)||^2. \tag{8.6}$$

Note that the foot contact of the robot is underactuated and thus there does not exist any set of joint torques completely cancel out the gravitational term, unless the foot

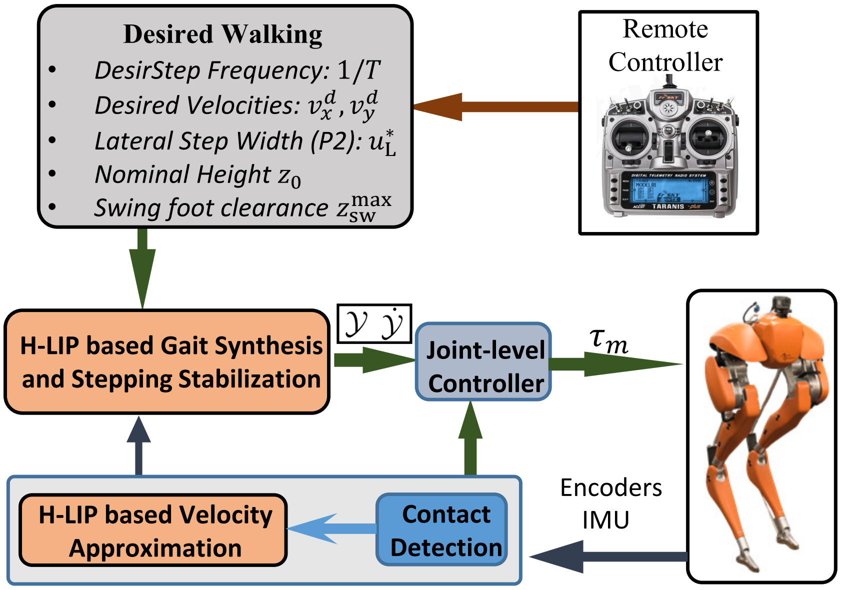
Desired Walking
DesirStep Frequency: $1/T$
Desired Velocities: v_x^d, v_y^d
Lateral Step Width (P2): u_{L}^*
Nominal Height z_0
Swing foot clearance $z_{\mathrm{sw}}^{\max}$
Remote Controller
H-LIP based Gait Synthesis and Stepping Stabilization
$\mathcal{Y}$ $\dot{\mathcal{Y}}$
Joint-level Controller
τ_m
Encoders IMU
H-LIP based Velocity Approximation
Contact Detection

Figure 8.3: Illustration of the hardware realization of the controller on Cassie.

is fully-actuated. This yields a least square problem of

$$\mathtt{min} : ||\mathbf{AX} - \mathbf{b}||^2, \tag{8.7}$$

where $\mathbf{A} = [B_m, J_h^T]$, $\mathbf{X} = [\tau_G^T, F_h^T]^T$, $\mathbf{b} = G(q) - B_s\tau_s$. Similarly, this problem can be solved via the pseudo-inverse of $\mathbf{A}$, i.e., $\mathbf{X} = \mathtt{pinv}(\mathbf{A})\mathbf{b}$, which yields the gravity compensation term τ_{G}.

Hardware Implementation Scheme

The robot is controlled via a remote controller that sends radio commands to the robot. The remote has two joysticks, four potentiometers, and several toggle switches. The on-board computer is programmed to interpret the radio signals, read all the sensors on the robot, and send torque commands to the robot. The implementation is illustrated in Fig. 8.3. The remote commands are processed to get the desired walking behaviors. The H-LIP based gait synthesis and stepping stabilization calculate the output based on the gait parameters, contact, and COM states. The joint-level controller then calculates the motor torques and sends them to the motor modules to stabilize the outputs. Based on the computation capacity of the main on-board computer, the control loop is set at 1kHz.

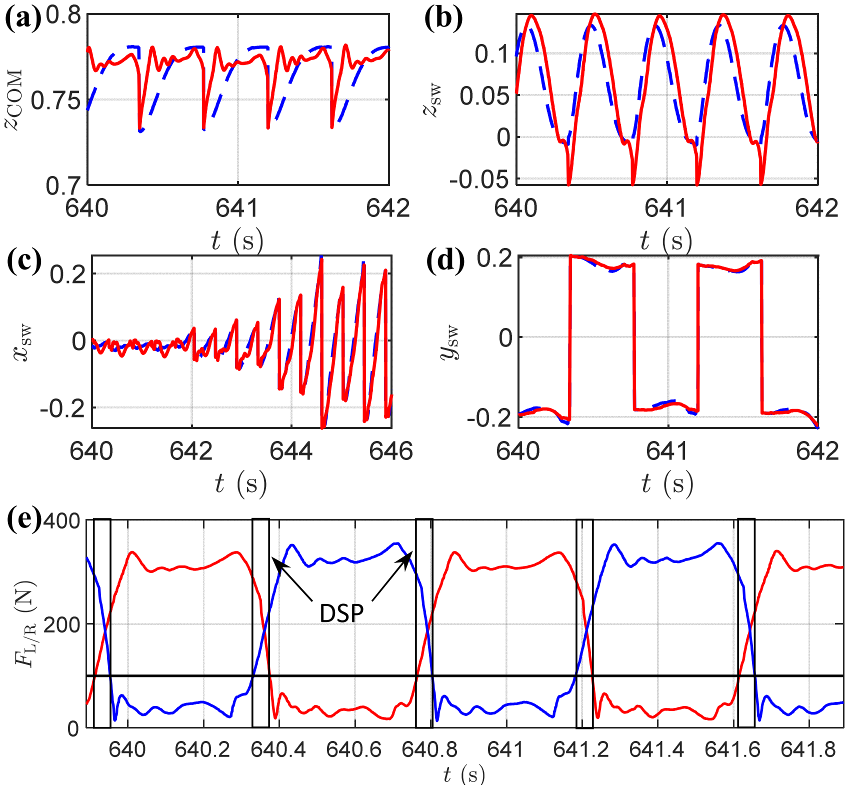

Figure 8.4: Illustrations of the output tracking and contact detection on the hardware: the desired output trajectories (the blue dashed lines) and the actual output trajectories (the red lines) of (a) the vertical COM position, and (b,c,d) the vertical, forward and lateral positions of the swing foot; (e) the contact detection via the GRF, where the boxed regions indicate the DSP.

8.2 Experiment Results

On the hardware, we not only focus on the realization of directional walking on Cassie, but also demonstrate the versatility and robustness of the walking that come from the H-LIP based direct synthesis. A video of the experiment results can be seen in `https://youtu.be/qEp1RUf6X-U`.

Directional Walking

We demonstrate directional walking behaviors on the robot by using the joysticks on the remote to steer the robot to its forward, backward, and lateral directions. Instead of driving the robot to periodic walking orbits with different velocities in the simulation, the desired walking velocity is continuously changing. The joystick values on the remote are used as the desired walking velocities $v^d_{x,y}$. We use low-pass filters to smooth the readings of the joysticks. Thus, the desired velocities between consecutive steps do not vary significantly. Additionally, the kinematic constraints

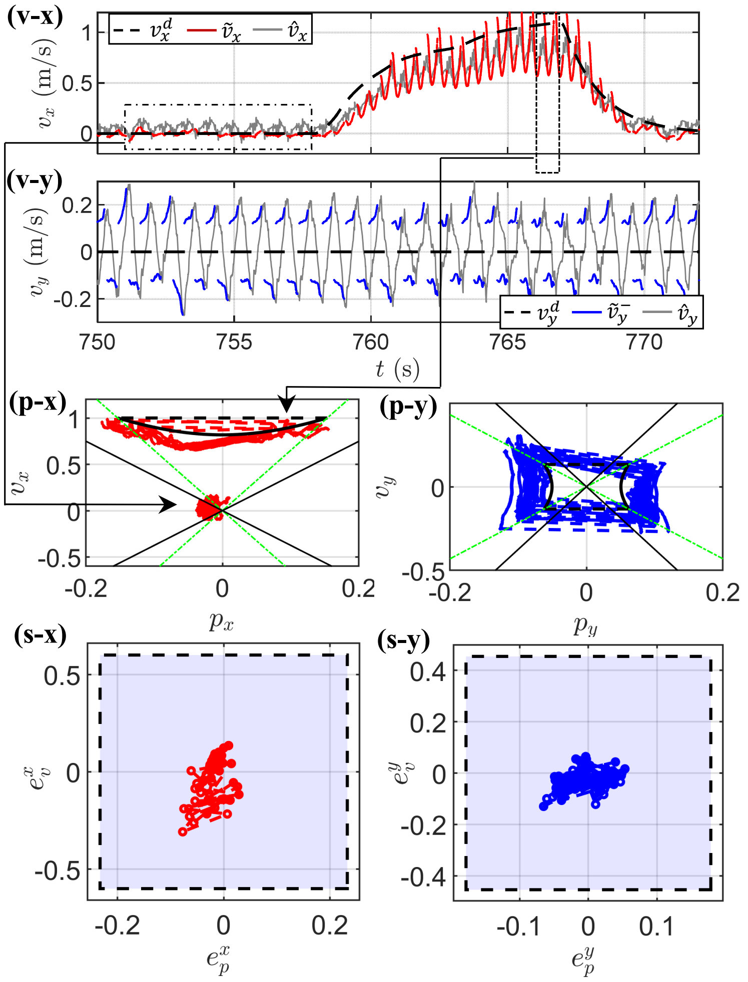

Figure 8.5: Trajectories of a forward walking with varying target velocities. (v-x, v-y) plot the horizontal COM velocities including the target velocity $v^t_{x,y}(t)$ (the black dashed lines), the velocity in the SSP from the H-LIP based approximation (the red lines), the predicted pre-impact velocity (the blue lines), and the estimated velocities (the gray lines). (p-x, p-y) plot the horizontal states in the sagittal plane (in different time segments) and the coronal plane, respectively. The black orbits are the desired orbits of the H-LIP. (s-x, s-y) plot the error state trajectories (red and blue circles) inside the calculated error invariant sets (transparent blue polytopes) in each plane.

on the robot provide a range of available step sizes at a certain stance height. If the step frequency is fixed, the step size limits produce limits on the achievable velocities. Thus the maximum command velocities in each plane are limited for safety. Fig. 8.6 demonstrates the snapshots of the robot walking in its sagittal and coronal plane.

In order to analyze the generated walking behaviors on the hardware, we use the

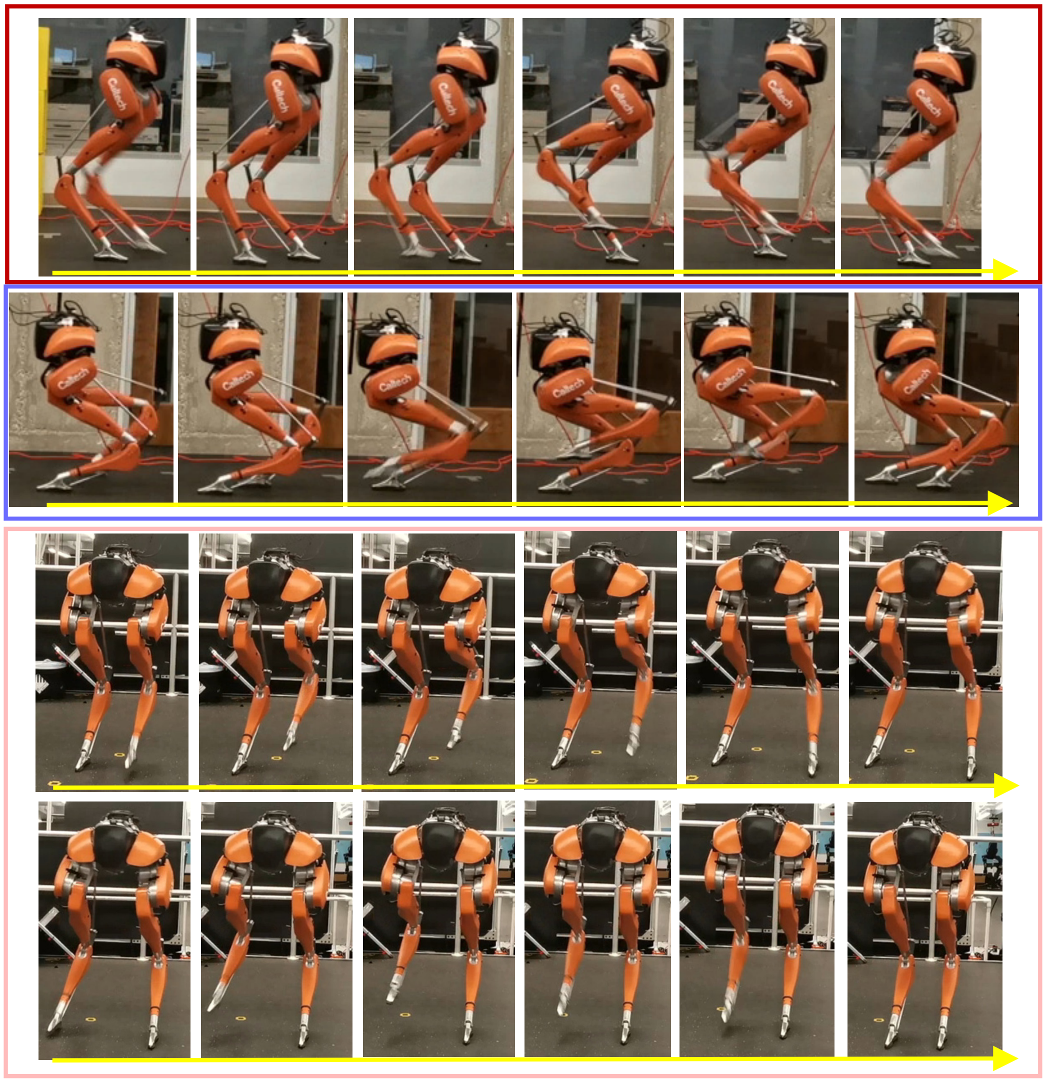

Figure 8.6: Walking tiles of the robot: the walking in the sagittal plane demonstrated P1 orbits and that in the coronal plane illustrated a P2 orbit.

extended Kalman filter offline to get a continuous estimation of the horizontal velocity of the COM on the robot. The estimated velocities are used as references but rather than the ground truth. This is because the estimation has non-neglectable errors that in nature come from the imperfection of the dynamics models and sensors.

Fig. 8.5 shows the horizontal COM states of a forward walking. The estimated velocities are compared with the approximated velocities from the H-LIP and the desired velocities. The desired walking velocities are tracked within reasonable errors. The error invariant sets are approximated in the same way (in Chapter 7), and the error states are all inside the invariant sets. Note that the error invariant sets are larger than those in Fig. 8.2. This is because the dynamics difference w is calculated using the estimated velocities; the estimation errors directly yield larger sets $W_{x,y}$ and thus larger sets $E_{x,y}$.

Additionally, the translational dynamics and transversal dynamics can be controlled separately. Therefore, we implement a turning controller that only changes the hip yaw angles and keeps the stepping controller intact. With turning, the robot can be joystick-controlled easily in confined environments.

Versatile Walking

To demonstrate the versatility in the gait design, we utilize the potentiometers on the remote to vary the gait parameters in real-time. As indicated previously, we select the step duration, swing foot clearance, step width in the P2 orbit, and the COM height. There are four potentiometers on the remote controller, and each potentiometer corresponds to one parameter. The reading on the potentiometers can oscillate, and we do not low-pass filter the values to show the robustness of the control implementation. Fig. 8.7 demonstrates a stepping-in-place walking with varying the four parameters. The ranges of the parameters are listed in Table 8.1. All the parameters can be varied continuously, and the H-LIP based stepping still can stabilize the walking. Fig. 8.8 demonstrates the changes of the values in the experiment. Additionally, forward walking behaviors with different COM heights are realized.

Table 8.1: Versatile Walking Parameters

Variables	Definition	Range
Step Duration	T	0.3 – 0.5s
Desired Swing Foot Clearance	$z_{\mathrm{sw}}^{\max}$	0.04 – 0.25m
Desired Step Width	u_{L}^{*}	0.08 – 0.45m
Desired COM Height	z_0	0.5 – 1m

For parameters outside of the range, the walking can be infeasible or destabilized. For instance, if the step width is too small or too larger, the walking becomes kinematically infeasible. If the COM is too tall or too low, each leg can extend or retract outside of its kinematic range of motion. If the foot clearance is extremely low, the robot then has a trivial SSP and cannot stabilize its walking via stepping; if it is too high, the vertical swing trajectory then requires large accelerations to lift-off and touch-down and thus exceeds the joint actuation limits. Similarly, the actuation

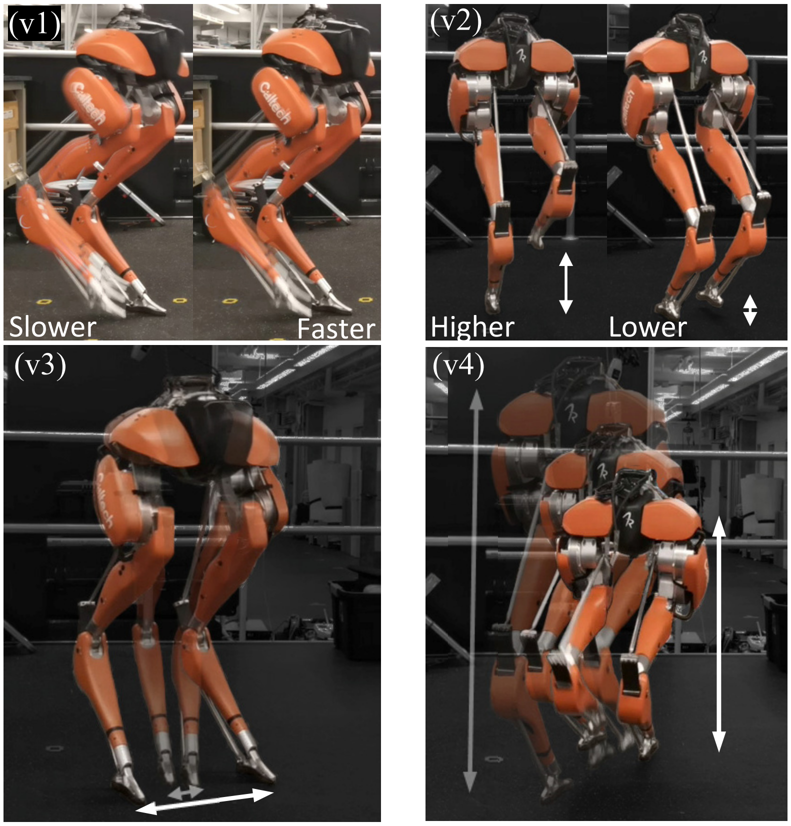

Figure 8.7: Illustration of the versatility of the realized walking: varying step frequency (v1), step clearance (v2), step width (v3), and COM height (v4).

limits also prevent the walking duration from being too small to track the swing trajectories. If the duration is too long, the robot can fall over before the swing leg strikes the ground to stabilize it.

The changes of the COM height and the step duration change the S2S approximation of the H-LIP (e.g., Eq. (3.12)). The implementation of the H-LIP stepping directly responds to the new S2S dynamics. Note that the vertical COM height is assumed constant on the H-LIP and that on the robot is controlled approximately constant. The height, however, can change between steps, as long as the vertical dynamics is not causing significant disturbance to the horizontal dynamics. The change of the swing foot clearance can change the impact velocity and potentially change w. Similarly, the variation on the step frequency also changes the integration of the dynamics error in the continuous domains, which then change w. In the experiment, the qualitative and quantitative effects of these parameters on w and then E are not analyzed due to the existence of the horizontal velocity error (from the H-LIP based velocity approximation in the control or the state estimation in the analysis).

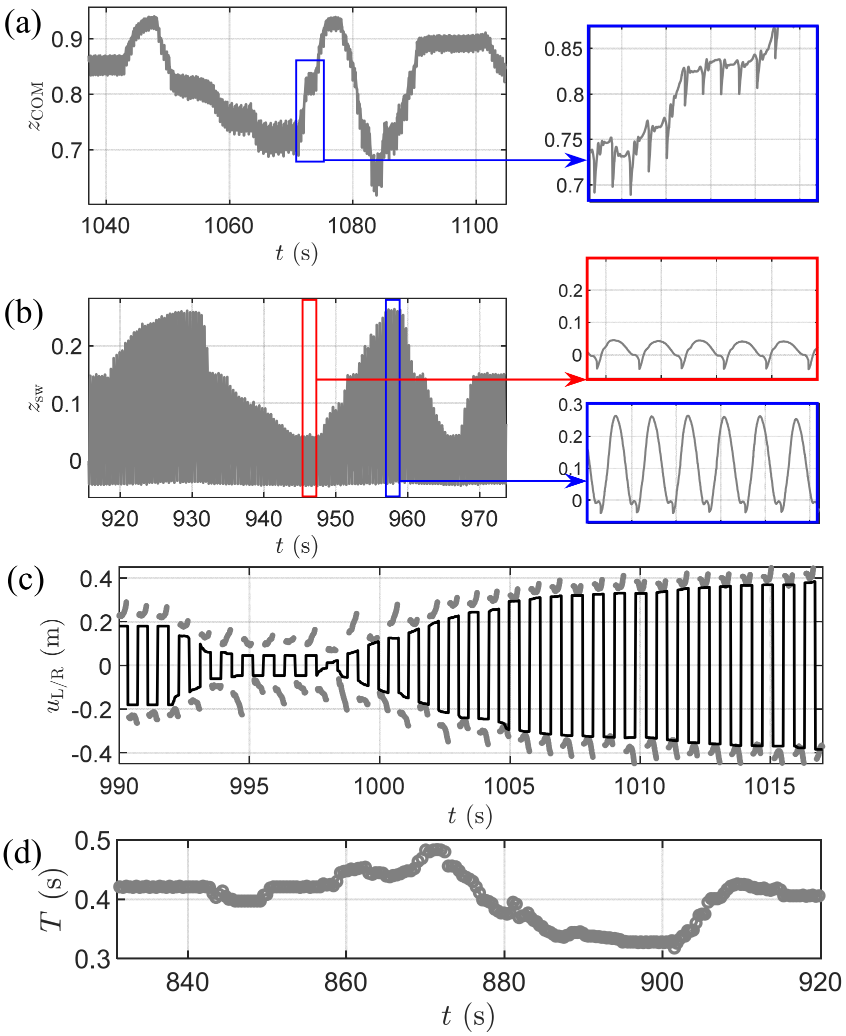

Figure 8.8: The trajectories of the (a) COM height and (b) vertical height of the swing foot in terms of the actual outputs, (c) the desired step width compared with the target step size $u^*_{\mathrm{L/R}}$ (black lines), and (d) the duration of the walking.

Instead, the experiment shows that versatile walking behaviors are stably generated with variations in the parameters.

Disturbance Rejection

Lastly, we demonstrate the robustness of the walking with disturbance rejection on the hardware. Since the H-LIP stepping provides COM state-dependent step size planning, the robot instantaneously reacts to external disturbances. We consider two types of disturbances: external pushes and ground variations. The external pushes directly disturb the S2S dynamics of the robot; the ground variations change the

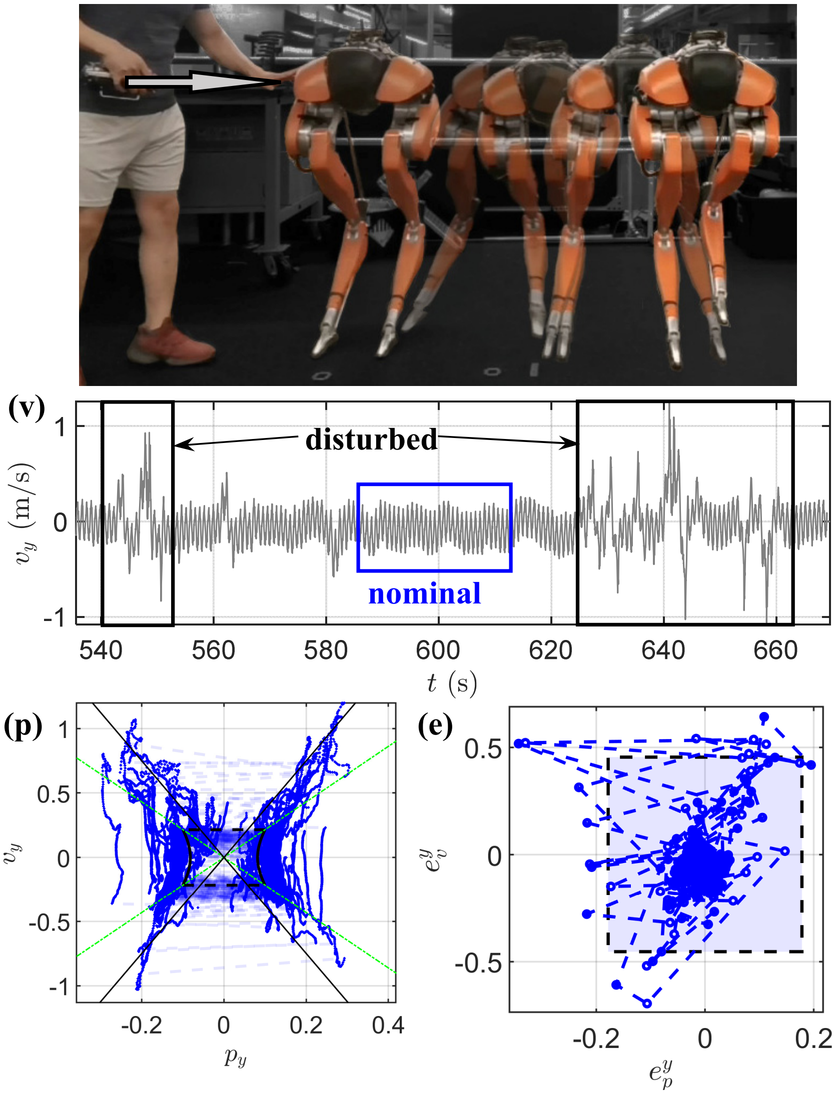

Figure 8.9: Illustration of the disturbance rejection on Cassie: (v) the estimated horizontal velocity in the coronal plane, (p) the horizontal state trajectory (the blue) in the coronal plane compared with the desired orbit of the H-LIP (the black), and (e) the error state trajectory (blue dots) compared with the error invariant set (transparent blue polytope).

domain durations, impact, and vertical COM behaviors, which indirectly disturb the S2S dynamics. We demonstrate walking with lateral pushes from a human operator (Fig. 8.9) and walking on grassy and uncertain terrain (Fig. 8.10).

With the push disturbances, the error state $\mathbf{e}^y$ can temporarily go outside of the invariant set E_y. The stepping controller then brings $\mathbf{e}^y$ back in E_y. In terms of the horizontal velocity, the robot is pushed to have large velocities and then the stepping controller drives the robot back to its nominal walking behavior. Since the

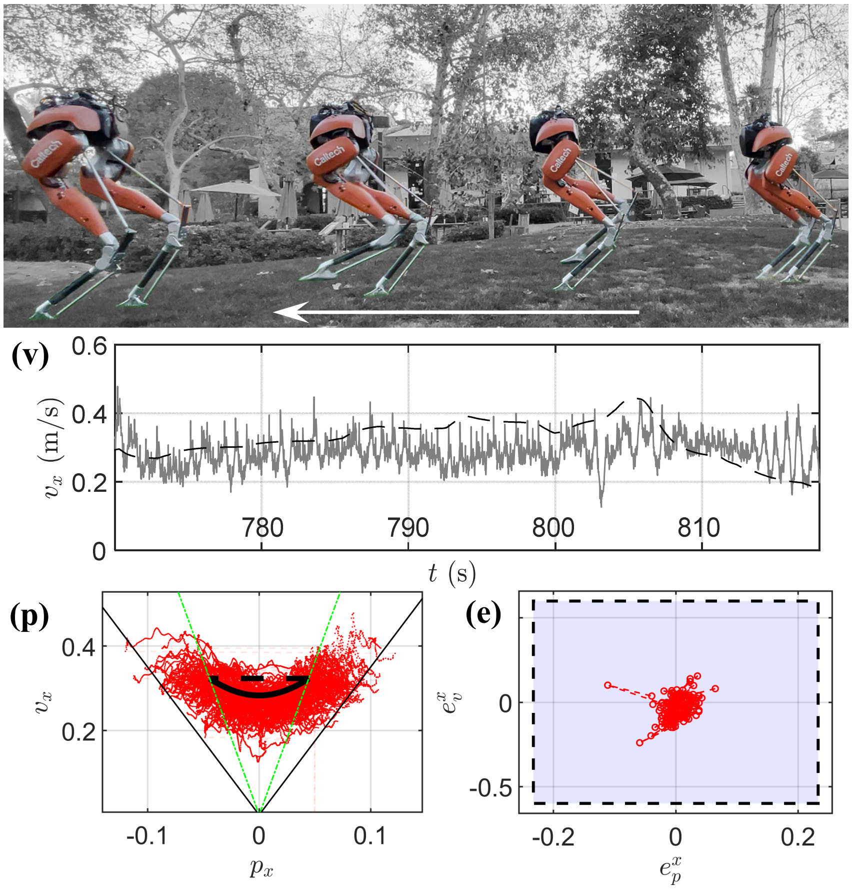

Figure 8.10: Walking on grassy and uncertain terrain: (v) the estimated forward velocity (gray line) and the desired velocity (black dashed line), (p) the horizontal state trajectory (the red) in the sagittal plane compared with the desired orbit of the H-LIP (the black) for a walking segment with an approximately constant v_x^d of 0.3m/s, and (e) the error state trajectory compared with the error invariant set.

kinematically feasible step width is very limited, the robot may take several steps to recover. If the push is excessive, the robot can fall over due to infeasible u_y^{des}.

When walking on grassy terrain, the horizontal dynamics of the robot are disturbed as the soil deforms and the height of the terrain varies. The continuous horizontal velocities thus have large variations as shown in Fig. 8.10; the error states, however, still lie inside the invariant sets.

Chapter 9

EXTENSION ⊃ GLOBAL POSITION CONTROL

The H-LIP based walking synthesis can be used not only for controlling local walking behaviors on bipedal robots, but also for controlling the global position of the robot. In this chapter, we demonstrate one of the extensions of the H-LIP based approach for global position control (GPC) [5, 147]. We start in Section 9.1 by including the global position in the H-LIP model to get a S2S dynamics with the global position. Following the similar formulation of the H-LIP based stepping controller, we demonstrate the GPC on the 3D-aSLIP walking in Section 9.2 and on Cassie in Section 9.3. We show that with the aid of MPC [53], the robot can follow various global trajectories with different shapes. Moreover, the robot can avoid obstacles and reject external disturbances during walking.

9.1 S2S Dynamics with Global Position Control

Besides the periodic orbits of the H-LIP, we are also interested in global/non-periodic behaviors, where the H-LIP can follow some custom specified global trajectories. To realize this kind of behavior, we first present the S2S dynamics that includes the global position in the formulation and then present the optimal control formulation to realize the desired walking behaviors on the H-LIP and finally on the robot via H-LIP based stepping.

S2S Dynamics with Global Position

We use the planner H-LIP as an example. Let c^x denote the global position of the point mass in the $x - z$ plane. The global position between steps at the end of the SSP is related by

$$c^x_{k+1} = c^x_k - p^x_k + p^x_{k+1} + u^x_k, \tag{9.1}$$

where the superscript $^-$ is omitted since we only deal with the states at the end of the SSP in this section. The superscript ^{x} is used to indicate the element in the $x - z$ plane; we will use it as a subscript when it is convenient. The global position can be put into the S2S formulation, which yields

$$\underbrace{\begin{bmatrix} c^x_{k+1} \\ p^x_{k+1} \\ v^x_{k+1} \end{bmatrix}}_{\tilde{\mathbf{x}}_{k+1}} = \underbrace{\begin{bmatrix} 1 & A_{1,1}-1 & A_{1,2} \\ 0 & A_{1,1} & A_{1,2} \\ 0 & A_{2,1} & A_{2,2} \end{bmatrix}}_{\tilde{A}} \underbrace{\begin{bmatrix} c^x_{k} \\ p^x_{k} \\ v^x_{k} \end{bmatrix}}_{\tilde{\mathbf{x}}_{k}} + \underbrace{\begin{bmatrix} B_1+1 \\ B_1 \\ B_2 \end{bmatrix}}_{\tilde{B}} u^x_k, \tag{9.2}$$

where the subscripts indicate the elements of the matrices A and B in Eq. (3.12) and Eq. (3.12). We use the overhead tilde to denote the elements in the S2S dynamics that includes the global position. In the latter, we call $\tilde{\mathbf{x}}$ the *extended pre-impact state* in the $x - z$ plane. For 3D walking, $\tilde{\mathbf{y}} = [c^y, p^y, v^y]^T$ is used to denote the extended pre-impact state in the $y - z$ plane.

Global Position Control

We consider the bipedal robot walks on flat ground in a 3D environment. The robot is given a walking path with a terminal location that it should reach. We assume the path is planned via a high-level planner from all the sensors on the robot. The path is supposed to be relatively smooth and obstacle-free. Additionally, the path is also generated with a speed profile, which is assumed to be feasible for the robot to realize. We parameterize the desired path by $\mathbf{r}(t)$ as

$$\mathbf{r}(t) = [r^d_x(t), r^d_y(t), r^d_\theta(t)]^T, \tag{9.3}$$

where $r^d_x(t), r^d_y(t)$ are the positions in the global frame and $r^d_\theta(t)$ be the angle of the tangent line to the path. The task for the walking is to drive the robot (depicted by its COM) to follow the path with a given time.

For underactuated robotic walking, it is not possible to exactly track the given path. To address the GPC problem on bipedal robots, we consider applying the H-LIP based synthesis: the GPC problem is first solved on the H-LIP, the motion of which is then realized on the bipedal robot approximately via the H-LIP based stepping controller. Note that since the extend state in the S2S dynamics has three elements, the deadbeat gain $\tilde{K}$ is solved from $(\tilde{A} + \tilde{B}\tilde{K})^3 = \mathbf{0}$, and the H-LIP based stepping for generating the desired step size on the robot becomes

$$u^{\text{des}} = u^{\text{H-LIP}} + \tilde{K}(\tilde{x} - \tilde{x}^{\text{H-LIP}}). \tag{9.4}$$

Similarly, the application of the H-LIP based stepping makes sure that the robot stays closely to the H-LIP and thus follows the path.

MPC on H-LIP

We solve the GPC on the H-LIP by formulating a Model Predictive Control (MPC) problem to online generate optimal step location for best-tracking the desired path. The superscript $^{\text{H-LIP}}$ is omitted in this part.

Optimization Variables: At each step, the next N steps are planned. Thus, all the states $\tilde{\mathbf{x}}_k$, $\tilde{\mathbf{y}}_k$ and all the inputs $u_k^{x,y}$ are selected as the optimization variables, where $k = 1, \dots, N$. The current step is indexed as 1.

Cost Function: The cost function of the MPC includes two parts: one is encoding the tracking performance as the distance between the mass states and the desired global trajectory, and the other past is penalizing the step sizes. Again let $T = T_{\text{SSP}} + T_{\text{DSP}}$ be the period of the walking. Then the look-ahead time horizon is NT. Recall that, inside each step, the trajectory of the mass at a specific time instant t can be expressed by a linear function of the states:

$$\left[c^x(t), v^x(t)\right]^T = A_t(t)\tilde{\mathbf{x}}_k + B_t(t)u_k^x, \tag{9.5}$$

$$\left[c^y(t), v^y(t)\right]^T = A_t(t)\tilde{\mathbf{y}}_k + B_t(t)u_k^y, \tag{9.6}$$

where $t \in [kT, (k+1)T]$, and $A_t(t), B_t(t)$ are derived from the piecewise linear dynamics of the H-LIP. Supposing n points are sampled in the time horizon $[t_0, t_0 + NT]$ to represent the trajectory of the states, the cost function on tracking performance is

$$J_t = \textstyle\sum_{k=1}^n (c^x(t_k) - r_x^d(t_k))^2 + (v^x(t_k) - v_x^d(t_k))^2 + (c^y(t_k) - r_y^d(t_k))^2 + (v^y(t_k) - v_y^d(t_k))^2,$$

where $t_k = t_0 + k\frac{NT}{n}$. Additionally, we add cost on the input to penalize large step sizes:

$$J_u = \sum_{k=1}^{N} u_k^{x^2} + u_k^{y^2}. \tag{9.7}$$

The final cost function is a combination of the two:

$$J_{\text{MPC}} = J_t + \alpha J_u, \tag{9.8}$$

which is a quadratic function of all the variables. $\alpha \in \mathbb{R}$ is an coefficient to leverage the tracking and planned step sizes.

Constraint Encoding: The MPC on the H-LIP should be cognitive about the potential constraints that the robot has such as limited step sizes. Theoretically, this can be set by $u^{\text{H-LIP}} \in U \ominus KE$. The easiest way is to let the robot face the direction of walking since the step size in the sagittal plane is much less limited than that in the lateral plane. As the point mass model of the H-LIP has no definition of orientation, we add a trivial torso with no inertia on the H-LIP (Fig. 9.1 (a)) to indicate the orientation of the model. The transversal dynamics of the torso is trivial and does not affect the horizontal dynamics of the COM. Suppose the horizontal dynamics is always expressed in the inertial frame for the consistency of global path tracking, the turning motion affects available step size in the $x - z$ plane and $y - z$ plane. Assuming the torso is controlled to be $r_\theta^d(t)$, the sagittal plane is aligned with the tangent line. The step length in the sagittal plane and the step width in the lateral plane can be expressed as

$$s_{\mathrm{L}} = u^x \cos(r_\theta^d) + u^y \sin(r_\theta^d(t)), \tag{9.9}$$

$$s_{\mathrm{W}} = -u^x \sin(r_\theta^d) + u^y \cos(r_\theta^d), \tag{9.10}$$

where u^x, u^y are the step sizes projected to the $x - z$ and $y - z$ planes, and $s_{\mathrm{L}}, s_{\mathrm{W}}$ are the step length and step width in its sagittal and lateral planes, respectively. Then the step size constraints in the MPC are modified by the following two inequality constraints:

$$s_{\mathrm{L}}^{\min} \leq u_k^x \cos(r_\theta^d) + u_k^y \sin(r_\theta^d) \leq s_{\mathrm{L}}^{\max}, \tag{9.11}$$

$$s_{\mathrm{W}}^{\min} \leq -u_k^x \sin(r_\theta^d) + u_k^y \cos(r_\theta^d) \leq s_{\mathrm{W}}^{\max}, \tag{9.12}$$

where $s_{\mathrm{L}}^{\mathrm{min/max}}, s_{\mathrm{W}}^{\mathrm{min/max}}$ are the available step sizes in each plane, which are linear functions of the states. Additionally, the robot should avoid kinematic conflicts for foot stepping. It can be easily specified through enforcing a finite minimum step width in the lateral plane, which can be added into the above constraint by changing $s_{\mathrm{W}}^{\mathrm{min/max}}$.

The dynamics in each step are encoded via linear equality constraints. Additional constraints include the initial state constraint, step size limits (input limits), and the step size difference constraint, which are constraints considered for the application of the robots. The step size limits come from the physical kinematic feasibility of the robot. The step size difference is that, $|u_{k+1} - u_k| \leq \Delta U$ where ΔU is a constant. This constraint avoids the system to dramatically change step sizes consecutively, which may lead to undesirable behaviors on the robot.

MPC: We compactly present the MPC formulation for the H-LIP. At each step, a

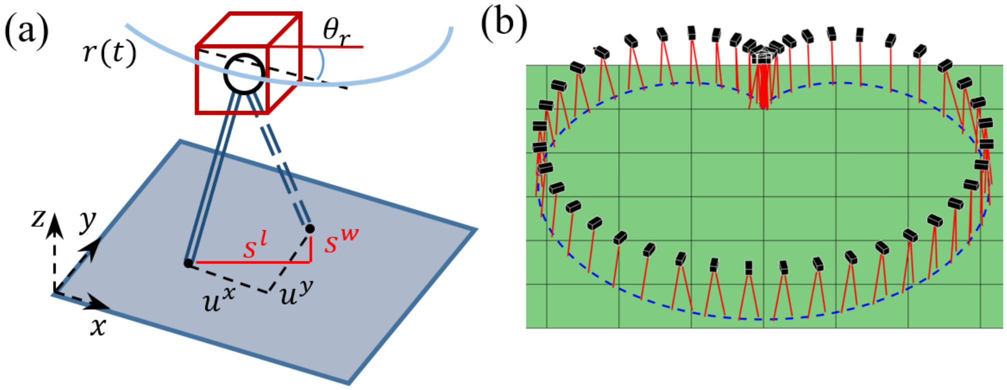

Figure 9.1: (a) The H-LIP with a trivial torso walking in 3D. (b) An example of path tracking of the H-LIP in 3D (blue dashed line is the desired path)

constrained quadratic program (QP) is formulated and solved. The QP is as follows:

$$\begin{aligned} u^{x,y}_{1,\dots,N}, \tilde{\mathbf{x}}_{1,\dots,N}, \tilde{\mathbf{y}}_{1,\dots,N} = & \underset{\{u^{x,y}_{1,\dots,N}, \tilde{x}_{1,\dots,N}, \tilde{y}_{1,\dots,N} \in \mathbb{R}^{8\times N}}{\operatorname{argmin}} J_{\text{MPC}} \\ \text{s.t.} \quad & \tilde{\mathbf{x}}_{k+1} = \tilde{A}\tilde{\mathbf{x}}_k + \tilde{B}u^x_k, k \in \mathbb{K} \\ & \tilde{\mathbf{y}}_{k+1} = \tilde{A}\tilde{\mathbf{y}}_k + \tilde{B}u^y_k, k \in \mathbb{K} \\ & |u^x_{k+1} - u^x_k| \leq \Delta U, k \in \mathbb{K} \\ & \text{Eq. (9.11), (9.12)}, \; k \in \mathbb{K} \\ & \tilde{\mathbf{x}}_1 = \tilde{\mathbf{x}}_{\text{now}}, \tilde{\mathbf{y}}_1 = \tilde{\mathbf{y}}_{\text{now}}, \end{aligned} \tag{9.13}$$

where $\mathbb{K} = \{1, \dots, N-1\}$, $\tilde{\mathbf{x}}_{\text{now}}, \tilde{\mathbf{y}}_{\text{now}}$ are the states of current step of the H-LIP. The first solution of the step sizes $[u^x_1, u^y_1]$ is applied at current optimization, which are used as the nominal step sizes in the H-LIP based stepping in Eq. (3.17).

Fig. 9.1 (b) shows an example of the optimized walking of the H-LIP with a torso in 3D for GPC. Here we assume that the torso orientation is aligned with the path direction but it can be freely decided. The feasibility of the MPC is not guaranteed in general, which should be cognitive by the high-level planer on the global trajectory. Additionally, we should note that the H-LIP based approximation reduces the available step sizes in the MPC. Robust MPC [83] can also be directly applied to the robot to directly generate the step size on the robot. We do not present these modifications in this chapter. Instead, we show that the proposed MPC with the H-LIP based stepping can be realized on the bipedal walking systems to track global trajectories in the following two sections.

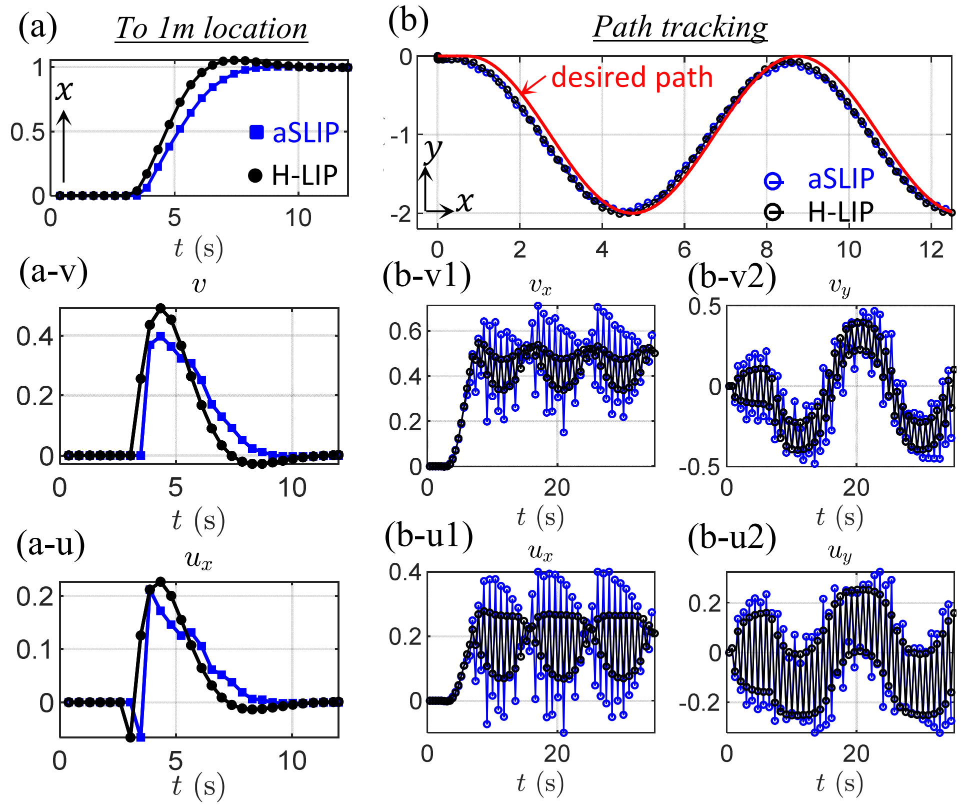

Figure 9.2: (a) Results of the fixed location tracking in terms of the forward positions x, forward velocities v and step sizes u_x. (b) Results of the path tracking with the velocities and step sizes in each plane. The black and blue indicate the results of the H-LIP and the aSLIP, respectively.

9.2 Global Position Control on 3D-aSLIP

We use the indirect approach in Chapter 6 to realize the GPC on the 3D-aSLIP. Similarly, a periodic stepping-in-place trajectory is first optimized and then the leg length is controlled to track the optimized leg length trajectory. The MPC is formulated and solved using YALMIP [78] at the step-level to determine the nominal step, and the H-LIP based stepping in Eq. (9.4) is applied to determine the current desired step size in the SSP. Here we demonstrate the approach via the following scenarios of walking.

Fixed Location Control: Suppose the task is to walk to a fixed location. Using the MPC approach, the task can be encoded as equality constraints: $\tilde{\mathbf{x}}_N = \tilde{\mathbf{x}}^d, \tilde{\mathbf{y}}_N = \tilde{\mathbf{y}}^d$ or as a part of the cost:

$$J_t = \textstyle\sum_{k=1}^{N}(\tilde{\mathbf{x}}_k - \tilde{\mathbf{x}}^d)^T Q(\tilde{\mathbf{x}}_k - \tilde{\mathbf{x}}^d) + (\tilde{\mathbf{y}}_k - \tilde{\mathbf{y}}^d)^T Q(\tilde{\mathbf{y}}_k - \tilde{\mathbf{y}}^d), \tag{9.14}$$

where Q is the parameter matrix. Here we illustrate this by using one example of

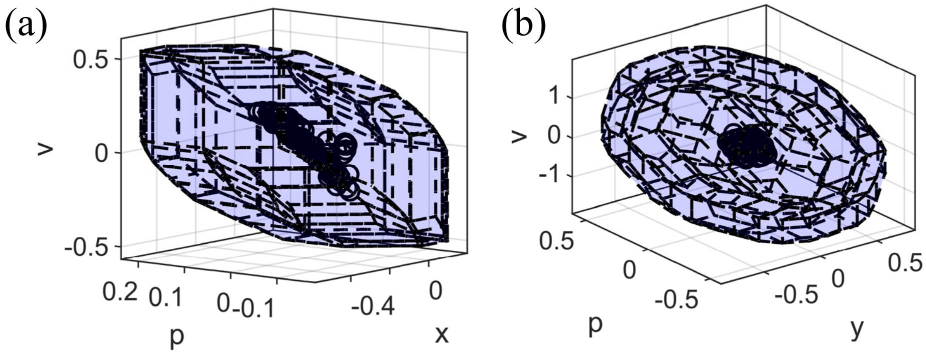

Figure 9.3: The disturbance invariant set E in blue transparent polytopes and the state difference $\mathbf{e}$ in black dots in the $x-z$ plane (a) and the $y-z$ plane (b).

controlling the aSLIP to a forward location in the $x-z$ plane where $\tilde{\mathbf{x}}^d = [1,0,0]^T$. The optimized result on the H-LIP and the generated walking on the aSLIP are shown in Fig. 9.2. Behaving like the H-LIP, the aSLIP walks to the desired location and stays there.

Trajectory Tracking: Now we consider the task is to track a trajectory on the ground. We use an example of tracking a sinusoidal path on the ground. Fig. 9.2 shows the results, where the aSLIP behaves closely to the H-LIP and tracks the desired trajectory closely.

The simulation results on the aSLIP indicate the success of applying the H-LIP based walking synthesis for GPC. One thing to note is that the initial H-LIP state should be close to that of the horizontal state of the aSLIP. In the examples, we select the H-LIP state to be identical to that of the aSLIP in the beginning, which indicates $\mathbf{e}_0 = \tilde{\mathbf{x}} - \tilde{\mathbf{x}}_{\text{H-LIP}} = \mathbf{0} \in E$. Then $\mathbf{e}_k \in E$ for all $k \in N$. To verify this, we first numerically calculate w in the simulated walking, and then approximate W via a polytope. Note that since the indirect approach in Section 6.2 is applied, W is not small. Fig. 9.3 shows the approximation of E and the error $\mathbf{e}$ for the example of trajectory tracking. $\mathbf{e} \in E$, which verifies the application of the stepping controller that keeps the error small.

9.3 Global Position Control on Cassie

In this section, we present the application of the H-LIP based synthesis to control the global position of Cassie. Similar to the application on the aSLIP, we apply the MPC with H-LIP based stepping for GPC. Since Cassie is a complex 3D underactuated robot, practical considerations are needed on generating the behavior on Cassie. We illustrate the modification on the output construction for GPC, and finally present the results and the analysis of this approach.

GPC on Cassie

The walking synthesis frame is identical to that in Chapter 7 with some exceptions: the nominal step sizes are calculated from the MPC, the desired step sizes are calculated via Eq. (9.4), and turning should be enabled to let the robot walk to the direction of the desired global trajectory. With turning, the robot can 'face' towards the direction of walking for observing and planning the path using its sensors if necessary. Moreover, the robot typically has different ranges of motion in its sagittal and lateral planes. With turning, the robot can flexibly align its plane with a larger range of motion to the direction of walking. Thus we want the pelvis angle to be $\phi_{\text{yaw}} \rightarrow r^d_\theta(t)$ by defining the desired yaw angle of the pelvis in the output construction as

$$\phi^{\text{des}}_{\text{yaw}} = r^d_\theta(t). \tag{9.15}$$

The desired pitch and roll angles of the pelvis are still set to 0. The desired yaw angle of the swing foot is constructed as

$$\psi^{\text{des}}_{\text{yaw}}(t) = (1 - \mathcal{B}_h(t))\psi^+_{\text{yaw}} + \mathcal{B}_h(t) r^d_\theta(t), \tag{9.16}$$

where ψ^+_{yaw} is the yaw angle of the swing foot in the beginning of the SSP. The rest of the output construction is identical to those in Chapter 7. Both direct and indirect approaches can be applied.

Results

To evaluate the GPC on Cassie, we designed various shapes of paths, including a circle, a cardioid, a square, and a sinusoid. Trapezoidal speed profiles are designed for tracking the cardioid, the circle, and the sinusoid path. A triangle speed profile is used on the square path. For the experimental results, we consider the square path. This will be further discussed after describing the simulation results. A video of the simulation and experiment results can be seen in `https://youtu.be/06efo-U1mrw`.

Tracking Results: Using the proposed approach, the robot can track all global trajectories well in terms of the position and velocity profiles (Fig. 9.4). The circle and cardioid can be tracked easily even with relatively large speeds. The sinusoid and square paths are designed as challenging examples. On the sinusoid, the robot walks and turns significantly. While on the square, the path is not smooth, and

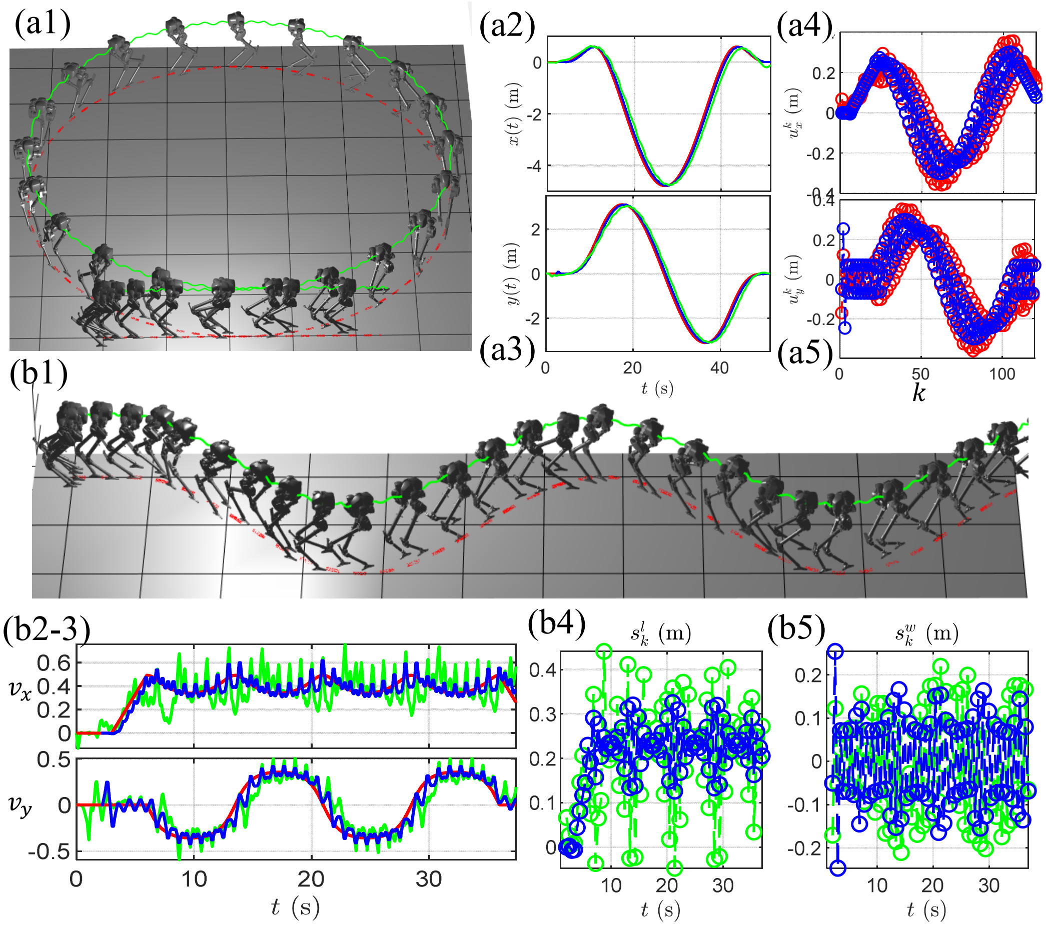

Figure 9.4: Simulation results (Red lines are the desired, blue lines are the planned trajectories of the H-LIP, and green lines are the actual trajectories of Cassie): Tracking a circle path (a) and a sinusoidal path (b) in terms of the global position/velocity trajectories (2,3) and the step sizes in the different planes (4, 5).

thus the robot has to come to a stop and turn. More dynamical maneuvers, such as sharp turning with high speeds, are challenging to realize on the robot considering the existence of the kinematic limits, torque bounds, and spring underactuation. Undoubtedly, the tracking can fail when the required walking and turning speeds are too large simultaneously, e.g., over 1m/s and 45deg/s. The failure often happens when the step sizes are too large for the robot to track due to limited motor torques. This can be easily avoided by setting limits on the allowed forward and turning velocities.

Obstacle Avoidance: The given path should be obstacle-free. In the case of existing obstacles on the path, we can avoid obstacles by adding constraints in the MPC formulation, e.g., in the form of $|[c^x(t), c^y(t)]^T - \mathbf{c}_{\text{obstacle}}| > d$, with $\mathbf{c}_{\text{obstacle}}$ being

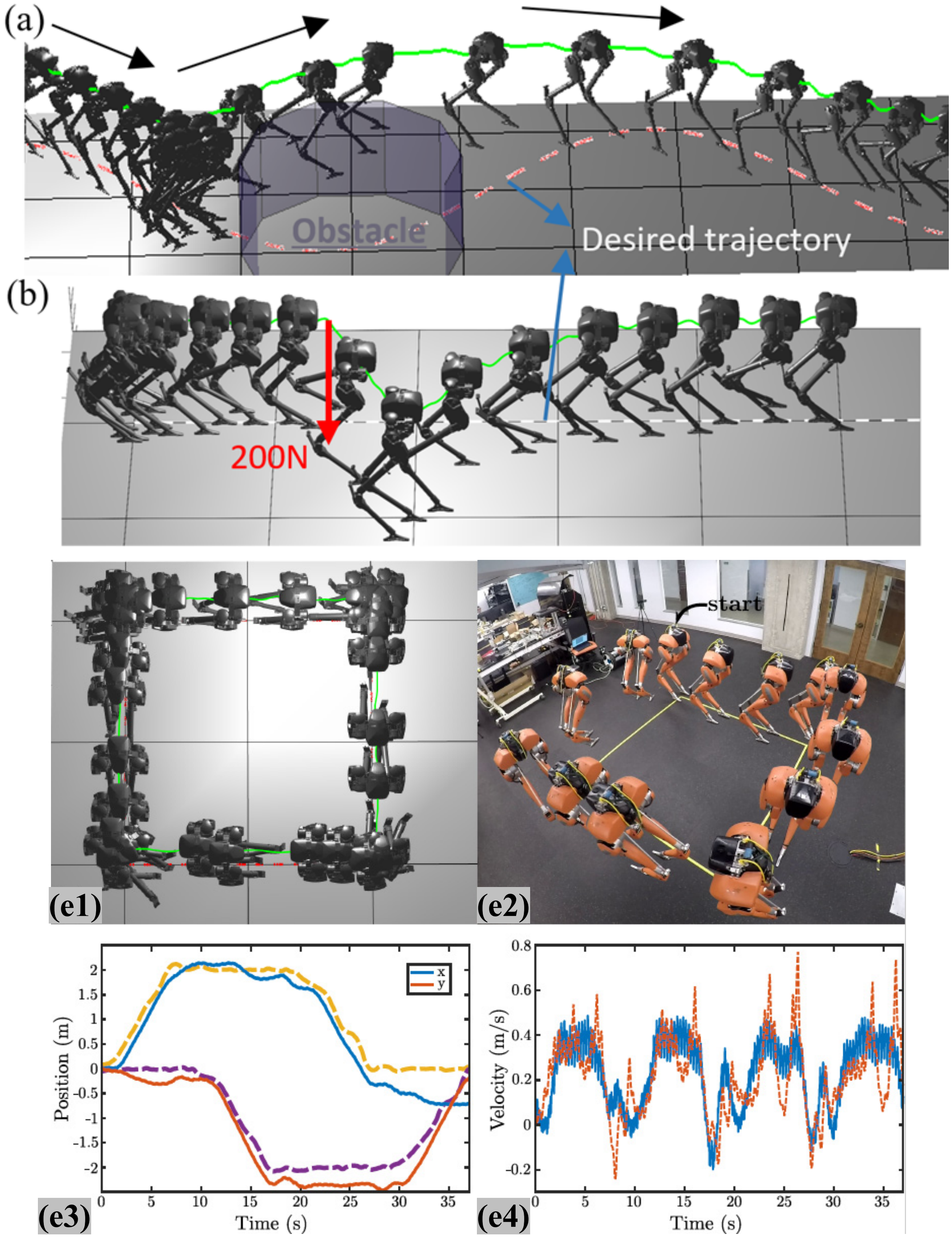

Figure 9.5: The simulated walking with (a) avoiding an obstacle on the path and (b) disturbance rejection. Cassie follows the square path in simulation (e1) and in experiment (e2). The pelvis global positions (e3) and heading velocities (e4) from simulation (dashed) and the experiment (solid).

the position of the obstacle and d being a minimum distance between the robot and the obstacle. This changes the MPC from a QP into a quadratic constrained QP, which is also fast to solve on the low dimensional dynamics of the H-LIP. Fig. 9.5 shows an example of avoiding an obstacle while tracking a sinusoidal path.

Disturbance Rejection: The step planning approach can also handle unknown external disturbances on the robot during walking. Fig. 9.5 (b) shows an example where an external lateral force of 200N is added on the pelvis at $t = 10$s for 0.1s duration. The robot was pushed away from the path but then walked back to its original path.

Experiment: The tracking for a square path was implemented on Cassie as a preliminary experiment to demonstrate that the generated motion from simulation using this approach is dynamically achievable on hardware. Due to the missing of sensors to detect the robot location globally, we have not yet implemented the closed-loop controller as being presented here on the hardware. Instead, as a proof-of-concept, we use the simulated walking velocities and turning rates of the robot as targets, which were tracked on the robot with another controller in [105]. The robot ultimately followed the desired walking profiles on hardware, with a comparison of the positions and velocities of the experiment and simulation shown in Fig. 9.5. As the robot is tracking the commanded velocities, there is a slight drift in the position and heading as it executes the later segments of the shape. Note that this experiment was conducted before the implementation of the controller in Chapter 8. In the future, we will faithfully realize the proposed GPC on the hardware using the H-LIP based direct synthesis by incorporating global sensing.

Chapter 10

EXTENSION ⊃ WALKING ON ROUGH TERRAIN

In this chapter, we show another extension of the H-LIP based synthesis for generating walking on rough terrains, which is an important behavior to realize on bipedal robots. Since the terrain height changes during walking, the vertical COM trajectory has to be planned accordingly. We first present the dynamics model of the H-LIP for walking on rough terrains in Section 10.1. With appropriate assumptions and choice of the coordinates, the S2S dynamics of the walking is also a linear controlled system. As a result, the stepping stabilization can be applied for a similar bipedal walking system on rough terrain. In Section 10.2, we use the aSLIP model as an example to show the direct synthesis with the aid from BBF-QP in Chapter 6. Lastly, we show the simulation results of enabling aSLIP to walk on slopes, stairs, sinusoidal terrains, and general rough terrains with uncertainties on the terrain height.

10.1 S2S Dynamics of H-LIP on Rough Terrain

To extend the H-LIP stepping to rough terrains, we first consider the walking of the H-LIP on a slope, as shown in Fig. 10.1 (b). Let z_0 denote the vertical distance between the mass and the slope, θ_s be the degree of the slope, and θ_x be the angle between the leg and the slope. Let p_x be the forward position of the mass along the direction of the slope, and p be that in the inertial frame. We assume z_0 is constant, and thus the point mass moves on the line which is parallel to the slope. The dynamics of the point mass are derived as follows.

SSP dynamics: The dynamics of the H-LIP in the SSP is identical to that of the LIP in [61] with zero ankle torque. One can directly apply the results of Eq. (12) in [61] to get the dynamics. The motion is caused by the leg force F and the gravitational force mg, and thus the Newton-Euler equations are

$$m\ddot{p}_x = F\cos(\theta_x) - mg\sin(\theta_s),$$
$$F\sin(\theta_x) = mg\cos(\theta_s).$$

Solving for $\ddot{p}_x$ yields

$$\ddot{p}_x = \lambda^2 p_x - g\sin(\theta_s).$$

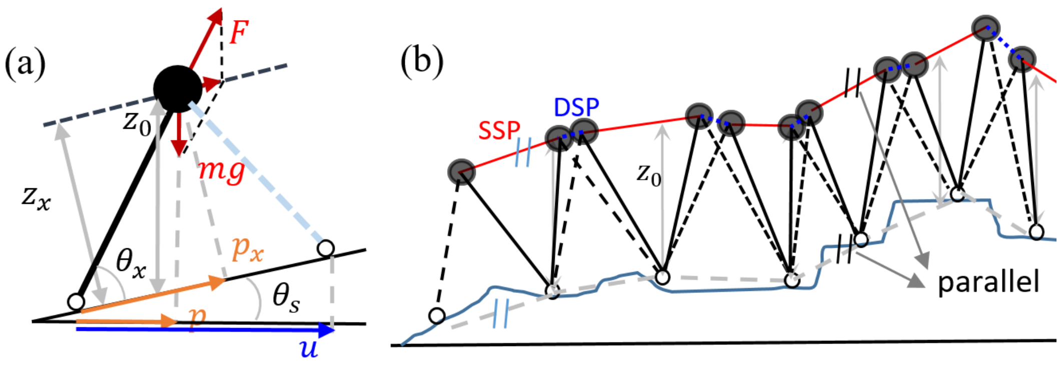

Figure 10.1: (a) Illustration of the walking on the H-LIP model on a slope in the SSP. (b) H-LIP walking on rough terrain, where the dashed lines indicate the leg that is about to lift off or strike the ground.

Since $p = p_x \cos(\theta_s) - z_x \sin(\theta_s)$, we have

$$\ddot{p} = \ddot{p}_x \cos(\theta_s) = \lambda^2 p, \tag{10.1}$$

which is identical to the dynamics of walking in the SSP on flat ground.

DSP dynamics: Similarly, in the DSP, two leg forces act on the point mass. Since the mass is assumed to move in parallel to the slope, the net acceleration should point to the direction which is parallel to the slope. Unlike the case in the SSP, the Newton-Euler equations cannot yield deterministic leg forces in the DSP. In other words, the magnitude of the net acceleration is controllable from the leg forces. For simplicity, we assume that the acceleration is zero.

S2S dynamics: As a result, by describing the state and input in the inertial frame, the dynamics of the H-LIP walking on the slope is identical to the dynamics on flat ground, so is the S2S dynamics and the resulting stepping controller.

On rough terrain: The walking model of the H-LIP on general non-flat terrains is based on the walking on slopes. Given a sequence of steps of the H-LIP walking on rough terrain, the walking is equivalent to walking on piecewise continuous slopes, as illustrated in Fig. 10.1 (c). Since the slope changes with each step, the assumption has to be made to enable the change of the slope. For instance, in [62], an impulse of the leg force is assumed to change the slope rate when the leg is strictly vertical in the SSP. Here we assume the slope changes in the DSP, where the leg forces can simultaneously create zero horizontal acceleration and change the vertical trajectory. Therefore, the dynamics in both domains remain the same, and

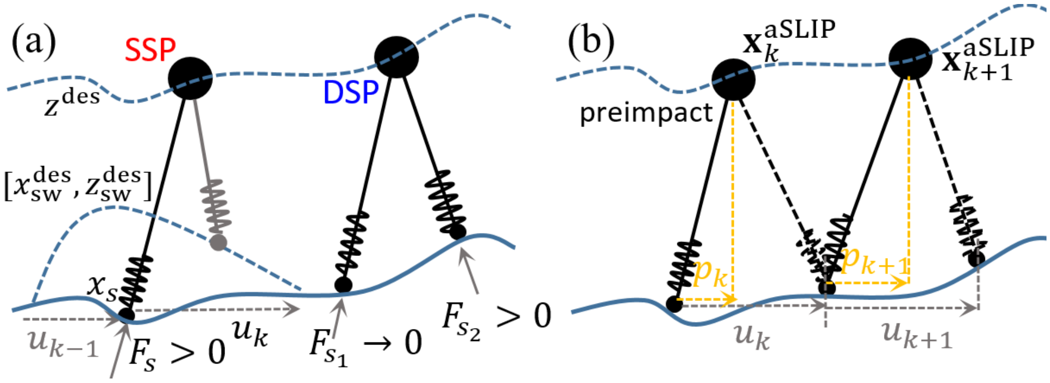
(a)
SSP
z^{des}
DSP
$[x_{\mathrm{sw}}^{\mathrm{des}}, z_{\mathrm{sw}}^{\mathrm{des}}]$
x_s
u_{k-1}
$F_s > 0$
u_k
$F_{s_1} \to 0$
$F_{s_2} > 0$
(b)
$\mathbf{x}_k^{\mathrm{aSLIP}}$
$\mathbf{x}_{k+1}^{\mathrm{aSLIP}}$
preimpact
p_k
p_{k+1}
u_k
u_{k+1}

Figure 10.2: (a) The control specifications and (b) the step-to-step dynamics.

so does the S2S dynamics. The assumptions on the H-LIP walking are designed to have a linear S2S dynamics to approximate and control the actual walking of a bipedal walking system. Thus, the H-LIP based stepping in Eq. 3.17 can still be applied to generate the desired step size for realizing desired walking behaviors.

10.2 aSLIP Walking on Rough Terrain

In this section, we extend the H-LIP based synthesis for realizing aSLIP walking on rough terrain using the direct approach in Chapter 6. We consider that the walking requires three specifications as shown in Fig. 10.2 (a): it keeps a vertical distance from the ground, the swing foot periodically lifts off and strikes the ground to switch support legs, and the swing foot steps to certain locations to produce a desired horizontal behavior. The corresponding controls are briefly explained as follows.

Vertical Mass Control via BBF-QP: The vertical state z is expected to follow the desired trajectory z^{des} which has an approximately constant distance from the ground (see Fig. 10.2). The vertical tracking and the leg force conditions are solved via the BBF-QPs, which has been explained in Chapter 6.

Vertical Swing Foot Control: The vertical position of the swing foot is controlled to lift off, avoid scuffing, and strike on the ground to finish the SSP at appropriate timing. The desired vertical swing foot position is constructed as

$$z_{\mathrm{sw}}^{\mathrm{des}} = z_{\mathrm{sw}}^{\mathrm{time}}(t) + z^{\mathrm{terrain}}(x_{\mathrm{sw}}), \tag{10.2}$$

where $z_{\mathrm{sw}}^{\mathrm{time}}(t)$ is the time-dependent component and $z^{\mathrm{terrain}}(x_{\mathrm{sw}})$ is the terrain profile. $z_{\mathrm{sw}}^{\mathrm{time}}(t)$ is constructed so that the swing foot lifts off from the ground, reaches to a maximum height $z_{\text{sw}}^{\max}$ and then strikes the ground at $t = T_{\text{SSP}}$. An example of the time-dependent component is $z_{\text{sw}}^{\text{time}}(t) = z_{\text{sw}}^{\max}\cos(\frac{t}{T_{\text{SSP}}}\pi - \frac{1}{2}\pi)$. The spring compression is assumed to go to zero on the swing leg, thus we select $z_{\text{sw}} = z - L_{\text{sw}}\cos(\beta_{\text{sw}})$ and apply a feedback linearizing controller to drive $z_{\text{sw}} \rightarrow z_{\text{sw}}^{\text{des}}$. Note that the terrain profile may not be exactly known in practice, for which we assume an estimated version is available. The uncertainty on the terrain height creates an uncertainty on the duration of the SSP: early strike causes a shorter duration and late strike produces a longer duration. The result section will show that the proposed approach is robust to the terrain uncertainty.

Horizontal Mass Control via H-LIP Stepping: The horizontal state $\mathbf{x}$ should be controlled for walking. Again, due to the point-foot underactuation, the horizontal state cannot be continuously controlled. We apply the H-LIP based stepping by using the S2S dynamics of the H-LIP on rough terrain to approximate the S2S of the aSLIP model (see Fig. 10.2 (b)). Then, the desired step size is continuously determined in the SSP. Again, we simply construct a smooth trajectory of the horizontal swing foot position to transit from the previous step location to the desired location.

10.3 Results and Discussion

The control parameters and system parameters are chosen identically to those in Chapter 6. The stepping gain K is chosen to be the deadbeat gain for all the walking for consistency. The BBF-QP based controller is solved at 1kHz. The aSLIP starts from an initial static configuration and walks to a desired pre-impact velocity v^*. A video of the results can be seen in `https://youtu.be/fUZu6y-Gu4g`.

Slopes

We first evaluate the approach for walking on slopes, for which, the desired vertical trajectory is parallel to the terrain; then it is similar to walking on flat ground. Under this circumstance, the aSLIP best matches the original assumption of the H-LIP walking. Fig. 10.3 shows the walking on slopes up to ±30°: (b) the converged velocities are still close to the desired one, (c) the vertical trajectories are controlled closely to the desired ones, and (d) the error states are inside E. The walking performance does not vary significantly on different slopes due to the trivial foot-ground impact.

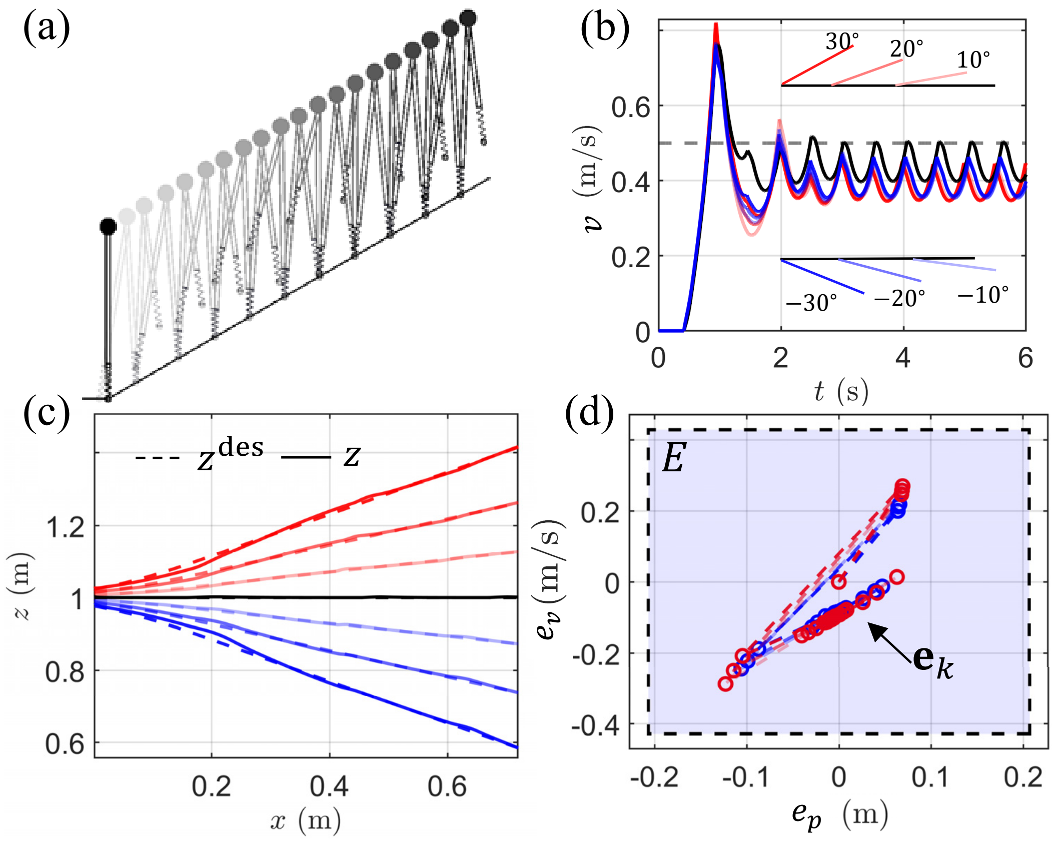

Figure 10.3: Walking on slopes with $v^* = 0.5$m/s: (a) an illustration of the walking on an incline; (b) the velocity trajectories, (c) the vertical trajectories, and (d) the error state trajectories for walking on different slopes (the red plots indicate the inclines, the blue plots indicates the declines and the black plots represent the flat ground).

Sin Waves, Stairs, and Rough Terrains

We then evaluate the walking on sin waves, stairs, and general rough terrains, as shown in Fig. 10.4. The desired vertical trajectories z^{des} are not necessarily piecewise linear and do not directly match the local slope assumption of the H-LIP, which presumably creates a larger w and thus a looser tracking performance on the horizontal state.

On sine waves, $z^{\mathrm{des}} = z_0 + z^{\mathrm{terrain}}$. For walking on stairs, we apply a moving averaging filter on z^{terrain} to generate a smooth z^{des}. For walking on general rough terrains, we assume the terrain height is not exactly known. The terrain is generated with a combination of slopes, stairs, and sine waves plus a uniformly distributed noise with a maximum magnitude δz. We apply the moving averaging on the noise-free

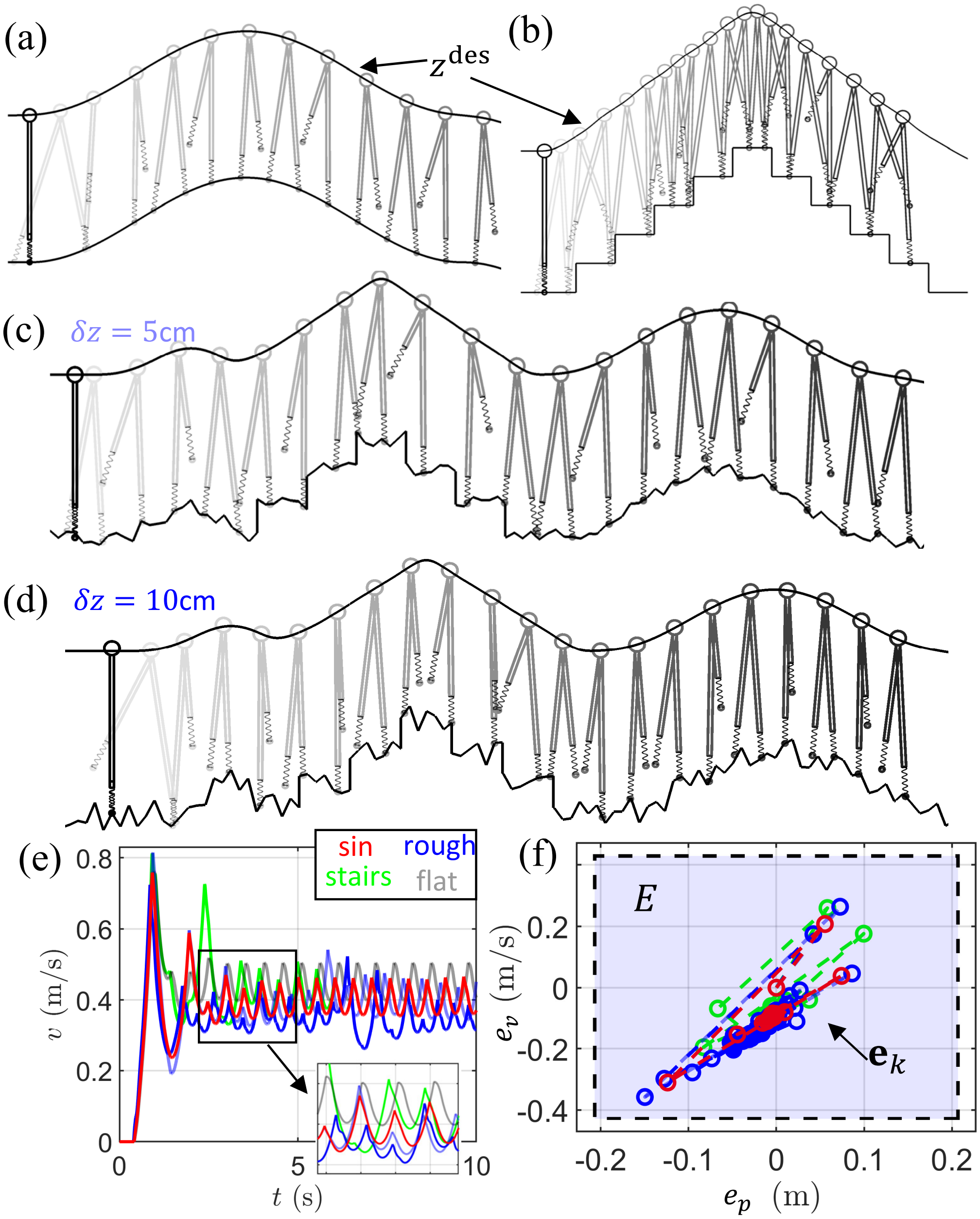

Figure 10.4: Walking with $v^* = 0.5$m/s on (a) sinusoidal terrain, (b) stairs, (c,d) rough terrains; (e) the velocity trajectories and (f) the error state trajectories.

profile to get z^{des}. The noise can be viewed as the measurement error from sensors on a physical robot. We tested the cases with $\delta z = 0, 5, 10$cm. For even larger (unrealistic) noises, kinematic violation starts to happen, i.e., leg collides on the edges of the terrains during walking.

The results are shown in Fig. 10.4: the vertical trajectories are well-tracked, and the horizontal velocities (compared with the walking on flat ground) are tracked approximately. As expected, the walking on the rough terrains generated larger oscillations (larger $\delta z \rightarrow$ larger oscillations). This is in part because the noise creates variations on T_{SSP}, which contributes to w. Despite the velocity oscillations, the error states (blue dots in Fig. 10.4(f)) are all inside E.

Discussion

There are existing approaches in the literature [27, 60, 77, 81, 91] which can enable bipedal robot walking on rough terrains. Compared to those, the proposed approach is highly efficient in computation and easy to implement. The continuous controls via BBF-QPs are convex optimizations. The discrete control via H-LIP stepping is in closed form. Additionally, compared to [60, 77, 81], our controller tolerates significantly larger height variations of the terrain. Moreover, the terrain does not need to be exactly known, thus robustness is promoted. The application of this approach on other robots besides the aSLIP model is also promising. Based on the evaluation on AMBER in Chapter 5, the vertical COM can be directly controlled, which should facilitate the realization of this H-LIP based synthesis for walking on rough terrain.

Similar to previous applications, the H-LIP stepping relies on the condition that the aSLIP dynamics is close to that of the H-LIP. This is mainly ensured by three components of the control synthesis. The first is the direct control of the vertical state of the mass, so the vertical trajectory is not distant from that of the H-LIP. The second is the time-based component of the vertical swing foot trajectory, which makes sure that the duration of the SSP does not vary unless the terrain noise δz is too large. The third is the control barrier function in DSP that guides one leg force to cross 0 at an appropriate timing, thus the DSP duration does not significantly vary.

Chapter 11

EXTENSION ⊃ TEMPLATE EMBEDDING ON HUMANOIDS

In previous chapters, we have evaluated the proposed H-LIP based walking synthesis on underactuated bipedal systems. Without doubts, they also can be used for generating walking on fully-actuated humanoid robots, since the fully-actuated humanoids become underactuated by deactivating the ankle actuation. Given ankle actuation, we seek frameworks that methodically use ankle actuation on humanoid robots. Therefore, in this chapter, we present two approaches of applying the H-LIP based stepping controller and the aSLIPs on the humanoid robots. The first approach focuses on the 3D motion decoupling and composition from different planar ROMs, which separately creates periodic motion and non-periodic/transitional motion in the sagittal and lateral plane. The composed 3D ROM motion is then embedded on a humanoid robot. The second approach directly embeds the walking on the 3D-aSLIP model on a humanoid robot. Since point-footed ROMs are embedded on the humanoids, overuses of the ankle actuation can be avoided.

Recall that we model the walking of the humanoid as a hybrid dynamical system with two domains. The description of the hybrid model is similar to that of the robot Cassie. We omit the detailed description here. We also assume that the humanoid has 6 DoFs in each leg. The extra DoFs on the arms are neglected for simplification. Thus there are 18 internal DoFs (including 6 floating-base coordinates) on the humanoid robot.

We first present in Section 11.1 the general output definition for humanoid walking and the contact force embedding within the optimization-based controllers. Then we illustrate the motion decoupling and composition method via ROMs in Section 11.2. The approach of 3D-aSLIP embedding is described in Section 11.3. Finally, we conclude this section in Section 11.4.

11.1 Dynamics Embedding with Contact Forces

We envision that the dynamics embedding of the ROMs on the fully-actuated humanoid is through constraining the COM and feet of the humanoid to follow the behavior of the ROMs. Thus we think about controlling the COM of the humanoid to the desired COM trajectory of the walking of the ROM. The swing foot is controlled to the step location of the ROM as well. Thus we first define the output for the walking of the humanoid.

Output Definition

In the DSP, foot contacts introduce 12 holonomic constraints. There are 6 outputs required for the robot with 18 DoFs. Except for the COM position, we also control the pelvis orientation. Thus the outputs for the DSP are defined as

$$\mathcal{Y}_{\text{DSP}}(q,t) = \begin{bmatrix} x_{\text{COM}}(q) \\ y_{\text{COM}}(q) \\ z_{\text{COM}}(q) \\ \phi_{\text{roll}}(q) \\ \phi_{\text{pitch}}(q) \\ \phi_{\text{yaw}}(q) \end{bmatrix} - \begin{bmatrix} x^{\text{des}}_{\text{COM}}(t) \\ y^{\text{des}}_{\text{COM}}(t) \\ z^{\text{des}}_{\text{COM}}(t) \\ \phi^{\text{des}}_{\text{roll}}(t) \\ \phi^{\text{des}}_{\text{pitch}}(t) \\ \phi^{\text{des}}_{\text{yaw}}(t) \end{bmatrix}, \tag{11.1}$$

where the meaning of individual element is self-explanatory. Similar to the walking on Cassie, the desired roll, pitch, and yaw angles of the pelvis are zeros except that the desired yaw angle changes for the application of turning.

In the SSP, only one foot contacts the ground. Thus we define 12 outputs with additional 6 being on the swing foot position and orientation:

$$\mathcal{Y}_{\text{SSP}}(q,t) = \begin{bmatrix} x_{\text{COM}}(q) \\ y_{\text{COM}}(q) \\ z_{\text{COM}}(q) \\ \phi_{\text{roll}}(q) \\ \phi_{\text{pitch}}(q) \\ \phi_{\text{yaw}}(q) \\ x_{\text{sw}}(q) \\ y_{\text{sw}}(q) \\ z_{\text{sw}}(q) \\ \psi_{\text{roll}}(q) \\ \psi_{\text{pitch}}(q) \\ \psi_{\text{yaw}}(q) \end{bmatrix} - \begin{bmatrix} x^{\text{des}}_{\text{COM}}(t) \\ y^{\text{des}}_{\text{COM}}(t) \\ z^{\text{des}}_{\text{COM}}(t) \\ \phi^{\text{des}}_{\text{roll}}(t) \\ \phi^{\text{des}}_{\text{pitch}}(t) \\ \phi^{\text{des}}_{\text{yaw}}(t) \\ x^{\text{des}}_{\text{sw}}(t) \\ y^{\text{des}}_{\text{sw}}(t) \\ z^{\text{des}}_{\text{sw}}(t) \\ \psi^{\text{des}}_{\text{roll}}(t) \\ \psi^{\text{des}}_{\text{pitch}}(t) \\ \psi^{\text{des}}_{\text{yaw}}(t) \end{bmatrix}, \tag{11.2}$$

where $\psi_{\text{roll/pitch/yaw}}$ is the RPY angle of the swing foot. The desired roll and pitch angles of the swing foot are designed to be zeros. Similar to the pelvis yaw angle, the desired yaw angle of the swing foot is designed to change if turning is required. The swing foot position is constructed smoothly from the initial to the final position of the swing foot in the SSP.

Optimization-based Controller with Contact Force Embedding

Recall in Chapter 2, both the TSC-QP and the CLF-QP can be applied for zeroing the outputs. As for the fully-actuated humanoid, it is possible to regulate the magnitude of the contact forces. With an eye on embedding the walking forces of the aSLIP on the humanoid, we think about constraining the contact forces of the humanoid to be similar to these of the aSLIP.

Recall Eq. (2.2) in Chapter 2 that the ground contact forces are affine functions w.r.t. the motor torques τ:

$$F_h = A_h \tau + b_h, \tag{11.3}$$

where F_h represents the contact force vector. $F_h \in \mathbb{R}^{12}$ in the DSP, and $F_h \in \mathbb{R}^6$ in the SSP. Suppose we would like the realized walking to exhibit the aSLIP walking behavior in terms of ground reaction forces. For instance, if the contact forces behave identically to those of the aSLIP walking, the transitions from the DSP to the SSP can happen naturally as the normal force goes to zero. The transition from the SSP to the DSP also behaves smoothly when the normal force increases as that on the aSLIP. Thus the force control can be realized via enforcing

$$S_h F_h = F_z^{\text{aSLIP}}, \tag{11.4}$$

where S_h is the selection matrix to select the vertical force in the vector; e.g., in the SSP, $S_h = [0, 0, 1, 0, 0, 0]$ and F_h is the reaction force vector with first three being the linear forces in the x, y, z direction. F_z^{aSLIP} is the vertical GRF on the stance leg of the aSLIP during walking. With Eq. (11.3), the equality becomes

$$S_h A_h \tau = F_z^{\text{aSLIP}} - S_h b_h. \tag{11.5}$$

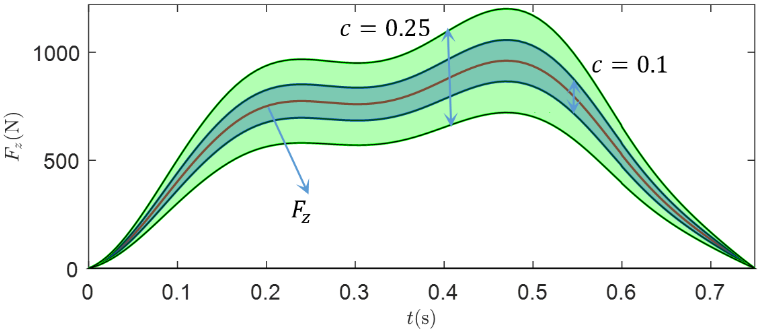

Figure 11.1: Illustration of the relaxation on contact force embedding with different different relaxation coefficient c.

In case when we control the vertical COM oscillation of the humanoid to be identical to that of the aSLIP simultaneously, the vertical GRF embedding needs to be relaxed. Otherwise, strictly matching the GRF can destabilize the tracking. Thus, we relax the equality in Eq.(11.5) by the inequality:

$$\left|S_h F_h - F_z^{\text{aSLIP}}\right| \leq c F_z^{\text{aSLIP}}, \tag{11.6}$$

resulting in

$$\underbrace{(1-c)F_z^{\text{aSLIP}} - S_h b_h}_{c_{lb}} \leq S_h A_h \tau \leq \underbrace{(1+c)F_z^{\text{aSLIP}} - S_h b_h}_{c_{ub}}, \tag{11.7}$$

where $c \in (0, 1)$ is a coefficient of the relaxation. Fig. 11.1 shows the valid range of the F_z with different c for the GRF embedding. It is important to note that this relaxation shrinks as the force decreases, which enforces the smooth transitions between the DSP and the SSP with the vertical GRF decreasing to 0 or increasing from 0.

QP-based Controller: Since the force embedding is realized via an inequality constraint, it is easy to include it in the QP-based controllers in Chapter 2. Here we show an example including the force embedding into the CLF-QP:

$$\begin{aligned}
\tau^* = \underset{\tau\in\mathbb{R}^{12},\ \delta\in\mathbb{R}}{\text{argmin}} \ & \tau^T \tau^T \mathcal{A}^T \mathcal{A} \tau + 2 \mathcal{L}_{\mathbf{f}}{}^T \mathcal{A} \tau + p\delta^2, && (11.8)\\
\text{s.t.} \quad & A^{\text{CLF}}(q, \dot{q})\tau \leq b^{\text{CLF}}(q, \dot{q}) + \delta, && \text{(CLF)}\\
& A^{\text{GRF}}\tau \leq b^{\text{GRF}}, && \text{(GRF)}\\
& \tau_{lb}(\dot{q}) \leq \tau \leq \tau_{ub}(\dot{q}), && \text{(Torque Limit)}\\
& c_{lb} \leq S_h A_h \tau \leq c_{ub}. && \text{(Force Embedding)}
\end{aligned}$$

Solving this QP yields the control input that can not only stablize the output but also perform the contact force embedding to create smooth domain transitions.

11.2 Motion Decoupling and Composition via ROMs

In this section, we present the template dynamics generation via motion decoupling and composition on ROMs. We first describe the walking motion via a sequential composition of periodic walking and transitional walking. Both are further decoupled into walking in the sagittal and lateral planes, each of which is characterized by a ROM. Each ROM is modified from the canonical ROMs to best match the type of motion (Fig. 11.2). Specifically, we use the aSLIP and the H-LIP model for sagittal

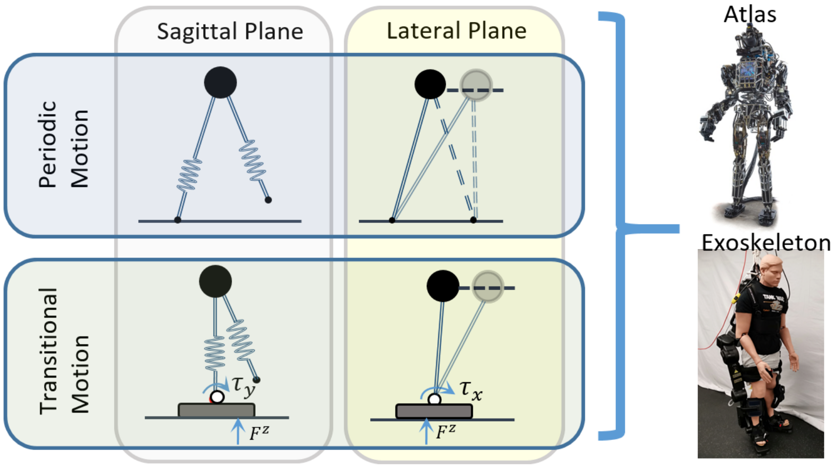

Figure 11.2: Illustration of the Motion Decoupling & Composition approach.

and lateral walking, respectively. Point-footed models of them, i.e., the original models presented in previous chapters, are used for synthesizing periodic walking, and the footed models (with ankle actuation) are applied for generating transitional motion. The 3D coupling between planar models comes from the ground normal forces.

To generate optimal walking behaviors, we formulate the trajectory optimization problems on the aSLIP models for the periodic and transitional motion. The periodic orbits of the point-footed H-LIP can be directly identified. The transitional motion of the footed H-LIP is synthesized via a QP. ZMP constraint and ankle actuation limit are included in the transitional optimizations.

The composition of ROMs naturally combines the benefit of using each ROM. The aSLIP model enables the walking to be dynamic, and the H-LIP model facilitates solvable QP for fast online planning. The optimization of the aSLIP model is also fast to solve due to its low dimensionality. Moreover, the optimization for each transition is only required to be solved once, which potentially can be implemented online with the recent progresses [31] [30] on the online optimization.

Periodic Walking Generation via Point-Footed ROMs

The periodic motion on the aSLIP is first optimized given desired walking behaviors such as velocities and step frequency. Note that the canonical SLIP model can also be used for finding different walking motions [42]. However, the system energy of canonical SLIP is conserved, which is not preferred for generating versatile motions. Thus we use the aSLIP model. After performing the optimization, the durations

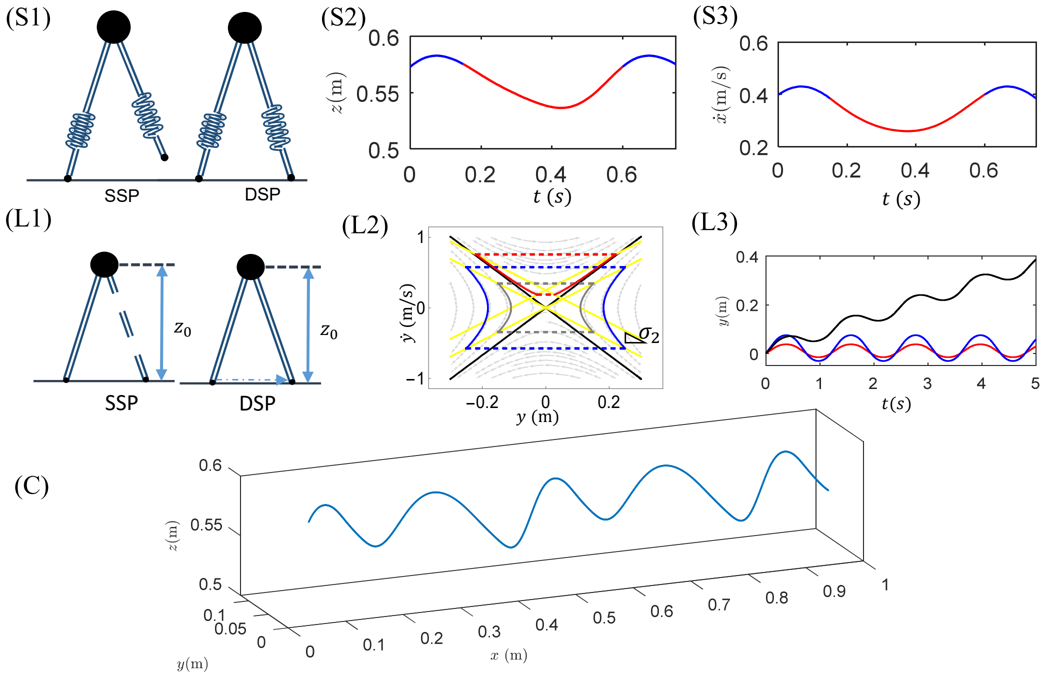

Figure 11.3: 3D composition of planar periodic walking: (S1) the aSLIP model in Sagittal plane, (S2) the vertical trajectory of a periodic walking on the aSLIP, (S3) the forward velocity of the aSLIP; (L1) the H-LIP in the Lateral place, (L2) periodic orbits of the H-LIP, (L3) global positions of the mass in different orbits. (C) An example of the composed trajectory of the point mass in 3D.

of each domain, i.e. T_{SSP} and T_{DSP}, are thus fixed. Then we apply the H-LIP for generating periodic lateral motion given the domain durations. The periodic orbit for the lateral motion can be identified in closed-form as described in Chapter 4. We limit the choice to the P2 orbits to make sure the step width is not smaller than the foot size to avoid kinematic violation on the humanoid. Fig. 11.3 (L2) illustrates different P2 orbits in the phase portrait.

Orbit Composition to 3D: Composition of the mass trajectory of the aSLIP in the sagittal plane and that of the H-LIP in the lateral plane yields the desired COM trajectory in 3D. The point foot nature of the two ROMs locates the center of pressure at the center of the foot, which is expected to provide a large margin of foot stability for embedding the COM on the humanoid. Fig. 11.3 (C) shows an example of the composed trajectory in 3D.

Transitional Motion Generation via Footed Reduced Order Models

Now we describe footed ROMs for fast and dynamic motion transition to enable versatile walking behaviors. The foot actuation is directly applied on the ROMs with the torque limits and the constraint that the center of pressure (COP) lies within the feet.

The transitional behavior is defined as the motion that transits the robot states between standing configurations to periodic motions or between periodic motions themselves. Optimization of the transition motion of the full-dimensional robot dynamics is still a difficult problem to solve due to complex specifications on the dynamics constraints, contact sequences, and actuation limits. The transition optimization on the footed ROMs can simplify the optimization while providing the desired transition COM dynamics. The application of ankle actuation also provides fast and smooth transitions.

We assume that the transition motion is specifically realized within two domains, one DSP and one SSP. In other words, we pre-specify the contact sequence and number of domains for the transition motion. One should note that it is possible to formulate the optimization into a contact implicit optimization using the linear complementarity problem formulation [82, 96] or using potential smooth techniques [132, 133] for the hybrid dynamics.

Footed aSLIP for Transition in Sagittal Plane: The footed aSLIP model is the aSLIP with actuated foot (Fig. 11.4 (S1)). The foot actuation resembles the ankle pitch on the humanoid. The foot length is equal to that of the humanoid. The closed-form dynamics can be found in the Appendix A. The transitional trajectory optimization is similar to the aSLIP optimization described in Chapter 6. The differences are as follows.

<u>Cost Function:</u> With the foot actuation, the cost for the transitional optimization includes both the leg length actuation and the ankle torques:

$$J = \sum_{i=1}^{N} \frac{\Delta t}{2} (\ddot{L}_{\mathrm{L}}^{i2} + \ddot{L}_{\mathrm{R}}^{i2} + \tau_{\mathrm{L}_y}^{i2} + \tau_{\mathrm{R}_y}^{i2}), \tag{11.9}$$

where $\tau_{\mathrm{L}_y}^{i}, \tau_{\mathrm{R}_y}^{i}$ are the ankle pitch torques.

<u>ZMP and Actuation Constraints:</u> To ensure the optimized COM trajectories are re-

alistic, the ZMP and physical actuation constraints must be included:

$$- \mathrm{d}_{\mathrm{heel}} F^i_{\mathrm{L/R}} < \tau^i_{\mathrm{L/R}_y} < \mathrm{d}_{\mathrm{toe}} F^i_{\mathrm{L/R}}, \tag{11.10}$$

$$\left|\tau^i_{\mathrm{L/R}_y}\right| < \tau_{y_{\max}}, \tag{11.11}$$

where $\mathrm{d}_{\mathrm{hee/toel}}$ are the distance between the projected ankle pitch axis to the heel and to the toe, $F^i_{\mathrm{L/R}}$ is the normal reaction force of foot contact, and $\tau_{y_{\max}}$ is the maximum ankle pitch torque of the humanoid. The interpretation of the ZMP constraint is simply that the COP must be in the foot so that the foot does not rotate on the ground.

Initial/Final States: The initial and final states must be identical to the ones in the beginning and at the end of the motion transition. The specification of the states comes from the humanoid. The real leg length r, leg angle β, and their velocities can directly mapped from the COM states of the robot. The leg length state can be solved by force balancing on the point mass with its acceleration. For instance, the states of the aSLIP for the humanoid with static standing on its left foot statically are

$$r_{\mathrm{L}} = z_{\mathrm{COM}}(q), \tag{11.12}$$

$$\beta_{\mathrm{L}} = \tan^{-1}\left(\frac{x_{\mathrm{COM}}(q)}{z_{\mathrm{COM}}(q)}\right), \tag{11.13}$$

$$L_{\mathrm{L}} = \{L \in \mathbb{R} : \mathrm{K}(L)(L - r) = mg\}, \tag{11.14}$$

$$s_{\mathrm{L}} = L_{\mathrm{L}} - r_{\mathrm{L}}, \tag{11.15}$$

where $\mathrm{K}(L)$ is the stiffness function of the aSLIP.

Footed LIP for Transition via Quadratic Programs: The durations of each domain in the transitional motion are determined by the footed aSLIP optimization. For generating the corresponding motion in the lateral plane, we use the footed H-LIP model, which is the H-LIP with foot actuation (Fig. 11.4 (L1)). The ankle actuation on the foot becomes the input to the system. Its continuous dynamics can be written compactly in each domain as

$$\ddot{y}_{\mathrm{SSP}} = \lambda^2 y + \frac{1}{m z_0} \tau_{\mathrm{L/R}_x}, \tag{SSP}$$

$$\ddot{y}_{\mathrm{DSP}} = \frac{1}{m z_0} (\tau_{\mathrm{L}_x} + \tau_{\mathrm{R}_x}), \tag{DSP}$$

where τ_{L_x} and τ_{R_x} are the ankle roll actuations. $\tau_{\mathrm{L/R}_x} = 0$ if the foot is not in contact with the ground. z_0 is the averaged height of the mass on the footed aSLIP in transition motion.

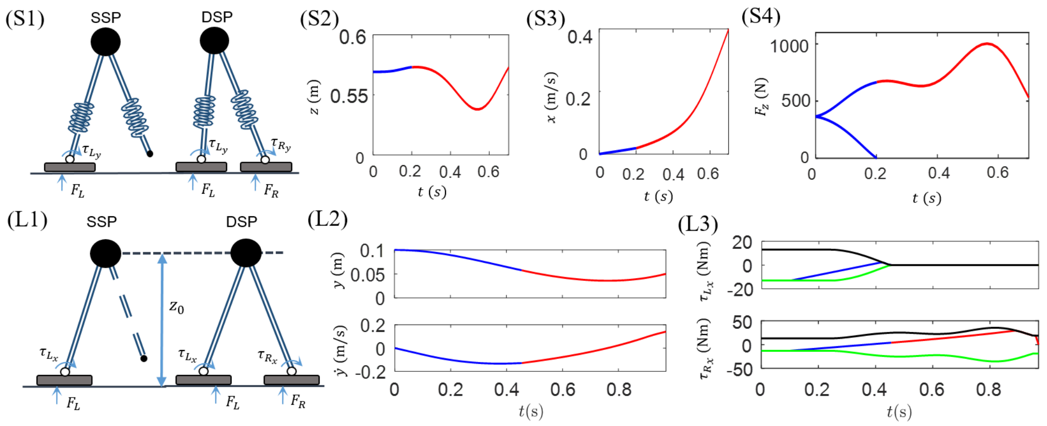

Figure 11.4: Transitional composition from planar footed ROMs: (S1, L1) illustration of the footed aSLIP and footed H-LIP. (S2-4) and (L2-3) are the results of optimizations of the footed ROMs transiting from standing to periodic walking. The blue lines indicate those in DSP, and the red indicates those in SSP. The yellow and black indicate the lower and upper bounds respectively on the ZMP constraint in the H-LIP optimization.

The linear dynamics motivates a quadratic program (QP) formulation for optimizing the transition from an initial state $[y_0; \dot{y}_0]$ to a final state $[y_f; \dot{y}_f]$. We discrete the trajectory over time with the discretized linear dynamics being

$$\underbrace{\begin{bmatrix} y^{k+1} \\ \dot{y}^{k+1} \end{bmatrix}}_{Y^{k+1}} = A^{\text{DSP/SSP}} \underbrace{\begin{bmatrix} y^{k} \\ \dot{y}^{k} \end{bmatrix}}_{Y^{k}} + \underbrace{\begin{bmatrix} \frac{\delta_T^2}{2mz_0} \\ \frac{\delta_T}{mz_0} \end{bmatrix}}_{B} \tau_x^{\text{DSP/SSP}}, \tag{11.16}$$

where δ_T is the time period in the discretization, k is the index in the discretization, and

$$A^{\text{SSP}} = \begin{bmatrix} 1 & \delta_T \\ \lambda\delta_T & 1 \end{bmatrix}, \quad A^{\text{DSP}} = \begin{bmatrix} 1 & \delta_T \\ 0 & 1 \end{bmatrix}, \tag{11.17}$$

$$\tau^{\text{SSP}} = \tau_{\text{L/R}_x}, \quad \tau^{\text{DSP}} = \tau_{\text{L}_x} + \tau_{\text{R}_x}. \tag{11.18}$$

<u>ZMP and Actuation Constraints:</u> With an eye towards the embedding on the humanoid, we also enforce the ZMP and actuation constraints in the lateral transition as

$$-\,\mathrm{d}_1 F^k_{\text{L/R}} \leq \tau^k_{\text{L/R}_x} \leq \mathrm{d}_2 F^k_{\text{L/R}}, \tag{11.19}$$

$$\left|\tau^k_{\text{L/R}_x}\right| < \tau_{x_{\max}}, \tag{11.20}$$

where $\mathrm{d}_1, \mathrm{d}_2$ are the distances between the projected ankle roll axis to each edge of the foot, and $\tau_{x_{\max}}$ is the maximum ankle roll torque.

Coupling with Footed aSLIP: The footed ROMs in each plane are coupled by the ground normal forces. The vertical reaction forces in Eq. (11.19) are from $F_{\mathrm{L/R}}(t)$ of the footed aSLIP, which makes sure the lateral ZMP is satisfied more realistically. One should note that the sum of the vertical reaction forces on the H-LIP, i.e. $F_{\mathrm{L}}(t) + F_{\mathrm{R}}(t)$, does not equal to mg, since the COM height varies from the footed aSLIP.

Quadratic Program: The final QP formulation with minimizing τ for efficiency is as follows:

$$
\begin{align}
\tau_{\mathrm{L}_x}, \tau_{\mathrm{R}_x}, Y = & \underset{\{\tau_{\mathrm{L}_x}, \tau_{\mathrm{R}_x}, Y\} \in \mathbb{R}^{N \times N \times 2N}}{\operatorname{argmin}} \sum\nolimits_{k=1}^{N} \tau_{\mathrm{L}_x}^{k^2} + \tau_{\mathrm{R}_x}^{k^2} \tag{11.21} \\
\text{s.t.} \quad & Y^{k+1} = A^{\mathrm{SSP/DSP}} Y^k + B \tau_x^{\mathrm{SSP/DSP}}, \tag{H-LIP} \\
& \left| \tau_{\mathrm{L/R}_x}^{k} \right| < \tau_{\max}^{x}, \tag{Torque Limit} \\
& - \mathrm{d}_1 F_{\mathrm{L/R}}^{k} \leq \tau_{\mathrm{L/R}_x}^{k} \leq \mathrm{d}_2 F_{\mathrm{L/R}}^{k}, \tag{ZMP} \\
& [y_N; \dot{y}_N] = [y_f; \dot{y}_f], \tag{Final State} \\
& [y_1; \dot{y}_1] = [y_0; \dot{y}_0]. \tag{Initial State}
\end{align}
$$

Comparison with ZMP Approaches: Quadratic programs [64] [20] have been widely used on the LIP dynamics for controlling humanoid walking. In our approach, planning on the ZMP trajectory is not required and the QP is only required to be solved once instead of online recursively.

Remark. *The optimizations on the footed ROMs are connected by the ground reaction force. Therefore the footed aSLIP optimization and the footed H-LIP QP can be combined into a single nonlinear optimization. The procedures are straightforward by combining the variables, constraints, and cost functions.*

Results

With the constructed COM dynamics from ROMs, one can embed the dynamics on the fully actuated humanoid robot by tracking the desired COM trajectory in Section 11.1. Here we evaluate the approach on a lower-body exoskeleton (Exo) [3] in simulation. The implementation steps are summarized in Algorithm 4. The stiffness and damping functions are scaled from Cassie's leg parameter based on the weight ratio of the Exo's over Cassie's. This eliminates the trial-and-error process

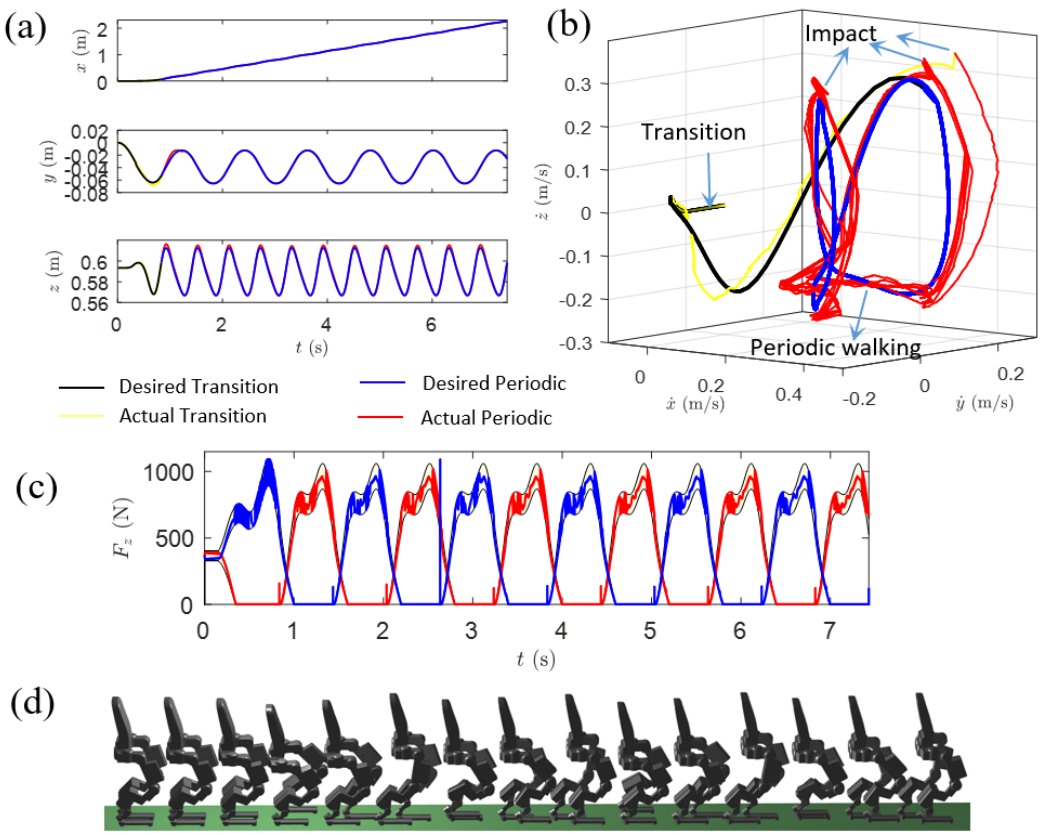

Figure 11.5: Simulation results of the walking from standing. (a) The tracking on the desired COM position. (b) The evolution of COM velocity. (c) Snopshots of the walking of the Exo. (d) Ground normal reaction forces are with in the relaxation under the CLF-QP control.

for identifying appropriate spring parameters. The trajectory optimization problems on the aSLIPs are solved by IPOPT [128]. The QP for the footed H-LIP is solved by qpOASES [38] with active set method. We mainly evaluate the method for two walking scenarios, i.e. periodic walking from standing and transition between two periodic walking behaviors. A video of the simulation results can be seen in `https://youtu.be/L3XW-dRE3E0`.

For both cases, periodic walking is first composed of the periodic orbits of the aSLIP and the H-LIP. Then the nonlinear program on the footed aSLIP and the QP for the footed H-LIP are solved for generating the transition in each plane. The ground normal forces of the aSLIP are used as references in Eq. (11.7). The composed trajectories of the COM are set to the desired outputs for the CLF-QP to generate desired torques in the control loop. Fig. 11.5 shows one of the simulation results.

Algorithm 4 Motion Decoupling and Composition for Humanoids

```
Input: Desired behavior: step length and width, durations
  $x^{\text{des}}_{\text{COM}}(t), z^{\text{des}}_{\text{COM}}(t), F_z(t), T_{\text{SSP}}, T_{\text{DSP}} \leftarrow$ aSLIP periodic optimization
  $y^{\text{des}}_{\text{COM}}(t) \leftarrow$ LIP periodic planning
  aSLIP transition optimization
  LIP transition planning $\leftarrow$ Eq. (11.21)
  while Simulation/Control loop do
    $\mathcal{Y}^{\text{des}}_{\text{SSP/DSP}}(t) \leftarrow$ Eq. (11.1), (11.2)
    Force Embedding $\leftarrow$ Eq. (11.7)
    $\tau \leftarrow$ Eq. (11.8)
  end while
```

The COM of the Exo follows the desired COM trajectory well. The ground normal forces are within the range of the relaxation. Note that the impact of the foot-ground contact still exists in the walking since we do not necessarily require the foot to strike the ground with zero velocity. It does not destabilize the system due to contact force embedding. The simulated walking thus verified the proposed walking generation via motion decoupling and composition.

11.3 3D-aSLIP Walking Embedding on the Humanoid

In this section, we propose a different approach of embedding where the H-LIP based stepping is applied. Recall that in Chapter 6 and 9, the H-LIP based stepping is applied on the 3D-aSLIP using an indirect approach to perturb a stepping-in-place walking into versatile walking behaviors. Therefore, we consider a direct embedding

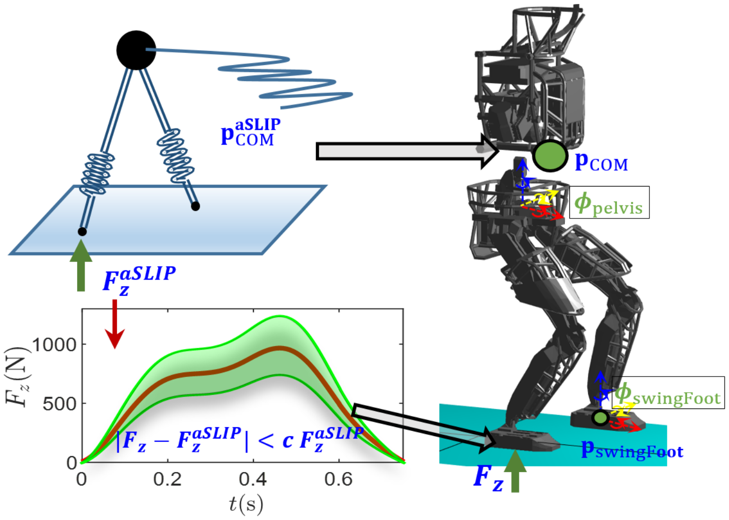
$\mathbf{p}^{\text{aSLIP}}_{\text{COM}}$
$\mathbf{p}_{\text{COM}}$
ϕ_{pelvis}
F_z^{aSLIP}
$F_z(N)$
1000
500
0
$|F_z - F_z^{aSLIP}| < c\, F_z^{aSLIP}$
0
0.2
0.4
0.6
$t(s)$
$\phi_{\text{swingFoot}}$
$\mathbf{p}_{\text{swingFoot}}$
F_z

Figure 11.6: The dynamics embedding scheme on Atlas.

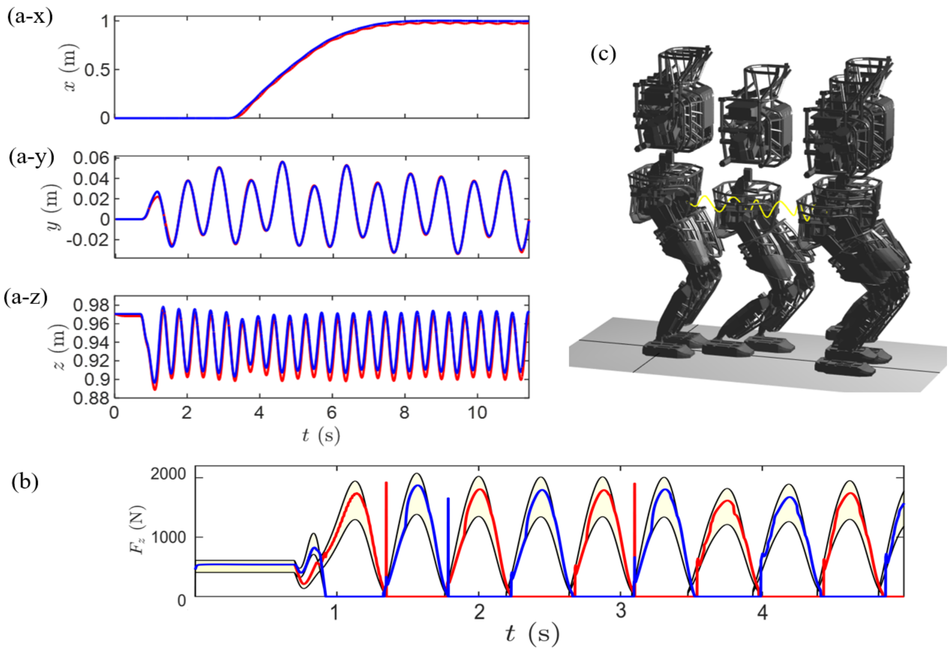

Figure 11.7: Simulation results of the fixed location control: (a-xyz) the tracking of the mass trajectory (red are that of Atlas, and blue are that of the aSLIP), (b) the ground normal force during walking inside the relaxation, and (c) the snapshots of the generate walking.

of the 3D-aSLIP model on the humanoid with certain practical considerations. On the humanoid, the COM height cannot exceed certain ranges. For instance, the maximum height of the COM is fixed and becomes lower during walking. The maximum and minimum step sizes are also bounded. These are specified when generating the walking on the 3D-aSLIP. The advantage of the approach is mainly on the highly efficient walking generation. Trajectory optimization is only required to perform once on the 3D-aSLIP to generate a periodic walking motion. The stepping controller is solved in closed-form for periodic walking or via fast solvable MPC on the H-LIP for GPC. Computation request is minimum at the planning level.

The embedding of the aSLIP walking on the humanoid is the same as the previous section (see Fig. 11.6). We use the robot model of Atlas [74] (without arms) to evaluate this approach. The embedding is realized in simulation. A video of the simulated results can be seen in `https://youtu.be/Xt6XEkxz1_I`.

Fig. 11.7 shows one simulated walking of fixed location control. The force relaxation is selected as $c = 0.2$. Additional walking behaviors including the periodic walking

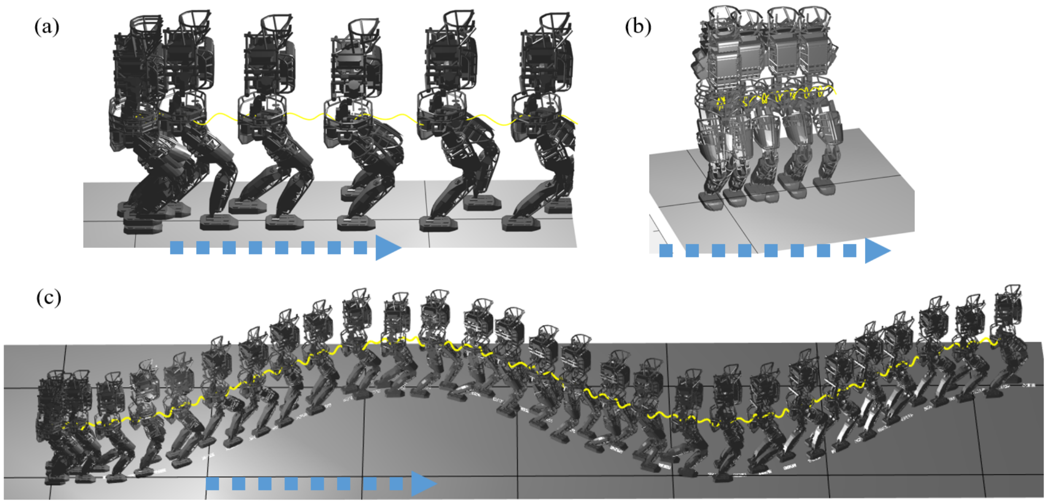

Figure 11.8: Simulation snapshots of the generate walking: (a) forward periodic walking, (b) lateral periodic walking, and (c) global trajectory tracking.

in forward and lateral direction and trajectory tracking are demonstrated in Fig. 11.8. Note that for global trajectory tracking, we also enable turning on the robot by letting the pelvis turn towards the direction of walking, so is the swing foot orientation. The method that enables turning is identical to the method in Section 9.3 of Chapter 9. By driving the outputs to zero via Eq. (11.8), the walking on Atlas can be generated. The simulated versatile walking verifies the proposed walking generation via embedding the aSLIP walking with H-LIP based stepping.

11.4 Comparison

In this chapter, we proposed and evaluated two approaches that embed walking of ROMs on fully-actuated humanoids. The first approach, motion decoupling and composition (MDC) of planar ROMs, leverages the benefits of low-dimensional (planar) models and trajectory optimization to generate and compose desired 3D template COM dynamics for the humanoid. The second approach, 3D-aSLIP embedding, directly embeds on the humanoid the COM of the aSLIP walking, which is efficiently generated via the H-LIP based stepping. Both approaches generate non-constant COM height walking and embed the ground reaction forces of aSLIP walking, which makes the transition between domains to be smooth. The optimization-based controllers in Chapter 2 are applied to further embed the template dynamics with the desired contact forces. Both methods are evaluated on humanoid robots on which different walking motions have been generated successfully.

The two approaches also differ from each other. The MDC still relies on trajectory optimization on the ROMs to generate different periodic walking and transitional walking motion. It is not highly efficient compared to the 3D-aSLIP embedding but can generate walking with different step frequencies. Additionally, the trajectory optimization is not daunting as it is on the planar ROMs. The 3D-aSLIP embedding is highly efficient for generating a variety of walking motions. The generated walking is perturbed from a periodic walking, thus it always has the same frequency of stepping. This can be avoided by generating different periodic walking with different frequencies or changing the step frequency during walking. As the walking is generated via H-LIP based stepping, the generated behavior is not going to be 100% accurate, which might not be desirable for certain applications where accuracy is important. The comparison is summarized as follows with existing representative approaches of template dynamics embedding on humanoids.

Table 11.1: Comparison of Dynamics Embedding Approaches on Humanoids

	MDC	3D-aSLIP	ZMP-LIP	Canonical SLIP
Reference	[140]	[143]	[64]	[77]
COM height variation	✓	✓	X	✓
Versatile behaviors	✓	✓	✓	X
Nonconvex optimization	✓(>1)	✓(1)	X	✓(>1)
Performance accuracy	Medium	Medium	High	Medium
Computational cost	Medium	Low	Low	Medium

Chapter 12

CONCLUSION

In this thesis, we propose a S2S dynamics based approximation via the H-LIP model to plan step sizes for realizing dynamic walking behaviors on bipedal robots with both versatility and robustness. By describing walking as a motion planning and feedback control problem, we innovate a walking synthesis that converts planning of step sizes into a feedback control problem on the S2S dynamics. The walking of the H-LIP is applied to approximate the S2S dynamics of the walking of the robot, and thus feedback step planning can be realized online for generating desired walking behaviors on the robot. The realization is extremely simple in computation. There are no non-convex optimizations to be solved offline or online. The periodic orbits of the H-LIP and its stepping are all in closed-form. After demonstrating the applications and extensions of the approach, we conclude with the contributions, implications, and future directions.

12.1 Contributions

We designed a low-dimensional model (H-LIP) with its comprehensive orbit characterizations to approximate underactuated bipedal walking. We took the underactuated version of the canonical LIP and apply it to *approximate* the underactuated bipedal walking. We provably and geometrically characterized all the P1 and P2 orbits of the H-LIP in its state space and proposed 3D walking via orbit composition.

We showed a versatile walking synthesis with stepping stabilization based on the H-LIP for realizing 3D underactuated bipedal walking. The walking synthesis method directly maps the features of the H-LIP walking to the robotic walking. We also derived stepping stabilization of the robotic walking to the desired walking of the H-LIP; the stepping stabilization is based on the approximation of the S2S dynamics of the robot via the S2S dynamics of the H-LIP.

We realized the walking synthesis on various robotic bipedal walking systems and experimentally implemented it on the physical hardware of the complex 3D underactuated bipedal robot Cassie with passive compliance. Moreover, we showed extensions of the approach to global position control and walking on rough terrain.

12.2 Implications

Gait Library: The orbit characterization of the H-LIP can be viewed as providing an approximated analytical "gait library" for the horizontal COM states of the bipedal robot. The "gait library" of the H-LIP is continuous, i.e., fully occupying the state-space of the horizontal COM. Although the horizontal COM of the robot does not necessarily evolve identically to the orbit, it converges closely to the orbit under the H-LIP based stepping controller. More importantly, transitions between "gaits" or non-periodic walking behaviors can be easily realized via the H-LIP stepping (e.g. the case of tracking a varying desired velocity).

Gait Characterization: The 3D composition of planar orbits offers a way of synthesizing and characterizing 3D bipedal walking gaits. The gait synthesis and characterization via composition of planar orbits can potentially be extended to other multi-legged systems, e.g., the bounding behavior on quadrupedal locomotion [32, 40] can be viewed as producing a P2 orbit in its sagittal plane. The extension appears to be non-trivial but possible.

Model-free Planning: The H-LIP based approach can be viewed as a "model-free" approach, where the robot model is not used in the planning. The walking of the H-LIP is shown to approximate the general hybrid nature of alternating support legs in bipedal walking. The planning on the hybrid dynamics of all the DoFs is encapsulated into the control on the horizontal dynamics of the COM; the individual dynamics of each DoF is not specifically described. As a result, the approach can tolerate the imperfections of the robot modeling in the planning of walking.

Stability of Balancing: The stability of underactuated bipedal walking is typically understood and analyzed on the periodic orbit of the robotic walking [50]. The S2S dynamics formulation provides a different perspective towards understanding the stability of walking. Assuming that the strongly actuated dynamics (the outputs) can be stabilized, the underactuated/weakly-actuated dynamics (the horizontal COM states) are shown to be directly controlled by the step sizes in the S2S dynamics at the step level. Stabilization of the underactuated dynamics can thus be directly synthesized. The stability of the walking is no longer on the periodic orbits but on the discrete horizontal COM states.

Trajectory Tracking: In this thesis, we mainly focused on the planning problem and simply applied optimization-based controllers to solve the trajectory tracking problem. One should note that trajectory tracking (output stabilization) is not a trivial problem in practice, but it is still a traditional feedback control problem. As has been shown before, the key component of balancing in terms of not-falling is on the realization of the desired step sizes during walking. In the presence of large tracking errors, the robot will not fall over as long as the realized step sizes drive the horizontal COM state to its desired set.

12.3 Future Work

We also identify future directions with an eye on addressing some limitations of the proposed approach.

Vertical COM Behavior: The vertical COM height is controlled approximately constant in each step. As we have shown, it permits gradual variations of the COM height between steps. It is not yet known if it is possible to dramatically change the COM height within a step. One possible solution to enable this is to employ a model (e.g. a height-varying pendulum model [22, 23, 72]) that captures both the vertical and horizontal COM behaviors.

Non-unique Trajectory Synthesis: In our approach, the pelvis/upper-body trajectories are fixed, and the swing foot trajectory is designed in the simplest way possible. Both are not optimized in terms of any criteria, e.g., energy consumption. It is possible to apply data-driven approaches to find a low-dimensional representation of the energy consumption in terms of parameterized trajectories of the swing foot or the pelvis. Optimal trajectories can then be constructed on the swing foot and the pelvis.

Performance Improvement: The error state $\mathbf{e}$ directly describes the performance of the stepping controller which drives the robot to a desired walking of the H-LIP. The error is not controlled to zero but in the error invariant set E. There are two ways to further improve the performance in terms of reducing $\mathbf{e}$. The first is to develop a better approximation of the S2S dynamics so that the model difference w is smaller. Different dynamics quantities such as the angular momentum [45] can also be explored for improvement. In our recent work [146], we applied a data-driven approach to learn the S2S dynamics of the robot, which provided better approximations and thus improved the performances on the stepping stabilization. Similar approaches also appeared in [16, 85]. The second is to employ a controller that can directly reduce the error $\mathbf{e}$; e.g., integral control is potentially able to mitigate the error.

Feasibility: The robot joints are designed with limited ranges of motion. Thus, the ranges of the available step sizes are also bounded, which then limits the behaviors (i.e. the walking speeds and orbit compositions) on the robot. Additionally, the legs can internally collide with each other within their ranges of motion. This is more evident on Cassie due to its complex design. The stepping controller presented in this thesis does not systematically take this into consideration. Instead, the kinematic feasibility is reflected on the choices of the desired walking of the H-LIP. In practice, this is sufficient to produce safe (despite conservative) walking on the robot. A more theoretically sound approach involves a systematic identification of the kinematic feasibility. In our recent work [146], we applied the System Level Synthesis (SLS) approach to address this kinematic feasibility under push disturbances. Advanced controllers such as robust MPC [83] can be also explored to include the disturbance and input bounds.

The realized walking behavior is also assumed to be dynamically feasible. In other words, the desired trajectories of the outputs are assumed to be trackable given the limitation of the motor design. In the optimization-based controllers, the torque bounds are included. However, theoretically, it does not guarantee the trajectories (especially the swing foot trajectories) to be well-tracked, e.g., when the walking duration is chosen too small, the motor joints may not be able to move fast enough to drive the swing foot to the desired location. In practice, this can be identified empirically on the hardware despite the loss of theoretical soundness.

Walking over Irregular Terrain: The H-LIP model is assumed to walk on flat terrain and has been extended for walking on non-flat terrain. However, we assumed the step locations are not constrained. In the cases of walking on discontinuous terrains such as the stepping stone problem [28, 90], fast online planning via dynamics approximation is still an open problem.

On Fully-actuated Humanoid Walking: For the walking of a humanoid, the foot is then actuated but with limited controls, which comes from the ankle actuation and ZMP constraint on the support polygon. The foot actuation provides a continuous control input to the horizontal COM state; in comparison, the step sizes are discrete control inputs. The addition of an upper-body with arms also provides continuous control inputs to the horizontal COM state via momentum-based approaches [57, 65, 71]. In Chapter 11, we explored the idea of dynamics embedding via point-footed ROMs. Variations of the integration with step planning and continuous control via ankle actuation and momentum-based approaches can potentially open a large design space of the control synthesis for dynamic humanoid walking in a challenging environment. Future work should explore in this direction.

BIBLIOGRAPHY

[1] Agility Robotics `http://www.agilityrobotics.com`.

[2] Boston Dynamics `https://www.bostondynamics.com`.

[3] Wandercraft `https://www.wandercraft.eu`.

[4] Andrew M Abate. Mechanical design for robot locomotion. 2018.

[5] Mohamadreza Ahmadi, Xiaobin Xiong, and Aaron D Ames. Risk-sensitive path planning via cvar barrier functions: Application to bipedal locomotion. In *Arxiv Preprint: https://arxiv.org/abs/2011.01578*, 2020.

[6] Mojtaba Ahmadi and Martin Buehler. Controlled passive dynamic running experiments with the arl-monopod ii. *IEEE Transactions on Robotics*, 22: 974–986, 2006.

[7] Aaron D Ames. Human-inspired control of bipedal walking robots. *IEEE Transactions on Automatic Control*, 59(5):1115–1130, 2014.

[8] Aaron D Ames and Matthew Powell. Towards the unification of locomotion and manipulation through control lyapunov functions and quadratic programs. In *Control of Cyber-Physical Systems*, pages 219–240. Springer, 2013.

[9] Aaron D Ames, Ramanarayan Vasudevan, and Ruzena Bajcsy. Human-data based cost of bipedal robotic walking. In *Proceedings of the 14th international conference on Hybrid systems: computation and control*, pages 153–162. ACM, 2011.

[10] Aaron D Ames, Kevin Galloway, Koushil Sreenath, and Jessy Grizzle. Rapidly exponentially stabilizing control lyapunov functions and hybrid zero dynamics. *IEEE Transactions on Automatic Control*, 59(4):876–891, 2014.

[11] Aaron D Ames, Xiangru Xu, Jessy W Grizzle, and Paulo Tabuada. Control barrier function based quadratic programs for safety critical systems. *IEEE Trans. on Autom. Control*, 62(8):3861–3876, 2016.

[12] Taylor Apgar, Patrick Clary, Kevin Green, Alan Fern, and Jonathan W Hurst. Fast online trajectory optimization for the bipedal robot cassie. In *Robotics: Science and Systems*, volume 101, page 14, 2018.

[13] Pranav A Bhounsule, Jason Cortell, Anoop Grewal, Bram Hendriksen, JG Daniël Karssen, Chandana Paul, and Andy Ruina. Low-bandwidth reflex-based control for lower power walking: 65 km on a single battery charge. *The International Journal of Robotics Research*, 33(10):1305–1321, 2014.

[14] Pranav A Bhounsule, Andy Ruina, and Gregg Stiesberg. Discrete-decision continuous-actuation control: balance of an inverted pendulum and pumping a pendulum swing. *J. Dyn. Syst. Meas. Control*, 137(5), 2015.

[15] Pranav A Bhounsule, Ezra Ameperosa, Scott Miller, Kyle Seay, and Rico Ulep. Dead-beat control of walking for a torso-actuated rimless wheel using an event-based, discrete, linear controller. In *ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pages V05AT07A042–V05AT07A042. American Society of Mechanical Engineers, 2016.

[16] Pranav A Bhounsule, Myunghee Kim, and Adel Alaeddini. Approximation of the step-to-step dynamics enables computationally efficient and fast optimal control of legged robots. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. American Society of Mechanical Engineers, 2020.

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

[18] Michael Bloesch, Marco Hutter, Mark A Hoepflinger, Stefan Leutenegger, Christian Gehring, C David Remy, and Roland Siegwart. State estimation for legged robots-consistent fusion of leg kinematics and imu. *Robotics: Science and Systems*, 17:17–24, 2013.

[19] Stephen P Boyd and Craig H Barratt. *Linear controller design: limits of performance*. Prentice Hall Englewood Cliffs, NJ, 1991.

[20] Camille Brasseur, Alexander Sherikov, Cyrille Collette, Dimitar Dimitrov, and Pierre-Brice Wieber. A robust linear mpc approach to online generation of 3d biped walking motion. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 595–601. IEEE, 2015.

[21] Eduardo F Camacho and Carlos Bordons Alba. *Model predictive control*. Springer Science & Business Media, 2013.

[22] S. Caron. Biped stabilization by linear feedback of the variable-height inverted pendulum model. In *2020 IEEE Int. Conf. on Rob. and Autom. (ICRA)*, pages 9782–9788, 2020. doi: 10.1109/ICRA40945.2020.9196715.

[23] Stéphane Caron and Abderrahmane Kheddar. Dynamic walking over rough terrains by nonlinear predictive control of the floating-base inverted pendulum. In *2017 IEEE/RSJ Int. Conf. on Intell. Rob. and Sys.*, pages 5017–5024.

[24] Christine Chevallereau, Hamed Razavi, Damien Six, Yannick Aoustin, and Jessy Grizzle. Self-synchronization and self-stabilization of 3d bipedal walking gaits. *Robotics and Autonomous Systems*, 100:43–60, 2018.

[25] Xingye Da and Jessy Grizzle. Combining trajectory optimization, supervised machine learning, and model structure for mitigating the curse of dimensionality in the control of bipedal robots. *Int J Rob Res*, 38(9):1063–1097, 2019.

[26] Behnam Dadashzadeh, Hamid Reza Vejdani, and Jonathan Hurst. From template to anchor: A novel control strategy for spring-mass running of bipedal robots. In *2014 IEEE/RSJ Int. Conf. on Intell. Rob. and Sys.(IROS)*, pages 2566–2571, 2014.

[27] Hongkai Dai and Russ Tedrake. Optimizing robust limit cycles for legged locomotion on unknown terrain. In *2012 IEEE 51st IEEE Conf. on Decision and Control (CDC)*, pages 1207–1213, 2012.

[28] Min Dai, Xiaobin Xiong, and Aaron Ames. Bipedal walking on constrained footholds: Momentum regulation via vertical com control. *arXiv preprint arXiv:2104.10367*, 2021.

[29] Mathew DeDonato, Felipe Polido, Kevin Knoedler, Benzun PW Babu, Nandan Banerjee, Christoper P Bove, Xiongyi Cui, Ruixiang Du, Perry Franklin, Joshua P Graff, et al. Team wpi-cmu: Achieving reliable humanoid behavior in the darpa robotics challenge. *Journal of Field Robotics*, 34(2):381–399, 2017.

[30] Jared Di Carlo, Patrick M Wensing, Benjamin Katz, Gerardo Bledt, and Sangbae Kim. Dynamic locomotion in the mit cheetah 3 through convex model-predictive control. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1–9, 2018.

[31] Yanran Ding, Abhishek Pandala, and Hae-Won Park. Real-time model predictive control for versatile dynamic motions in quadrupedal robots. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8484–8490. IEEE, 2019.

[32] Yanran Ding, Abhishek Pandala, Chuanzheng Li, Young-Ha Shin, and Hae-Won Park. Representation-free model predictive control for dynamic motions in quadrupeds. *arXiv preprint arXiv:2012.10002*, 2020.

[33] Adrien Escande, Nicolas Mansard, and Pierre-Brice Wieber. Hierarchical quadratic programming: Fast online humanoid-robot motion generation. *The International Journal of Robotics Research*, 33(7):1006–1028, 2014.

[34] Salman Faraji, Soha Pouya, and Auke Ijspeert. Robust and agile 3d biped walking with steering capability using a footstep predictive approach. In *Robotics science and systems (RSS)*, 2014.

[35] Siyuan Feng, X Xinjilefu, Weiwei Huang, and Christopher G Atkeson. 3d walking based on online optimization. In *2013 13th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pages 21–27. IEEE, 2013.

[36] Siyuan Feng, X Xinjilefu, Christopher G Atkeson, and Joohyung Kim. Optimization based controller design and implementation for the atlas robot in the darpa robotics challenge finals. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 1028–1035, 2015.

[37] Siyuan Feng, X Xinjilefu, Christopher G Atkeson, and Joohyung Kim. Robust dynamic walking using online foot step optimization. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5373–5378. IEEE, 2016.

[38] H.J. Ferreau, C. Kirches, A. Potschka, H.G. Bock, and M. Diehl. qpOASES: A parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, 6(4):327–363, 2014.

[39] Robert J Full and Daniel E Koditschek. Templates and anchors: neuromechanical hypotheses of legged locomotion on land. *Journal of experimental biology*, 202(23):3325–3332, 1999.

[40] Zhenyu Gan, Ziyuan Jiao, and C David Remy. On the dynamic similarity between bipeds and quadrupeds: A case study on bounding. *IEEE Robotics and Automation Letters*, 3(4):3614–3621, 2018.

[41] Mariano Garcia, Anindya Chatterjee, Andy Ruina, and Michael Coleman. The simplest walking model: stability, complexity, and scaling. *Journal of biomechanical engineering*, 120(2):281–288, 1998.

[42] Gianluca Garofalo, Christian Ott, and Alin Albu-Schäffer. Walking control of fully actuated robots based on the bipedal slip model. *2012 IEEE International Conference on Robotics and Automation*, pages 1456–1463, 2012.

[43] Hartmut Geyer and Uluc Saranli. Gait based on the spring-loaded inverted pendulum. *In: Goswami A., Vadakkepat P. (eds) Humanoid Robotics: A Reference*, 2017.

[44] Hartmut Geyer, Andre Seyfarth, and Reinhard Blickhan. Compliant leg behaviour explains basic dynamics of walking and running. *Proceedings of the Royal Society of London B: Biological Sciences*, 273(1603):2861–2867, 2006.

[45] Yukai Gong and Jessy Grizzle. Angular momentum about the contact point for control of bipedal locomotion: Validation in a lip-based controller. *arXiv preprint arXiv:2008.10763*, 2020.

[46] Yukai Gong, Ross Hartley, Xingye Da, Ayonga Hereid, Omar Harib, Jiunn-Kai Huang, and Jessy Grizzle. Feedback control of a cassie bipedal robot: Walking, standing, and riding a segway. *arXiv preprint arXiv:1809.07279*, 2018.

[47] Ambarish Goswami and Prahlad Vadakkepat. *Humanoid robotics: a reference*. Springer, 2019.

[48] Kevin Green, Ross L Hatton, and Jonathan Hurst. Planning for the unexpected: Explicitly optimizing motions for ground uncertainty in running. *arXiv preprint arXiv:2001.10629*, 2020.

[49] Jesse A Grimes and Jonathan W Hurst. The design of atrias 1.0 a unique monopod, hopping robot. In *Adaptive Mobile Robotics*, pages 548–554. World Scientific, 2012.

[50] Jessy Grizzle, Christine Chevallereau, Ryan W Sinnet, and Aaron D Ames. Models, feedback control, and open problems of 3d bipedal robotic walking. *Automatica*, 50(8):1955–1988, 2014.

[51] Jessy W Grizzle, Jonathan Hurst, Benjamin Morris, Hae-Won Park, and Koushil Sreenath. Mabel, a new robotic bipedal walker and runner. In *2009 American Control Conference*, pages 2030–2036. IEEE, 2009.

[52] Ross Hartley, Maani Ghaffari Jadidi, Jessy W Grizzle, and Ryan M Eustice. Contact-aided invariant extended kalman filtering for legged robot state estimation. *arXiv preprint arXiv:1805.10410*, 2018.

[53] Martin Herceg, Michal Kvasnica, Colin N. Jones, and Manfred Morari. Multi-Parametric Toolbox 3.0. In *Proc. of the European Control Conference*, pages 502–510, Zürich, Switzerland, July 17–19 2013. `http://control.ee.ethz.ch/~mpt`.

[54] Ayonga Hereid and Aaron D Ames. Frost: Fast robot optimization and simulation toolkit. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 719–726. IEEE, 2017.

[55] Ayonga Hereid, Matthew J Powell, and Aaron D Ames. Embedding of slip dynamics on underactuated bipedal robots through multi-objective quadratic program based control. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pages 2950–2957. IEEE, 2014.

[56] Ayonga Hereid, Eric A Cousineau, Christian M Hubicki, and Aaron D Ames. 3d dynamic walking with underactuated humanoid robots: A direct collocation framework for optimizing hybrid zero dynamics. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 1447–1454. IEEE, 2016.

[57] Andreas Hofmann, Marko Popovic, and Hugh Herr. Exploiting angular momentum to enhance bipedal center-of-mass control. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 4423–4429. IEEE, 2009.

[58] Christian Hubicki, Jesse Grimes, Mikhail Jones, Daniel Renjewski, Alexander Spröwitz, Andy Abate, and Jonathan Hurst. Atrias: Design and validation of a tether-free 3d-capable spring-mass bipedal robot. *Int J Rob Res*, 35(12): 1497–1521, 2016.

[59] Christian M Hubicki, Jeff Aguilar, Daniel I Goldman, and Aaron D Ames. Tractable terrain-aware motion planning on granular media: An impulsive jumping study. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 3887–3892. IEEE, 2016.

[60] Fumiya Iida and Russ Tedrake. Minimalistic control of biped walking in rough terrain. *Autonomous Robots*, 28(3):355–368, 2010.

[61] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa. The 3d linear inverted pendulum mode: a simple modeling for a biped walking pattern generation. In *Proceedings 2001 IEEE/RSJ Int. Conf. on Intell. Rob. and Sys. (IROS)*, volume 1, pages 239–246, 2001.

[62] Shuuji Kajita and Kazuo Tani. Study of dynamic biped locomotion on rugged terrain-derivation and application of the linear inverted pendulum mode. In *Proceedings. 1991 IEEE Int. Conf. on Rob. and Autom (ICRA)*, pages 1405–1406, 1991.

[63] Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kazuhito Yokoi, and Hirohisa Hirukawa. A realtime pattern generator for biped walking. In *IEEE Int. Conf. on Rob. and Autom. (ICRA)*, volume 1, pages 31–37, 2002.

[64] Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kensuke Harada, Kazuhito Yokoi, and Hirohisa Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In *Robotics and Automation, IEEE International Conference on*, volume 2, pages 1620–1626, 2003.

[65] Shuuji Kajita, Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kensuke Harada, Kazuhito Yokoi, and Hirohisa Hirukawa. Resolved momentum control: Humanoid motion planning based on the linear and angular momentum. In *Proceedings 2003 IEEE/RSJ Int. Conf. on Intell. Rob. and Sys. (IROS)(Cat. No. 03CH37453)*, volume 2, pages 1644–1650. IEEE, 2003.

[66] Hassan K Khalil. Noninear systems. *Prentice-Hall, New Jersey*, 2(5):5–1, 1996.

[67] P. V. Kokotovic. The joy of feedback: nonlinear and adaptive. *IEEE Control Systems Magazine*, 12(3):7–17, 1992.

[68] Shishir Kolathaya and Aaron D Ames. Parameter to state stability of control lyapunov functions for hybrid system models of robots. *Nonlinear Analysis: Hybrid Systems*, 25:174–191, 2017.

[69] Ilya Kolmanovsky and Elmer G Gilbert. Theory and computation of disturbance invariant sets for discrete-time linear systems. *Mathematical problems in engineering*, 4(4):317–367, 1998.

[70] Twan Koolen, Tomas De Boer, John Rebula, Ambarish Goswami, and Jerry Pratt. Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models. *Int J Rob Res*, 31(9): 1094–1113, 2012.

[71] Twan Koolen, Sylvain Bertrand, Gray Thomas, Tomas De Boer, Tingfan Wu, Jesper Smith, Johannes Englsberger, and Jerry Pratt. Design of a momentum-based control framework and application to the humanoid robot atlas. *International Journal of Humanoid Robotics*, 13(01):1650007, 2016.

[72] Twan Koolen, Michael Posa, and Russ Tedrake. Balance control using center of mass height variation: limitations imposed by unilateral contact. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 8–15. IEEE, 2016.

[73] Manuel Krause, Johannes Englsberger, Pierre-Brice Wieber, and Christian Ott. Stabilization of the capture point dynamics for bipedal walking based on model predictive control. *IFAC Proceedings Volumes*, 45(22):165–171, 2012.

[74] Scott Kuindersma, Robin Deits, Maurice Fallon, Andrés Valenzuela, Hongkai Dai, Frank Permenter, Twan Koolen, Pat Marion, and Russ Tedrake. Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot. *Autonomous robots*, 2016.

[75] Arthur D Kuo. Stabilization of lateral motion in passive dynamic walking. *The Int. J. of Rob. Res.*, 18(9):917–930, 1999.

[76] Yiping Liu, Patrick M Wensing, David E Orin, and Yuan F Zheng. Dynamic walking in a humanoid robot based on a 3d actuated dual-slip model. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5710–5717, 2015.

[77] Yiping Liu, Patrick M Wensing, James P Schmiedeler, and David E Orin. Terrain-blind humanoid walking based on a 3-d actuated dual-slip model. *IEEE Robotics and Automation Letters*, 1(2):1073–1080, 2016.

[78] J. Löfberg. Yalmip : A toolbox for modeling and optimization in matlab. In *In Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[79] Qiuyue Luo, Victor De-León-Gómez, Anne Kalouguine, Christine Chevallereau, and Yannick Aoustin. Self-synchronization and self-stabilization of walking gaits modeled by the three-dimensional lip model. *IEEE Robotics and Automation Letters*, 3(4):3332–3339, 2018.

[80] Kevin M Lynch and Frank C Park. *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, 2017.

[81] Ian R Manchester, Uwe Mettin, Fumiya Iida, and Russ Tedrake. Stable dynamic walking over uneven terrain. *I. J. Rob. Res.*, 30(3):265–279, 2011.

[82] Zachary Manchester and Scott Kuindersma. Variational contact-implicit trajectory optimization. In *Robotics Research*. Springer, 2020.

[83] D.Q. Mayne, M.M. Seron, and S.V. Raković. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2): 219 – 224, 2005. ISSN 0005-1098. doi: https://doi.org/10.1016/j.automatica.2004.08.019.

[84] Igor Mordatch, Martin De Lasa, and Aaron Hertzmann. Robust physics-based locomotion using low-dimensional planning. In *ACM Transactions on Graphics (TOG)*, volume 29, page 71. ACM, 2010.

[85] Jun Morimoto, Christopher G Atkeson, Gen Endo, and Gordon Cheng. Improving humanoid locomotive performance with learnt approximated dynamics via gaussian processes for regression. In *2007 IEEE/RSJ Int. Conf. on Intell. Rob. and Sys. (IROS)*, pages 4234–4240, 2007.

[86] Benjamin Morris, Matthew J Powell, and Aaron D Ames. Sufficient conditions for the lipschitz continuity of qp-based multi-objective control of humanoid robots. In *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, pages 2920–2926. IEEE, 2013.

[87] Benjamin J Morris, Matthew J Powell, and Aaron D Ames. Continuity and smoothness properties of nonlinear optimization-based feedback controllers. In *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on*, pages 151–158. IEEE, 2015.

[88] Richard M Murray, Zexiang Li, S Shankar Sastry, and S Shankara Sastry. A mathematical introduction to robotic manipulation, 1994. *Boca Raton, FL: CRC*, 1994.

[89] Maximilien Naveau, Manuel Kudruss, Olivier Stasse, Christian Kirches, Katja Mombaur, and Philippe Souères. A reactive walking pattern generator based on nonlinear model predictive control. *IEEE Robotics and Automation Letters*, 2(1):10–17, 2016.

[90] Quan Nguyen, Ayush Agrawal, Xingye Da, William C Martin, Hartmut Geyer, Jessy W Grizzle, and Koushil Sreenath. Dynamic walking on randomly-varying discrete terrain with one-step preview. In *Robotics: Science and Systems*, 2017.

[91] Quan Nguyen, Ayush Agrawal, William Martin, Hartmut Geyer, and Koushil Sreenath. Dynamic bipedal locomotion over stochastic discrete terrain. *Int. J. Robot. Res.*, 37(13-14):1537–1553, 2018.

[92] M. Olivares and P. Albertos. On the linear control of underactuated systems: The flywheel inverted pendulum. In *2013 10th IEEE International Conference on Control and Automation (ICCA)*, pages 27–32, 2013. doi: 10.1109/ICCA.2013.6564905.

[93] David E Orin and Ambarish Goswami. Centroidal momentum matrix of a humanoid robot: Structure and properties. In *Intelligent Robots and Systems, 2008. IROS 2008. IEEE/RSJ International Conference on*, pages 653–659. IEEE, 2008.

[94] Giulia Piovan and Katie Byl. Reachability-based control for the active slip model. *The International Journal of Robotics Research*, 34(3):270–287, 2015.

[95] Giulia Piovan and Katie Byl. Approximation and control of the slip model dynamics via partial feedback linearization and two-element leg actuation strategy. *IEEE Trans. on Rob.*, 32(2):399–412, 2016.

[96] Michael Posa, Cecilia Cantu, and Russ Tedrake. A direct method for trajectory optimization of rigid bodies through contact. *The International Journal of Robotics Research*, 33(1):69–81, 2014.

[97] Ioannis Poulakakis and Jessy Grizzle. The spring loaded inverted pendulum as the hybrid zero dynamics of an asymmetric hopper. *IEEE Transactions on Automatic Control*, 54(8):1779–1793, 2009.

[98] Matthew J Powell and Aaron D Ames. Mechanics-based control of underactuated 3d robotic walking: Dynamic gait generation under torque constraints. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 555–560. IEEE, 2016.

[99] Jerry Pratt, John Carff, Sergey Drakunov, and Ambarish Goswami. Capture point: A step toward humanoid push recovery. In *2006 6th IEEE-RAS international conference on humanoid robots*, pages 200–207. IEEE, 2006.

[100] Jerry Pratt, Twan Koolen, Tomas De Boer, John Rebula, Sebastien Cotton, John Carff, Matthew Johnson, and Peter Neuhaus. Capturability-based analysis and control of legged locomotion, part 2: Application to m2v2, a lower-body humanoid. *The Int. J. of Rob. Res.*, 31(10):1117–1133, 2012.

[101] Marc H Raibert. *Legged robots that balance*. MIT press, 1986.

[102] Sasa V Rakovic, Eric C Kerrigan, Konstantinos I Kouramas, and David Q Mayne. Invariant approximations of the minimal robust positively invariant set. *IEEE Transactions on automatic control*, 50(3), 2005.

[103] Hamed Razavi, Anthony M Bloch, Christine Chevallereau, and Jessy W Grizzle. Restricted discrete invariance and self-synchronization for stable walking of bipedal robots. In *2015 Proc Am Control Conf (ACC)*, pages 4818–4824. IEEE, 2015.

[104] Jacob P Reher, Ayonga Hereid, Shishir Kolathaya, Christian M Hubicki, and Aaron D Ames. Algorithmic foundations of realizing multi-contact locomotion on the humanoid robot durus. In *Algorithmic Foundations of Robotics XII*, pages 400–415. Springer, 2020.

[105] Jenna Reher, Claudia Kann, and Aaron D Ames. An inverse dynamics approach to control lyapunov functions. *arXiv preprint arXiv:1910.10824*, 2019.

[106] Siavash Rezazadeh, Christian Hubicki, Mikhail Jones, Andrew Peekema, Johnathan Van Why, Andy Abate, and Jonathan Hurst. Spring-mass walking with atrias in 3d: Robust gait control spanning zero to 4.3 kph on a heavily underactuated bipedal robot. In *ASME 2015 dynamic systems and control Conf.*, 2015.

[107] Siavash Rezazadeh, Andy Abate, Ross L Hatton, and Jonathan W Hurst. Robot leg design: A constructive framework. *IEEE Access*, 6:54369–54387, 2018.

[108] Nelson Rosa and Kevin M. Lynch. Extending equilibria to periodic orbits for walkers using continuation methods. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3661–3667, 2014. doi: 10.1109/IROS.2014.6943076.

[109] Juergen Rummel, Yvonne Blum, H Moritz Maus, Christian Rode, and Andre Seyfarth. Stable and robust walking with compliant legs. In *Robotics and automation (ICRA), 2010 IEEE international conference on*, pages 5250–5255. IEEE, 2010.

[110] Hamid Sadeghian, Luigi Villani, Mehdi Keshmiri, and Bruno Siciliano. Task-space control of robot manipulators with null-space compliance. *IEEE Transactions on Robotics*, 2013.

[111] Yoshiaki Sakagami, Ryujin Watanabe, Chiaki Aoyama, Shinichi Matsunaga, Nobuo Higaki, and Kikuo Fujimura. The intelligent asimo: System overview and integration. In *IEEE/RSJ international conference on intelligent robots and systems*, volume 3, pages 2478–2483. IEEE, 2002.

[112] Uluc Saranlı, Ömür Arslan, M Mert Ankaralı, and Ömer Morgül. Approximate analytic solutions to non-symmetric stance trajectories of the passive spring-loaded inverted pendulum with damping. *Nonlinear Dynamics*, 62(4): 729–742, 2010.

[113] Shankar Sastry. *Nonlinear systems: analysis, stability, and control*, volume 10. Springer Science & Business Media, 2013.

[114] Nicola Scianca, Daniele De Simone, Leonardo Lanari, and Giuseppe Oriolo. Mpc for humanoid gait generation: Stability and feasibility. *arXiv preprint arXiv:1901.08505*, 2019.

[115] Luis Sentis and Oussama Khatib. A whole-body control framework for humanoids operating in human environments. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.*, pages 2641–2648. IEEE, 2006.

[116] Andre Seyfarth, Hartmut Geyer, Michael Günther, and Reinhard Blickhan. A movement criterion for running. *Journal of biomechanics*, 35(5):649–655, 2002.

[117] André Seyfarth, Hartmut Geyer, and Hugh Herr. Swing-leg retraction: a simple control model for stable running. *Journal of Experimental Biology*, 206(15):2547–2555, 2003.

[118] Maziar A Sharbafi and Andre Seyfarth. Vbla, a swing leg control approach for humans and robots. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*, pages 952–957, 2016.

[119] Natan Shemer and Amir Degani. A flight-phase terrain following control strategy for stable and robust hopping of a one-legged robot under large terrain variations. *Bioinspir Biomim*, 12(4):046011, 2017.

[120] Bruno Siciliano and Oussama Khatib. *Springer handbook of robotics*. Springer Science & Business Media, 2008.

[121] Benjamin Stephens. Humanoid push recovery. In *2007 7th IEEE-RAS International Conference on Humanoid Robots*, pages 589–595.

[122] Toru Takenaka, Takashi Matsumoto, Takahide Yoshiike, and Shinya Shirokura. Real time motion generation and control for biped robot-2 nd report: Running gait pattern generation. In *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1092–1099, 2009.

[123] Russ Tedrake. Underactuated robotics: Learning, planning, and control for efficient and agile machines course notes for mit 6.832. *Working draft edition*, 3, 2009.

[124] Russ Tedrake, Scott Kuindersma, Robin Deits, and Kanako Miura. A closed-form solution for real-time zmp gait generation and feedback stabilization. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 936–940.

[125] Pat Terry and Katie Byl. Com control for underactuated 2d hopping robots with series-elastic actuation via higher order partial feedback linearization. In *2015 Proc IEEE Conf Decis Control*, pages 7795–7801, 2015.

[126] Hamid Reza Vejdani, Albert Wu, Hartmut Geyer, and Jonathan W Hurst. Touch-down angle control for spring-mass walking. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015.

[127] Miomir Vukobratović, Branislav Borovac, and Dragoljub Šurdilović. Zero moment point-proper interpretation and new applications. In *Int. Conf. on Humanoid Robots*, pages 237–244, 2001.

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

[129] P. M. Wensing, A. Wang, S. Seok, D. Otten, J. Lang, and S. Kim. Proprioceptive actuator design in the mit cheetah: Impact mitigation and high-bandwidth physical interaction for dynamic legged robots. *IEEE Trans. Robot.*, 33(3): 509–522, 2017. doi: 10.1109/TRO.2016.2640183.

[130] Patrick M Wensing and David E Orin. Generation of dynamic humanoid behaviors through task-space control with conic optimization. In *2013 IEEE International Conference on Robotics and Automation*, pages 3103–3109. IEEE, 2013.

[131] Patrick M Wensing and David E Orin. High-speed humanoid running through control with a 3d-slip model. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 5134–5140, 2013.

[132] Tyler Westenbroek, Xiaobin Xiong, Aaron D Ames, and S Shankar Sastry. Optimal control of piecewise-smooth control systems via singular perturbations. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pages 3046–3053, 2019.

[133] Tyler Westenbroek, Xiaobin Xiong, S Shankar Sastry, and Aaron D Ames. Smooth approximations for hybrid optimal control problems with application to robotic walking. In *IFAC Conference on Analysis and Design of Hybrid Systems (ADHS)*, 2021.

[134] Eric R Westervelt, Jessy W Grizzle, and Daniel E Koditschek. Hybrid zero dynamics of planar biped walkers. *IEEE transactions on automatic control*, 48(1):42–56, 2003.

[135] Eric R Westervelt, Jessy W Grizzle, Christine Chevallereau, Jun Ho Choi, and Benjamin Morris. *Feedback control of dynamic bipedal robot locomotion*, volume 28. CRC press, 2007.

[136] Pierre-Brice Wieber. Trajectory free linear model predictive control for stable walking in the presence of strong perturbations. In *2006 6th IEEE-RAS International Conference on Humanoid Robots*, pages 137–142. IEEE, 2006.

[137] Albert Wu and Hartmut Geyer. The 3-d spring-mass model reveals a time-based deadbeat control for highly robust running and steering in uncertain environments. *IEEE Trans. on Robot.*, 29(5):1114–1124, 2013.

[138] Xiaobin Xiong and Aaron D Ames. Bipedal hopping: Reduced-order model embedding via optimization-based control. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3821–3828. IEEE, 2018.

[139] Xiaobin Xiong and Aaron D Ames. Coupling reduced order models via feedback control for 3d underactuated bipedal robotic walking. In *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, pages 1–9. IEEE, 2018.

[140] Xiaobin Xiong and Aaron D Ames. Motion decoupling and composition via reduced order model optimization for dynamic humanoid walking with clf-qp based active force control. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1018–1024, 2019.

[141] Xiaobin Xiong and Aaron D Ames. Orbit characterization, stabilization and composition on 3d underactuated bipedal walking via hybrid passive linear inverted pendulum model. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4644–4651. IEEE, 2019.

[142] Xiaobin Xiong and Aaron D Ames. 3d underactuated bipedal walking via h-lip based gait synthesis and stepping stabilization. In *Submitted to IEEE Trans. On Robotics*, 2020.

[143] Xiaobin Xiong and Aaron D Ames. Dynamic and versatile humanoid walking via embedding 3d actuated slip model with hybrid lip based stepping. *IEEE Rob. and Automation Letters*, 5(4):6286–6293, 2020.

[144] Xiaobin Xiong and Aaron D. Ames. Sequential motion planning for bipedal somersault via flywheel slip and momentum transmission with task space control. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3510–3517, 2020. doi: 10.1109/IROS45743.2020.9341467.

[145] Xiaobin Xiong, Aaron D Ames, and Daniel I Goldman. A stability region criterion for flat-footed bipedal walking on deformable granular terrain. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4552–4559. IEEE, 2017.

[146] Xiaobin Xiong, Yuxiao Chen, and Aaron D Ames. Robust disturbance rejection for robotic bipedal walking: System-level-synthesis with step-to-step dynamics approximation. In *submitted to IEEE Conference on Decision and Control (CDC)*, 2021.

[147] Xiaobin Xiong, Jenna Reher, and Aaron D Ames. Global position control on underactuated bipedal robots: Step-to-step dynamics approximation for step planning. In *International Conference on Robotics and Automation (ICRA), Preprint: https://arxiv.org/abs/2011.06050*, 2021.

[148] Shishir Nadubettu Yadukumar, Murali Pasupuleti, and Aaron D Ames. Human-inspired underactuated bipedal robotic walking with amber on flat-ground, up-slope and uneven terrain. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2478–2483. IEEE, 2012.

[149] Haitao Yu, Haibo Gao, and Zongquan Deng. Toward a unified approximate analytical representation for spatially running spring-loaded inverted pendulum model. *IEEE Trans. on Rob.*, 2020.

Appendix A

EOM OF ASLIP

In this appendix, we present the closed-form dynamics of the aSLIP model. Generally, the dynamics of the aSLIP with actuated foot during walking in each domain are

$$\text{SSP}:\begin{cases}\ddot{r}_1 = \frac{F_1}{m} - g\cos(\beta_1) + r\dot{\beta}_1^2\\ \ddot{\beta}_1 = \frac{1}{r_1}(-2\dot{\beta}_1\dot{r}_1 + g\sin(\beta_1) + \frac{\tau_1}{mr_1})\\ \ddot{s}_1 = \ddot{L}_1 - \ddot{r}_1\end{cases},$$

$$\text{DSP}:\begin{cases}\ddot{r}_1 = \frac{F_1+F_2\cos(\delta_q)}{m} - g\cos(\beta_1) + r_1\dot{\beta}_1^2 + \frac{\tau_2}{mr_2}\sin(\delta_q)\\ \ddot{\beta}_1 = \frac{-2\dot{\beta}_1\dot{r}_1+g\sin(\beta_1)-\frac{F_2}{m}\sin(\delta_q)}{r_1} + \frac{\cos(\delta_q)}{mr_1r_2}\tau_2 + \frac{\tau_1}{mr_1^2}\\ \ddot{r}_2 = \frac{F_2+F_1\cos(\delta_q)}{m} - g\cos(\beta_2) + r_2\dot{\beta}_2^2 - \frac{\tau_1}{mr_1}\sin(\delta_q)\\ \ddot{\beta}_2 = \frac{-2\dot{\beta}_2\dot{r}_2+g\sin(\beta_2)+\frac{F_1}{m}\sin(\delta_q)}{r_2} + \frac{\cos(\delta_q)}{mr_1r_2}\tau_1 + \frac{\tau_2}{mr_2^2}\\ \ddot{s}_1 = \ddot{L}_1 - \ddot{r}_1\\ \ddot{s}_2 = \ddot{L}_2 - \ddot{r}_2\end{cases},$$

where $\delta_q = \beta_1 - \beta_2$, $\ddot{L}_1, \ddot{L}_2$ are assumed to be the virtual input [139], and τ_1, τ_2 are the ankle actuation. The spring forces F_1, F_2 come from the leg spring [138]. Note that here we use the subscripts $_{1,2}$ to denote the legs. The dynamics of point-footed aSLIP are these of the footed aSLIP with zero ankle actuation ($\tau_1 = 0, \tau_2 = 0$).

Impact Map Assumption: We assume there is an impact event happening when the swing leg strikes the ground. Since the leg is massless, the velocity on the mass is assumed to be continuous. The velocity of the swing foot becomes zero at touchdown. Let 2 index the swing leg. $\dot{r}_2$ and $\dot{\beta}_2$ are discontinuous at impact. Additionally, the holonomic constraint on the swing leg enforces $\dot{L}_2^+ = \dot{s}_2^+ + \dot{r}_2^+$. It is assumed that the leg length velocity is continuous, i.e. $\dot{L}_2^+ = \dot{L}_2^-$. This matches with the intuition that, when the leg is rigidly controlled, the impact instantaneously acts on the compliant spring. The impact map is

$$\Delta_{\text{SSP}\to\text{DSP}}:\begin{cases}\dot{q}_2^+ = \frac{1}{r_2}(\dot{q}_1\dot{r}_1\cos(\delta_q) + \dot{r}_1\sin(\delta_q))\\ \dot{r}_2^+ = \dot{r}_1\cos(\delta_q) - \dot{q}_1r_1\sin(\delta_q)\\ \dot{s}_2^+ = \dot{L}_2^- - \dot{r}_2^+\end{cases}$$

where we only specify the discontinuous states to save space. The transition map $\Delta_{\text{DSP}\to\text{SSP}}$ is smooth.

Appendix B

LEG SPRING ON CASSIE

In this appendix, we derive the leg spring model from the kinematics of the leg, which yields the aSLIP model [138, 139] in Chapter 6. Since the leg is designed with low inertia and mass, it is rational to approximate the full leg dynamics on the upper body through a much lower dimensional model. By inspection on the compliant components in the leg, we propose the approximation via a leg spring model, which is a telescopic leg with a prismatic spring in serial connection. It is expected on the robot that the stiffness of the leg spring changes with different robot configurations. For instance, the leg is presumably more compliant when the leg is in a configuration with a shorter length. Therefore, we explicitly derive the leg stiffness and damping as functions of leg configurations, i.e., joint angles. The analogy of the leg stiffness/damping is the end-effector stiffness/damping for robotic manipulators [120], thinking about fixing the pelvis and treat the toe and the end-effector. Then we approximate the axial component of the leg stiffness and damping by polynomial functions of leg length L, which finally provide the leg spring model.

Leg Stiffness and Damping: The leg stiffness $\mathrm{K}_{\mathrm{leg}}$ is the resistance of the leg to external forces. The complementary concept is called leg compliance $\mathrm{C}_{\mathrm{leg}} = \mathrm{K}_{\mathrm{leg}}^{-1}$. When the leg is under external load at the foot, the leg deforms due to compliant elements in the leg. Assuming that we only consider the transitional deformations, the external force can be calculated by

$$F_{\mathrm{ext}} = \mathrm{K}_{\mathrm{leg}}\delta, \tag{B.1}$$

where $F_{\mathrm{ext}} \in \mathbb{R}^3$, $\mathrm{K}_{\mathrm{leg}} \in \mathbb{R}^{3\times3}$ and $\delta \in \mathbb{R}^3$. Under the assumption that the deformation is small and only happens at the joints, the leg deformation δ can be mapped from joint deformations Δq by the foot Jacobian as $\delta = J\Delta q$, where $J \in \mathbb{R}^{3\times n}$ and $\Delta q \in \mathbb{R}^n$. Let τ denotes the moments at the joints caused by the external load, thus $\tau = J^T F_{\mathrm{ext}}$. If the stiffness at each joint is k_i with $i = 1, ..., n$, then the joint stiffness matrix is defined as $\mathrm{K}_J = \mathrm{diag}(\mathrm{k}_1, ..., \mathrm{k}_n)$, and $\tau = \mathrm{K}_J\Delta q$. The joint stiffness matrix K_J and leg stiffness $\mathrm{K}_{\mathrm{leg}}$ are hence related by the joint moments:

$$\mathrm{K}_J\Delta q = \tau = J^T F_{\mathrm{ext}} = J^T\mathrm{K}_{\mathrm{leg}}\delta = J^T\mathrm{K}_{\mathrm{leg}}J\Delta q. \tag{B.2}$$

Then the leg stiffness can be calculated from the joint stiffness matrix by

$$\mathrm{K}_{\mathrm{leg}}(q) = (J(q)\mathrm{K}_J^{-1}J(q)^T)^{-1}. \tag{B.3}$$

This indicates the leg stiffness is a function of the Jacobian and thus a function of the configuration q. The leg damping $\mathrm{D}_{\mathrm{leg}}(q)$ is derived in the same way by dropping the assumption on small deformation at joints, which yields

$$\mathrm{D}_{\mathrm{leg}}(q) = (J(q)\mathrm{D}_J^{-1}J(q)^T)^{-1}. \tag{B.4}$$

We now apply the calculation of leg stiffness and damping on Cassie. Note that the main difference is the closed kinematic chain inside the leg with pure passive joints and compliant joints. The pure passive joints have no contribution towards leg stiffness, so we need to derive the forward kinematics from active joints, i.e. the spring joints and motor joints. As there are two chains towards the toe, the velocity of the toe relative to the hip can be calculated as

$$v_{\mathrm{Toe}\leftarrow\mathrm{Hip}} = J_1(q_1)\dot{q}_1 = J_2(q_2)\dot{q}_2, \tag{B.5}$$

where $q_1 = [q_{\mathrm{hip}}; q_{\mathrm{knee}}; q_{\mathrm{shin}}; q_{\mathrm{tarsus}}; q_{\mathrm{toe}}]$ and $q_2 = [q_{\mathrm{r}_2}; q_{\mathrm{r}_1}; q_{\mathrm{heel}}; q_{\mathrm{toe}}]$. Eq. (B.5) can be rewritten as

$$\underbrace{\begin{bmatrix} J_1(q_1) & -J_2(q_2) \end{bmatrix}}_{E} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} = 0. \tag{B.6}$$

Then we can rearrange the matrix E and group q_1, q_2 into active joints $q_A = \{q_{\mathrm{spring}}, q_{\mathrm{motor}}\}$ and pure passive joints q_P. Eq. (B.5) becomes

$$\begin{bmatrix} E_A(q) & E_P(q) \end{bmatrix} \begin{bmatrix} \dot{q}_A \\ \dot{q}_P \end{bmatrix} = 0. \tag{B.7}$$

Then the passive joint velocity $\dot{q}_P = -E_P^{-1}E_A\dot{q}_A$. As there is only one passive joint, the tarsus, on the main kinematic chain, we can find $\dot{q}_{\mathrm{tarsus}} = -E_P^{-1}E_A\dot{q}_A$, where $q_A = \{q_{\mathrm{knee}}, q_{\mathrm{shin}}, q_{\mathrm{heel}}, q_{\mathrm{toe}}\}$. Then the forward kinematics and the leg stiffness can be expressed in terms of q_A:

$$\begin{aligned} v_{\mathrm{Toe}\leftarrow\mathrm{Hip}} &= J_A(q_A)\dot{q}_A, && \text{(B.8)} \\ \mathrm{K}_{\mathrm{leg}}(q_A) &= (J_A(q_A)\mathrm{K}_A^{-1}J_A(q_A)^T)^{-1}, && \text{(B.9)} \end{aligned}$$

where $\mathrm{K}_A = \mathrm{diag}(\infty, \mathrm{k}_{\mathrm{shin}}, \mathrm{k}_{\mathrm{heel}}, \infty)$ as we assume the motor joints being rigidly controlled to fixed positions. Assuming the spring joints have small deflections under

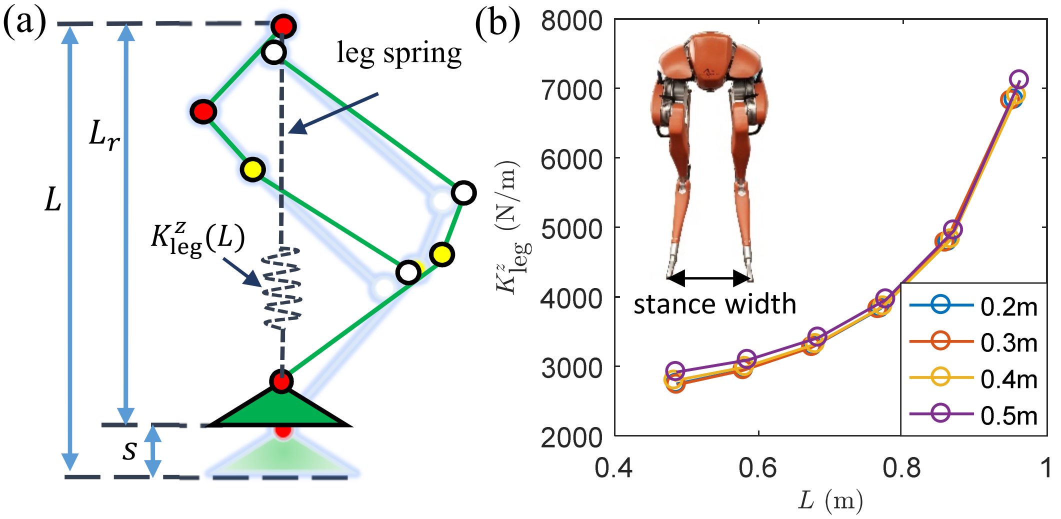

Figure B.1: (a) Illustration of the leg spring from the compliant joints. (b) The vertical leg stiffness v.s virtual leg length for different stance width.

normal load, $\text{K}_{\text{leg}}(q_A)$ can be approximated as $\text{K}_{\text{leg}}(q_{\text{knee}}, q_{\text{shin}} = 0, q_{\text{heel}} = 0, q_{\text{toe}})$. q_{toe} has trivial contribution in terms of J_A and K_{leg}. Thus $\text{K}_{\text{leg}}(q_A) \approx \text{K}_{\text{leg}}(q_{\text{knee}})$.

This naturally inspires a definition of **virtual leg length** (uncompressed leg length) $L(q_{\text{knee}})$ to approximate $\text{K}_{\text{leg}}(q_{\text{knee}})$ by $\text{K}_{\text{leg}}(L)$. $L_r(q_{\text{knee}}, q_{\text{shin}}, q_{\text{tarsus}}, q_{\text{heel}})$, i.e. the compressed leg length, is defined as the distance between the hip pitch joint and the toe joint, whereas the virtual leg length is the real leg length with zero spring deflections as being illustrated in Fig. B.1 (a); see Appendix C for details. Due to Cassie's specific leg design, the compliance mainly appears in the direction of the leg. As we are interested in the axial behavior, the last element in K_{leg}, denoted by K^z_{leg}, is taken as the stiffness of the leg. Fig. B.1 (b) shows how K^z_{leg} changes with L at different static stance configurations. We apply a polynomial regression in the form of

$$\text{K}^z_{\text{leg}}(L) = a_0 + a_1 L + a_2 L^2 + a_4 L^4 \tag{B.10}$$

to approximate the function $\text{K}^z_{\text{leg}}(L)$. The leg damping is approximated in the same way. The coefficients of the regressed functions can be found in Table B.1.

Leg Spring: Given the leg stiffness and damping calculation, we have a nonlinear spring in the leg. More importantly, the leg spring model is actuated. The spring also changes with the change of the virtual leg length. Since the internal dynamics of the leg is ignored, we view the actuation of the leg length at the input to the leg

Table B.1: Coefficients of the Parametric Regression

	a_0	a_1	a_2	a_4
On $\mathrm{K}^z_{\mathrm{leg}}$	-9451	48657	-55230	23309
On $\mathrm{D}^z_{\mathrm{leg}}$	-141	726	-824	348

spring model and thus also the input to the aSLIP in Chapter 6 with the same leg spring model. On the robot, one can approximately view the leg as a serial-elastic actuator, where the springs compose the elastic element at the output of the leg, and the actuation is to change the uncompressed/virtual leg length L.

Appendix C

UNCOMPRESSED OUTPUT ON CASSIE

The compliance springs create challenges to preciously control the vertical COM and swing foot positions. If the output contains the compliant DoFs, the spring can create undesired resonance, which then destabilizes the output, especially in the vertical direction. Note that the DoFs of the spring joints are also underactuated. Therefore, accommodations have to be made for the compliance in the output design. In Chapter 7 and 8, the rigid representations of the vertical COM, the leg length, the swing foot positions are used. Here, we define these outputs rigorously.

By definition, the vertical position of the COM w.r.t. the stance foot is a function of $\{\phi_{\text{roll}}, \phi_{\text{pitch}}, \phi_{\text{yaw}}, q_{\text{motor}}, q_{\text{tarsus}}, q_{\text{spring}}\}$. The COM height with uncompressed springs is defined as

$$\tilde{z}_{\text{COM}} = z_{\text{COM}}(\phi_{\text{roll}}, \phi_{\text{pitch}}, \phi_{\text{yaw}}, q_{\text{motor}}, q_{\text{tarsus}} \to q^{\text{rigid}}_{\text{tarsus}}, q_{\text{spring}} \to 0).$$

$q^{\text{rigid}}_{\text{tarsus}}$ is the uncompressed tarsus angle under the holonomic constraint of the push-rod:

$$q^{\text{rigid}}_{\text{tarsus}}(q_{\text{knee}}) = \texttt{Root}(h_{\text{rod}}(q_{\text{knee}}, q_{\text{shin}} \to 0, q_{\text{heel}} \to 0, q_{\text{tarsus}}) = 0),$$

which is solved via Newton-Raphson method. Thus z_{COM} in Eq. (7.9) is approximated by $\tilde{z}_{\text{COM}}$. Similarly, the position of the swing foot w.r.t. the stance foot are approximated in the same way by $\{\tilde{x}, \tilde{y}, \tilde{z}\}_{\text{sw}}$. Since the springs on the stance leg are expected to oscillate less, we only set the springs on the swing legs to 0 in $\{\tilde{x}, \tilde{y}, \tilde{z}\}_{\text{sw}}$ for better approximations. For the leg length of each leg, it is a function of $\{q_{\text{knee}}, q_{\text{shin}}, q_{\text{tarsus}}, q_{\text{heel}}\}$. The rigid representation of the leg length is thus

$$L(q_{\text{knee}}) = L_r(q_{\text{knee}}, q_{\text{shin}} \to 0, q_{\text{tarsus}} \to q^{\text{rigid}}_{\text{tarsus}}, q_{\text{heel}} \to 0).$$

Appendix D

BÉZIER POLYNOMIALS

The Bézier polynomials are used to design the desired output trajectories. The Bézier polynomials are defined as

$$b(\bar{t}) := \sum_{k=0}^{M} \alpha_k \frac{M!}{k!(M-k)!} \bar{t}^k (1-\bar{t})^{M-k}, \tag{D.1}$$

where $\bar{t} \in [0, 1]$ and α_k are the coefficients of the Bézier polynomial. M is the degree. In the application of walking, we define $\bar{t} = \frac{t}{T}$ with T being the duration of a walking cycle, and t is the time, which is reset to 0 when $t = T$. The $\mathcal{B}_h$ used in the thesis has coefficients of $[0, 0, 1, 1, 1]$.