2021 IEEE International Conference on Robotics and Automation (ICRA) | 978-1-7281-9077-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICRA48506.2021.9560906

Inverse Dynamics Control of Compliant Hybrid Zero Dynamic Walking

Jenna Reher and Aaron D. Ames[1]

Abstract— **We present a trajectory planning and control architecture for bipedal locomotion at a variety of speeds on a highly underactuated and compliant bipedal robot. A library of compliant walking trajectories are planned offline, and stored as compact arrays of polynomial coefficients for tracking online. The control implementation uses a floating-base inverse dynamics controller which generates dynamically consistent feedforward torques to realize walking using information obtained from the trajectory optimization. The effectiveness of the controller is demonstrated in simulation and on hardware for walking both indoors on flat terrain and over unplanned disturbances outdoors. Additionally, both the controller and optimization source code are made available on GitHub.**

I. INTRODUCTION

The inclusion of model detail in walking controllers for underactuated bipedal locomotion can be used to accurately capture the underactuated dynamics of the system. However, implementing model-based planning and control for locomotion is non-trivial due to inherent model inaccuracies, dynamically changing constraints, and possibly conflicting objectives for the robot which can arise. It is due to these challenges that bipedal robots which exhibit simultaneously robust, efficient, and agile motions are rare in practice [21].

A significant subset of the bipedal robotics literature mitigates the complexity of humanoids and bipeds by viewing walking as a problem wherein the real world dynamics are assumed to be governed by the evolution of a simpler system, such as a LIP models (Linear Inverted Pendulum [8], [15]), SLIP models (Spring Loaded Inverted Pendulum [25]), and the ZMP (Zero Moment Point [27]). These methods can reduce computational complexity for fast planning and experimental success. Despite its viability in practical implementation, this local representation of the system can limit the agility of behaviors and compromise energy efficiency.

One area of planning and control which is particularly difficult to directly address is passive compliance. Some of the earliest inclusions of compliant hardware on bipedal robots was with spring flamingo and spring turkey [13], and more recently MABEL [18] and ATRIAS [25]. One of the latest robots in this category is the Cassie biped (shown in Fig. 1). From a mathematical standpoint, compliance increases the degree of underactuation which makes finding stable walking behaviors more difficult. Compliance can also increase numerical stiffness and model uncertainty. We attempt to address these challenges through two contributions.

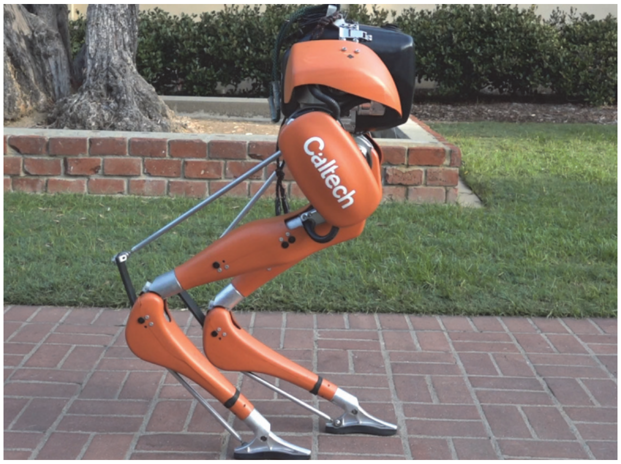

Fig. 1. The bipedal robot Cassie at Caltech, used as the experimental platform for demonstration of the methods presented in this work.

The first contribution of this paper is a motion library of walking behaviors for Cassie that leverage its full-body dynamics including compliance. To generate this library, we utilize the framework of Hybrid Zero Dynamics (HZD) [28] which has demonstrated success in developing controllers for underactuated walking behaviors. HZD has enabled a wide variety of dynamic behaviors such as underactuated humanoid walking [22], compliant running [26], and 3D bipedal walking with point-feet [20]. Directly related to this work, motion libraries on sagittal motions under the assumption of sufficient rigidity has been successfully realized on Cassie [9], but this work ignored the robot's inherent compliance. Tangentially, singular walking behaviors which consider passive compliance were realized in [24]. This work, therefore, is the first to combine motion libraries that consider the full-body compliant dynamics of Cassie.

The second contribution is the demonstration of a model-based controller on hardware that can exploit the HZD library to generate accurate feedforward torques. One of the limiting assumptions affecting HZD behaviors is that their formal stability guarantees are tied to exponential tracking of outputs. This can lead to the use of high-gain PD feedback controllers when applied to hardware. Instead, it would be more desirable to achieve tracking with some inherent control compliance to disturbances. In this direction, model-based control in the form of inverse dynamics [3], [16], control Lyapunov functions [4], or some combination of the two [23] may offer more desirable control properties. In this work, we consider an analytical solution to the floating-base inverse dynamics problem, following the work in [17]. The end result is an experimental demonstration on Cassie, with the result being stable and robust walking.

*This research is supported under NSF Grant Numbers 1544332, 1724457, 1724464 and Disney Research LA.

[1] The authors are with the Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125 {jreher,ames}@caltech.edu

The paper is structured as follows: Section II gives an overview of the Cassie robotic model and the hybrid domain structure which is prescribed for walking. Section III details the optimization problem which is used to produce a library of compliant walking trajectories. Section IV introduces a orthogonal decomposition based inverse dynamics controller, which uses the optimized accelerations and robot model to obtain a dynamically consistent feedforward term for the planned walking gaits. Walking on hardware and in simulation is then demonstrated, validating that the planned trajectories capture the passive dynamics of the robot.

II. Robotic Model

The Cassie biped is an approximately one meter tall walking robot designed and manufactured by Agility Robotics. The sensing on the robot is entirely proprioceptive and includes a 9-DOF IMU, torque sensing, and high-resolution absolute encoders to measure both compliance and actuated joints. The physical model of Cassie is complex, with 22 degrees of freedom, 4 of which are fiberglass leaf springs. There are 10 BLDC motors which control the joints through 8 low-friction cycloidal gearboxes and 2 harmonic gearboxes at the feet. The compliance in the model is contained to a 4-bar linkage, through which the knee motors effectively drive the leg length. While the feet of the robot are actuated, and offer some small control authority in the sagittal direction, contact is a reduced to a point in the coronal plane.

A. Floating Base Model

The robot itself is modeled as a tree structure composed of rigid links. As legged locomotion inherently involves intermittent sequences of rigid contacts with the environment, it is common practice to construct a floating-base Lagrangian model of the system and then add constraint forces via D'Alembert's principle [10]. We define the configuration space of the bipedal robot Cassie as $\mathcal{Q} \subset \mathbb{R}^n$, where n is the DOF (degrees of freedom) without considering any holonomic constraints that result in DOF reduction. A visualization of the prescribed coordinates is shown in Fig. 2, and an URDF of the robotic model detailed in this work is provided online as a part of a software package [1]. Let

$$q = (p_b, \phi_b, q_l) \in \mathcal{Q} := \mathbb{R}^3 \times SO(3) \times \mathcal{Q}_l,$$

where p_b is the global Cartesian position of the body fixed frame attached to the base linkage (the pelvis), ϕ_b is its global orientation, and $q_l \in \mathcal{Q}_l \subset \mathbb{R}^{n_l}$ are the local coordinates representing the joint angles between links. Further, the state space $\mathcal{X} = T\mathcal{Q} \subset \mathbb{R}^{2n}$ has coordinates $x = (q^\mathsf{T}, \dot{q}^\mathsf{T})^\mathsf{T}$. The local coordinates are denoted as $q_l^\mathsf{T} = \left(q^{\mathrm{L}}, q^{\mathrm{R}}\right)$, where the superscript L/R denotes the left/right leg, and

$$q^{i \in \{\mathrm{L},\mathrm{R}\}} = \left(\theta^i_{\mathrm{hr}}, \theta^i_{\mathrm{hp}}, \theta^i_{\mathrm{hy}}, \theta^i_{\mathrm{k}}, \theta^i_{\mathrm{s}}, \theta^i_{\mathrm{t}}, \theta^i_{\mathrm{hs}}, \theta^i_{\mathrm{a}}\right),$$

represents the joint of *hip roll, hip pitch, hip yaw, knee, shin, tarsus, heel spring and ankle* accordingly. Among these joints, $\theta_{\mathrm{hr}}, \theta_{\mathrm{hp}}, \theta_{\mathrm{hy}}, \theta_{\mathrm{k}}, \theta_{\mathrm{a}}$ are actuated by BLDC motors with associated control inputs $u \in \mathcal{U} \subset \mathbb{R}^m$ with $m = 10$ as previously described. The joints $\theta_{\mathrm{hs}}, \theta_{\mathrm{s}}$ are directly driven by leaf springs, which are treated as rigid links with rotational springs of stiffness 2300 and 2000 Nm/rad respectively at the pivot. There is one completely passive joint θ_{t} per leg, and an unconstrained model of the robot with 22 DOF.

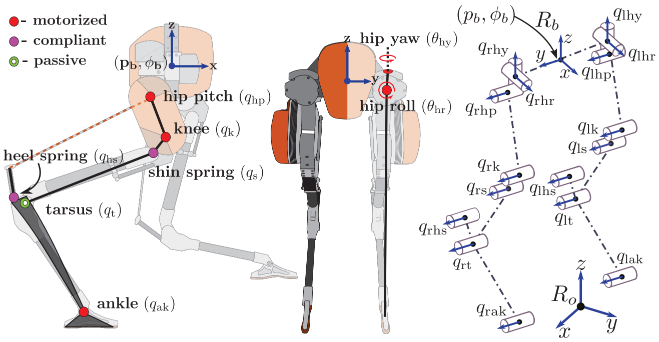

Fig. 2. The configuration coordinates of the Cassie robot: with the side and front views highlighting the compliant mechanism and the general morphology of the robot.

The Lagrangian floating-base model can be expressed as:

$$D(q)\ddot{q} + H(q, \dot{q}) = Bu + J(q)^\mathsf{T}\lambda + B_s\tau_s \qquad (1)$$

where $D(q)$ is the inertia matrix, $h(q, \dot{q})$ contains the Coriolis and gravity terms, B is the actuation matrix with gear reduction as its entries, and the Jacobian matrix of the holonomic constraints is $J(q) = \partial \frac{\eta}{\partial q}(q)$ with its corresponding constraint wrenches $\lambda \in \mathbb{R}^{m_\eta}$. Note that we introduced the spring forces as external forces $\tau_s = k_s q + k_b \dot{q}$ with k_s, k_b the stiffness and damping coefficients, which are distributed according to the associated spring selection matrix B_s.

B. Holonomic Constraints

Two types of holonomic constraints are commonly considered for legged robots, external contact constraints depending on the current configuration of the robot and it's interactions with the world, and internal constraints from the robot geometry. Both constraints are enforced in the same manner, by prescribing a closure equation, $\eta(q) = \mathrm{constant}$, which represents a positional constraint. This expression then yields a constraint on the accelerations when differentiated twice:

$$J(q)\ddot{q} + \frac{\partial}{\partial q}\left(\frac{\partial J(q)}{\partial q}\dot{q}\right)\dot{q} = 0. \qquad (2)$$

The enforcement of this constraint gives rise to the corresponding force terms, λ, in the equations of motion (1).

Contact Constraints. In this work, ground contact is enforced through a holonomic constraint on a stance foot's position and orientation, $\eta_{st}(q)$. Because the width of the feet on Cassie is negligible we enforce contact as a line, or as two collinear points of contact at the toe and heel. This gives rise to a 5-DOF ground contact constraint:

$$\eta_{st}(q)^\mathsf{T} := \left[p^x_{st}, p^y_{st}, p^z_{st}, \phi^y_{st}, \phi^z_{st}\right]^\mathsf{T}, \qquad (3)$$

where the first three components are the Cartesian position of the center of the foot and the last two correspond to the the foot pitch and yaw, shown in Fig. 3. Because λ^z_{st} is a normal force, it is unilateral. Additionally, the forces λ^x_{st}, λ^y_{st} are required to satisfy friction models to remain feasible. Ideally, a classical *Amontons-Coulomb model* of (dry) friction is used to avoid slippage and is represented as a friction cone. For a friction coefficient μ and a surface normal, the space of valid reaction forces is then:

$$\mathcal{C} = \left\{ (\lambda_{st}^x, \lambda_{st}^y, \lambda_{st}^z) \in \mathbb{R}^3 \middle| \, \lambda_{st}^z \geq 0; \sqrt{(\lambda_{st}^x)^2 + (\lambda_{st}^y)^2} \leq \mu \lambda_{st}^z \right\}.$$

The moment associated with the foot pitch λ_{st}^{my} can produce rotation of the foot over the forward edge if it is too large. It has been shown that this moment is limited by:

$$-\frac{l}{2}\lambda_{st}^z < \lambda_{st}^{my} < \frac{l}{2}\lambda_{st}^z, \tag{4}$$

where l is the length of the foot from heel to toe.

Loop Constraints. It is common practice to model robotic manipulators in branched tree structures. However, on Cassie, a *compliant multi-link mechanism* is used to transfer power from higher to lower limbs without allocating the actuators' major weight onto the lower limbs, and effectively acts as a pair of springy legs [24]. When the mechanism has a kinematic loop, this is often managed by cutting the loop at one of the joints and enforcing a holonomic constraint at the connection to form the closed-chain manipulator. In the Cassie leg the heel spring is attached to the rear of the tarsus linkage, with its end constrained via a pushrod affixed to the hip pitch linkage. For this work, we assume that pushrod attachment is a virtual holonomic distance constraint applied between the hip and heel spring connectors as:

$$\eta_{ach}(q_l) := d(q_l) - 0.5012 = 0, \tag{5}$$

where the attachment distance $d(q_l) \in \mathbb{R}$ is obtained via the forward kinematics between connectors at the hip and heel spring (see Fig. 2). We also assume that when a leg is in swing that the springs on that leg are rigidly fixed:

$$\eta_{sw}(q)^{\mathsf{T}} := [q_s, q_{hs}]^{\mathsf{T}} = 0. \tag{6}$$

This simplifies both the optimization and control implementations [26], and makes the dynamics less numerically stiff.

C. Hybrid Locomotion Model

A bipedal walking gait consists of one or more different continuous phases followed by discrete events that transition from one phase to another, motivating the use of a hybrid system formulation with a specific ordering of phases. This is traditionally described as a walking cycle, which is a directed cycle with a sequence of continuous domains (continuous dynamics) and edges (changes in contact conditions).

In this work, we structure the dynamics of walking on Cassie in a hybrid fashion. The walking consists of two single support domains, $\mathcal{D}_{\text{SS}}^{\{\text{L,R}\}}$, associated with stance on the respective left (L) or right (R) foot, with an instantaneous double-support (the swing leg lifts immediately on impact). An associated directed graph can be specified for the system which is depicted in Fig. 3. There is an additional assumption that the walking is symmetric for both left and right stance when the robot's average lateral speed over a step is zero, while the walking is allowed to be asymmetric otherwise. A transition from one single support domain to another occurs when the normal reaction force on the non-stance foot crosses zero. Therefore, the domain and guard are given by:

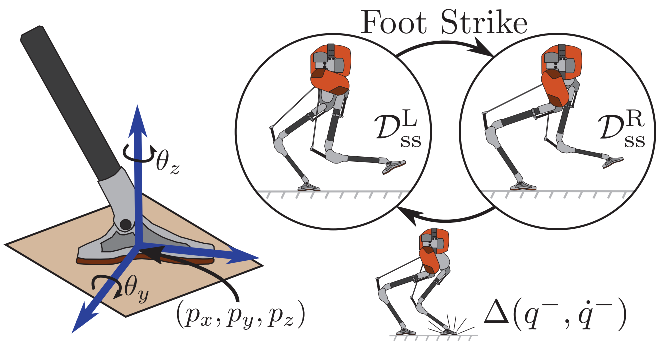
Foot Strike
θ_z
θ_y
(p_x, p_y, p_z)
$\mathcal{D}_{\text{SS}}^{\text{L}}$
$\mathcal{D}_{\text{SS}}^{\text{R}}$
$\Delta(q^-, \dot{q}^-)$

Fig. 3. Shown on the left is the contact geometry of the robot's foot, along with the constrained coordinates and associated contact frame. The directed graph of walking used in this paper is shown on the right, where we view walking on Cassie as consisting of two single-support domains with a compliant stance leg, and rigidly stiff swing leg.

$$\mathcal{D}_{\text{SS}}^{\{\text{L,R}\}} = \{(q, \dot{q}, u) : p_{nsf}^z(q) \geq 0, \lambda_{nsf}^z(q, \dot{q}, u) = 0\},$$
$$S_{\{\text{L}\to\text{R, R}\to\text{L}\}} = \{(q, \dot{q}) : p_{nsf}^z(q) = 0, \dot{p}_{nsf}^z(q, \dot{q}) < 0\},$$

where $\lambda_{nsf}^z(q, \dot{q}, u)$ is the vertical ground reaction force of the swing foot and $p_{nsf}^z(q)$ is the vertical position of the center of the swing foot from the ground. Impact occurs when the swing foot touches the ground, modeled here as an inelastic contact between two rigid bodies. In this contact model, the configurations of the robot are invariant through the impact and velocities will instantaneously change due to the introduction of new holonomic constraints [10], [14]. The associated reset map, Δ, is given as:

$$\Delta := \begin{bmatrix} q^+ \\ \dot{q}^+ \end{bmatrix} = \begin{bmatrix} \mathcal{R}q^- \\ \mathcal{R}\Delta^{\dot{q}}(q^-)\dot{q}^- \end{bmatrix}, \tag{7}$$

where $\mathcal{R}$ is a relabeling matrix, q^- and q^+ denote the pre and post-impact configuration, and $\Delta^{\dot{q}}(q)$ is obtained from the plastic impact equation:

$$\Delta^{\dot{q}}(q^-) = I - D^{-1}J^{\mathsf{T}}(JD^{-1}J^{\mathsf{T}})^{-1}J. \tag{8}$$

As in [26], the relabeling matrix accounts for the assumption that the spring on the swing leg is at its rest deflection.

III. Trajectory Planning for Walking

This section details the trajectory optimization approach used to design a collection of trajectories which can be implemented to obtain compliant walking behaviors on the physical system in both the sagittal and coronal directions.

A. Virtual Constraints

Analogous to holonomic constraints, virtual constraints are defined as a set of functions that regulate the motion of the robot with a desired behavior [28]. The term "virtual" comes from the fact that these constraints are enforced through feedback controllers instead of through physical constraints. The primary idea is to design a controller $u(x, \alpha)$ to regulate:

$$y(q, t) := y^a(q) - y^d(\alpha, t), \tag{9}$$

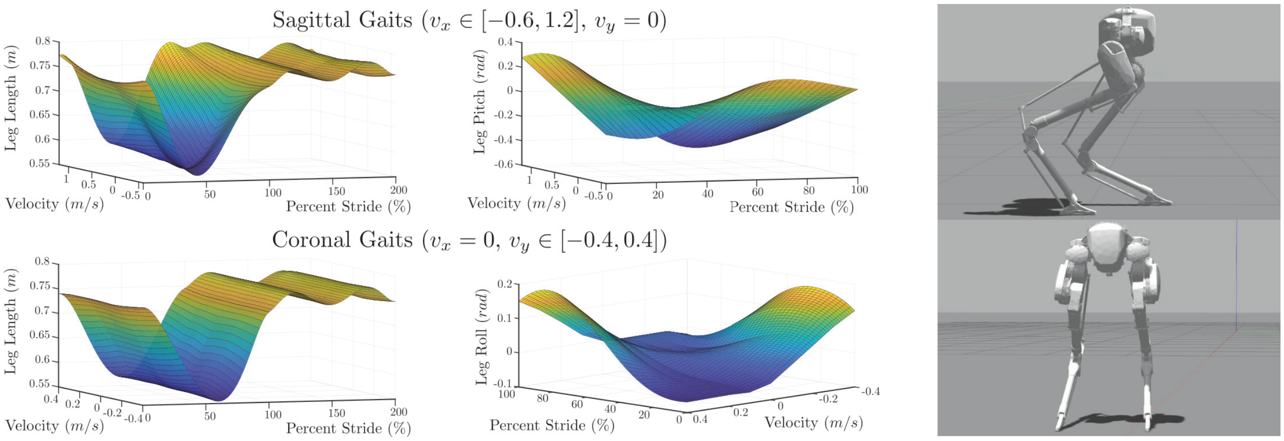

Fig. 4. Contour plots of the swing leg length, leg angle, and leg roll outputs over the library speeds in the sagittal and coronal directions, showing the forward and reverse sweep of the leg as it tracks the motions. Also shown is the corresponding motion executed in a Gazebo simulation.

for which a behavior can be encoded through the desired outputs, $y^d(\alpha, t)$. A 6th-order Bézier polynomial chosen for the desired outputs, for which α is a matrix of real coefficients that parameterize the curve. In this work, we consider nine outputs of vector relative degree 2:

$$y^a(q) := \left. \begin{bmatrix} \phi_x,\ \phi_y \\ \theta_{hy}^{st},\ \theta_{hy}^{sw} \\ \theta_{hr}^{sw} \\ ||\psi^{sw}||_2 \\ ||\psi^{st}||_2 \\ \text{atan2}\big(\psi_x^{sw}, \psi_z^{sw}\big) \\ \phi^y(q) \end{bmatrix} \right|_{\substack{q_{\text{s}=0} \\ q_{\text{hs}=0}}} \begin{pmatrix} \text{pelvis roll/pitch} \\ \text{stance/swing hip yaw} \\ \text{swing hip roll} \\ \text{swing leg length} \\ \text{stance leg length} \\ \text{swing leg pitch} \\ \text{swing foot pitch} \end{pmatrix}$$

where $\phi^y(q)$ is the swing foot Cartesian pitch and,

$$\psi^{st/sw}(q) = p_{hp}^{st/sw}(q) - p_{ak}^{st/sw}(q), \tag{10}$$

is the expression for the distance between the hip pitch (hp) and ankle pitch (ak) joints from the forward kinematics. Because the foot geometry is quite small, we leave the stance foot passive. It should be noted that we are controlling the undeflected leg length and angle of the legs by zeroing the spring deflections. By formulating the outputs in this way, the passive dynamics of the system will contain the additional dynamics associated with the compliant elements [26]. As a practical matter, this is also important as directly controlling the compliance in the leg is significantly more difficult.

B. Hybrid Zero Dynamics

The combined mechanical model (1) and (2) can be expressed in state variable form as,

$$\dot{x} = f(x) + g(x)u. \tag{11}$$

Additionally, because all outputs specified for the system have vector relative degree two, we assume the existence of a feedback control law $u^*(x, t)$ that drives $y \rightarrow 0$ exponentially, resulting in a closed loop system:

$$\dot{x} = f_{\text{cl}}(x) = f(x) + g(x)u^*(x). \tag{12}$$

Additionally, by driving the outputs to zero the controller u^* renders the *zero dynamics manifold*:

$$\mathbb{Z} = \{(q, \dot{q}) \in \mathcal{D} \mid y(q, t) = 0,\ L_{f_{\text{cl}}} y(q, \dot{q}, t) = 0\}. \tag{13}$$

forward invariant and attractive. Thus, the continuous dynamics (12) will then evolve on $\mathbb{Z}$. However, because (13) has been designed without taking into account the hybrid transition maps (7), it will not be impact invariant. In order to enforce impact invariance, the Bézier polynomials for the desired outputs can be shaped through the parameters α. This can be interpreted as the condition:

$$\Delta(\mathbb{Z} \cap S) \subset \mathbb{Z}, \tag{14}$$

and will be imposed as a constraint on the states through the impact (7). When (14) is satisfied, we say that the system lies on the *hybrid zero dynamics* (HZD) manifold.

C. Gait Library Optimization

The desired evolution of the outputs (9) must now be designed such that we can achieve locomotion, while also satisfying both the physical limitations of the hardware and other conditions such as (14). A nonlinear trajectory optimization problem is formed to solve this problem. Similar to [9], [29], we would like to design a variety of walking speeds for which the robot can operate. To accomplish this, a library of walking gaits at sagittal speeds of $v_x \in [-0.6, 1.2]$ m/s and coronal speeds of $v_y \in [-0.4, 0.4]$ m/s are generated in a grid of 0.1 m/s intervals. The impact-to-impact duration of each step is fixed at 0.4 seconds for ease of implementation.

Each hybrid optimization was performed over the two domains, $\mathcal{D}_{\text{SS}}^{\{\text{L,R}\}}$, with a constraint imposed such that when the discrete impact (7) is applied to the terminal state that it satisfies the hybrid invariance condition (14). It is critical that the motions respect the limitations of the physical system. In order to address this constraints for the friction cone, actuator limits, and joint limits are imposed. While these constraints alone ensure invariance and satisfaction of the physical constraints, additional constraints on the behavior were tuned for implementation on hardware, such as the

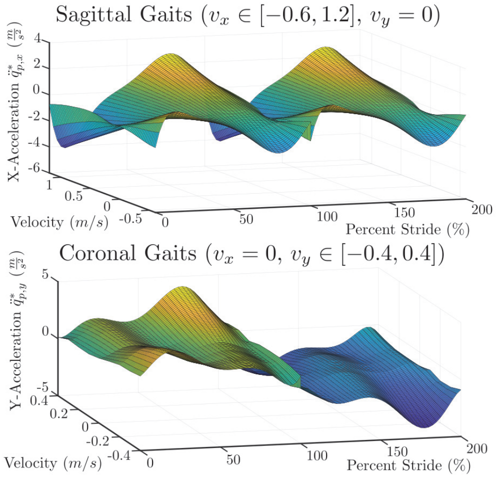

Fig. 5. The contours of the floating base x and y accelerations which are obtained from the trajectory optimization problem. These contribute to the control action of a feedforward controller which is dynamically consistent with the contact constraints of the underactuated robotic walking.

swing foot velocity, impact foot configuration, step symmetry when $v_y = 0$, and others outlined in Table I.

In order to minimize torque and to center the floating base coordinate movement around the origin, the following cost function was minimized:

$$\mathcal{J}(\mathbf{w}) := \int_{t=0}^{tf} \left(c_u |u|^2 + c_\phi |\phi_b|^2 \right) dt, \tag{15}$$

where $\mathbf{w} \in \mathbb{R}^{N_w}$ with N_w being the total number of optimization variables and $c_\square$ are weights applied to the respective terms. For the library presented in this work the weights used were $c_u = 0.0001$ and $c_\phi = (20, 1, 30)$.

An optimization package, FROST [12], was used to transcribe the constraints and cost of each of the 171 gaits into a nonlinear programming (NLP) problem that can be solved by a standard optimization solver such as IPOPT [5]:

$$\begin{aligned} \mathbf{w}^* = \underset{\mathbf{w}}{\text{argmin}} \quad & \mathcal{J}(\mathbf{w}) \\ \text{s.t.} \quad & \text{Closed loop dynamics: Eq.(12)} \\ & \text{HZD condition: Eq. (14)} \\ & \text{Physical feasibility: Table I} \end{aligned} \tag{16}$$

Each optimization was then solved through the C-FROST interface [11] on a laptop with an Intel Core i7-6820 HQ CPU @ 2.7 GHz with 16 GB RAM, and consisted of 8418 variables with 4502 equality and 5880 inequality constraints. Using each gait as an initial guess for the next speed in the library, the average number of iterations per run was 199 with an average total evaluation time of 263.8 seconds, and an average objective value of $\mathcal{J}(\mathbf{w}^*) = 4.12$.

TABLE I

OPTIMIZATION CONSTRAINTS AND PARAMETERS

Step duration	$= 0.4$	sec
Average step velocity, $\bar{v}_{x,y}$	$= v_{x,y}$	m/s
Pelvis height, p_z	≥ 0.80	m
Mid-step foot clearance, p^z_{nsf}	≥ 0.14	m
Vertical impact velocity, $\dot{p}^z_{sw}$	$\in (-0.40, -0.10)$	m/s
Step width, $p^y_{lf} - p^y_{lf}$	$\in (0.14, 0.35)$	m
Swing foot pitch, $\phi^y(q)$	$= 0$	rad
Friction cone, μ	< 0.6	

Extracting parameters for real-time control. The controller implemented on hardware needs both the feedback control objectives, defined by $y^d(\alpha, t)$, and acceleration information $\ddot{q}^*$ from the optimal path to complete the planned motions. The feedback parameters, α, already concisely parameterize the feedback control, and are placed in a large matrix which can be used in a bilinear interpolation routine. A subset of the resulting library of output parameters are shown in Fig. 4, where the leg length, leg pitch, and hip roll outputs are visualized over various walking speeds.

Generalized accelerations $\ddot{q}^*$ are extracted from the optimization variables, $\mathbf{w}$, as time-series data from each step. Regression is performed on each curve to obtain the parameters for a 6th order Bézier polynomial. They can then be stacked with the α parameters in the same bilinear interpolation routine for code efficiency. Plots of the accelerations for the floating base x and y coordinates are visualized in Fig. 5. Finally, the floating-base position $p^*_{x,y}$ and velocity $v^*_{x,y}$ relative to the stance foot is also extracted in the same manner as $\ddot{q}^*$, and shown in Fig. 7.

IV. Real-time Tracking of Optimized Motions

The optimization framework described in Sec. III creates a continuum of walking trajectories [7], which can be interpolated to obtain walking motions at a range of speeds. In previous work on Cassie [9], [24], PD control achieved reasonable tracking of the desired outputs with the passive compliance in the system matching the planned response. However, implementation relied on high gains with hand-tuned heuristic feedforward and trajectory modifications to account for the model mismatch. This section will focus on an inverse dynamics controller which can leverage nominal accelerations already found via the optimized plan.

A. Inverse Dynamics Controller

The use of inverse dynamics control for underactuated and floating-base robots is significantly more complex than for fixed-base manipulators. In developing controllers for these systems, there are many considerations to address such as numerical problems due to repeated matrix inversions of the inertia matrix, contact force distribution, and computational efficiency [23]. Here, we follow the approach of Mistry [17], which uses an orthogonal decomposition to compute the inverse dynamics torques in the null-space of the constraints.

The underlying idea of the method is to use a QR decomposition of the constraint Jacobian matrix:

$$J(q)^\mathsf{T} = Q \begin{bmatrix} R \\ 0 \end{bmatrix}, \tag{17}$$

where Q is an orthogonal matrix and R is an upper triangle matrix of rank m_η. The Q in the QR decomposition provides a coordinate transform which separates the system dynamics (1) into the constrained and unconstrained components:

$$Q^\mathsf{T}\left[D(q)\ddot{q}+H(q,\dot{q})\right]=Q^\mathsf{T}Bu+\begin{bmatrix}R\\0\end{bmatrix}\lambda. \tag{18}$$

Through the use of a selection matrix $S_u = \left[0_{(n-k)\times k} \quad I_{(n-k)\times(n-k)}\right]$, we can obtain the unconstrained system dynamics which are not dependent on the constraints:

$$S_uQ^\mathsf{T}\left[D(q)\ddot{q}+H(q,\dot{q})\right]=S_uQ^\mathsf{T}Bu. \tag{19}$$

Solving for u leads us to the inverse dynamics:

$$u_{ff}=(S_uQ^\mathsf{T}B)^\dagger S_uQ^\mathsf{T}\left[D(q)\ddot{q}^d+H(q,\dot{q})\right], \tag{20}$$

where † represents the appropriate Moore-Penrose inverse and $\ddot{q}^d$ is a target acceleration for the system to follow. For a more detailed analysis of the orthogonal decomposition and controller derivation, we again direct readers to [17]. In addition to the feedforward term provided by (20), a feedback correction is computed as a standard PD controller:

$$u=u_{ff}-\left(\frac{\partial y(t,q)}{\partial q_m}\right)^\dagger\left(K_py(t,q)+K_d\dot{y}(t,q,\dot{q})\right), \tag{21}$$

where $K_p = [900, 500, 300, 250, 200, 200, 200, 200, 25]$ and $K_d = [12, 6, 4, 6, 5, 6, 4, 4, 2]$ are the proportional and derivative gains applied to the output feedback on hardware.

In order to select the parameters which are used for output tracking and for the feedforward term, we use average velocity of the previous step, $\bar{v}^a_{k-1}$. Thus, our Bèzier polynomial array provides us at each time instant with the nominal acceleration $\ddot{q}^*(t,\bar{v}^a_{k-1})$, outputs $y^d(t,\bar{v}^a_{k-1})$, floating-base positions $p^*_{x,y}(t,\bar{v}^a_{k-1})$ and velocities $v^*_{x,y}(t,\bar{v}^a_{k-1})$.

In (18) the spring force term, $B_s\tau_s$, is not present. We originally developed the controller with the spring dynamics and were able to achieve walking. However, the robot would often overreact to perturbations. Instead, we apply a constraint on the springs (6) at their current deflection. Because we are using (18) with the accelerations from the compliant plan, the controller still produces actuated joint accelerations/torques according to the library (see Fig. 7).

B. Velocity Tracking and Regulation Heuristics

The presented controller required no heuristic tuning on the feedforward terms to achieve tracking of the outputs such as offsets or hand-tuned torque profiles. However, directly implementing this controller with the trajectory obtained from the NLP (16) can at best result in a marginally stable locomotion for experiments as it has no way to trend walking towards an overall target walking speed. Motivated by this, a PD controller is used to find an offset to the footstrike location, and translated to the desired outputs:

$$\Delta := \begin{bmatrix}\Delta_x\\\Delta_y\\\Delta_z\end{bmatrix} = \begin{bmatrix}\tilde{K}_{p,x}(\tilde{v}^a_x-v^d_x)+\tilde{K}_{d,x}(\tilde{v}^a_x-\bar{v}^a_{x,k-1})\\\tilde{K}_{p,y}(\tilde{v}^a_y-v^d_y)+\tilde{K}_{d,y}(\tilde{v}^a_y-\bar{v}^a_{y,k-1})\\0\end{bmatrix},$$

$$\begin{aligned}y^d_{sw,ll}&=\|p^*_{nsf}+\Delta\|_2,\\y^d_{lp}&=\sin^{-1}\left((p^*_x(y^d)+\Delta_x)/y^d_{sw,ll}\right)-y^d_{b,x},\\y^d_{lr}&=\sin^{-1}\left((p^*_y(y^d)+\Delta_y)/y^d_{sw,ll}\right)-y^d_{b,y},\end{aligned}$$

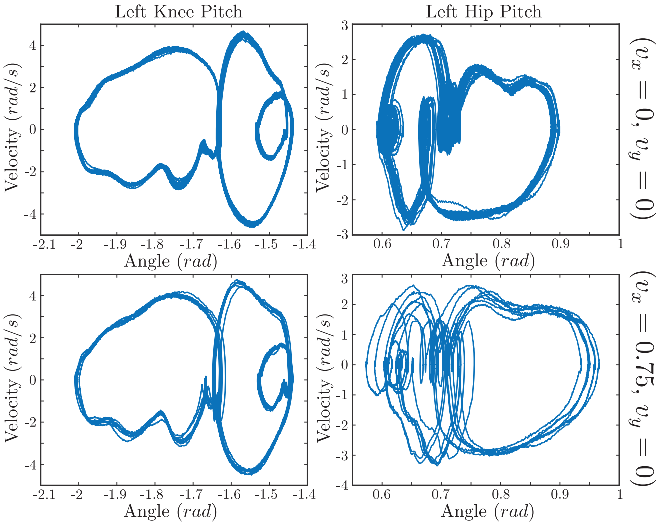

Fig. 6. Phase portraits of the left knee and hip pitch joints.

where $\tilde{v}^a_{x,y} = \bar{v}^a_{k-1} + \left(v^a_{x,y} - v^*_{x,y}\right)$ is the current step velocity, $v^d_{x,y}$ is the target step velocity from the user joystick, $v^a_{x,y}$ is the instantaneous velocity of the robot relative to the stance foot, and $p^*_{nsf}(y^d) = (p^*_{nsf,x}, p^*_{nsf,y}, p^*_{nsf,z})$ are the nominal Cartesian swing foot positions computed from the desired outputs. This style of regulator is inspired by early work of [19], and has been used widely in the literature.

An additional regulator is applied to modify the nominal accelerations of the floating base accelerations, which were pictured in Fig. 5. Through the application of a heuristic feedback controller, we can make the robot choose feedforward torques which trend the robot toward the target velocity while satisfying the contact constraints on the system:

$$\begin{aligned}\ddot{q}^d_{x,y}=&\ddot{q}^*_{x,y}+k_p\left(p^a_{x,y}(q)-p^*_{x,y}\right)+\\&k_v(\tilde{v}^a_{x,y}-v^d_{x,y})+k_i\int_0^t\gamma(\tilde{v}^a_{x,y}(t')-v^d_{x,y}(t'))dt',\end{aligned}$$

where $k_p = [1.25, 1.90]$ is a gain affecting the x and y position errors of the pelvis relative to the stance foot, $k_v = [0.80, 0.60]$ is a gain on the step velocity tracking error, and $k_i = [1.90, 0.0]$ is a gain on the accumulated step velocity error with a decay constant of $\gamma = 0.9995$ to avoid integral windup. Because the feedforward term (20) uses the full actuator matrix B, it does apply some control effort on the stance foot actuator. It is for that reason that the integral term is applied to the x direction, and not in the y.

C. Experimental Implementation and Results

The controller is implemented on the Intel NUC computer which comes installed in the Cassie torso, on which we added a PREEMPT_RT kernel. The software runs on two ROS nodes: one which communicates to the Simulink Real-Time xPC over UDP to send torques and receive sensor data and to perform estimation, and a second which runs the controllers. Each node is given a separate core on the CPU, and is elevated to real-time priority. The first node runs at 2 kHz and executes contact classification, inverse kinematics to obtain the heel spring deflection, and an EKF for velocity

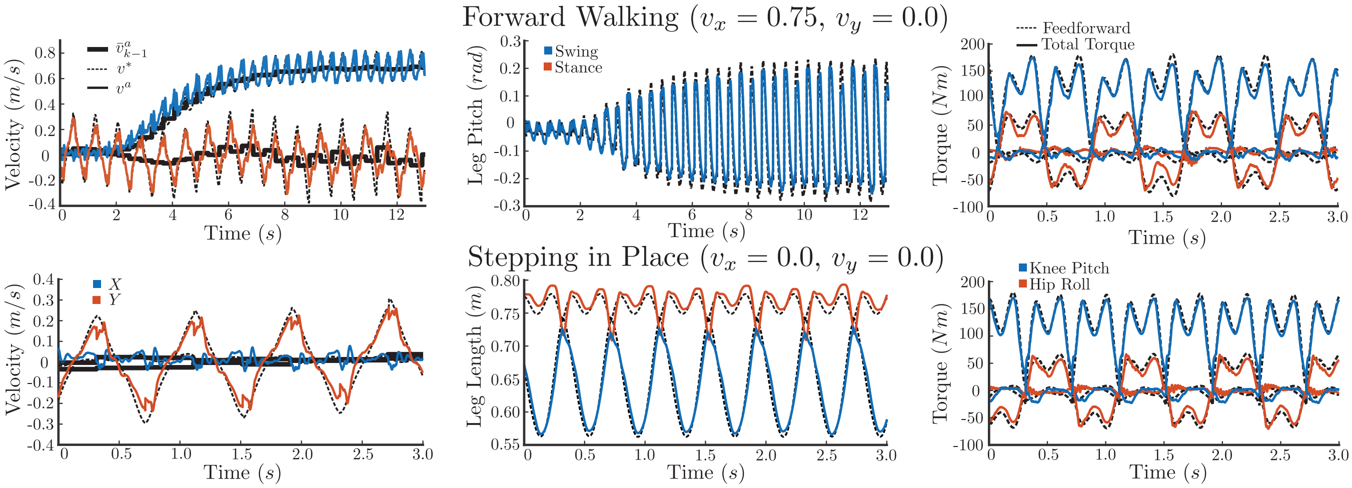

Fig. 7. Experimental results for walking (top) and stepping in place (bottom). On the left is a comparison of the desired velocities from the current gait, $\bar{v}^a_{k-1}$, compared with the actual velocity of the robot. Also depicted are the accuracy of the tracking on the leg length and swing leg pitch outputs (center), and the feedforward and total torques for the knee pitch and hip roll joints.

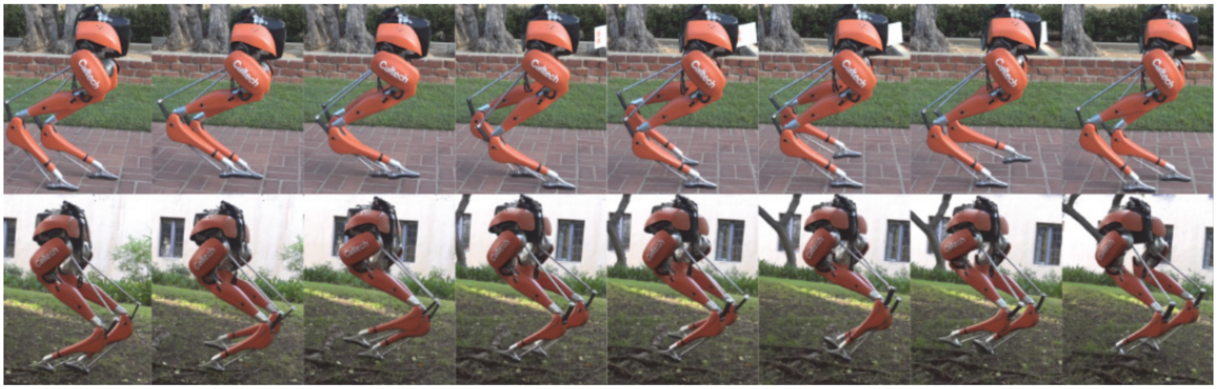

Fig. 8. Gait tiles of the robotic walking exhibited experimentally on the Caltech Cassie robot. The top half is two steps of flat-ground forward walking outside, while the bottom half shows several steps of the robot walking over a height variation caused by both a shallow grassy slope and exposed roots.

estimation [6], [24]. The second node runs at 1 kHz and receives the estimation and proprioceptive data over ROS. It then runs either the standing controller presented in [23], or executes the controllers presented in Sec. IV-A and IV-B before communicating the commanded torque over ROS.

These controllers were first tuned in a Gazebo simulation, shown in Fig. 4, before implementation on hardware. Cassie was then tested indoors for stepping in place and forward, backward, lateral, and diagonal walking. The velocity tracking data, output tracking, and feed-forward torques are shown in Fig. 7 for forward walking and stepping in place. Additionally, limit cycles of the knee and hip pitch joints are shown in Fig. 6, demonstrating the stability of the walking. Cassie was then taken outside to walk over several hills, raised roots, and a brick path. A video of both the simulation and experiments is provided at [2], with walking tiles of the robot on flat ground and traversing rough terrain is shown in Fig. 8. The controller, simulation, and trajectory optimization also made available in an open-source repository [1].

V. Conclusions and Future Work

The Cassie biped poses a unique challenge due to its compliant leg mechanism and the highly underactuated nature of the dynamics. In order to leverage these components in experiments, we constructed a hybrid model for walking dynamics based on a compliant model. A trajectory optimization was then developed to efficiently generate walking trajectories using the method of HZD and was then tracked through an inverse dynamics controller. The resulting experiments show that the optimization can effectively capture the passive dynamics on a highly complex robot, while providing accurate model-based feedforward information.

On the physical system, a trivial double-support domain is virtually impossible to attain due to the compliance present in both legs. While this assumption simplified the development of controllers and trajectory planning, the consideration of a continuous double-support domain can enhance the stability of the behaviors. Future work will focus on reintroducing a nontrivial double-support domain to our walking model. Additionally, the use of an analytical solution to the inverse dynamics cannot account for optimal distributions of contact forces, friction, torque limits, or provide convergence guarantees. Thus, our future controller development is focused on further developing real-time model based controllers for implementation on hardware, such as extending the control Lyapunov function based methods in [23] to walking.

References

[1] Open-sourced repository for the C++ controller code used in this work on hardware and for use in a Gazebo simulation. https://github.com/jpreher/cassie_documentation.

[2] Video of simulation and experimental results on Cassie. https://youtu.be/H9r-GKkYDRs.

[3] Farhad Aghili. A unified approach for inverse and direct dynamics of constrained multibody systems based on linear projection operator: applications to control and simulation. *IEEE Transactions on Robotics*, 21(5):834–849, 2005.

[4] Aaron D Ames, Kevin Galloway, Koushil Sreenath, and Jessy W Grizzle. Rapidly exponentially stabilizing control lyapunov functions and hybrid zero dynamics. *IEEE Transactions on Automatic Control*, 59(4):876–891, 2014.

[5] Lorenz T Biegler and Victor M Zavala. Large-scale nonlinear programming using ipopt: An integrating framework for enterprise-wide dynamic optimization. *Computers & Chemical Engineering*, 33(3):575–582, 2009.

[6] 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*, 17:17–24, 2013.

[7] Xingye Da, Ross Hartley, and Jessy W Grizzle. Supervised learning for stabilizing underactuated bipedal robot locomotion, with outdoor experiments on the wave field. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3476–3483. IEEE, 2017.

[8] 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.

[9] 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. In *2019 American Control Conference (ACC)*, pages 4559–4566. IEEE, 2019.

[10] Jessy Grizzle, Christine Chevallereau, Ryan Sinnet, and Aaron Ames. Models, feedback control, and open problems of 3D bipedal robotic walking. *Automatica*, 50(8):1955 – 1988, 2014.

[11] A. Hereid, O. Harib, R. Hartley, Y. Gong, and J. W. Grizzle. Rapid trajectory optimization using C-FROST with illustration on a Cassie-series dynamic walking biped. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4722–4729, 2019.

[12] 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.

[13] Ian W Hunter, John M Hollerbach, and John Ballantyne. A comparative analysis of actuator technologies for robotics. *Robotics Review*, 2:299–342, 1991.

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

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

[16] 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.

[17] Michael Mistry, Jonas Buchli, and Stefan Schaal. Inverse dynamics control of floating base systems using orthogonal decomposition. In *2010 IEEE international conference on robotics and automation*, pages 3406–3412. IEEE, 2010.

[18] Hae-Won Park, Koushil Sreenath, Jonathan W Hurst, and Jessy W Grizzle. Identification of a bipedal robot with a compliant drivetrain. *IEEE Control Systems*, 31(2):63–88, 2011.

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

[20] Alireza Ramezani, Jonathan W Hurst, Kaveh Akbari Hamed, and Jessy W Grizzle. Performance analysis and feedback control of ATRIAS, a three-dimensional bipedal robot. *Journal of Dynamic Systems, Measurement, and Control*, 136(2), 2014.

[21] Jenna Reher and Aaron D. Ames. Dynamic walking: Toward agile and efficient bipedal robots. *To appear in the Annual Review of Control, Robotics, and Autonomous Systems*, 2021. arXiv preprint arXiv:2010.07451.

[22] Jenna 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.

[23] Jenna Reher, Claudia Kann, and Aaron D Ames. An inverse dynamics approach to control Lyapunov functions. In *2020 American Control Conference (ACC)*, pages 2444–2451. IEEE, 2020.

[24] Jenna Reher, Wen-Loong Ma, and Aaron D Ames. Dynamic walking with compliance on a Cassie bipedal robot. In *2019 18th European Control Conference (ECC)*, pages 2589–2595. IEEE, 2019.

[25] 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 *Proceedings of the ASME 2015 Dynamic Systems and Control Conference*, 2015.

[26] Koushil Sreenath, Hae-Won Park, Ioannis Poulakakis, and Jessy W Grizzle. Embedding active force control within the compliant hybrid zero dynamics to achieve stable, fast running on MABEL. *The International Journal of Robotics Research*, 32(3):324–345, 2013.

[27] Miomir Vukobratović and Branislav Borovac. Zero-moment point—thirty five years of its life. *International journal of humanoid robotics*, 1(01):157–173, 2004.

[28] Eric R Westervelt, Jessy W Grizzle, Christine Chevallereau, Jun Ho Choi, and Benjamin Morris. *Feedback control of dynamic bipedal robot locomotion*. CRC press, 2018.

[29] Zhaoming Xie, Patrick Clary, Jeremy Dao, Pedro Morais, Jonanthan Hurst, and Michiel Panne. Learning locomotion skills for cassie: Iterative design and sim-to-real. In *Conference on Robot Learning*, pages 317–329, 2020.