Feedback Control of a Cassie Bipedal Robot: Walking, Standing, and Riding a Segway

Yukai Gong, Ross Hartley, Xingye Da, Ayonga Hereid, Omar Harib, Jiunn-Kai Huang, and Jessy Grizzle

Abstract— The Cassie bipedal robot designed by Agility Robotics is providing academics a common platform for sharing and comparing algorithms for locomotion, perception, and navigation. This paper focuses on feedback control for standing and walking using the methods of virtual constraints and gait libraries. The designed controller was implemented six weeks after the robot arrived at the University of Michigan and allowed it to stand in place as well as walk over sidewalks, grass, snow, sand, and burning brush. The controller for standing also enables the robot to ride a Segway. A model of the Cassie robot has been placed on GitHub and the controller will also be made open source if the paper is accepted.

I. Introduction and Related Work

The primary purpose of this paper is to introduce and document an *open-source controller* for the Cassie bipedal robot shown in Fig. 1 and video [5]. Agility Robotics delivered its first batch of Cassie robots to the University of Michigan, Caltech, and Harvard at the end of August, beginning of September, 2017. Since then, each team has been working to develop its own control laws [34].

A. Contributions

The Cassie robot comes with a proprietary control law in the form of .p MATLAB files (i.e., code is obfuscated) that run in Real-Time Simulink . The native controller provides for quiet standing, walking forward, backward, and sideways, and turning. The controller presented here provides these same features.

Among this paper's contributions are (a) the controller's design is documented and (b) its implementation in Real-Time Simulink will be available on GitHub if the paper is accepted. Additional contributions include the following: (c) The robot is operated in more challenging conditions than those demonstrated by Agility Robotics, such as walking in a controlled burn, walking in snow, and walking in soft sand [5]. (d) The robot is treated as being underactuated in the sense that stance ankle torque is not required to maintain balance when walking. Because of this, the robot's feet sinking into soft sand at a 30° pitch angle does not affect stability of the gait. (e) Finally, the robot is demonstrated riding a Segway, a task relevant to search and rescue operations where a robot could use a tool found at the scene to save energy or effect a faster arrival.

The authors are with the College of Engineering and the Robotics Institute, University of Michigan, Ann Arbor, MI 48109 USA {ykgong,rosshart,xda,ayonga,oharib,bjhuang, grizzle}@umich.edu

B. Related Work

The majority of walking gaits in 3D humanoid robots are stabilized by regulating the Zero-Moment Point (ZMP) [10], [17], [18], [20] (roughly the center of pressure of the stance foot). Because this method requires having a non-zero support polygon, it is not applicable to robots such as Cassie with line feet [31]. Simplifying the robot's dynamics to a point mass attached to a rod of fixed length, the so-called (inverted) pendulum models, is another ubiquitous means in the bipedal robotics literature to approach the gait design and stabilization problem [11], [21], [23], [25], [28]. The many challenges associated with the pendulum approach include, exploiting the full capability of the robot, respecting motor and joint limitations, and deciding how to associate the states of the full-order system with the motion of the low-dimensional pendulum.

An important feature of the pendulum models, nevertheless, is that the Cartesian velocity of their center of mass can be "stabilized by foot placement"; more precisely, by controlling the angle of the swing leg at ground contact, the robot's speed can be regulated to a desired value [22], [26], [30]. Indeed, the open-loop center of mass velocity of the linear inverted pendulum (LIP) evolves as an integrator in the longitudinal and lateral directions and can therefore be stabilized with a proportional controller.

The controller presented here is based on the *gait library method* introduced in [9]. It allows the full dynamic model of

Fig. 1: Cassie Blue, one of Michigan's Cassie-series robots designed by Agility Robotics. The robot is shown here participating in a controlled burn. It has 20 degrees of freedom, 10 actuators, joint encoders, and an IMU. The robot's serial number is 001.

the robot to be used for the design of periodic walking gaits so that key limits on joint angles, ground reaction forces, motor torque, and motor power can be directly incorporated as constraints into a parameter optimization problem that minimizes energy consumption per distance traveled for a given speed. The individual periodic gaits are realized on the robot via virtual constraints [7], which are in turn "gain scheduled" with the robot's current velocity, thereby creating a continuum of gaits. The resulting closed-loop system enjoys the property that its center of mass velocity evolves *approximately* as an integrator in the longitudinal and lateral directions [9], and hence, similar to the inverted pendulum models, walking speed can be regulated to a desired value via "foot placement" as in [27].

II. CASSIE ROBOT

The bipedal robot shown in Fig. 1, named Cassie Blue, is designed and built by Agility Robotics. It is all electric and can walk for approximately four hours on a single battery charge. The robot's morphology and name are inspired by the *Cassowary*, a flightless bird native to New Guinea; those without a biology background can liken it to an ostrich.

When standing up, Cassie is about one meter tall. The robot's total mass is 31 kg, of which the battery, housed in its torso, accounts for approximately 4 kg. The floating base model of the robot has 20 DoF. There are seven joints in each leg, five of which are actuated by electric motors and the other two are passive joints realized via a four-bar linkage where one of the links is a leaf spring [34, Fig. 2]. Three of the five actuated joints in each leg correspond to setting the roll, yaw, and pitch of the leg with respect to the torso, one motor sets the angle of the "knee", and the last motor regulates the pitch angle of the toe, hereafter called the foot. The robot is fully capable of 3D walking and quiet standing.

A. Coordinate and Actuation Description

A URDF of the robot Cassie is available online [13]; in addition, Agility Robotics has released two dynamic models
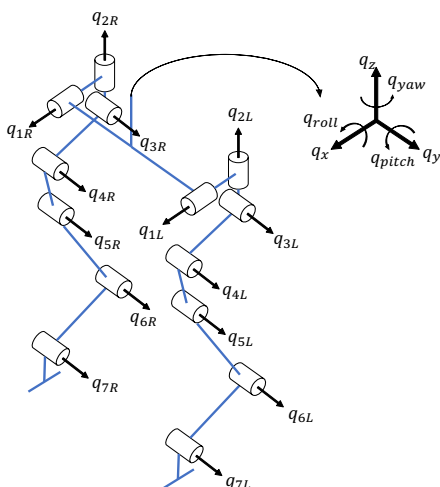

Fig. 2: A kinematic model of Cassie. The shin is the link between joints q_5 and q_6. The tarsus is the link between joints q_6 and q_7. The last link on the robot is anatomically a toe, but we will call it a foot because it functions as one. The joints $q_1, \ldots, q_4$ and q_7 are actuated and corresponding the motor torques are labeled $u_1, \ldots, u_4$ and u_5 in the obvious way.

of the robot [29]. Figure 2 shows a kinematic model of Cassie with the configuration variables labeled. The variables (q_x, q_y, q_z) are the Cartesian position of the torso, and $q_{yaw}, q_{pitch}, q_{roll}$ are the intrinsic Euler Angles taken in the z-y-x order. The coordinates q_1, q_2, q_3 are the hip roll, hip yaw, and hip pitch angles, respectively, q_4 is the knee pitch angle, q_5 is the shin pitch angle, q_6 is the tarsus pitch angle, and q_7 is the toe pitch angle. Very roughly speaking, in a human, the tarsus is the part of the foot from the heel to the metatarsus, which in turn is the part of the foot from which the toes radiate. When we walk on two legs, our tarsus is on the ground and we call it the foot. When we crawl on all fours, our tarsus is in the air, like Cassie's. From now on, we'll also call Cassie's toe a "foot" because it functions as a foot with respect to most other bipedal robots.

The coordinates are lumped into a vector

$$
\begin{aligned}
q :=[&q_x, q_y, q_z, q_{yaw}, q_{pitch}, q_{roll}, \\
&q_{1L}, q_{2L}, q_{3L}, q_{4L}, q_{5L}, q_{6L}, q_{7L}, \\
&q_{1R}, q_{2R}, q_{3R}, q_{4R}, q_{5R}, q_{6R}, q_{7R}]^{\mathsf{T}}.
\end{aligned}
\tag{1}
$$

The *body* or *shape* coordinates are

$$
q_{\mathrm{body}} := [q_{1L}, \ldots, q_{7L}, q_{1R}, \ldots, q_{7R}]^{\mathsf{T}},
\tag{2}
$$

the relative angles of the joints connecting the various links of the robot; these are measured via encoders while the Euler parameters in q are estimated by an EKF that comes with the IMU on Cassie, a VectorNav-100.

The joints $(q_1, q_2, q_3, q_4, q_7)$ are actuated. The joints (q_5, q_6) are passive, being formed by the introduction of (rather stiff) springs. When the springs are uncompressed,

$$
q_5 = 0, \text{ and } q_6 = -q_4 + 13^{\circ}.
\tag{3}
$$

B. Floating-base Model

The floating-base dynamic model is expressed in the form of Lagrange as

$$
D(q)\ddot{q} + H(q, \dot{q}) = Bu + J_{sp}(q)^{\top}\tau_{\mathrm{sp}} + J(q)^{\top}\lambda,
\tag{4}
$$

with q the vector of generalized coordinates given in (1), u the vector of motor torques, τ_{sp} the spring torques, and λ is the contact wrench. $D(q)$ is the mass-inertia matrix, $H(q, \dot{q})$ contains the velocity and gravitational terms, and B is the motor torque distribution matrix. The Jacobian for the springs is given by

$$
J_{sp}(q) := \frac{\partial}{\partial q}
\begin{bmatrix}
q_{5\mathrm{L}} \\
q_{4\mathrm{L}} + q_{5\mathrm{L}} + q_{6\mathrm{L}} \\
q_{5\mathrm{R}} \\
q_{4\mathrm{R}} + q_{5\mathrm{R}} + q_{6\mathrm{R}}
\end{bmatrix}.
\tag{5}
$$

The ground contact Jacobian is specified in the next subsection.

The open-source package FROST [14] is used to generate all of the terms in (4). The floating base model used for control design is available as open-source code [13]. In that model, the spring stiffness is taken to infinity, resulting in (3) holding. For simulations, the springs are kept.

C. Hybrid Model for Walking

Walking is modeled as a hybrid system corresponding to alternating phases of single support (one leg in contact with the ground and no slipping) and double support (both legs in contact with the ground). The double support phase is assumed to be instantaneous. This paper uses two domains; one for the right leg in stance, and one for the left leg in stance. The non-stance foot is also called the swing foot. Periodic gaits are computed over one step each on the right and left legs. The following assumes the *right leg is stance* and the *left leg is swing*. The labels R and L are swapped when the left leg is in stance.

1) Single Support: Ground contact is assumed to give rise to a holonomic constraint on foot position and orientation, $c_{\mathrm{R}}(q)$, that maintains the Cartesian position as well as the yaw and pitch angle of the stance foot to constant values during a step, while the foot's roll angle is unconstrained due to the narrow width of Cassie's feet.

$$c_{\mathrm{R}}(q)^\top := \left[p_{\mathrm{R}}^x, p_{\mathrm{R}}^y, p_{\mathrm{R}}^z, \psi_R^{\mathrm{yaw}}, \psi_R^{\mathrm{pitch}} \right]^\top. \quad (6)$$

With the right foot in stance, the contact Jacobian is $J_{\mathrm{R}}(q) := \frac{\partial c_{\mathrm{R}}(q)}{\partial q}$; the contact constraint is imposed by setting its acceleration to zero

$$J_{\mathrm{R}}(q)\ddot{q} + \dot{J}_{\mathrm{R}}(q, \dot{q})\dot{q} = 0. \quad (7)$$

The contact Jacobian has full row rank of five, so using (7) it is possible to eliminate five coordinates from (4) and the ground contact forces on the stance foot (the forces on the swing foot are identically zero), thereby obtaining a Lagrangian model with 15 DoF. We skip this step because later, for gait design, we once again use the package FROST which works directly with differential algebraic equations, such as (4) and (7).

2) Double Support: The transition from single support to double support is captured by the height of the swing foot from the ground decreasing to zero. Specifically, with the right foot in stance, the transition set is

$$\mathcal{S}_{\mathrm{R} \to \mathrm{L}} = \{ (q, \dot{q}) \in \mathcal{TQ} \mid p_L^z(q) = 0, \ \dot{p}_L^z(q, \dot{q}) < 0 \} \quad (8)$$

where $p_L^z(q)$ is the vertical component of the position of the swing foot; see Fig. 3a.

Remark: *During a step, the controller will maintain the pitch angle of the swing foot approximately level with the ground. Moreover, the applied torque is limited to a sufficiently low value that on solid ground, the weight of the robot easily and quickly places the foot level with the ground. Finally, impact is detected by the compression of the springs in joints q_{5L} and q_{6L}.*

The instantaneous impacts are modeled through a discrete map that results in a discontinuity in the velocity of the system $\dot{q}^-$ just before impact and the velocity of the system $\dot{q}^+$ just after impact, while the positions do not change [33]. Moreover, just after impact, the former swing foot is assumed to satisfy the same constraints as those imposed on the stance foot. Letting $c_{\mathrm{L}}(q)$ denote the corresponding holonomic constraint, the pre- and post-impact velocities
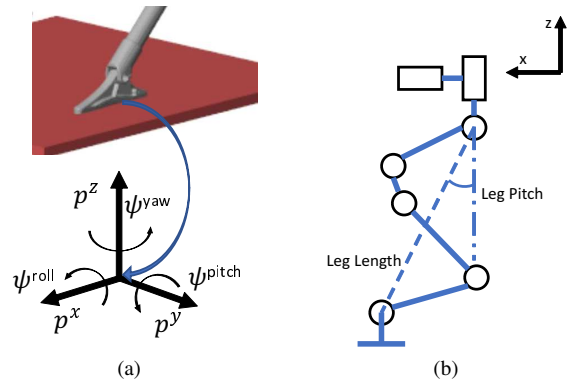

Fig. 3: (a) The coordinates on the feet are shown. When a foot is in contact with the ground, there are five independent constraints leaving only the roll angle of the foot free. (b) The virtual leg is the dotted line from the hip to the top of the foot; its length is called Leg Length. The relative leg pitch is its angle of the virtual leg relative to the hip while absolute leg pitch is the relative angle plus the pitch angle of the torso.

then satisfy

$$\begin{bmatrix} D(q) & -J_{\mathrm{L}}^T(q) \\ J_{\mathrm{L}}(q) & 0 \end{bmatrix} \begin{bmatrix} \dot{q}^+ \\ \delta F_{\mathrm{L}} \end{bmatrix} = \begin{bmatrix} D(q)\dot{q}^- \\ 0 \end{bmatrix}, \quad (9)$$

where δF_{L} is the vector of contact impulses. Because the Jacobian $J_{\mathrm{L}}(q)$ has full row rank and $D(q)$ is positive definite, the left hand side of (9) is invertible. Projecting the solution of (9) to the velocity components defines the impact map,

$$\dot{q}^+ =: \Delta_{\mathrm{R} \to \mathrm{L}}(\dot{q}^-). \quad (10)$$

3) Hybrid Model: The overall model is given as follows:

$$\Sigma : \begin{cases} \begin{aligned} D(q)\ddot{q} + H(q, \dot{q}) &= Bu + J_{sp}(q)^\top \tau_{\mathrm{sp}} + J_{\mathrm{R}}(q)^\top \lambda \\ J_{\mathrm{R}}(q)\ddot{q} + \dot{J}_{\mathrm{R}}(q, \dot{q})\dot{q} &= 0 \quad (q; \dot{q}^-) \notin \mathcal{S}_{\mathrm{R} \to \mathrm{L}} \\ \dot{q}^+ &= \Delta_{\mathrm{R} \to \mathrm{L}}(\dot{q}^-) \quad (q; \dot{q}^-) \in \mathcal{S}_{\mathrm{R} \to \mathrm{L}} \end{aligned} \\ \\ \begin{aligned} D(q)\ddot{q} + H(q, \dot{q}) &= Bu + J_{sp}(q)^\top \tau_{\mathrm{sp}} + J_{\mathrm{L}}(q)^\top \lambda \\ J_{\mathrm{L}}(q)\ddot{q} + \dot{J}_{\mathrm{L}}(q, \dot{q})\dot{q} &= 0 \quad (q; \dot{q}^-) \notin \mathcal{S}_{\mathrm{L} \to \mathrm{R}} \\ \dot{q}^+ &= \Delta_{\mathrm{L} \to \mathrm{R}}(\dot{q}^-) \quad (q; \dot{q}^-) \in \mathcal{S}_{\mathrm{L} \to \mathrm{R}}. \end{aligned} \end{cases} \quad (11)$$

D. Model for Standing

Cassie can stand quietly. The model is given by (4) with $J^\top = [J_{\mathrm{R}}^\top, J_{\mathrm{L}}^\top]$. If the feet are flat on the ground, 10 DoF are removed. In this case, when the spring stiffness is considered to be infinite, and the x-axes of the two feet do not lie on a common line, the model is fully actuated.

III. WALKING ON VARIOUS TERRAINS

This section presents the initial walking gait controller implemented on Cassie. The control design is based on the method of *virtual constraints*, a *gait library* limited to the sagittal plane, and *leg-angle adjustment* in the sagittal and frontal planes [9].

A. Virtual Constraints

Virtual constraints are functional relations among the generalized coordinates of the robot that are asymptotically

imposed on the system through feedback control. In particular, the virtual constraints are expressed as outputs of the model (4) in the form

$$y = h(q, \tau, \alpha) = h_0(q) - h_d(\tau, \alpha), \qquad (12)$$

where h_0 specifies the quantities being regulated, h_d encodes their desired evolution, and α is a matrix of real coefficients that parameterizes the spline h_d. The phasing variable, τ, satisfies

$$\dot{\tau} = \frac{1}{T}, \qquad (q; \dot{q}^-) \notin \mathcal{S}_{R \to L} \cup \mathcal{S}_{L \to R}$$
$$\tau^+ = 0 \qquad \text{otherwise}, \qquad (13)$$

where T is the nominal step duration.

A controller is then designed with the objective of zeroing the outputs, i.e., $y \equiv 0$, thereby achieving the virtual constraints. The zeroing of the output value will be at best accomplished asymptotically, and in practice, on a physical robot, only approximately.

B. Choice of What to Control

The most direct choice for the regulated quantities, h_0, would be the actuated joints of Cassie, which are a subset of the body coordinates. On previous planar robots we have studied, such as Rabbit and MABEL [24], [32], this made sense because these robots had a simple (human-inspired) morphology and the control objectives could be associated in an intuitive manner with hip and knee angles. On our first 3D biped, MARLO, the legs, and hence the actuated coordinates in the sagittal plane, were associated with four-bar linkages, which gave rise to synthesizing coordinates that were associated with a virtual leg connecting the hip to the end of the leg [12]. MARLO also had a rather tall torso that provided adequate inertia about the roll axis so that adjustments by the stance leg hip motor to maintain the torso roll angle approximately zero would not cause oscillations on the swing leg roll angle.

For the land-bird-inspired Cassie, if one is not a biologist, the actuated joints have limited physical meaning with respect to the walking behavior of the robot. Hence, we choose instead to regulate torso orientation, stance and swing leg lengths, swing leg orientation, and swing foot pitch angle. We find these quantities to be rather universal across bipedal platforms (whether human or bird inspired) and directly relatable to gait outcomes. For example, leg lengths are directly related to the height of the torso and foot clearance; swing leg pitch and roll angles at impact are commonly used in bipeds to regulate walking speed, and the yaw angle of the swing leg sets the direction of the robot for the next step. Moreover, because Cassie is roughly a basketball-sized sphere with one meter-long legs attached to it, its (spherical) torso provides little "mechanical filtering" between the motion of the stance and swing legs.

With this in mind, we define the following nine outputs,

with the stance foot pitch angle left passive[1],

$$h_0(q) = \begin{bmatrix} q_{\text{roll}} \\ q_{2 \text{ st}} \\ q_{\text{pitch}} \\ q_{\text{LL st}} \\ q_{\text{LR sw}} \\ q_{2 \text{ sw}} \\ q_{\text{LP sw}} \\ q_{\text{LL sw}} \\ q_{\text{FP sw}} \end{bmatrix} \begin{pmatrix} \text{torso roll} \\ \text{stance hip yaw} \\ \text{torso pitch} \\ \text{stance leg length} \\ \text{swing leg roll} \\ \text{swing hip yaw} \\ \text{swing leg pitch} \\ \text{swing leg length} \\ \text{swing foot pitch} \end{pmatrix}, \qquad (14)$$

where, when the right leg is stance and the left leg is swing,

$$q_{\text{LL st}} = \sqrt{0.5292 \cos(q_{4R} + 0.035) + 0.5301}$$
$$q_{\text{LR sw}} = q_{\text{roll}} + q_{1L}$$
$$q_{\text{LP sw}} = -q_{\text{pitch}} + q_{3L} \qquad (15)$$
$$\quad - \arccos\left(\frac{0.5(\cos(q_{4L} + 0.035) + 0.5292)}{\sqrt{0.5292 \cos(q_{4L} + 0.035) + 0.5301}}\right) + 0.1$$
$$q_{\text{LL sw}} = \sqrt{0.5292 \cos(q_{4L} + 0.035) + 0.5301}$$
$$q_{\text{FP sw}} = -q_{\text{pitch}} + q_{7L} + 1.1.$$

The forward kinematics of leg length and leg pitch are calculated based on configuration of the robot and shown in Fig. 3b. For clarity, leg pitch refers to the absolute pitch angle of the virtual leg when torso roll and yaw are zero; leg roll is defined in a similar manner.

For later use in control implementation, we note that the to-be-regulated quantities in (15) can be expressed in terms of the *actuated joints* via

$$q_1 = q_{\text{LR}} - q_{\text{roll}}$$
$$q_3 = q_{\text{LP}} + q_{\text{pitch}}$$
$$\quad + \arccos\left(\frac{0.9448 \, q_{\text{LL}}^2 - 0.0284}{q_{\text{LL}}}\right) - 0.1$$
$$q_4 = \arccos\left(1.8896 \, q_{\text{LL}}^2 - 1.0017\right) - 0.035 \qquad (16)$$
$$q_7 = q_{\text{FP}} + q_{\text{pitch}} - 1.1,$$

where the distinction between stance and swing has been dropped.

C. Set of Gaits for Walking at Various Speeds

The desired evolution of the virtual constraints is defined by h_d in the output equation (12). This function is constructed using linear interpolation of a discrete library of gaits, each encoding a particular forward walking speed. Here, seven gaits were generated where the average velocity in the sagittal plane, $\bar{v}_x$, ranged from -0.5 m/s to +1.0 m/s in 0.25 m/s increments. We assume that the virtual constraints of each of these "open-loop" gaits have a desired trajectory, $h_d^i(\tau, \alpha_i)$, that is parameterized by a set of 5th-order Bézier polynomials with the corresponding matrix of coefficients denoted as α_i. The step time for all gaits was chosen to be a constant. Trajectory optimization is then used to independently solve for each α_i.

---

[1]For now, it is simply observed that leaving it passive avoids "foot roll"; when walking on soft sand, we will come back to this feature.

The nonlinear optimization problems were constructed and solved using FROST [14], which internally uses the direct-collocation trajectory optimization framework developed by Hereid et. al. [15]. Each hybrid optimization was performed over two domains (right stance then left stance), where the following cost function was minimized:

$$\text{Domain Cost} = \int_{\tau=0}^{\tau=1} \left(||u||^2 + c\,|q_{pitch}|^2 + c\,|q_{roll}|^2 \right.$$
$$\left. + c\,|q_{1\mathrm{L}}|^2 + c\,|q_{2\mathrm{L}}|^2 + c\,|q_{1\mathrm{R}}|^2 + c\,|q_{2\mathrm{R}}|^2 \right) d\tau.$$

The addition of the torso pitch/roll and the hip roll/yaw angles into the cost function (multiplied by a large weight, $c = 10,000$) guides the optimizer to find gaits with minimal roll and yaw movement. Constraints are placed on the optimization problem to ensure that the optimized gait is periodic over two steps and that the left and right stance are symmetric[2]. Torque, joint angle, and joint velocity limits were imposed to ensure that the gait can be physically-realized on the actual robot. Additional constraints are outlined in Table I.

Average sagittal velocity, $\bar{v}_x$	$= v_i$ m/s
Average lateral velocity, $\bar{v}_y$	$= 0$ m/s
Step time	$= 0.4$ s
Torque for stance foot pitch	$= 0$ Nm
Friction cone, μ	< 0.6
Mid-step swing foot clearance	> 0.15 m
Absolute swing foot pitch	$= 0$ rad
Distance between feet	> 0.2 m
Distance between pelvis and stance foot	$\in (0.5, 1)$ m
Swing foot velocity on impact (x and y)	$= 0$ m/s
Swing foot velocity on impact (z)	$\in (-1, 0)$ m/s

TABLE I: Constraints used in gait optimizations. For each of the seven optimizations, the average sagittal velocity was constrained to a different value between -0.5 and 1 m/s.

Each of the 7 optimization problems yields a single parameter matrix, α_i, and takes approximately 3 min to solve using IPOPT in MATLAB.

Remark: *A C++ implementation of the optimization problem formed by FROST has been posted on GitHub [16]. It allows parallel computation of the gaits and cloud-based gait optimization.*

D. Approximately Implementing the Virtual Constraints

If the overall dynamic model and joint angular velocity estimates were sufficiently accurate, we could implement the virtual constraints via input-output linearization. Indeed, the outputs (12) have relative degree two [19] and the row rank of the decoupling matrix is full rank on the control-design model.

On the actual robot, however, the power amplifiers, motor dynamics, network delays, and walking surface are not

[2]Due to the enforced symmetry, it is possible to write this optimization problem using a single domain. However, the general two-step formulation allows for future design of non-symmetric gaits through simple removal of this particular constraint.
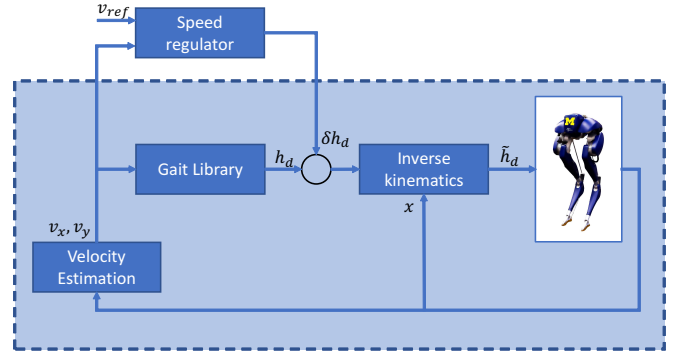

Fig. 4: Control Diagram for Walking. The feedback loop implementing the virtual constraints and the gait library (blue box) maintains the robot's posture and synchronizes the legs for walking. Moreover, the resulting closed-loop system renders the dynamics of the center of mass velocity close enough to that of an inverted pendulum that it can be regulated by adjusting the pitch and roll angles of the swing leg.

adequately characterized to allow model-based torque control (the mechanical model itself is not the main source of uncertainty). Consequently, the virtual constraints are approximately imposed through decoupled PD controllers, as in [12], [24], [32]. To do this, (16) is used to rewrite (12) as

$$\widetilde{y} = \widetilde{h}_0(q) - \widetilde{h}_d(\tau, q_{\mathrm{pitch}}, q_{\mathrm{roll}}, \alpha), \qquad (17)$$

with (right leg in stance)

$$\widetilde{h}_0(q)^\top = [q_{\mathrm{roll}}, q_{2\mathrm{R}}, q_{\mathrm{pitch}}, q_{4\mathrm{R}}, \qquad (18)$$
$$q_{1\mathrm{L}}, q_{2\mathrm{L}}, q_{3\mathrm{L}}, q_{4\mathrm{L}}, q_{7\mathrm{L}}]^\top$$

and

$$\widetilde{h}_d(\cdot) = \begin{bmatrix} h_{d\ 1}(\cdot) \\ h_{d\ 2}(\cdot) \\ h_{d\ 3}(\cdot) \\ \arccos\left(1.8896\ [h_{d\ 4}(\cdot)]^2 - 1.0017\right) - 0.035 \\ h_{d\ 5}(\cdot) - q_{\mathrm{roll}} \\ h_{d\ 6}(\cdot) \\ h_{d\ 7}(\cdot) + q_{\mathrm{pitch}} + 0.1 \\ \quad + \arccos\left(\frac{0.9448\ (h_{d\ 8}(\cdot))^2 - 0.0284}{h_{d\ 8}(\cdot)}\right) \\ \arccos\left(1.8896\ [h_{d\ 8}(\cdot)]^2 - 1.0017\right) - 0.035 \\ h_{d\ 9}(\cdot) + q_{\mathrm{pitch}} - 1.1 \end{bmatrix}.$$

$$(19)$$

Remark: *Though not proven here, (17) implements the same virtual constraints as (12) and (15); one is zero if, and only if, the other is zero.*

In (18) and (19), the outputs are ordered so that they correspond to the first four actuators on the stance leg followed by the five actuators on the swing leg. Recall that the torque on the stance foot is set to zero. The virtual constraints are approximately zeroed with a classical PD controller,

$$u = -K_P \widetilde{y} - K_D \dot{\widetilde{y}}, \qquad (20)$$

where the 9×9 matrices K_P and K_D are diagonal.

Remark: *The torso pitch and roll angles are world frame coordinates. When the stance foot is firmly on the ground (i.e., not slipping), they can be controlled through the hip motors. The remaining outputs are directly actuated. When the springs are ignored, the output has vector relative degree two.*

E. Gait Library and Stabilization by Leg Angle Adjustment

The *Gait Library* is an interpolation of the seven discrete gaits into a continuum of gaits valid for $-0.5 \leq v_x \leq 1.0$ m/s. The interpolation parameter is the robot's filtered sagittal velocity. The implementation of the gait library is done exactly as in [9, Eq. (8)-(10)] and does not introduce any new parameters into the controller.

With the gait library implemented, the closed-loop system is unstable in the sense that the $(\dot{x}, \dot{y})$ Cartesian coordinates are approximate integrators; see [9, Sect. III-C] for the explanation. Leg angle adjustment is added to stabilize the closed-loop system. The implementations for longitudinal and lateral velocity stabilization are based on [9, Eq. (13) and (17)]. These controllers add four more control parameters. The overall control strategy is shown in Fig. 4.

F. Parameter Tuning

The right and left legs of the robot are sufficiently symmetric that control parameters for the left and right legs are the same. The controller was implemented in `Real-Time Simulink` and the parameters tuned by hand on a `SimMechanics` model provided to us by Agility Robotics. A process for tuning the 18 joint-level PD parameters and the 4 leg-angle PD parameters on the robot is posted with the code on GitHub.

G. Experiments

A first version of the walking controller was implemented on Cassie Blue six weeks after arrival on campus and was demonstrated to the Associated Press (AP) on October 23rd, 2017 [1]. On June 2, 2018, we damaged a leg on Cassie Blue and sent her back for repairs. While the robot was in the shop, Agility Robotics upgraded the hip roll and yaw joints to match those on their current production model, significantly reducing friction in them. At the same time, Agility also modified the MATLAB environment in which a user's controller is implemented, breaking our controller. Because we would be soon modifying the robot with the addition to a 15 kg torso, we did not spend much time retuning the controller.

The remainder of the section discusses some of the many terrains on which the robot has been challenged to operate over the past 11 months as documented in [5]. The dates of the experiments are noted below. In each experiment, the robot is being directed by an operator via an RC Radio with commands "stand quietly" or "walk". When walking, the robot is sent desired $\bar{v}_x^{tg}$, $\bar{v}_y^{tg}$, and turn rate.

1) Initial Testing in the Laboratory: After the controller was successfully working in closed-loop with the `SimMechanics` model, it was transferred to the robot and the PD gains tuned over a period of a few days. During this process, an overhead gantry was used to catch the robot in case of a fall. The gains were initially tuned for walking in place. Once that milestone was achieved, walking at various speeds came quickly. A typical limit cycle is shown in Fig. 5.

2) North Campus Grove: Cassie Blue was taken outdoors for the first time and demonstrated to the Associated Press [1] on 23 October 2017. Initially, a safety gantry was used. After walking on a sidewalk with no difficulty, Cassie was released from the gantry. After initial walking on a level concrete area, Cassie was steered onto the grass. Due to a nervous operator hitting the wrong button on the RC-controller, the robot sped up and had to be stabilized by a researcher. The softer nature of the grass caused no difficulties in detecting foot impacts via spring deflection. For the next 11 minutes, the robot was steered on and off grass and hard surfaces, onto sloped grass surfaces, turned, and walked in place. When the (unplanned) trajectory brought the robot and researchers near a grassy knoll, the decision was made to see if the robot could handle it. The robot unexpectedly sped up when heading up the slope, leaped over a bench, and fell at least 1.5 m onto the concrete. This put a slight dent in the battery, but caused no other observable damage as the robot was quickly rebooted and walked in place. This ended the experiment.

3) Waxed Floors and Snow: Cassie Blue was taken to the UM Dental School on 11 December 2017 at the invitation of Dean Laurie McCauley. The video can be found here [2]. The robot handled well on surfaces with reduced friction. An unplanned bump into the pillar and walking in the snow are shown here [5].

4) Controlled Burn for Wild Grasses: On 22 April 2018, a controlled burn was conducted on the UM campus to promote the growth of native grasses. After clearing it with the personnel conducting the burn, Cassie Blue walked over sloped ground, in heavy smoke, and over short burning grass, branches, and leaves [6]. The robot never fell. While the exercise demonstrated our general confidence in the robot's controller, it was done to drive home the fact that a battery-powered robot does not suffer from smoke inhalation and can take some heat. We'll return in 2019 with a full perception
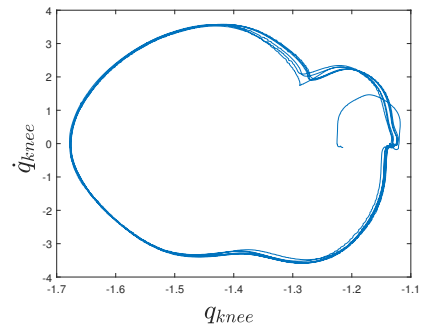

Fig. 5: The phase portrait of the left knee when Cassie is walking in place, starting from a standing position. The units are rad and rad/s.
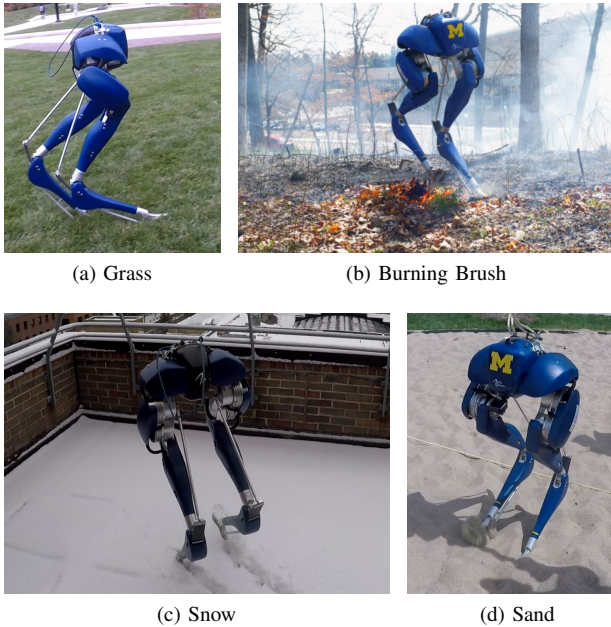
6

(a) Grass

(b) Burning Brush

(c) Snow

(d) Sand

Fig. 6: Cassie Blue walking on various unmodeled terrains.

package on the robot.

5) Sand Volley Ball Court: On 09 May 2018, the Discovery Channel filmed Cassie Blue. The Segway riding, reported later, was their main interest. Since we were near the sand volley ball court, we challenged the robot to walk on it [4]. The narrow feet sunk into the sand, with the "heel" digging in the most. Because the stance foot is passive, the robot's gait remained quite stable. The robot walked more slowly than on grass (possibly due to foot slip) and it traversed the entire course, passing under the net. For the second pass, a pair of tennis shoes was placed on the robot. Cassie then walked no differently than when on grass or concrete.

IV. Quiet Standing and Riding a Segway

For standing, all ten actuators are used. Foot actuation is required to prevent rotation of the robot about the y-axis in the body frame. The standing controller also allows Cassie to ride a Segway; yes, a robot riding another robot.

A. Quiet Standing

The standing condition is here assumed to be reached either from a stepping-in-place gait or by a user booting up the robot with the torso suspended approximately a half meter off the ground. The feet are assumed to be flat on the ground and beneath the torso.

Let $p^{\mathrm{CoM}} = (p_x^{\mathrm{CoM}}, p_y^{\mathrm{CoM}}, p_z^{\mathrm{CoM}})$ be the center of mass of the robot. With the feet flat on the ground and $(p_x^{\mathrm{CoM}}, p_y^{\mathrm{CoM}})$ within the convex hull of the feet, the robot can maintain a static pose. To maintain the desired center of mass position, p_x^{CoM} is regulated with the actuators for the pitch angle of the feet, while p_y^{CoM} is regulated in an indirect way: the roll angle, instead of p_y^{CoM}, is controlled to be zero. The roll is controlled by adjusting the leg length difference in the two legs. A zero roll angle is equivalent to a centered p_y^{CoM} when all other joints are symmetric. One reason we
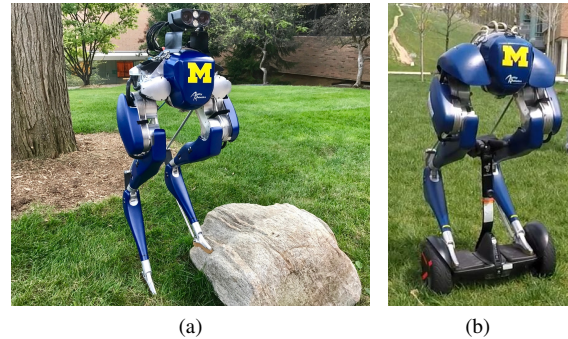

(a)

(b)

Fig. 7: (a) Cassie standing on an uneven surface. (b) A stack of robots. Cassie is riding a Segway miniPro. The Segway will accelerate forward if the foot platform leans forward. Turning is controlled by pushing against the central vertical bar.

are using roll angle for feedback is because its value is less noisy than p_y^{CoM}, which is calculated via a kinematic chain. Another reason is that if the ground is sloped in the frontal plane, causing the robot to lean to the right[3], the right leg will automatically be extended and the left leg compressed, as shown in Fig. 7. The height of the standing pose p_z^{CoM} is set by adjusting the average of the two leg lengths. The hip roll and yaw joints are regulated to constant values. With this controller, and the two feet roughly 0.3 m apart, Cassie is able to squat almost flat on the ground and stand to a height of approximately one meter. Quiet standing, lowering to a squat, and standing back up are illustrated in the video associated with the paper [5].

B. Riding a Segway

Figure 7 shows Cassie Blue riding a Segway miniPro (robot). The dynamics of the Segway are unknown and its states are not measured. The acceleration and direction of the Segway are determined by body lean, that is, by adjusting the target center of mass position. As elsewhere, the commands are sent by an operator via radio control.

To accelerate or decelerate, p_x^{CoM} is shifted forward or backward, respectively. To turn, Cassie needs to lean into the center bar with her legs, which is accomplished by shifting p_y^{CoM}. With the nominal standing controller, the Segway would oscillate when Cassie was placed on its platform. The feedback gains on the feet were turned down and the oscillations ceased.

With the crouched posture seen in Fig. 7, Cassie Blue was able to ride on sidewalks and grassed areas at roughly 4 m/s [3]. To be clear, the robot was placed on the Segway by an operator. Mounting and dismounting the Segway were not addressed.

V. Conclusion

The Cassie biped has the morphology of a large flightless bird. The full 20 DoF dynamic model and optimization

---

[3]This feature is useful even on flat ground. The PD control of leg length should be thought of as soft springs. A bit of lean to the right places more weight on the right leg, which causes further compression of the spring in the right leg, which causes further leaning, etc., until p_y^{CoM} moves to the right of the foot and the robot falls. With high PD gains, this cascading effect can be avoided, but regulating p_y^{CoM} solves the problem with lower gains.

were used to design seven gaits for walking in place, forward, and backward, while meeting key physical constraints. The complicated morphology of the robot was translated into "universal," physically meaningful control objectives involving torso orientation, leg orientation, and leg length. Moreover, it was shown how to practically implement these control objectives via decoupled PD controllers on the robot.

The gait-library method of [9] was used to gain-schedule the discrete gaits into a continuum of walking gaits. The final controller was demonstrated both in and out of the laboratory, including walking on sidewalks, grass, sand, waxed floors, snow, and a hill with short brush [5]. The control software will be made open-source if the paper is accepted for publication.

In terms of next steps, we are working on a data-driven method to design feed forward torques to improve tracking performance. We are also seeking gaits that allow fast motion with turning, perhaps using the method introduced in [8].

ACKNOWLEDGMENT

REFERENCES

[1] Cassie Blue and Associated Press. https://youtu.be/h6hnfCo4a00. Accessed: 2018-09-12.

[2] Cassie Blue at the UM Dental School. https://youtu.be/jhQMq6vpnAo. Accessed: 2018-09-12.

[3] Cassie Blue Hones Her Segway Riding Skills. https://youtu.be/0gauVSUJzd0. Accessed: 2018-09-12.

[4] Cassie Blue Plays in the Sand. https://youtu.be/VoD7hbssu-M. Accessed: 2018-09-12.

[5] Cassie Blue Summary Video. https://youtu.be/UhXly-5tEkc. Accessed: 2018-09-14.

[6] Cassie Blue Walks Through Fire. https://youtu.be/cGb3bE6ZwrQ. Accessed: 2018-09-12.

[7] Carlos Canudas-de-Wit. On the concept of virtual constraints as a tool for walking robot control and balancing. *Annual Reviews in Control*, 28(2):157–166, 2004.

[8] Xingye Da and Jessy W Grizzle. Combining Trajectory Optimization, Supervised Machine Learning, and Model Structure for Mitigating the Curse of Dimensionality in the Control of Bipedal Robots. *arXiv preprint arXiv:1711.02223*, 2017.

[9] Xingye Da, Omar Harib, Ross Hartley, Brent Griffin, and Jessy W Grizzle. From 2D design of underactuated bipedal gaits to 3D implementation: Walking with speed tracking. *IEEE Access*, 4:3469–3478, 2016.

[10] Hongkai Dai and Russ Tedrake. Planning robust walking motion on uneven terrain via convex optimization. In *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*, pages 579–586. IEEE, 2016.

[11] Salman Faraji, Soha Pouya, Christopher G Atkeson, and Auke Jan Ijspeert. Versatile and robust 3D walking with a simulated humanoid robot (Atlas): a model predictive control approach. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1943–1950. IEEE, 2014.

[12] Brent Griffin and Jessy Grizzle. Nonholonomic virtual constraints and gait optimization for robust walking control. *The International Journal of Robotics Research*, page 0278364917708249, 2016.

[13] Ross Hartley. Cassie URDF and FROST Model. https://github.com/UMich-BipedLab/Cassie_Model, 2017.

[14] Ayonga Hereid and Aaron D Ames. Frost: Fast robot optimization and simulation toolkit. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, pages 719–726. IEEE, 2017.

[15] 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.

[16] Ayonga Hereid, Omar Harib, Ross Hartley, Yukai Gong, and Jessy W Grizzle. Rapid bipedal gait design using c-frost with illustration on a cassie-series robot. *arXiv preprint arXiv:1807.06614*, 2018.

[17] Kazuo Hirai, Masato Hirose, Yuji Haikawa, and Toru Takenaka. The development of honda humanoid robot. In *Robotics and Automation, 1998. Proceedings. 1998 IEEE International Conference on*, volume 2, pages 1321–1326. IEEE, 1998.

[18] Qiang Huang, Shuuji Kajita, Noriho Koyachi, Kenji Kaneko, Kazuhito Yokoi, Hirohiko Arai, Kiyoshi Komoriya, and Kazuo Tanie. A high stability, smooth walking pattern for a biped robot. In *Robotics and Automation, 1999. Proceedings. 1999 IEEE International Conference on*, volume 1, pages 65–71. IEEE, 1999.

[19] A. Isidori. *Nonlinear Control Systems*. Springer-Verlag, Berlin, third edition, 1995.

[20] 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 *ICRA*, volume 3, pages 1620–1626, 2003.

[21] Kenji Kaneko, Fumio Kanehiro, Mitsuharu Morisawa, Kazuhiko Akachi, Go Miyamori, Atsushi Hayashi, and Noriyuki Kanehira. Humanoid robot HRP-4 – humanoid robotics platform with lightweight and slim body. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 4400–4407. IEEE, 2011.

[22] T. Koolen, T. de Boer, J. Rebula, A. Goswami, and J. Pratt. Capturability-based analysis and control of legged locomotion, Part 1: Theory and application to three simple gait models. *The International Journal of Robotics Research*, 31(9):1094–1113, July 2012.

[23] 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.

[24] Hae-Won Park, Alireza Ramezani, and JW Grizzle. A finite-state machine for accommodating unexpected large ground-height variations in bipedal robot walking. *IEEE Transactions on Robotics*, 29(2):331–345, 2013.

[25] J. Pratt, T. Koolen, T. de Boer, J. Rebula, S. Cotton, J. Carff, M. Johnson, and P. Neuhaus. Capturability-based analysis and control of legged locomotion, Part 2: Application to M2V2, a lower-body humanoid. *The International Journal of Robotics Research*, 31(10):1117–1133, August 2012.

[26] Jerry E. Pratt and Russ Tedrake. Velocity based stability margins for fast bipedal walking. In *Proceedings of the First Ruperto Carola Symposium on Fast Motions in Biomechanics and Robotics: Optimization and Feedback Control. Heidelberg, Germany*, September 2005.

[27] 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 conference*, pages V001T04A003–V001T04A003. American Society of Mechanical Engineers, 2015.

[28] Siavash Rezazadeh, Christian Hubicki, Mikhail Jones, Andrew Peekema, Johnathan Van Why, Andy Abate, and Jonathan W Hurst. Spring-mass walking with ATRIAS in 3D: Robust gait control spanning zero to 4.3 kph on a heavily underactuated bipedal robot. *ASME Dynamic Systems and Control Conference (DSCC)*, page 23, 2015.

[29] Agility Robotics. Cassie Simulators. http://www.agilityrobotics.com/sims/, 2018.

[30] Alexander Schepelmann, Yin Zhong, Jessica Austin, Kathryn A Geberth, and Hartmut Geyer. Experimental evaluation of robust swing-leg placement controls in robotic limb testbeds. *preprint, Sept. 8, 2018*.

[31] Miomir Vukobratovic and Branislav Borovac. Zero-moment point: Thirty five years of its life. *International Journal of Humanoid Robotics*, 01(01):157–173, 2004.

[32] E. R. Westervelt, G. Buche, and J. W. Grizzle. Experimental validation of a framework for the design of controllers that induce stable walking in planar bipeds. *International Journal of Robotics Research*, 24(6):559–582, June 2004.

[33] Eric R Westervelt, Christine Chevallereau, Jun Ho Choi, Benjamin Morris, and Jessy W Grizzle. *Feedback control of dynamic bipedal robot locomotion*. CRC press, 2007.

[34] Xiaobin Xiong and Aaron Ames. Bipedal hopping: Reduced-order model embedding via optimization-based control. *arXiv preprint arXiv:1807.08037*, 2018.