Dynamic Humanoid Locomotion: A Scalable Formulation for HZD Gait Optimization

Ayonga Hereid , *Member, IEEE*, Christian M. Hubicki, *Member, IEEE*, Eric A. Cousineau, *Member, IEEE*, and Aaron D. Ames , *Senior Member, IEEE*

Abstract—Hybrid zero dynamics (HZD) has emerged as a popular framework for dynamic walking but has significant implementation difficulties when applied to the high degrees of freedom humanoids. The primary impediment is the process of gait design—it is difficult for optimizers to converge on a viable set of virtual constraints defining a gait. This paper presents a methodology that allows for fast and reliable generation of dynamic robotic walking gaits through the HZD framework, even in the presence of underactuation. Specifically, we describe an optimization formulation that builds upon the novel combination of HZD and direct collocation methods. Furthermore, achieving a scalable implementation required developing a defect-variable substitution formulation to simplify expressions, which ultimately allows us to generate compact analytic Jacobians of the constraints. We experimentally validate our methodology on an underactuated humanoid, DURUS, a spring-legged machine designed to facilitate energy-economical walking. We show that the optimization approach, in concert with the HZD framework, yields dynamic and stable walking gaits in hardware with a total electrical cost of transport of 1.33.

A. Hereid is with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: ayonga@ umich.edu).

C. M. Hubicki is with the Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332 USA (e-mail: hubicki@ gatech.edu).

E. A. Cousineau is with the Toyota Research Institute, Cambridge, MA 02139 USA (e-mail: eric.cousineau@tri.global).

A. D. Ames is with the Mechanical and Civil Engineering Department, California Institute of Technology, Pasadena, CA 91125 USA (e-mail: ames@ caltech.edu).

I. INTRODUCTION

HUMANOID robots have long held the promise of walking around in the human world the dynamic way that people walk. While humans and other biological bipeds can perform these motions with relative ease, translating such dynamic behaviors to three-dimensional (3-D) humanoids is a challenging task. The rapid development of mechanical and actuation capabilities of modern robots has already made more dynamic locomotion possible. The faster and more nimble we demand these machines to be, the more the robot needs to reason about its full-order dynamics. It can be helpful to tilt on the edges of its feet for toe-off or heel-strike maneuvers, exploit soft compliant linkages for impact reduction, or embrace the underactuated dynamics of falling forward to the advantage of locomotion. Due to the nonlinearities and high degrees of freedom of the multibody systems, however, planning dynamic motion that reconciles the full body dynamics of the complex robot model has been primarily prevented by a particular computational bottleneck: gait synthesis.

A. Related Works

Many existing methods for planning humanoid locomotion typically use simplified reduced-order models as a basis to mitigate the complexity of the full-order gait planning operation. These approaches plan trajectories for a low-dimensional dynamical model that approximates the full-order robot dynamics by enforcing specific kinematic constraints. For instance, the arguably most commonly used linear inverted pendulum model assumption requires a constant center of mass (COM) velocity in the vertical direction and typically fully actuated systems. By enforcing the zero moment point (ZMP) criteria, which requires the ZMP position always rest within the support polygon of the robot feet [1], motion generators can plan the COM position to ensure fall-free motions [2]–[4], or quickly compute recovery steps [5], [6]. The whole body motions are then generated via inverse kinematics or inverse dynamics techniques by conforming the robot to these analytically tractable models. The simple, often linear, form of system dynamics expedites planning the walking gait online to accommodate the changes in the surrounding environment [7], [8]. The maturity and reliability of the simplified model based planning made it a prevalent component in control approaches at the DARPA Robotics

Challenge [9]–[11]. While these model-simplification methods have many implementation advantages, they also limit the flexibility of the walking behaviors, often resulting in a rather artificial motion when compared to natural human walking. Furthermore, these methods typically require a fully actuated foot design, a property not present in some of the most agile [12], robust [13], or efficient [14] bipedal robots.

Recently, the development of advanced optimization techniques permits the use of more complicated dynamic models as the foundation of gait planning [15], [16]. In particular, Posa *et al.* have successfully generated multicontact walking gaits for a simulated model of the 3-D ATLAS robot using direct collocation-based constrained whole body dynamic optimization [17]. Similar optimization-based approaches have also been deployed to generate versatile and dynamic motions by using the machine's centroidal dynamics and whole-body contact forces as a foundation for humanoid planning [18], [19]. In those applications, the whole body dynamics are represented by the 6-D COM dynamics using the augmented linear and angular momentum at the COM due to the ground contact forces, while joint torques are assumed away as internal forces of the system [6]. Furthermore, researchers have started to consider the contact wrench sum (CWS) criteria as the stable balancing condition instead of the traditional ZMP approach to realize more versatile locomotion. It requires that the CWS must lie within the contact wrench cone to keep the dynamic balance of the robot [20]. The stability criterion is checked by examining if the CWS is equal to the rate of change in the linear and angular momentum of the robot. This condition is coupled with the COM dynamics to formulate a nonlinear constrained optimization problem to generate more dynamic gaits for full-order humanoids [10], [21].

On the other end of the spectrum, the hybrid zero dynamics (HZD) control framework is designed to mathematically support stable control of dynamic maneuvers of hybrid dynamical systems, such as bipedal locomotion [22], [23]. HZD defines the gait by designing a set of *virtual constraints* enforced via feedback control of the actuated joints. If these virtual constraints are invariant through impact, all of the stability properties of the high-dimensional system are captured in a lower-dimensional representation, termed as the *HZD*, without making any simplification assumptions of the model. Since its inception, HZD has built a strong history of success in planar robot implementations for bipedal walking [24]–[29] and running [30], [31]. Recent work has begun expanding the method into 3-D applications with highly underactuated robots [32], [33]. Robust and stable walking gaits were realized using systematic virtual constraints optimization approaches to accommodate the unknown variation of terrain heights and lateral balancing of point-feet 3-D bipeds [34], [35].

B. Problem Statement

This vital task of finding an appropriate set of virtual constraints (and parameters thereof) is typically relegated to a nonlinear optimization problem. When robots have as many linkages as humanoids, optimizing motions that meet HZD criteria becomes increasingly tricky for nonlinear programming (NLP)

tools to solve. This has been a significant impediment toward applying HZD to full humanoid robots, which have far more degrees of freedom than planar bipeds. In a broader view, this is where the HZD approach has had to pay the piper for its admissibility of highly dynamic gaits. Traditional HZD gait optimization approaches optimize only parameters and boundary state values, reflecting an instinctive desire to minimize the number of design variables for the optimization [36], [37]. Intuitively, one might assume that such minimization of the NLP problem's dimensionality would be an advisable practice for maximizing an optimization's speed and reliability. However, this formulation is prone to many issues, such as nonsmooth approximations of the constraint Jacobian [38] and the "tail wagging the dog" phenomenon [39], which can introduce pseudominima or merely cause the algorithm to fail to find a solution. Given the nonlinearity of bipedal robot dynamics, it can be difficult to achieve reliable convergence via this optimization and often relies on expert users to seed it.

With a goal of removing the limitations of applying the HZD control framework to high-dimensional humanoids, our approach unifies virtual constraint optimization with a technique from the trajectory optimization community: *direct collocation* [40], [41]. A direct collocation formulation represents both the time-varying states and inputs as parameterized curves, where system dynamics are enforced as equality constraints (called *defect constraints*) in a nonlinear program. By eschewing the need for time-marching integration schemes (as per shooting methods) in favor of these local defect constraints, open the possibility of expressing all optimization constraints in closed form. Fully algebraic constraint expressions allow for symbolic Jacobians with fast evaluation times and high accuracy, which is critical for the scalability of the gait optimization. By necessity, this paper also takes care in addressing the scalability of constraint expression sizes, which can quickly explode to intractable proportions. Specifically, we present our formulae by systematically introducing *defect variables* into the constraints that avoid symbolically verbose operations, keeping the resulting expressions simple. Furthermore, by carefully indexing the optimization variables and constraints, we further simplify the Jacobian matrix to have a banded structure, enabling efficient evaluation and use by standard large-scale NLP solvers. All such components of this process were crucial in engineering fast and reliable optimizations that synthesize HZD gaits for humanoid robots.

As validation of this approach, we use this computationally scalable framework to DURUS, a 23-DOF spring-legged humanoid robot. DURUS is designed for energy-economical locomotion, including soft distal springs at the ankles to absorb hard impacts. This significant non-joint-collocated compliance serves as an appropriate platform for testing dynamic gaits. To show the flexibility of the synthesis process (illustrated in Fig. 1), we present a set of hardware experiments demonstrating planar multicontact "heel-toe" walking and 3-D flat-footed dynamic locomotion on DURUS. We achieved sustained 3-D walking that continues for hours with a single battery pack without falling. We further report this energy-optimized gait as having high-energy economy in the experiment (COT: 1.33), demonstrated that we can successfully control the robot hardware while
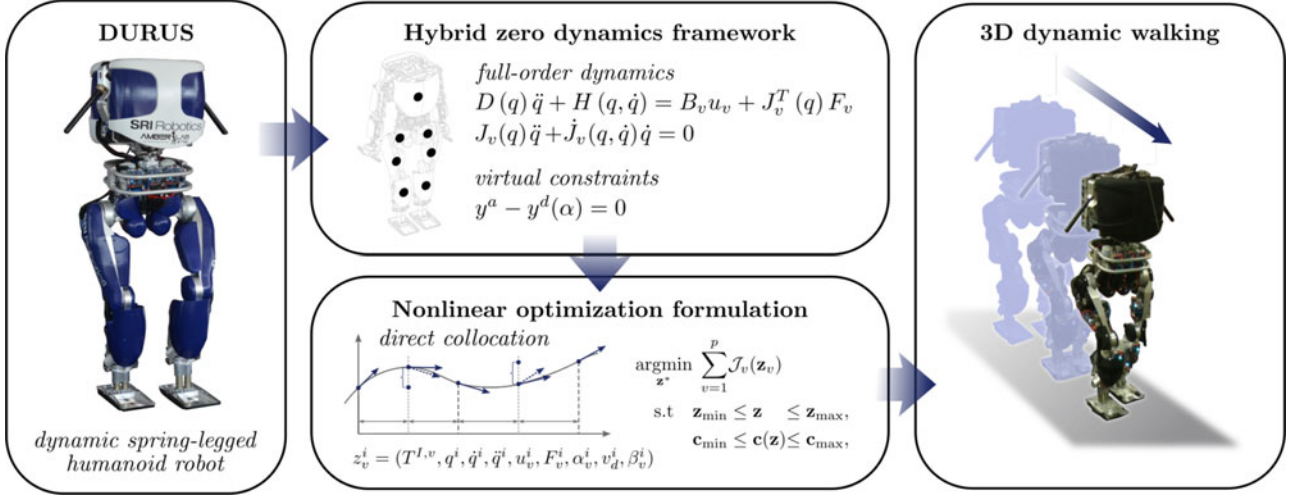
Fig. 1. Illustration of the process used to generate dynamic 3-D walking with DURUS. This direct collocation framework parses a multibody model of the robot and a set of virtual constraints into a large and sparse NLP with upwards of 10 000 design variables and constraints. Large-scale algorithms typically solve this NLP in under 10 min, thereby optimizing a dynamic gait for the 3-D humanoid that exploits the full body dynamics of the machine, despite underactuation.

still taking advantage of its energy-conscious underactuated design.

A preliminary presentation of this paper was outlined in [42], but with a less-complete description of the formulation and fewer validating experiments. This manuscript elaborates on the total generalized framework, as well as more specific formulation that allow for fast execution (such as indexing methods). This paper further supports the scalability and generality of the approach through application to additional multidomain bipedal locomotion cases (e.g., heel-toe walking).

The structure of this paper is as follows. Section II reviews the formal definition of the general multidomain hybrid control system, and Section III introduces the design of virtual constraints for bipedal locomotion. In Section IV, we propose a virtual constraints optimization framework based on direct collocation methods. Section V uses the proposed framework to design two different types of walking for DURUS and generates energy-efficient gaits for each case respectively. Experimental results of two different walking gaits on DURUS are presented in Section VI, including sustained 3-D walking, followed by the discussion and conclusion in Section VII.

II. BIPEDAL LOCOMOTION AS HYBRID SYSTEMS

Bipedal locomotion consists of a collection of continuous phases (or domains) with discrete events triggering transitions between these continuous phases; formally modeling this interplay of continuous and discrete dynamics results in a multidomain hybrid system model.

A. Formal Definition of Multidomain Hybrid Systems

During a steady-state walking gait, the transitions between different phases become ordered and periodic; this motivates the use of a multidomain hybrid system with a predetermined ordering of domains as represented by a *directed cycle*, i.e., a cyclic directed graph.
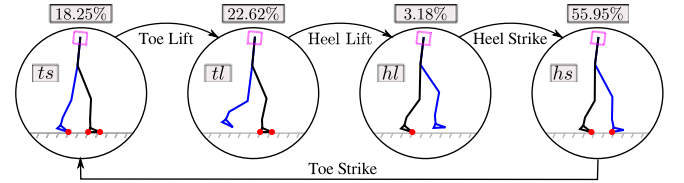

Fig. 2. Typical periodic human walking pattern can be represented as a *directed cycle* of four discrete domains with different contact conditions (red circles in the figure).

Definition 1: A *directed cycle* is a directed graph $\Gamma = (V, E)$, with $V = \{v_1, v_2, \ldots, v_{n_p}\}$ a set of vertices and $E = \{e_1 = (v_1 \rightarrow v_2), e_2 = (v_2 \rightarrow v_3), \ldots, e_{n_p} = (v_{n_p} \rightarrow v_1)\}$ a set of edges. Let $\mathrm{sor} : E \rightarrow V$ and $\mathrm{tar} : E \rightarrow V$ be the maps that determine the source vertex and target vertex of an edge, respectively. In other words, any $e \in E$ can be represents as $e = \{\mathrm{sor}(e) \rightarrow \mathrm{tar}(e)\}$. For a directed cycle, sor and tar are one-to-one and onto. Hence, their inverse maps $\mathrm{sor}^{-1} : V \rightarrow E$ and $\mathrm{tar}^{-1} : V \rightarrow E$ exist and are well defined.

Example 1: In Fig. 2 , a four-domain directed cycle illustrates the domain structure of a typical human walking gait pattern that consists of four discrete phases depending on different contact conditions [43]. This directed cycle $\Gamma = (V, E)$ consists of four vertices and four edges:

$$V = \{\mathrm{ts, tl, hl, hs}\}$$

$$E = \{\mathrm{ts} \rightarrow \mathrm{tl}, \mathrm{tl} \rightarrow \mathrm{hl}, \mathrm{hl} \rightarrow \mathrm{hs}, \mathrm{hs} \rightarrow \mathrm{ts}\}.$$

Definition 2: A *hybrid control system* is a tuple

$$\mathscr{HC} = (\Gamma, \mathcal{D}, \mathcal{U}, S, \Delta, FG) \tag{1}$$

where $\Gamma = \{V, E\}$ is a *directed cycle*, $\mathcal{D} = \{\mathcal{D}_v\}_{v \in V}$ is a set of admissible *domains*, $\mathcal{U}$ is a set of admissible control inputs, $S = \{S_e\}_{e \in E}$ is a set of *guards* or *switching surfaces*, $\Delta = \{\Delta_e\}_{e \in E}$ is a set of *reset maps* that dictate the discrete transitions triggered at S_e, and $FG = \{FG_v\}_{v \in V}$ is a set of *control systems*

that determine the continuous dynamics of the system on a domain $\mathcal{D}_v$.

Utilizing the formal definition of multidomain hybrid systems, we have the framework necessary to discuss how the Lagrangian and contact constraints—such as foot contacts with the ground—of the mechanical system of a bipedal robot are used to determine each element of the hybrid system model.

B. Hybrid System Models for Bipedal Locomotion

In this section, we review the mathematical formulation of each element of the hybrid control system model for dynamic bipedal locomotion based on a generalized robot model. The multibody system of a robot is often modeled as a kinematic tree of rigid links. Motivated by the desire to consider robots in a generalized position, i.e., not impose assumptions on contact constraints, we use the floating base coordinates of a robot (see [44]). Let R_0 is a fixed inertial frame and R_b is a reference frame rigidly attached to the base link of the robot, then the Cartesian position $p_b \in \mathbb{R}^3$ of the origin and the orientation $\phi_b \in SO(3)$ of R_b with respect to R_0, respectively, compose the floating base coordinates of the robot. Let $q_r \in \mathcal{Q}_r$ be the joint coordinates of a robot, the floating-base generalized coordinates can be defined as

$$q = (p_b, \phi_b, q_r) \in \mathcal{Q} = \mathbb{R}^3 \times SO(3) \times \mathcal{Q}_r \subseteq \mathbb{R}^n \quad (2)$$

where n is the total degrees of freedom of the system.

Domains: A specific admissible domain is determined by the associated contact conditions. In this paper, we use holonomic constraints to model the robot's physical contacts with the external environment. With $\mathcal{C}_v$ as an indexing set of all holonomic constraints defined on $\mathcal{D}_v$, we state the holonomic constraints of this domain as

$$\eta_v = \{\eta_c\}_{c \in \mathcal{C}_v} \equiv \text{constant}$$

and the associated kinematic constraints as $J_v(q)\dot{q} = 0$, where $J_v(q)$ is the Jacobian matrix of η_v, i.e., $J_v(q) = \frac{\partial \eta_v}{\partial q}$. It was shown in [45] and [44] that the foot contact with the ground is unilateral in essence. Hence, a certain set of conditions should be imposed on the contact wrenches, λ_v, in order to satisfy the holonomic constraints assumption. Specifically, we state these conditions as

$$\nu_v(q)\lambda_v(q, \dot{q}, u) \geq 0 \quad (3)$$

where $\nu_v(q)$ depends on the geometric parameters of the contacts, such as the size of the robot feet and the friction coefficient with the ground. It can be noted that here we explicitly show the dependence of contact wrenches on the system states and control inputs. For some domains, additional unilateral constraints related to the robot postures, denoted by $h_v(q) > 0$, should also be considered. Combining (3) and unilateral constraints together yields the domain of admissibility

$$\mathcal{D}_v = \{(q, \dot{q}, u) \in T\mathcal{Q} \times \mathcal{U} | A_v(q, \dot{q}, u) \geq 0\} \quad (4)$$
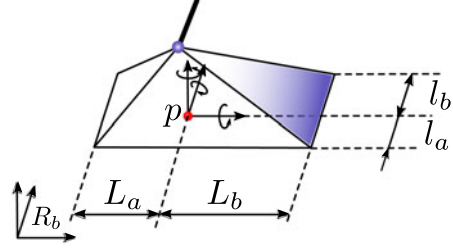

Fig. 3. Geometric illustration of the flat foot ground contact.

for $v \in V$, where

$$A_v(q, \dot{q}, u) = \begin{bmatrix} \nu_v(q)\lambda_v(q, \dot{q}, u) \\ h_v(q) \end{bmatrix} \geq 0 \quad (5)$$

defines the boundary condition of the domain manifold.

Example 2: During the single support domain of a typical flat-footed walking gait, the stance foot should lie flat on the ground. Thus, the associated holonomic constraints can be defined as the Cartesian position of a fixed point p, as shown in Fig. 3, on the stance foot link and the orientation of the stance foot link is constant. The corresponding contact wrenches consist of three constraint forces, $(\lambda_c^{fx}, \lambda_c^{fy}, \lambda_c^{fz})$, and three constraint moments, $(\lambda_c^{mx}, \lambda_c^{my}, \lambda_c^{mz})$, respectively. Conditions on contact wrenches include:

1) the ground reaction force should not be negative;
2) feet should not slide on the ground;
3) the robot should not roll over the edge of the feet.

These conditions can be stated as

$$\lambda_c^{fz} \geq 0 \quad (6)$$

$$\sqrt{(\lambda_c^{fx})^2 + (\lambda_c^{fy})^2} < \mu_c \lambda_c^{fz}, \lambda_c^{mz} < \gamma_c \lambda_c^{fz} \quad (7)$$

$$-l_a \lambda_c^{fz} < \lambda_c^{mx} < l_b \lambda_c^{fz}, -L_b \lambda_c^{fz} < \lambda_c^{my} < L_a \lambda_c^{fz} \quad (8)$$

where μ_c and γ_c are the linear and torsional friction coefficient, respectively. The inequalities in (8) is also referred as the ZMP conditions [1], [44]. In addition, we also require that the height of the swing foot should be positive, so that the swing foot is always above the ground. This can be formulated as a unilateral constraint: $p_{\text{nsf}}^z(q) \geq 0$. These unilateral constraints on contact wrenches and robot kinematics form the domain of admissibility condition in (5).

Continuous Dynamics: With the mass, inertia, and length properties of each link of the robot, the equation of motion of the constrained dynamical system for a given domain $\mathcal{D}_v$ is determined by the classical Euler–Lagrange equation and holonomic constraints of the domain [46]

$$D(q)\ddot{q} + H(q, \dot{q}) = B_v u + J_v^T(q)\lambda_v \quad (9)$$

$$J_v(q)\ddot{q} + \dot{J}_v(q, \dot{q})\dot{q} = 0 \quad (10)$$

where $D(q)$ is the inertia matrix, $H(q, \dot{q}) = C(q, \dot{q})\dot{q} + G(q)$ is the vector containing the Coriolis and gravity term, B_v is the actuator distribution matrix. The wrenches λ_v can be determined by solving (9) and (10) simultaneously [44]. Substituting the

closed-form solution of λ_v into (9) yields the affine control system of the form

$$\dot{x} = f_v(x) + g_v(x)u \tag{11}$$

with $x = (q, \dot{q}) \in \mathcal{X} = T\mathcal{Q}$ being the state of the system.

Guards: A guard S_e is a proper subset of the boundary of the domain, $\mathcal{D}_{\mathrm{sor}(e)}$, determined by an edge condition associated with the transition from $\mathcal{D}_{\mathrm{sor}(e)}$ to the following domain, $\mathcal{D}_{\mathrm{tar}(e)}$. Let $H_e(q, \dot{q}, u)$ be an appropriate element from the vector in (5) corresponding to a transition, then the guard is defined as

$$S_e = \{(q, \dot{q}, u) \in \mathcal{D}_{\mathrm{sor}(e)} | H_e(\cdot) = 0, \dot{H}_e(\cdot) < 0\}. \tag{12}$$

Discrete Dynamics: When a guard is reached, it indicates a change of the contact. For bipedal locomotion, it could be when new contacts are established, e.g., the heel or sole of the swing foot hits the ground, or when existing contacts break, e.g., the lift-off event of the swing foot. As a consequence, the states of the robot will undergo a discrete change. This discrete dynamics of the system can be captured as a reset map that projects the current states of the system, at the current phase's guard, to the following domain. In particular, we model the robot and guard as rigid bodies. Hence, given the preimpact states $(q_e^-, \dot{q}_e^-)$ on the guard, the postimpact states $(q_e^+, \dot{q}_e^+)$ are computed by assuming a perfectly plastic impact (if an impact occurs) [47], [48]. Following the presentation in [44], the robot configuration is invariant through impact, i.e., $q_e^+ = q_e^-$. Because the impact occurs instantaneously due to the rigid body assumption, the generalized momentum of the system should be conserved, i.e.,

$$D(q_e^+)(\dot{q}_e^+ - \dot{q}_e^-) = J_{\mathrm{tar}(e)}^T(q_e^+)\delta F_{\mathrm{tar}(e)} \tag{13}$$

where $\delta F_{\mathrm{tar}(e)}$ is a vector of the intensity of impulsive contact wrenches over the infinitesimal impact event. This plastic impact equation together with the holonomic constraints of the subsequent domain determines the discrete jump of the joint velocities, represented as $\dot{q}_e^+ = \Delta_{\dot{q}}(q)\dot{q}_e^-$. Therefore, the reset map of a given guard can be written as

$$(q_e^+, \dot{q}_e^+) = \Delta_e(q_e^-, \dot{q}_e^-) := \begin{bmatrix} \Delta_q(q_e^-) \\ \Delta_{\dot{q}}(q)\dot{q}_e^- \end{bmatrix} \tag{14}$$

where $\Delta_q(q_e^-)$ represents the change in the robot configuration, which is often an identity map (see Remark 1).

Remark 1: In the study of symmetric walking gaits, a bipedal robot is often modeled regarding "stance" and "nonstance" leg angles instead of physical "left" and "right" leg angles to reduce the number of discrete domains. In these cases, the robot configuration needs to be relabeled if there is a change in the "stance" and "nonstance" leg, i.e., when the "nonstance" leg becomes the "stance" leg. As a result, $\Delta_q(q_e^-)$ is no longer an identity map. This relabeling process can be denoted as

$$\Delta_q(q_e^-) := \mathcal{R}(q_e^-) \tag{15}$$

where $\frac{\partial \mathcal{R}(q)}{\partial q}$ has full rank. It is important to note that this map is a linear map in many applications [49], [50].

III. VIRTUAL CONSTRAINTS BASED FEEDBACK CONTROL

In this section, virtual constraints are introduced as a means to synthesize feedback controllers that realize dynamic locomotion of a walking robot. Enforcing virtual constraints results in a reduced dimensional representation of the full order system that captures the natural dynamics of the robot.

A. Virtual Constraints

Analogous to holonomic constraints, virtual constraints (also termed *outputs* in the control literature [49], [51]) are defined as a set of functions that modulate the behavior of a robot in order to achieve particular desired trajectories via state-based feedback controllers. The term "virtual" comes from the fact that these constraints are enforced via joint actuators instead of mechanical constraints.

Definition 3: Given $v \in V$, $y_v^a = (y_{1,v}^a, y_{2,v}^a)$ is an admissible combination of robot outputs consisting of velocity-modulating outputs, $y_{1,v}^a : \mathcal{Q} \to \mathbb{R}^{n_{1,v}}$, and position-modulating outputs, $y_{2,v}^a : \mathcal{Q} \to \mathbb{R}^{n_{2,v}}$. With m_v as the total number of admissible controls and n_v as the total number of holonomic constraints, the total number of position-modulating outputs, $n_{2,v}$, is determined by

$$n_{2,v} = \begin{cases} m_v - n_{1,v}, & \text{if } m_v \leq n - n_v \\ n - n_v - n_{1,v}, & \text{if } m_v > n - n_v. \end{cases} \tag{16}$$

Let $\mathcal{O}_v$ be an indexing set for y_v^a whereby $y_v^a(q) = \{y_{1,o}^a(q), y_{2,o}^a(q)\}_{o \in \mathcal{O}_v}$. A output combination is *independent* if the Jacobian of $y_v^a(q)$ has a full rank.

Remark 2: The idea of the velocity modulating output originates from the study of human-inspired control. By analyzing human locomotion data, Ames proposed that the forward hip velocity appear to be an approximately constant value [52]. Hence, if there are enough admissible actuators present in the bipedal robot, the forward velocity can be controlled via feedback controllers as a velocity-modulating output. The admissible condition is determined by whether or not the forward velocity of the hip is fully controllable. For example, we define the forward velocity as the velocity-modulating output only when the stance foot is flat on the ground, and both the ankle and knee joints are actuated.

Given a group of actual robot outputs as in Definition 3, the *virtual constraints* are defined as the difference between the actual and desired outputs of the robot

$$y_{1,v}(q, \dot{q}, \alpha_v) = \dot{y}_{1,v}^a(q, \dot{q}) - y_{1,v}^d(\tau(q), \alpha_v) \tag{17}$$

$$y_{2,v}(q, \alpha_v) = y_{2,v}^a(q) - y_{2,v}^d(\tau(q), \alpha_v) \tag{18}$$

for $v \in V$, where $y_{1,v}$ and $y_{2,v}$ are relative degree 1 and relative degree 2 by definition, respectively, and $\alpha_v := \{\alpha_o\}_{o \in \mathcal{O}_v}$ is the set of parameters for the desired outputs. Although the desired output can be represented in various function forms, we typically define desired outputs as follows: the desired velocity-modulating output (if present) is assumed to be a *constant*, and the desired position-modulating outputs are given in term of Bézier polynomials. Furthermore, the desired outputs often are defined as functions of a state-based parameterization of time

$\tau(q)$ so as to create an autonomous control system, which is more robust than nonautonomous systems [23]. $\tau(q)$ must be a strictly monotonic (increasing or decreasing) function over a specific duration of time, such as a step cycle.

B. Partial Hybrid Zero Dynamics (PHZD)

With the goal of driving the virtual constraints $y_v = (y_{1,v}, y_{2,v}) \to 0$ exponentially, consider the feedback linearization control law with a control gain $\varepsilon > 0$

$$u_v^\varepsilon = -\mathcal{A}_v^{-1} \left(\begin{bmatrix} L_{f_v} y_{1,v} \\ L_{f_v}^2 y_{2,v} \end{bmatrix} + \begin{bmatrix} \varepsilon y_{1,v} \\ 2\varepsilon \dot{y}_{2,v} + \varepsilon^2 y_{2,v} \end{bmatrix} \right) \qquad (19)$$

where L_f and L_f^2 are the first- and second-order Lie derivatives and $\mathcal{A}_v$ is the *decoupling matrix* that is invertible due to the specific choice of virtual constraints [49]. Applying this control law yields linear output dynamics of the form

$$\dot{y}_{1,v} = -\varepsilon y_{1,v} \qquad (20)$$

$$\ddot{y}_{2,v} = -2\varepsilon \dot{y}_{2,v} - \varepsilon^2 y_{2,v} \qquad (21)$$

which are exponentially attractive to the origin, i.e., $y_v \to 0$. Applying the feedback controllers given in (19) in each domain of the hybrid control system (1) yields a hybrid system model, given as

$$\mathscr{H} = (\Gamma, \mathcal{D}^X, S^X, \Delta, F^X) \qquad (22)$$

where $\mathcal{D}^X = \{\mathcal{D}_v^X\}_{v \in V}$ is a set of admissible domains with $\mathcal{D}_v^X \subseteq \mathcal{X}$ a smooth submanifold of the state space $\mathcal{X}$ only, $S^X = \{S_e^X\}_{e \in E}$ is a set of *guards* with $S_e^X \subset \mathcal{D}_v^X$, and $F^X = \{F_v^X\}_{v \in V}$ is a set of restricted dynamical systems defined on $\mathcal{D}_v^X$, i.e., $\dot{x} = f_v^X(x)$ with $x \in \mathcal{D}_v^X$. The control law in (19) renders the reduced-dimensional *zero dynamics* submanifold

$$\mathcal{Z}_v = \{(q, \dot{q}) \in \mathcal{D}_v^X | y_{1,v} = 0, y_{2,v} = 0, \dot{y}_{2,v} = 0\} \qquad (23)$$

invariant during the continuous domains [49]. However, it is not necessarily invariant through discrete dynamics due to impacts. Moreover, if the reset map involves a plastic impact, it is impossible to guarantee the velocity-modulating output to be *constant* due to the change in velocities caused by the impact. In this case, we instead consider the *partial zero dynamics* manifold on which only the position-modulating outputs have zero errors, i.e.,

$$\mathcal{PZ}_v = \{(q, \dot{q}) \in \mathcal{D}_v^X | y_{2,v} = 0, \dot{y}_{2,v} = 0\}. \qquad (24)$$

If there exists a set of parameters $\alpha = \{\alpha_v\}_{v \in V}$ so that for any edge $e \in E$, the submanifold $\mathcal{PZ}_v$ is impact invariant if

$$\Delta_e(x) \in \mathcal{PZ}_{\text{tar}(e)} \qquad \forall x \in S_e^X \cap \mathcal{PZ}_{\text{sor}(e)}. \qquad (25)$$

A manifold $\mathcal{PZ} = \bigcup_{v \in V} \mathcal{PZ}_v$ is called hybrid invariant if it is invariant over all domains of continuous dynamics and impact invariant through all discrete dynamics, i.e., solutions that start in $\mathcal{PZ}$ remain in $\mathcal{PZ}$, even after impulse effects (see Fig. 4). If a feedback control law renders $\mathcal{PZ}$ hybrid invariant, then we say that the multidomain hybrid control system has a *partial HZD* (PHZD). By enforcing PHZD, the full order dynamics of the hybrid system can be represented as a reduced-dimension
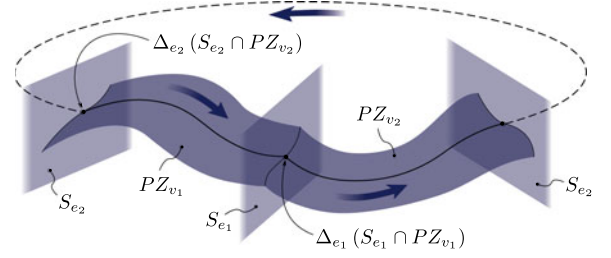

Fig. 4. Illustration of the PHZD periodic orbit in the case of a two-domain hybrid system.

dynamical system that is independent of control inputs. Moreover, the stability properties of periodic solutions of the full order dynamics can also be determined by this low-dimensional representation [53].

Remark 3: The PHZD is not a mandatory requirement if a velocity-regulating output is not present in the design. Furthermore, one could still consider the *HZD* if the desired velocity output is not a constant. However, introducing the partial zero dynamics not only allows us to command a constant desired velocity, but also the evolution of y_1 is now solely determined via the linear output dynamics given in (20) and is independent of τ. This allows for a driving element that pushes the robot forward regardless of the state of the phase variable. Hence, in the remainder of this paper, we will focus our discussion only on the PHZD.

Designing a dynamic walking gait for humanoids using virtual constraints based feedback controllers requires determining a valid set of gait parameters α that satisfies the PHZD requirements and the robot's physical constraints. Finding such parameters is typically regulated as a nonlinear optimization problem. Existing approaches often use *direct shooting* methods, such as single shooting [23], [54] or multiple shooting [55], to solve such an optimization problem. However, these approaches become increasingly intractable for 3-D humanoids or highly underactuated robots. The reasons are twofold: the HZD becomes increasingly unstable for robots with high degrees of underactuation and obtaining equations of motion for the reduced dimensional system explicitly is computationally challenging for systems with as many linkages as humanoids. Consequently, the direct shooting methods run into scalability issues with increasing degrees of freedom robots due to the necessity of explicit forward integration of the zero dynamics.

IV. DIRECT COLLOCATION BASED HZD OPTIMIZATION

In this section, we present the core contribution of the paper—a novel scalable optimization formulation of multidomain humanoid locomotion based on the *direct collocation* method. The direct collocation method works by replacing the explicit forward integration of the dynamical systems with a series of defect constraints via implicit Runge–Kutta methods. The usage of implicit Runge–Kutta methods, which have better convergence properties for unstable systems than explicit methods, enables the direct collocation method to optimize dynamic gaits for highly underactuated robots. Furthermore, we tackle the

scalability issue by decoupling complicated constraints into several simpler constraints by introducing *defect variables*, which are supplementary decision variables that could have been computed in closed-form solutions.

A. Constrained Dynamics

An important feature of direct collocation method is that it allows us for expressing the system dynamics in an implicit differential algebraic equations (DAEs) form. This motivates us to use the full-order constrained dynamics on PHZD manifolds in the optimization. Recall that the PHZD represents a restricted submanifold on which both holonomic constraints and virtual constraints vanish. Based on the previous discussions in Sections II and III, we can state the constrained dynamics as an index-1 DAEs problem

$$F_v(\cdot) := \begin{bmatrix} D(q)\ddot{q} + H(q,\dot{q}) - B_v u - J_v^T(q)\lambda_v \\ J_v(q)\ddot{q} + \dot{J}_v(q,\dot{q})\dot{q} \\ \dot{y}_{1,v} + \varepsilon y_{1,v} \\ \ddot{y}_{2,v} + 2\varepsilon \dot{y}_{2,v} + \varepsilon^2 y_{2,v}, \end{bmatrix} = 0 \quad (26)$$

subject to the initial value conditions at $t = t_0$, given as

$$\eta_v(q(t_0)) = \bar{\eta}_v, \quad J_v(q(t_0))\dot{q}(t_0) = 0 \quad (27)$$

$$y_{2,v}(q(t_0), \alpha_v) = 0, \quad \dot{y}_{2,v}(q(t_0), \dot{q}(t_0), \alpha_v) = 0 \quad (28)$$

where $\bar{\eta}_v$ is a vector of constants determined by the contact conditions.

Lemma 1: Supposing that $\phi_v(t) \subset \mathcal{X}$ is a solution of the DAE system, $F_v(\cdot) = 0$, subject to initial conditions specified in (27) and (28), then $\phi_v(t) \subset \mathcal{PZ}_v$. That is, $\phi_v(t)$ is also a solution on the PHZD.

Proof: The result follows immediately from the construction of the constrained dynamics. By assumption, $\phi_v(t_0) = [q(t_0), \dot{q}(t_0)]^T$ satisfies the initial conditions in (27). Furthermore, the second equation in (26) guarantees that $\eta_v(q) \equiv \bar{\eta}_v$ along any solution of (26). Let $\mathcal{X}_v$ be the canonical projection of $\mathcal{D}_v$ onto the state space $\mathcal{X}$, then $\phi_v(t) \subset \mathcal{X}_v$. Similarly, the third and fourth equations in (26) stabilize the virtual constraints exponentially to the origin. Considering that $y_{2,v}(t_0) = 0$ and $\dot{y}_{2,v}(0) = 0$, therefore, we have

$$y_{2,v}(q(t), \alpha_v) = 0$$

$$\dot{y}_{2,v}(q(t), \dot{q}(t), \alpha_v) = 0$$

for all $t_0 \leq t \leq t_f$ until the solution $\phi_v(t)$ reaches a guard, i.e., $\phi_v(t_f) \in \mathcal{X}_v \cap S_{\text{sor}^{-1}(v)}$. By the definition of the PHZD in (24), we conclude that $\phi_v(t) \subset \mathcal{PZ}_v$. ∎

Remark 4: The implicit DAEs given in (26) yields a equivalent representation of the reduced dimensional robot dynamics on the PHZD manifold. Instead of computing the reduced order dynamics in terms of zero dynamics coordinates as in traditional HZD literature [23], [49], we express the restricted dynamics in the form of implicit DAEs. From a technical perspective, computing the symbolic expressions of $F_v(\cdot)$ would be easier and less time consuming compared to the traditional zero dynamics equations (cf., Eq. (63) in [49]). As we shown in later sections,
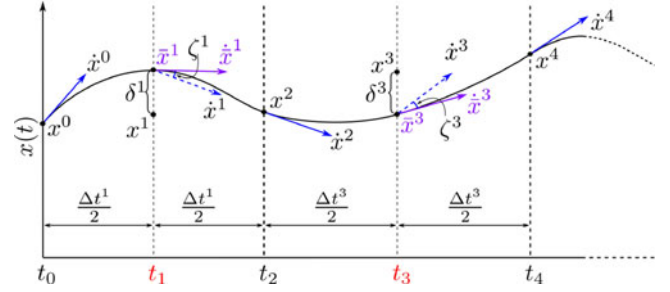

Fig. 5.   Illustration of defect constraints and node distribution.

fully utilizing this fact in the direct collocation optimization is the key to unifying the HZD and direct collocation methods.

B. Modified Direct Collocation Optimization

In this section, we modify the classic Hermite–Simpson collocation scheme so that the system dynamics can be imposed as implicit forms, e.g., (26). Specifically, we first discretize each continuous domain into N_v intervals, then introduce $q^{(i)}$, $\dot{q}^{(i)}$, $\ddot{q}^{(i)}$, $u^{(i)}$, and $\lambda_v^{(i)}$ as NLP decision variables at each discrete node. We denote $\boldsymbol{q}_v$, $\dot{\boldsymbol{q}}_v$, $\ddot{\boldsymbol{q}}_v$, $\boldsymbol{u}_v$, and $\boldsymbol{\lambda}_v$ as collections of variables defined on all nodes, and α_v^* as the virtual constraint parameters that need to be determined for a specific domain $\mathcal{D}_v$. Assuming $T_{I,v} > 0$ is the time at which the system reaches the guard associated with the given domain, the time discretization is defined as

$$0 = t_0 < t_1 < t_2 < \ldots < t_{N_v} = T_{I,v} \quad (29)$$

with $N_v = 2(N_v^c - 1)$, where the even points are called *cardinal nodes* and the odd points are called *interior nodes* (see Fig. 5). The total number of cardinal nodes specified per domain, N_v^c, must be greater than 1, and an interior point (IP) must be placed at the center of two adjacent cardinal nodes.

Remark 5: In the classic Hermite–Simpson method, the states at interior nodes and the slope $\dot{x}$ at cardinal nodes are computed in closed form and are not considered as variables. For a high-dimensional dynamical system, particularly when the system dynamics cannot be determined explicitly, such formulation will harm the convergence of the overall optimization problem. Moreover, u and λ_v can be also computed in closed form in the HZD framework, but it requires inverting matrices such as $D(q)$ and $\mathcal{A}_v(q)$. In our formulation, we introduce them all as defect variables, so that the collocation constraints and system dynamics will be decoupled and complicated closed-form calculations can be avoided.

Collocation Constraints: Given the discretization, we first use Hermite interpolation polynomials to approximate the solution within two neighboring cardinal nodes using the estimated states $x^{(i)} = (q^{(i)}, \dot{q}^{(i)})$ and slopes $\dot{x}^{(i)} = (\dot{q}^{(i)}, \ddot{q}^{(i)})$. The following two defect constraints at each interior node, i, must be satisfied to ensure that the polynomials are indeed accurate approximations of the dynamical system solutions [39], [40]: 1) the difference between $x^{(i)}$ and the interpolated states $\bar{x}^{(i)}$ from the approximated polynomial ($\delta^{(i)}$ in Fig. 5), and 2) the

difference between $\dot{x}^{(i)}$ and the slope of the polynomial ($\zeta^{(i)}$ in Fig. 5). These constraints can be stated as[1]

$$\boldsymbol{\zeta}_v(\boldsymbol{q}_v, \dot{\boldsymbol{q}}_v, \ddot{\boldsymbol{q}}_v) := \begin{bmatrix} \zeta^{(1)} & \zeta^{(3)} & \cdots, & \zeta^{(N_v-1)} \end{bmatrix}^T = \boldsymbol{0} \quad (30)$$

$$\boldsymbol{\delta}_v(\boldsymbol{q}_v, \dot{\boldsymbol{q}}_v, \ddot{\boldsymbol{q}}_v) := \begin{bmatrix} \delta^{(1)} & \delta^{(3)} & \cdots, & \delta^{(N_v-1)} \end{bmatrix}^T = \boldsymbol{0} \quad (31)$$

for each $v \in V$, where

$$\zeta^{(i)} := \dot{x}^{(i)} - \frac{3}{2\Delta t^{(i)}}(x^{(i+1)} - x^{(i-1)}) + \frac{1}{4}(\dot{x}^{(i-1)} + \dot{x}^{(i+1)})$$

$$\delta^{(i)} := x^{(i)} - \frac{1}{2}(x^{(i+1)} + x^{(i-1)}) - \frac{\Delta t^{(i)}}{8}(\dot{x}^{(i-1)} - \dot{x}^{(i+1)})$$

with $\Delta t^{(i)} = t_{i+1} - t_{i-1}$ for $i \in \{1, 3, 5, \ldots, N_v - 1\}$.

PHZD Constraints: To ensure that the approximated solution is indeed a solution of the restricted PHZD of the robot, we enforce that the dynamic equations in (26) and the domain of admissibility conditions as in (4) at all nodes. Let $\boldsymbol{F}_v(\boldsymbol{q}_v, \dot{\boldsymbol{q}}_v, \ddot{\boldsymbol{q}}_v, \boldsymbol{u}_v, \boldsymbol{\lambda}_v, \alpha_v^*)$ and $\boldsymbol{A}_v(\boldsymbol{q}_v, \dot{\boldsymbol{q}}_v, \boldsymbol{\lambda}_v)$ be the vectors that are obtained by stacking $F_v(q^{(i)}, \dot{q}^{(i)}, \ddot{q}^{(i)}, u^{(i)}, \lambda_v^{(i)}, \alpha_v^*)$ and $A_v(q^{(i)}, \dot{q}^{(i)}, \lambda_v^{(i)})$ for all $i \in [0, N_v]$, respectively, these constraints can be stated as

$$\boldsymbol{F}_v(\boldsymbol{q}_v, \dot{\boldsymbol{q}}_v, \ddot{\boldsymbol{q}}_v, \boldsymbol{u}_v, \boldsymbol{\lambda}_v, \alpha_v^*) = \boldsymbol{0} \quad (32)$$

$$\boldsymbol{A}_v(\boldsymbol{q}_v, \dot{\boldsymbol{q}}_v, \boldsymbol{\lambda}_v) \geq \boldsymbol{0}. \quad (33)$$

for all $v \in V$. Moreover, the initial value conditions in (27) and (28) should also be satisfied at the first node $i = 0$ of each domain. In addition, it must be guaranteed that the system reaches the associated guard of a domain $v \in V$ at $T_{I,v}$. This is equivalent to imposing the guard condition in (12) at the last node of each domain. While the continuous dynamics is satisfied by collocation constraints, the discrete dynamics given in (42) can be directly imposed as an equality constraint that connects the solutions of two neighboring domains. Because this constraint involves variables defined on two different domains, we denote $\square^{(0_v+)}$ as a variable defined on the first node of the next domain of $\mathcal{D}_v$.

Let $\mathscr{Z} = \{T_{I,v}, \boldsymbol{q}_v, \dot{\boldsymbol{q}}_v, \ddot{\boldsymbol{q}}_v, \boldsymbol{u}_v, \boldsymbol{\lambda}_v, \alpha_v^*, \bar{\eta}_v\}_{v\in V}$ be the set of NLP variables, the direct collocation based HZD optimization problem can be stated as

$$\mathscr{Z}^* = \underset{\mathscr{Z}}{\operatorname{argmin}} \sum_{v\in V} \mathcal{J}_v(\cdot)$$

$$\text{s.t} \quad \boldsymbol{\zeta}_v(\boldsymbol{q}_v, \dot{\boldsymbol{q}}_v, \ddot{\boldsymbol{q}}_v) = 0 \quad (34a)$$

$$\boldsymbol{\delta}_v(\boldsymbol{q}_v, \dot{\boldsymbol{q}}_v, \ddot{\boldsymbol{q}}_v) = 0 \quad (34b)$$

$$\boldsymbol{F}_v(\boldsymbol{q}_v, \dot{\boldsymbol{q}}_v, \ddot{\boldsymbol{q}}_v, \boldsymbol{u}_v, \boldsymbol{\lambda}_v, \alpha_v^*) = 0 \quad (34c)$$

$$\boldsymbol{A}_v(\boldsymbol{q}_v, \dot{\boldsymbol{q}}_v, \boldsymbol{\lambda}_v) \geq 0 \quad (34d)$$

$$H_e(q^{(N_v)}, \dot{q}^{(N_v)}, \lambda_v^{(N_v)}) = 0 \quad (34e)$$

$$\dot{H}_e(q^{(N_v)}, \dot{q}^{(N_v)}, \lambda_v^{(N_v)}) < 0 \quad (34f)$$

$$\Delta_e(q^{(N_v)}, \dot{q}^{(N_v)}) - (q^{(0_v+)}, \dot{q}^{(0_v+)}) = 0 \quad (34g)$$

$$\eta_v(q^{(0)}) - \bar{\eta}_v = 0 \quad (34h)$$

$$J_v(q^{(0)})\dot{q}^{(0)} = 0 \quad (34i)$$

$$y_{2,v}(q^{(0)}, \alpha_v) = 0 \quad (34j)$$

$$\dot{y}_{2,v}(q^{(0)}, \dot{q}^{(0)}, \alpha_v^*) = 0 \quad (34k)$$

for all $v \in V$ and $e = \text{sor}^{-1}(v) \in E$, where $\mathcal{J}_v(\cdot)$ is a cost function. In particular, physical constraints (such as torque limits, joint velocity, and angle limits, etc.) can be imposed directly as the limiting values of corresponding variables.

A cost function that consists of function integrals, which is quite common in trajectory optimization problems, can be approximated with Simpson's quadrature rule [56]. Let $\mathbf{L}(\cdot)$ be a function that needs to be integrated over the continuous domains, which is also termed as a *running cost*, we have

$$\int_{t_v^0}^{t_v^f} \mathbf{L}_v(\cdot)dt = \sum_{i=0}^{N_v} w_i \mathbf{L}_v(\cdot) \quad (35)$$

where w_i is the integration weight of node i, which can be determined by the Simpson's quadrature rule. Specifically, $w_i = \frac{1}{6}\Delta t^{(i+1)}$ if $i = 0$ or $i = N_v$, $w_i = \frac{2}{3}\Delta t^{(i)}$ if i is an interior node, and $w_i = \frac{1}{3}(\Delta t^{(i-1)} + \Delta t^{(i+1)})$ if i is a cardinal node other than 0 and N_v. With the quadrature approximation, the total cost function can be computed as

$$\mathcal{J}_v(\cdot) = \mathbf{E}_v(\cdot) + \sum_{i=0}^{N_v} w_i \mathbf{L}_v(\cdot) \quad (36)$$

where $\mathbf{E}_v$ is a *terminal cost* that does not requires integration.

Remark 6: It is straightforward to verify from (26) that the constraint wrenches $\lambda_v^{(i)}$ are determined (implicitly) via the second equation, and the control inputs $u^{(i)}$ are determined (also implicitly) from the linear output dynamics stated in the third and fourth equations in (26). By definition, the control inputs determined from the optimization are equal to the feedback controllers defined in (19), which provides us with a set of parameters $\alpha = \{\alpha_v\}_{v\in V}$ that represent the optimal gait behavior for the bipedal robot, rather than open-loop control inputs that result in optimal trajectories. Hence, the control inputs from the trajectory optimization problem are compatible with the feedback control law defined on the PHZD manifold. This feature is different from classic trajectory optimization formulations, in which the control inputs are often assumed to be open-loop and piecewise constant or linear.

Remark 7: The reset map constraint in (34g) can be simplified with the introduction of defect variable δF_v. Specifically, these constraints can be imposed for all $v \in V$ as

$$q^{(0_v+)} - \mathcal{R}(q^{(N_v)}) = 0 \quad (37)$$

$$D(q^{(0_v+)})(\dot{q}^{(0_v+)} - \mathcal{R}(\dot{q}^{(N_v)})) - J_{v+}^T(q^{(0_v+)})\delta F_v = 0. \quad (38)$$

Remark 8: In particular, we include the constant vectors $\bar{\eta}_v$ as decision variables for the (desired) holonomic constraint values so that they can be determined by the optimization. These constants often include the gait properties such as the step length and width. By including $\bar{\eta}_v$ as optimization variables, we have

direct control over these properties in the optimization. Moreover, if specific holonomic constraints are defined on multiple domains, then they should be consistent over the entire gait cycle. Similarly, if some virtual constraints are defined on multiple domains, we often require that they use the same parameter set α_o to have smoother desired outputs. Hence, we impose that

$$\alpha_v[o] - \alpha_{v+}[o] = 0 \quad \forall o \in (\mathcal{O}_v \cap \mathcal{O}_{v+}) \tag{39}$$

$$\bar{\eta}_v[c] - \bar{\eta}_{v+}[c] = 0 \quad \forall c \in (\mathcal{C}_v \cap \mathcal{C}_{v+}) \tag{40}$$

for all vertices except the last one, i.e., $v \in V \backslash \{v_{n_p}\}$. In particular, o and c represent the indexing of the virtual and holonomic constraints defined on the particular domains.

With the direct collocation formulation of HZD gait optimization, we can now state the main result of the paper.

Theorem 1: Solving the constrained NLP problem in (34) yields a set of optimal parameters α^* and a hybrid invariant periodic flow $\phi^*(t)$ such that $\phi^*(t) \subset \mathcal{PZ}_{\alpha^*}$.

Proof: Let $\tilde{\phi}_v^*(t)$ be a piecewise continuous polynomial determined by the solution $\{T_{I,v}^*, \boldsymbol{q}_v^*, \dot{\boldsymbol{q}}_v^*, \ddot{\boldsymbol{q}}_v^*\}$ from the optimization (34) for each domain $\mathcal{D}_v$. Then from (34a)–(34c), $\tilde{\phi}_v^*(t)$ is a approximated solution of the continuous constrained dynamics (26) given relatively small time steps, and by the exponential convergence of collocation methods [57], $\tilde{\phi}_v^*(t) \subset \mathcal{X}_v$ can be considered as the exact solution, $\phi^*(t) \subset \mathcal{X}_v$, of (26), as $N_v \to \infty$ for each $v \in V$. Furthermore, from (34e) and (34f), we can conclude that

$$(q^{(N_v)}, \dot{q}^{(N_v)}) \in S_e \cap \mathcal{X}_v \tag{41}$$

and from (34g), the solutions of two adjacent continuous domains are connected via the reset map Δ_e, i.e.,

$$\phi_{v+}^*(0) = \Delta_e(q^{(N_v)}, \dot{q}^{(N_v)}). \tag{42}$$

Both F_v and Δ_e are C^1 continuous by definition, therefore, there exists an unique solution for some given initial condition $x(0)$. Moreover, with the reset map constraints, we have

$$\phi^*(T_I) = \Delta_e(q^{(N_{v_{n_p}})}, \dot{q}^{(N_{v_{n_p}})}) = (q^{(0_{v_1})}, \dot{q}^{(0_{v_1})}) = \phi^*(0).$$

Hence, $\phi^*(t)$ is periodic due to the uniqueness of the solution. In particular, $T_I = \sum_{v \in V} T_{I,v}$ is the period of the periodic solution. The hybrid invariant of the periodic solution can be verified by constraints (34i)–(34k). In other words, $\phi^*(t) \subset \mathcal{PZ}_v$ by Lemma 1, and $(q^{(N_v)}, \dot{q}^{(N_v)}) \in S_e \cap \mathcal{PZ}_v$. Moreover from (34g), we could easily conclude that (42) holds for each discrete transition. By Lemma 1, $\phi_{v+}(0) \in \mathcal{PZ}_{v+}$. Therefore,

$$\Delta_e(q^{(N_v)}, \dot{q}^{(N_v)}) \in \mathcal{PZ}_{v+} \tag{43}$$

for all $v \in V$. This shows that the solution is impact invariant over all discrete dynamics. We also know that the solution is forward invariant under the feedback controller $\boldsymbol{u}_v^*$. As a result $\phi^*(t) \subset \mathcal{PZ}_{\alpha^*}$, where the PHZD manifold, $\mathcal{PZ}_{\alpha^*} := \mathcal{PZ}_{v_1} \cup \mathcal{PZ}_{v_2} \cdots \cup \mathcal{PZ}_{v_{n_p}}$, depends on the parameters α^*. ∎

C. Sparse NLP Formulation

It is known that the direct collocation formulations significantly increase the number of constraints and optimization variables, leading to a *large* NLP problem. Yet, the Jacobian matrix

of constraints is very sparse; the density of the matrix is far less than 1% in many cases. This feature allows the problem to be solved efficiently using large sparse NLP solvers. To promote the convergence properties of the problem, we will further exploit this sparsity structure of the formulation.

Defect Variables: In the previous discussion, we have introduced many defect variables to simplify the constraints. Now we extend this idea to variables that affect the entire domain, namely the duration $T_{I,v}$ and parameters α_v. Specifically, we define these variables at each node despite the fact that they should be constant on a given domain. While at first glance this modification seems counterintuitive and superfluous, it bears distinct advantages. First, it distributes the "decision weight" of these variables, so each design variable only affects constraints on the neighboring points, not the entire domain. This attribute is helpful for NLP solvers that iterate on linear approximations of the problem (e.g., sequential quadratic programming and IP). To ensure that these variables are indeed the same at all nodes, we additionally enforce the following linear constraints:

$$T_{I,v}^{(i)} - T_{I,v}^{(i+1)} = 0 \tag{44}$$

$$\alpha_v^{(i)} - \alpha_v^{(i+1)} = 0 \tag{45}$$

for all $i \in \{0, 1, 2, \ldots, N_v - 1\}$.

Analytic Jacobian: For a gradient-based NLP solver, providing more accurate information is vital to its convergence. Typically, the Jacobian of constraints and cost function are computed via finite differencing or automatic differentiation of functions. Despite being straightforward to compute, the finite difference approach is very slow to evaluate numerically and often has very low accuracy. Automatic differentiation provides good accuracy, however, it often suffers from limitations caused by restrictions on the tools available. On the other hand, our formulation yields much simpler closed-form cost and constraints, therefore generating the analytic Jacobian (or gradient) of these expressions becomes feasible. In this paper, we use a custom-developed Wolfram Mathematica package to symbolically compute the closed-form expression and the analytic Jacobian of these functions. These symbolic expressions then can be exported into C++ source codes and compiled as static libraries that could be called by the NLP solver during the optimization evaluation. The whole process is executed *a priori* and only needs to generate once for the functions that will be called multiple times at each optimization iteration. Therefore, the overhead time of generating the symbolic expressions will not affect the execution of NLP iterations.

Sparse Jacobian Construction: To expedite the optimization evaluation, we exploit the sparsity pattern of the Jacobian matrix further. First, we group variables and constraints that defined at each node together, then assign indices for each of them based on their locations at the entire variables and constraints. For example, let i^c be the indices of an arbitrary constraint c, and j^x be the indices of dependent variables x^c of the constraint c. Then the Jacobian of this constraint is given by a $n^c \times n^x$ matrix with $n^c = \text{Dim}(i^c)$ and $n^x = \text{Dim}(j^x)$. Based on the indices of variables and constraints, the large sparse Jacobian matrix can be constructed based on the compressed column storage
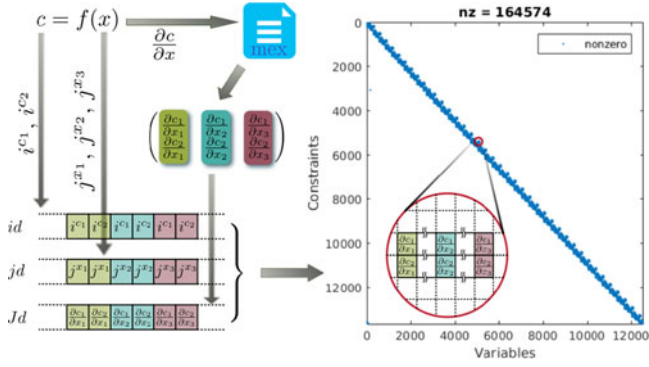
Fig. 6. Illustration of sparse Jacobian matrix construction.

format [58]. Combining indices and values of nonzero elements of all constraints, the whole Jacobian matrix of constraints is created using the MATLAB function, `sparse`. See Fig. 6 for an illustration of this process.

V. APPLICATIONS ON AN UNDERACTUATED HUMANOID

In this section, we apply the HZD framework and direct collocation optimization to explicitly generate two different types of dynamic and energy-efficient walking gaits on DURUS—a spring-legged humanoid robot.

A. DURUS Model

DURUS is a three-dimensional humanoid robot designed and built by SRI International to implement efficient and dynamic locomotion. DURUS consists of 15 actuated joints and two passive springs. The passive springs, which are rigidly fixed to and perpendicular to the bottom of each foot, are designed to reduce the energy loss during foot impact while walking. Here we use the floating base coordinates, assuming that the origin of the 6-D base coordinates is located at the center of the pelvis link (see Fig. 7). The generalized coordinates, $q \in \mathcal{Q} \subset \mathbb{R}^{n^R}$ with $n^R = 23$, of the robot are then determined by (2) where q_r consists of 15 actuated joints and 2 passive springs shown in Fig. 7. For the convenience of conventions, we define the generalized coordinates of the robot in terms of stance and nonstance leg angles, instead of left and right leg angles. Followed by this definition is a relabeling of coordinates at foot impact due to the change of stance leg, which can be done by a linear map, $\mathcal{R} : \mathcal{Q} \to \mathcal{Q}$, in which left and right leg angles are switched accordingly and the sign of all roll and yaw angles, as well as the base position in $y-$axis direction, are "flipped." For the sake of simplicity, we assume the right leg is the stance leg in the remainder of the section.

B. 3-D Flat-Footed Walking

We start with the 3-D flat-footed walking—one of the most commonly seen robotic walking behaviors. The term "flat-footed" indicates that the feet remain flat with respect to the ground plane. It is used to distinguish walking from the multi-contact case—which will be discussed later—where feet can be angled in any number of ways. In the following discussions, we
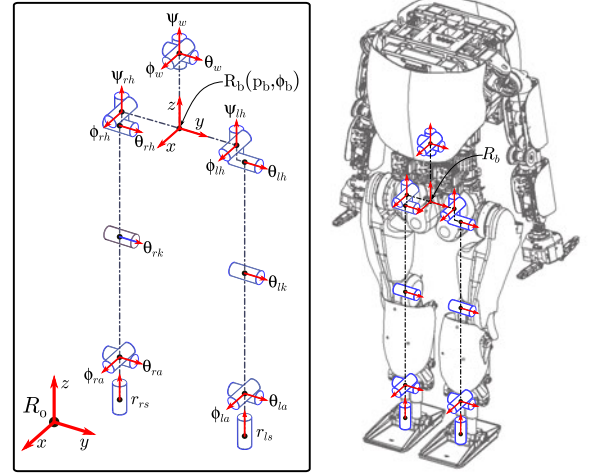

Fig. 7. Generalized coordinates of the DURUS robot, where R_0 is the inertial frame, R_b is the robot base frame located at the center of the pelvis with p_b, ϕ_b is the position and orientation of R_b. ψ_w, ϕ_w, and θ_w are the waist yaw, roll, and pitch angles, ψ_{lh}, ϕ_{lh}, θ_{lh}, θ_{lk}, θ_{la}, ϕ_{la}, and r_{ls} are the left hip yaw, hip roll, hip pitch, knee pitch, ankle pitch, ankle roll angles, and spring deflection, respectively, and ψ_{rh}, ϕ_{rh}, θ_{rh}, θ_{rk}, θ_{ra}, ϕ_{ra}, and r_{rs} are the right hip yaw, hip roll, hip pitch, knee pitch, ankle pitch, ankle roll angles, and spring deflection, respectively. The red arrow of each joint represents the positive rotation (or translation) axis of the corresponding joint using the right-hand rule.
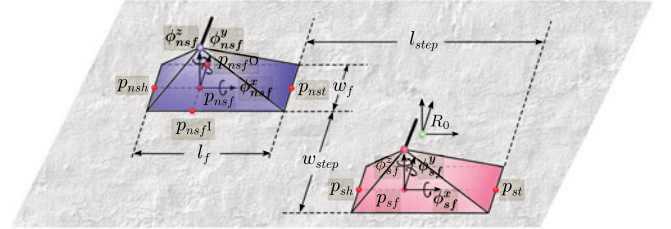

Fig. 8. Illustration of the location of foot contacts.

denote $p_\square(q) = [p_\square^x, p_\square^y, p_\square^z]^T : \mathcal{Q} \to \mathbb{R}^3$ as the 3-D Cartesian position of a point in R_0, and $\phi_\square(q) = [\phi_\square^x, \phi_\square^y, \phi_\square^z]^T : \mathcal{Q} \to SO(3)$ as the 3-D orientation of the link with respect to R_0. The subscript $\square$ indicates the name of the contact point.

Hybrid System Model: The design of passive springs at the end of each leg permits a nontrivial double support phase, therefore, the hybrid system model of 3-D flat-footed walking of DURUS consists of two domains: a *double-support* domain, $\mathcal{D}_{ds}$, and a *single-support* domain, $\mathcal{D}_{ss}$. Specifically, the holonomic constraints for each domain of 3-D flat-footed walking are given by

$$\eta_{ds}(q) := (p_{sf}, \phi_{sf}, p_{nsf}, \phi_{nsf}) \in \mathbb{R}^{12}$$

$$\eta_{ss}(q) := (p_{sf}, \phi_{sf}) \in \mathbb{R}^6$$

where sf and nsf be a point on the stance foot and nonstance foot, respectively (see Fig. 8). The admissibility conditions for each domain are determined according to the discussion in Example 2. In addition, the nonstance foot should be always above the ground during the single-support domain, which could be formulated as an unilateral constraint of the domain, defined as: $h_{ss}(q) := p_{nsf}^z(q) \geq 0$.

Accordingly, a transition from double-support to single-support domain takes place when the normal force on nonstance foot reaches zero, and a transition from single-support to double-support domain occurs when the nonstance foot strikes the ground, i.e.,

$$H_{\text{ds}\to\text{ss}}(q, \dot{q}, u) := \lambda_{\text{nsf}}^{fz}(q, \dot{q}, u) \quad (46)$$

$$H_{\text{ss}\to\text{ds}}(q, \dot{q}, u) := p_{\text{nsf}}^{z}(q). \quad (47)$$

No impact or coordinate change occurs when transitioning from a double-support to single-support domain, i.e., $\Delta_{\text{ds}\to\text{ss}} = I$ where I is an identity matrix. On the other hand, the reset map, $\Delta_{\text{ss}\to\text{ds}}$, needs to incorporate the impact of the nonstance foot strikes and the change of coordinates caused by switching of the stance leg.

Virtual Constraints: The choice of virtual constraints is inspired by authors' previous work on human-inspired bipedal locomotion [43]. Different from [43], we use 4th-order Bézier polynomials as desired outputs. Considering that the robot has actuated ankle and knee joints, we pick the linearized hip position given by

$$\delta p_{\text{hip}}(q) = L_a \theta_{ra} + (L_a + L_c)\theta_{rk} + (L_a + L_c + L_t)\theta_{rh} \quad (48)$$

as the velocity-modulating output for both the double-support and single-support domains, where L_a, L_c, and L_t are the length of ankle, calf, and thigh link of the robot, respectively.

In addition, we pick following position-modulating outputs for the double-support domain:

1) stance knee pitch: $y_{2,\text{skp}}^a = \theta_{rk}$;
2) stance torso pitch: $y_{2,\text{stp}}^a = -\theta_{ra} - \theta_{rk} - \theta_{rh}$;
3) stance ankle roll: $y_{2,\text{sar}}^a = \phi_{ra}$;
4) stance torso roll: $y_{2,\text{str}}^a = -\phi_{ra} - \phi_{rh}$;
5) stance hip yaw: $y_{2,\text{shy}}^a = \psi_{rh}$;
6) waist roll: $y_{2,\text{wr}}^a = \phi_w$;
7) waist pitch: $y_{2,\text{wp}}^a = \theta_w$;
8) waist yaw: $y_{2,\text{wy}}^a = \psi_w$;
9) non-stance knee pitch: $y_{2,\text{nskp}}^a = \theta_{lk}$.

yielding $\mathcal{O}_{\text{ds}} = \{\text{skp}, \text{stp}, \text{sar}, \text{str}, \text{shy}, \text{wr}, \text{wp}, \text{wy}, \text{nskp}\}$. Due to the holonomic constraints imposed on the nonstance foot, nonstance leg joints should not be controlled via virtual constraints. Otherwise, the system will be over-constrained. An exception is the nonstance knee pitch angle due to the fact that the passive spring introduces one additional degree of freedom in the nonstance leg.

For the single-support domain, we define five outputs in addition to the outputs listed above, considering the fact that the nonstance foot is no longer constrained in contact with the ground. These outputs are:

1) nonstance slope: $y_{2,\text{nsl}}^a = -\theta_{ra} - \theta_{rk} - \theta_{rh} + \frac{L_c}{L_c+L_t}\theta_{lk} + \theta_{lh}$;
2) nonstance leg roll: $y_{2,\text{nslr}}^a = \phi_{rh} - \phi_{lh}$;
3) nonstance foot roll: $y_{2,\text{nsfr}}^a = p_{\text{nsf}^{\text{I}}}^z(q) - p_{\text{nsf}^{\text{O}}}^z(q)$;
4) nonstance foot pitch: $y_{2,\text{nsfp}}^a = p_{\text{nst}}^z(q) - p_{\text{nsh}}^z(q)$;
5) non-stance foot yaw: $y_{2,\text{nsfy}}^a = p_{\text{nst}}^y(q) - p_{\text{nsh}}^y(q)$.
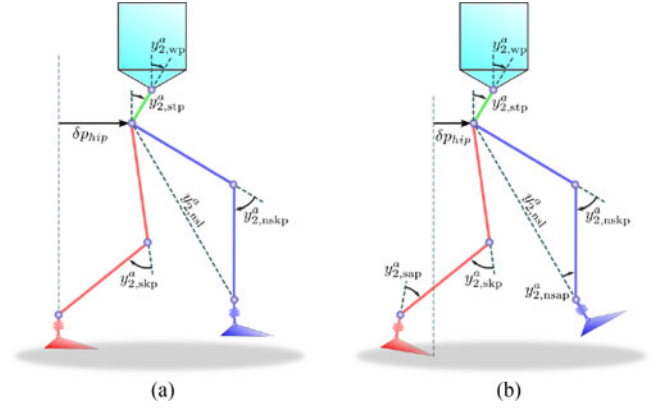

Fig. 9.   Illustration of outputs associated with pitch angles. (a) Flat-footed outputs. (b) Heel–toe outputs.

See Fig. 9(a) for the illustration of some outputs defined above. Consequently, we have the output indexing set $\mathcal{O}_{\text{ss}} = \mathcal{O}_{\text{ds}} \cup \{\text{nsl}, \text{nslr}, \text{nsfr}, \text{nsfp}, \text{nsfy}\}$. The last three outputs are nonlinear outputs equivalently representing the orientations of the nonstance foot. The locations of points nst, nsh, nsf^{I}, and nsf^{O} are shown in Fig. 8. These outputs were chosen over Euler angles in order to avoid expressions that contain inverse trigonometric functions. To guarantee that the nonstance foot remains flat, the desired outputs associated with these three outputs should be zero.

Gait Generation: We apply the direct collocation based HZD optimization framework in Section IV to design efficient and dynamic walking gaits for this hybrid system model. The number of cardinal nodes is chosen as 10 and 20 for the double-support and single-support domain, respectively. To achieve efficient walking, we set the objective function to minimize the mechanical cost of transport (CoT) of the walking gait. Hence, the running cost of the problem is defined as

$$\mathbf{L}(\dot{q}, u, \bar{\eta}_v) := \frac{1}{mgd(\bar{\eta}_v)}\|P_v(\dot{q}, u)\| \quad (49)$$

where mg is the robot weight, $d(\bar{\eta}_v)$ is the distance traveled during a gait which could be determined from the desired holonomic constraints, and $P_v(\dot{q}, u)$ is the total power consumed assuming no power-regeneration (see [59]). In practice, we enforce additional physical constraints to achieve sustainable 3-D flat-footed walking gaits based on observations of the actual implementation on physical hardware.

Restricting torso movement: The robot tends to fall more easily when the upper body wobbles. This can be prevented by constraining the torso movement in the gait design. With $\phi_{\text{tor}}(q) : \mathcal{Q} \to SO(3)$ being the orientation of the upper torso link, we restrict this orientation within a small range specified by $[\phi_{\text{tor}}^{\min}, \phi_{\text{tor}}^{\max}]$, i.e.,

$$\phi_{\text{tor}}^{\min} \leq \phi_{\text{tor}}(q) \leq \phi_{\text{tor}}^{\max}. \quad (50)$$

Constraining impact velocities: It becomes apparent through testing that the swing foot impacting too hard on the ground will destabilize the robot's balance. Hence, we enforce that the Cartesian velocities of the swing foot at impact $\dot{h}_{\text{nsf}}(q^{(N_{\text{ss}})}, \dot{q}^{(N_{\text{ss}})})$

TABLE I
BOUNDARY VALUES FOR PHYSICAL CONSTRAINTS

Constraint	Minimum	Maximum
Torso	$[-0.2, -0.1, -0.1]$	$[0.2, 0.4, 0.1]$
Impact velocity	$[-0.1, -0.1, -0.25]$	$[0.3, 0.1, 0]$
Leg separation angle	-0.1	0

must be restricted within the region defined by $\{v^{\min}, v^{\max}\}$

$$v^{\min} \leq \dot{h}_{\text{nsf}}(q^{(N_{\text{ss}})}, \dot{q}^{(N_{\text{ss}})}) \leq v^{\max}. \tag{51}$$

Avoiding swing leg collision: Due to the existence of compliance in the mechanical system, the swing leg will hit the stance leg if they are not separated enough. The separation of legs can be expressed as the difference between stance and swing hip roll angles. Therefore, during the single-support domain, the nonstance leg roll output is constrained as

$$\phi^{\min} \leq y^a_{2,\text{nslr}}(q) \leq \phi^{\max} \tag{52}$$

where $\phi^{\max} > \phi^{\min} \geq 0$ are the maximum and minimum allowable separation angles.

The minimum and maximum values for the above physical constraints are chosen heuristically based on observing the behavior of actual robot during experiments. The final values used in this paper are listed in Table I. The total number of optimization variables and constraints are 8974 and 9893, respectively, with the sparsity of the constraints Jacobian matrix being 0.13%. The performance of the gait optimization and the simulation and experimental validations of an optimal HZD gait will be discussed in Section VI.

C. 2-D Heel–Toe Walking

In this paper, we use the term "2-D walking" to indicate that the behavior of the robot is restricted to the sagittal plane only. It is important to point out that we still use the full-order 3-D model of DURUS in this application despite the motion being planar. The planar motion is realized by connecting a supporting boom to the torso via a prismatic joint. This linear boom is designed to freely slide along a straight gantry mounted on the ceiling, allowing the robot to move in the sagittal plane only. The prismatic joint connection between the torso and the boom introduces a 3-D holonomic constraint. Let ts be the point at the connecting joint, these constraints are p^y_{ts}, ϕ^x_{ts}, and ϕ^z_{ts}. The inertial effect of the boom mass on the system dynamics is modeled by adding the kinetic energy to the system Lagrangian as in [60].

Hybrid System Model: Different from the flat-footed walking, 2-D heel-toe walking allows the feet to rotate about their toes or heels, resulting in heel- or toe-only contacts with the ground. This walking closely emulates normal human walking pattern that we discussed in Example 1. This hybrid system model has four domains in total: a *toe-strike* domain, $\mathcal{D}_{\text{ts}}$, a *toe-lift* domain, $\mathcal{D}_{\text{tl}}$, a *heel-lift* domain, $\mathcal{D}_{\text{hl}}$, and a *heel-strike* domain, $\mathcal{D}_{\text{hs}}$, as illustrated in Fig. 2. We assume that a walking gait cycle starts from the toe-strike domain, and ends at the heel-

strike domain. Based on the contact conditions, we specify the holonomic constraints for each domain as

$$\eta_{\text{ts}}(q) := (p^y_{\text{ts}}, \phi^x_{\text{ts}}, \phi^z_{\text{ts}}, p_{\text{st}}, \phi_{\text{sf}}, p_{\text{nst}}, \phi^x_{\text{nsf}}, \phi^z_{\text{nsf}}) \in \mathbb{R}^{14}$$

$$\eta_{\text{tl}}(q) := (p^y_{\text{ts}}, \phi^x_{\text{ts}}, \phi^z_{\text{ts}}, p_{\text{st}}, \phi_{\text{sf}}) \in \mathbb{R}^{9}$$

$$\eta_{\text{hl}}(q) := (p^y_{\text{ts}}, \phi^x_{\text{ts}}, \phi^z_{\text{ts}}, p_{\text{st}}, \phi^x_{\text{sf}}, \phi^z_{\text{sf}}) \in \mathbb{R}^{8}$$

$$\eta_{\text{hs}}(q) := (p^y_{\text{ts}}, \phi^x_{\text{ts}}, \phi^z_{\text{ts}}, p_{\text{st}}, \phi^x_{\text{sf}}, \phi^z_{\text{sf}}, p_{\text{nsh}}, \phi^x_{\text{nsf}}, \phi^z_{\text{nsf}}) \in \mathbb{R}^{13}$$

where st, nst, and nsh are the position of the stance toe, the nonstance toe, and the nonstance heel, respectively (see Fig. 8). The contact wrench constraints are similar to the flat-footed walking case with an exception that the ZMP constraints are no longer required due to the supporting boom. The unilateral constraints are determined so that toe or heel position of the foot should be above the ground, i.e.,

$$h_{\text{ts}}(q) := (p^z_{\text{nsh}}), \qquad\qquad h_{\text{tl}}(q) := (p^z_{\text{nsh}}, p^z_{\text{nst}})$$

$$h_{\text{hl}}(q) := (p^z_{\text{sh}}, p^z_{\text{nsh}}, p^z_{\text{nst}}), \qquad h_{\text{hs}}(q) := (p^z_{\text{sh}}, p^z_{\text{nst}}).$$

Correspondingly, the guard conditions are defined as

$$H_{\text{ts}\to\text{tl}}(q, \dot{q}, u) := \lambda^{fz}_{\text{nst}}(q, \dot{q}, u)$$

$$H_{\text{tl}\to\text{hl}}(q, \dot{q}, u) := \lambda^{fz}_{\text{sh}}(q, \dot{q}, u)$$

$$H_{\text{hl}\to\text{hs}}(q, \dot{q}, u) := p^z_{\text{nsh}}(q)$$

$$H_{\text{hs}\to\text{ts}}(q, \dot{q}, u) := p^z_{\text{nst}}(q).$$

There is no impact or coordinate change involved with the toe-lift and the heel-lift event, therefore, the associated reset map is an identity map for these two transitions. The other two events involve an impact, and among them the toe-strike event requires coordinate change, or relabeling, due to the switching of the stance and the nonstance foot.

Virtual Constraints: Outputs for each domain are selected from the outputs that we have defined for the flat-footed walking plus two new outputs, namely [see Fig. 9(b)]

1) stance ankle pitch: $y^a_{2,\text{sap}} = \theta_{ra}$
2) non-stance ankle pitch: $y^a_{2,\text{nsap}} = \theta_{la}$.

We determine the position-modulating outputs of each domain in terms of the following output indexing sets:

$$\mathcal{O}_{\text{ts}} = \{\text{skp, stp, wr, wp, wy, nskp}\}$$

$$\mathcal{O}_{\text{tl}} = \mathcal{O}_{\text{ts}} \cup \{\text{nsl, nsap, nslr, nsfr, nsfy}\}$$

$$\mathcal{O}_{\text{hl}} = \mathcal{O}_{\text{ts}} \cup \{\text{sap, nsl, nsap, nslr, nsfr, nsfy}\}$$

$$\mathcal{O}_{\text{hs}} = \mathcal{O}_{\text{ts}} \cup \{\text{sap, nsap}\}.$$

For domains $\mathcal{D}_{\text{ts}}$ and $\mathcal{D}_{\text{tl}}$, we define the same velocity-modulating output as stated in (48). Because the nonflat stance foot will make it difficult (sometimes impossible) to *directly* control the forward speed of the hip, there is no velocity-modulating output defined on $\mathcal{D}_{\text{hl}}$ and $\mathcal{D}_{\text{hs}}$.

Gait Generation: Dynamic and energy efficient 2-D heel-toe walking gaits will be generated using the same procedures as the 3-D flat-footed walking. Specifically, we pick the number of cardinal nodes as 10, 15, 20, and 12 for the toe-strike, toe-lift,

heel-strike, and toe-strike domain, respectively. To achieve a planar motion also with the other parts of the body, we particularly require the desired outputs that are associated with roll and yaw angles to be zero. These outputs include $y^a_{2,\text{wr}}$, $y^a_{2,\text{wy}}$, $y^a_{2,\text{nslr}}$, $y^a_{2,\text{nsfr}}$, and $y^a_{2,\text{nsfy}}$. This requirement can be achieved by setting the upper and lower bound of parameter sets α_o associated with these outputs to zero. Also, the impact velocity constraint in (51) is now imposed on heel-strike and toe-strike, respectively. Considering this heel-toe walking behavior has four domains, the total number of optimization variables and constraints is 21309 and 22721, respectively.

VI. Experimental Results

In this section, we present simulation and experimental results of two types of walking gaits on DURUS generated from the proposed HZD optimization method. The performance of this method under different conditions is also evaluated and discussed.

A. Setup

The HZD gait optimization problem is solved by IPOPT [61] interfaced with MATLAB on a laptop computer with an Intel Core i7-6820HQ processor (2.7 GHz × 8) and 8 GB of RAM. In addition, we use the linear solver ma57 for IPOPT. Optimal gaits generated from the optimization are then implemented on the DURUS hardware. For more details, we refer the readers to [33].

During the experiments, the robot walked on a $5' \times 8'$ treadmill platform. The speed of the treadmill was automatically controlled to match the walking speed of the robot. For 2-D restricted walking, the top of the robot torso was pinned to a specially designed boom supported by a straight gantry parallel to the treadmill. The boom allows the robot to freely move up and down, forward and backward, and to rotate about the $y-$axis, but restricts its motion to the sagittal plane only. For the 3-D humanoid walking gait, there was no supporting boom attached, rather a safety catch rope was tied to the robot and ceiling. The rope is loose enough not to support the robot as it walks, but catches before the robot hits the ground in case of a fall.

B. 3-D Flat-Footed Walking

In this section, we present the simulation and experimental results of one of many stable periodic gaits obtained from the optimization. The step length of this particular gait is 0.13 m and the step width is 0.24 m. The total elapsed time of one complete gait step is 0.48 s. The desired linearized hip velocity is 0.3 m/s for this gait.

Simulation Results: To demonstrate the convergence of actual outputs to given desired trajectories under the feedback controllers, we simulate the system starting from a disturbed rest position. This initial condition is determined by slightly disturbing the fixed point of the original periodic orbit on the Poincaré section, and set all joint velocities to be zero. The feedback controllers drive the system to periodic limit cycles even
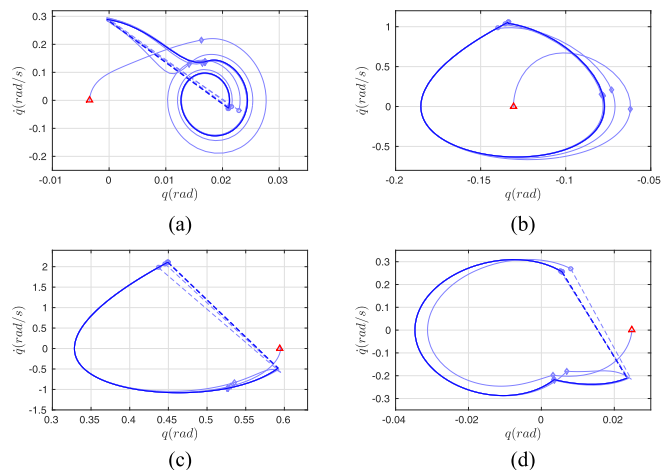

Fig. 10. Phase portraits of representative joints in the 3-D flat-footed walking simulation starting from the rest. The red △ shows the initial point of the simulation. The dashed lines represent the discrete jump in system states at the end of each step. The domain transitions are represented by ◇ and blue ○, respectively. (a) Stance spring. (b) Waist pitch. (c) Stance knee pitch. (d) Nonstance hip roll.

starting from a point that is not on the orbit. This is demonstrated in Fig. 10, where we show phase portrait plots of four representative joints. As shown in these figures, both uncontrolled states (stance spring) and controlled states (waist pitch, stance knee pitch, and nonstance hip roll) converge to periodic limit cycles under the feedback controllers. The maximum magnitude of the eigenvalues of the Jacobian of the Poincaré return map is 0.24, which further indicates that the gait is stable.

Experimental Results: The 3-D flat-footed gait walked stably for hours on multiple occasions. The stable walking of DURUS was showcased at the robot endurance test at the DARPA robotics challenge finals, during which DURUS exhibited sustained walking over large distances with a consistently low CoT. Fig. 11 shows the periodic phase portraits of each of the actuated joint angles in one of the experiments. Particularly, we show the periodic limit cycles in terms of left/right leg angles, rather than in stance and nonstance leg angles. The comparison of the corresponding periodic orbit in simulation is also plotted in the figure, which shows very close match between experimental and simulation trajectories despite discrepancies between simulation and reality. These periodic limit cycles, in turn, represent a cyclic behavior of the robot, i.e., periodic walking gaits. We show tiled images of one gait step of the stable 3-D walking in the experiment and simulation in Fig. 12. This successful stable walking in 3-D is a strong indicator of the practicality of the presented optimization approach on humanoid robots. A video of this 3-D flat-footed walking gait in simulation and experiment can be found in the attached video and [62].

Remark 9: The passive springs of DURUS is only considered in the gait optimization process in Section V-B to generate dynamically compatible motion for the full-order model of the robot, but not explicitly incorporated within the feedback control in experiments due to the fact that our current choice of outputs is independent of the vertical springs in this paper. Moreover, due to the sufficient damping and the natural frequency of the
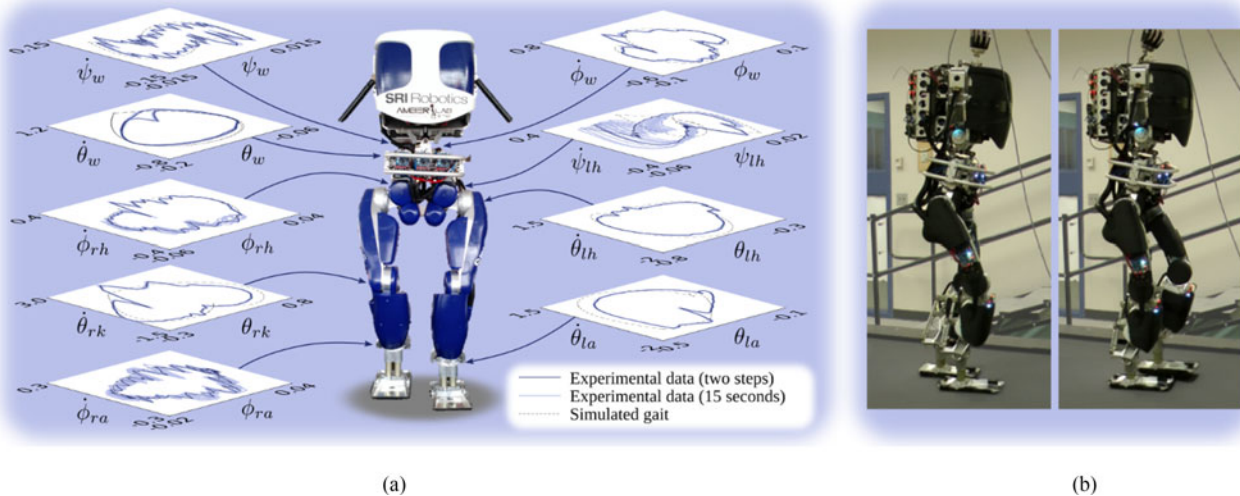
Fig. 11. Periodic limit cycles of actuated joints in the experiment and overlaid on the simulated gait (units: rad and rad/s; symmetric joints omitted for clarity).
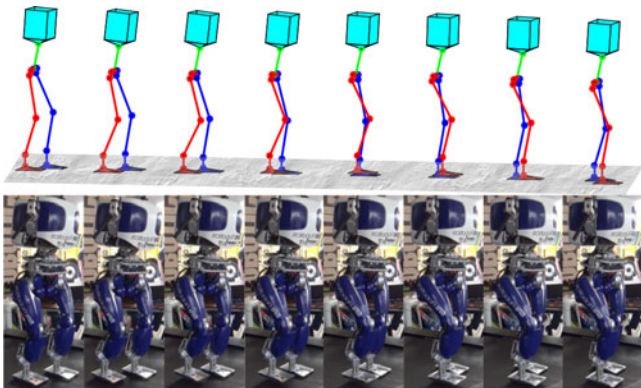

Fig. 12. Tiled still images from the simulation and experiment of DURUS flat-footed walking in 3-D at 0.3 m/s.

Fig. 13. Electrical CoT for a 3-D flat-footed walking experiment over 450 steps with a mean CoT of 1.33.

vertical springs, the actual operating frequency of these virtual springs (typically around 2 Hz) is significantly smaller than the resonant frequency of the springs. Hence, we have not noticed any excessive resonant motion of the springs induced from the motion and control in our experiments.

3-D Walking Efficiency: DURUS uses an onboard battery pack to supply power to all electrical components of the robot, including the central control computer, motor drivers, and controllers, motors, and sensors. The specific cost of electrical transport c_{et} was calculated as in [63], where the total energy consumed over the weight and distance traveled for step i as

$$c_{\mathrm{et},i} = \frac{1}{mgd_i} \int_{t_i^+}^{t_i^-} P_{\mathrm{el}} \qquad (53)$$

where P_{el} is total consumed electrical power and d_i is the x-position traveled by the nonstance foot of the robot through the duration of the ith step. In particular, Fig. 13 shows measurements of the electrical CoT of 3-D flat-footed walking gait over 450 steps. Power data were computed directly from current and voltage measured on the output of the battery pack.
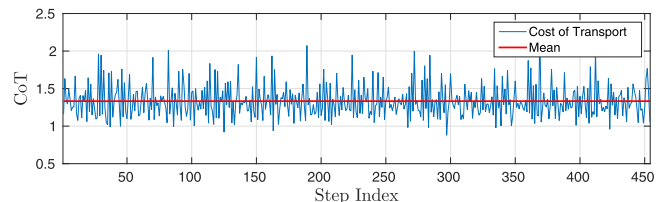
The average total transport cost was measured to be 1.33, which demonstrates a significant improvement in humanoid locomotion economy. Notable comparable examples include ASIMO, which is estimated to have an electrical CoT of 3.2 [64]. Notably, other nonhumanoid bipeds have been built specifically to demonstrate more efficient locomotion, yielding transport costs under 0.2 [14]. By these results, we demonstrate that gait economy can be advanced in more-traditional 3-D humanoid forms, at least in part as a result of our scalable and energy-optimized gait generation.

In addition, the average electrical CoT accounts for the power only consumed on actuation—excluding energy consumption on other electrical components, such as control computer and sensors—is 0.83. In comparison, the total mechanical CoT of the ideal robot model from our optimization is 0.16. That is, we have observed a significant increase in the power consumption in experiment due to the efficiency of actuator and transmission as well as extra control efforts of tracking the desired trajectories due to the model mismatch between simulation and experiment.

C. 2-D Heel–Toe Walking

An optimized gait from the heel–toe walking with the desired velocity 0.35 m/s and the step length 0.36 m is shown in Figs. 14 and 15. The maximum magnitude of the eigenvalues of the Jacobian of the Poincaré return map of this gait is 0.96, which indicates theoretical stability of the walking gait. In particular,
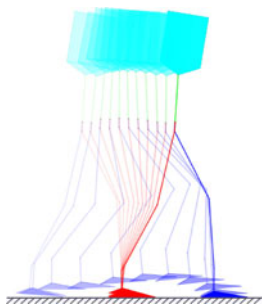
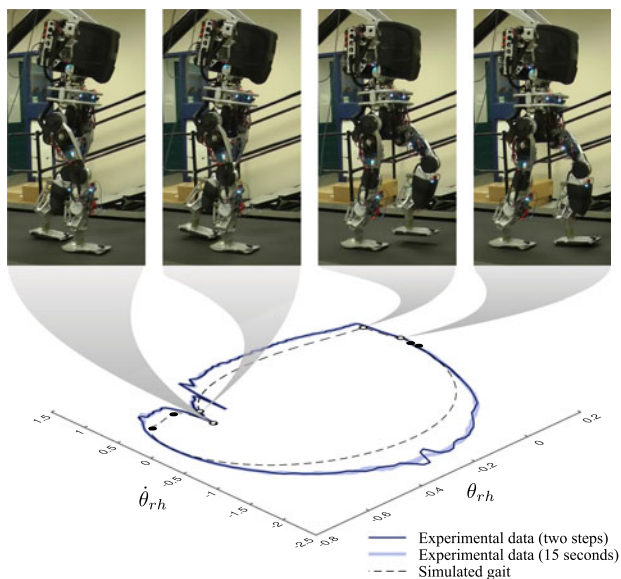Fig. 14. Stroboscopic figure of the 2-D heel–toe walking.

Fig. 15. Experimental results of multicontact heel–toe walking with DURUS in 2-D, showing phase plot of a representative joint angle both in simulation and experiment (units: rad and rad/s). Four tiled images of DURUS near times of multicontact domain switches are shown in correspondence with their location in the phase plot.

TABLE II
RUNTIME AND CONVERGENCE TEST OF THE OPTIMIZATION

Init. Guess	Bound.	Iteration	Feasibility	Runtime (s)
Random	Relaxed	856	2.24×10^{-9}	675.49
	Regular	1209	7.89×10^{-13}	911.48
	Restricted	2811	3.08×10^{-9}	2027.08
Old gait	Relaxed	193	4.91×10^{-09}	150.28
	Regular	357	3.68×10^{-11}	375.9
	Restricted	693	5.21×10^{-13}	620.03

optimization tends to converge more quickly. However, some physical constraints must be enforced strictly due to the hardware limitations. Hence, not every constraint or variables can be relaxed. The initial guess seeded to the optimizer also affects the convergence of the problem. The optimization converges quickly to a solution if seeded with a "better" initial guess, such as a solution from the previous optimization. Table II shows partial statistical results of the optimization of 3-D flat-footed walking with DURUS using different optimization parameters. There are two different seeds: the first set of initial guesses is randomly picked and the second set is obtained from a previous solution that does not satisfy all NLP constraints. We also considered three different boundary conditions for NLP constraints: a "relaxed" condition indicates the reachable set of optimization variables set to be large, for example, larger joint velocity, higher actuator torques, and relaxed configuration limits, etc.; a "regular" condition suggests that the boundaries of variables are set to reasonable values, mostly set to slightly tighter limits than what the hardware is capable of; a "restricted" condition restricts the reachable set of the optimization variables within a very small region so as to achieve walking gaits with certain fixed behaviors, for instance maintaining a straight torso.

Our results showed that adjusting the initial guess and relaxing the constraints can improve the convergence time of the optimization. However, we note that even a completely random initial guess subject to strict hardware constraints still converges to a feasible solution successfully. This suggests that creative seeding is a helpful but perhaps unnecessary measure for solving such high-dimensional problems.

Comparison Versus Existing HZD Optimization Approaches: Finally, we compared the computational performance of the proposed approach and existing HZD optimization approaches using direct single shooting and multiple shooting methods. Considering both shooting methods are not capable of solving HZD optimization for high-dimensional humanoids, we particularly used a 5-link planar biped used in [49] for the single shooting optimization and 7-link spring-leg planar biped in [55] for the multiple shooting optimization. Specifically, the following tests are evaluated with MATLAB's fmincon NLP solver instead of IPOPT. Moreover, Considering that the shooting methods are not robust enough to handle random initial guesses, we take the parameters from previous results as the initial guess. The comparison results under the same physical constraints and initial guesses are shown in Tables III and IV.

The results demonstrate that our proposed approach based on direct collocation methods converges significantly faster than

the phase plot of a representative joint angle in both simulation and experiment shows the existence of stable periodic orbits of the system. The stroboscopic figure of simulated 2-D heel–toe walking gaits is shown in Fig. 14. Compared to flat-footed walking, both feet are no longer flat on the ground or in the air during the 2-D heel–toe walking. The most apparent benefit of the heel–toe walking is a longer step length than its counterpart gaits, likely because the feet can stretch further out in front by landing on the heel, exhibiting more qualitatively human-like behaviors. The simulation and experimental results of the 2-D heel–toe walking gait can be found in the attached video (Supplementary material) and [65].

D. Performance Evaluation of the Optimization method

Optimization Performance on DURUS: The run time and convergence rate of the optimization greatly depends on a number of factors. These include the reachable set of the optimization variables as determined by the boundary values and constraints, and the initial guess used to seed the solution. With more relaxed upper and lower limits of variables and constraints, the

TABLE III
COMPARISON RESULTS BETWEEN THE (LOCAL) DIRECT COLLOCATION
OPTIMIZATION VERSUS THE SINGLE SHOOTING OPTIMIZATION

Method	CPU time (s)	Iteration	Function Calls
Single shooting (`fmincon`)	162.59	21	612
Direct collocation (`fmincon`)	5.17	23	55
Direct collocation (`IPOPT`)	1.60	47	59

TABLE IV
COMPARISON RESULTS BETWEEN THE (LOCAL) DIRECT COLLOCATION
OPTIMIZATION VERSUS THE MULTIPLE SHOOTING OPTIMIZATION

Method	CPU time (s)	Iteration	Function Calls
Multiple shooting	5027.45	1155	16 24 687
Direct collocation (`fmincon`)	84.30	44	123
Direct collocation (`IPOPT`)	41.47	207	845

the existing HZD gait optimization formulations using shooting methods. The direct collocation method has approximately 30 to 100 times faster than the single shooting method depends on which solver has been used, and is approximately 60 to 100 times faster than the multiple shooting method. That is because using numerical finite difference methods to compute the gradient information requires an extensive amount of computing resources. It can also be observed from the total number of function calls of different optimization methods used. In our direct collocation formulation, the use of the analytic Jacobian reduces the function calls significantly compared to the numerical finite differentiation used for direct shooting methods.

The direct collocation method also reduces the number of iterations to a feasible solution compared to the direct multiple shooting optimization, as shown in Table IV. It is due to the relatively linear relationships between the constraints and NLP variables in the direct collocation optimization. In principle, the same result should be observed in the comparison versus the single shooting optimization. However, the particular case of the 5-link point feet planar model allows explicitly solving the preimpact states and the closed-form solution of the two-dimensional zero dynamics, which in turn benefit the overall convergence of the single shooting optimization. Unfortunately, the same approach cannot extend to high-dimensional humanoid robots or highly underactuated systems. Besides, using numerical finite difference methods to compute the gradient information requires the extensive amount of computing resources. As a result, the overall CPU time used in the single shooting optimization is still comparatively more than the collocation method. With that being said, even when the partial zero dynamics can be computed in closed form as it appears in many fully actuated humanoids [66], [67], the single shooting optimization requires incredible amounts of computing time. Interestingly, even though the number of iterations is, in fact, greater when using `IPOPT`, yet still it uses the least amount of wall time. This indicates the fact that `IPOPT` performs better for large-scale sparse problems. For the 7-link spring-leg planar robot as in [55], the closed-form solution of the HZD cannot be obtained explicitly. Therefore, the numerical forward

integration of zero dynamics must be performed using explicit Runge–Kutta methods. As a result, the number of iterations and functions calls is notably more than that of the collocation method. Moreover, the multiple shooting method tends to be significantly less reliable; the number of iterations is far more than the direct collocation optimization.

VII. CONCLUSION

We presented a generalized optimization framework for synthesizing formally stable locomotion on underactuated robots as complex as humanoids. This framework blends the theoretical foundation of HZD with direct collocation trajectory optimization technique. As a result of this process, we formulate our gait design problem as a nonlinear program that can be solved in under ten min with standard algorithms on a laptop computer. Encouragingly, despite the high dimensionality of the 10 000-variable problem, the optimization even converges with random initial guesses. Furthermore, this method optimizes the interactions of the full order multibody dynamics of multidomain humanoid hybrid system models, without conforming motions to simpler more-tractable dynamics.

Using the spring-legged humanoid platform, DURUS, this method produced both flat-footed and heel–toe walking. By optimizing for efficient locomotion, we achieved an average CoT of 1.33, significantly lower than that reported by other human-scale humanoid robots, such as ASIMO [64]. It is important to note that energy economy is a product of many factors, including mechanics as well as control. While it is difficult to determine that the control is necessarily responsible for this reported economy, these results demonstrate that the presented method was capable of controlling a machine with underactuated features designed to facilitate efficiency (such as soft prismatic ankle compliance). More pointedly, we believe that this is an indicator that the control works in concert with DURUS' energy-conscious underactuated features via optimization, and not working with them via compensation methods. These efficient and stable walking gaits were exhibited in public at the DARPA Robotic Challenges finals, where it competed against the Sandia National Labs robot, STEPPR [68], in the *Robot Endurance Test*. Highlights of DURUS walking at the DRC finals can be found at [69]. Recently, the proposed scalable HZD gait optimization method has been extended to 3-D multicontact heel–toe walking to achieve natural human-like locomotion on DURUS, and further improved the energy efficiency [70], [71]. We believe that these 3-D walking results signal that HZD approaches have overcome a significant technical hurdle, and are, therefore, sufficiently equipped to tackle the complexity of humanoid locomotion.

References

[1] M. Vukobratović and B. Borovac, "Zero-moment point: thirty five years of its life," *Int. J. Humanoid Robot.*, vol. 1, no. 1, pp. 157–173, 2004.

[2] K. Kaneko, F. Kanehiro, M. Morisawa, K. Miura, S. Nakaoka, and S. Kajita, "Cybernetic human HRP-4C," in *Proc. 9th IEEE-RAS Int. Conf. Humanoid Robots*, Dec. 2009, pp. 7–14.

[3] Y. Sakagami, R. Watanabe, C. Aoyama, S. Matsunaga, N. Higaki, and K. Fujimura, "The intelligent ASIMO: System overview and integration," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2002, vol. 3, pp. 2478–2483.

[4] E. Yoshida, C. Esteves, I. Belousov, J.-P. Laumond, T. Sakaguchi, and K. Yokoi, "Planning 3D collision-free dynamic robotic motion through iterative reshaping," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1186–1198, Oct. 2008.

[5] J. Pratt *et al.*, "Capturability-based analysis and control of legged locomotion, part 2: Application to M2V2, a lower-body humanoid," *Int. J. Robot. Res.*, vol. 31, no. 10, pp. 1117–1133, 2012.

[6] B. J. Stephens and C. G. Atkeson, "Dynamic balance force control for compliant humanoid robots," *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2010, pp. 1248–1255.

[7] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3D linear inverted pendulum model: A simple modeling for a biped walking pattern generation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2001, pp. 239–246.

[8] I. W. Park, J. Y. Kim, J. Lee, and J. H. Oh, "Online free walking trajectory generation for biped humanoid robot KHR-3(HUBO)," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2006, pp. 1231–1236.

[9] M. DeDonato *et al.*, "Human-in-the-loop control of a humanoid robot for disaster response: A report from the DARPA robotics challenge trials," *J. Field Robot.*, vol. 32, no. 2, pp. 275–292, 2015.

[10] S. Kuindersma *et al.*, "Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot," *Auton. Robots*, vol. 40, no. 3, pp. 429–455, 2016.

[11] M. Johnson *et al.*, "Team IHMC's lessons learned from the DARPA robotics challenge," *J. Field Robot.*, vol. 32, no. 2, pp. 192–208, 2015.

[12] M. H. Raibert *et al.*, *Legged Robots That Balance*, vol. 3. Cambridge, MA, USA: MIT Press, 1986.

[13] H. Park, K. Sreenath, A. Ramezani, and J. W. Grizzle, "Switching control design for accommodating large step-down disturbances in bipedal robot walking," in *Proc. IEEE/RSJ Int. Conf. Robot. Autom.*, May 2012, pp. 45–50.

[14] P. A. Bhounsule *et al.*, "Low-bandwidth reflex-based control for lower power walking: 65 km on a single battery charge," *Int. J. Robot. Res.*, vol. 33, no. 10, pp. 1305–1321, 2014.

[15] K. H. Koch, K. Mombaur, and P. Soueres, "Optimization-based walking generation for humanoid robot," *IFAC Proc. Vol.*, vol. 45, no. 22, pp. 498–504, 2012.

[16] S. Lengagne, J. Vaillant, E. Yoshida, and A. Kheddar, "Generation of whole-body optimal dynamic multi-contact motions," *Int. J. Robot. Res.*, vol. 32, no. 9/10, pp. 1104–1119, 2013.

[17] M. Posa, S. Kuindersma, and R. Tedrake, "Optimization and stabilization of trajectories for constrained dynamical systems," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2016, pp. 1366–1373.

[18] J. Carpentier, S. Tonneau, M. Naveau, O. Stasse, and N. Mansard, "A versatile and efficient pattern generator for generalized legged locomotion," in *Proc. IEEE Int. Conf. Robot. Autom.*, Stockholm, Sweden, May 2016, pp. 3555–3561.

[19] M. Kudruss *et al.*, "Optimal control for whole-body motion generation using center-of-mass dynamics for predefined multi-contact configurations," in *Proc. IEEE-RAS 15th Int. Conf. Humanoid Robots.*, Nov. 2015, pp. 684–689.

[20] H. Dai, A. Valenzuela, and R. Tedrake, "Whole-body motion planning with centroidal dynamics and full kinematics," in *Proc. 14th IEEE-RAS Int. Conf. Humanoid Robots*, 2014, pp. 295–302.

[21] A. Herzog, N. Rotella, S. Schaal, and L. Righetti, "Trajectory generation for multi-contact momentum control," in *Proc. IEEE-RAS 15th Int. Conf. Humanoid Robots*, 2015, pp. 874–880.

[22] J. W. Grizzle, G. Abba, and F. Plestan, "Asymptotically stable walking for biped robots: Analysis via systems with impulse effects," *IEEE Trans. Automat. Control*, vol. 46, no. 1, pp. 51–64, Jan. 2001.

[23] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback Control of Dynamic Bipedal Robot Locomotion*. Boca Raton, FL, USA: CRC Press, 2007.

[24] C. Chevallereau *et al.*, "Rabbit: A testbed for advanced control theory," *IEEE Control Syst.*, vol. 23, no. 5, pp. 57–79, Oct. 2003.

[25] C. Chevallereau, J. Grizzle, and C. Shih, "Asymptotically stable walking of a five-link underactuated 3D bipedal robot," *IEEE Trans. Robot.*, vol. 25, no. 1, pp. 37–50, Feb. 2009.

[26] A. Hereid, S. Kolathaya, M. S. Jones, J. Van Why, J. W. Hurst, and A. D. Ames, "Dynamic multi-domain bipedal walking with ATRIAS through SLIP based human-inspired control," in *Proc. 17th Int. Conf. Hybrid Syst., Comput. Control*, 2014, pp. 263–272.

[27] A. E. Martin, D. C. Post, and J. P. Schmiedeler, "Design and experimental implementation of a hybrid zero dynamics-based controller for planar bipeds with curved feet," *Int. J. Robot. Res.*, vol. 33, no. 7, pp. 988–1005, 2014.

[28] C. Shih, J. Grizzle, and C. Chevallereau, "Asymptotically stable walking of a simple underactuated 3D bipedal robot," in *Proc. 33rd Annu. Conf. IEEE Ind. Electron. Soc.*, Taipei, Taiwan, Nov. 2007, pp. 2766–2771.

[29] K. Sreenath, H. Park, I. Poulakakis, and J. W. Grizzle, "A compliant hybrid zero dynamics controller for stable, efficient and fast bipedal walking on MABEL," *Int. J. Robot. Res.*, vol. 30, no. 9, pp. 1170–1193, 2011.

[30] W.-L. Ma, A. Hereid, C. M. Hubicki, and A. D. Ames, "Efficient HZD gait generation for three-dimensional underactuated humanoid running," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2016, pp. 5819–5825.

[31] K. Sreenath, H.-W. Park, I. Poulakakis, and J. Grizzle, "Embedding active force control within the compliant hybrid zero dynamics to achieve stable, fast running on MABEL," *Int. J. Robot. Res.*, vol. 32, no. 3, pp. 324–345, Mar. 2013.

[32] B. G. Buss, K. A. Hamed, B. A. Griffin, and J. W. Grizzle, "Experimental results for 3D bipedal robot walking based on systematic optimization of virtual onstraints," in *Proc. IEEE Amer. Control Conf.*, 2016, pp. 4785–4792.

[33] J. Reher, E. A. Cousineau, A. Hereid, C. M. Hubicki, and A. D. Ames, "Realizing dynamic and efficient bipedal locomotion on the humanoid robot DURUS," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2016, pp. 1794–1801.

[34] X. Da, R. Hartley, and J. W. Grizzle, "Supervised learning for stabilizing underactuated bipedal robot locomotion, with outdoor experiments on the wave field," in *Proc. IEEE Int. Conf. Robot. Autom.*, Singapore, May 29–Jun. 3, 2017, pp. 3476–3483.

[35] B. Griffin and J. Grizzle, "Walking gait optimization for accommodation of unknown terrain height variations," in *Proc. IEEE Amer. Control Conf.*, 2015, pp. 4810–4817.

[36] A. D. Ames, "First steps toward underactuated human-inspired bipedal robotic walking," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 1011–1017.

[37] E. R. Westervelt and J. W. Grizzle, "Design of asymptotically stable walking for a 5-link planar biped walker via optimization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, vol. 3, pp. 3117–3122.

[38] M. Srinivasan and A. Ruina, "Computer optimization of a minimal biped model discovers walking and running," *Nature*, vol. 439, no. 7072, pp. 72–75, 2006.

[39] J. T. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, vol. 19. Philadelphia, PA, USA: SIAM, 2010.

[40] C. R. Hargraves and S. W. Paris, "Direct trajectory optimization using nonlinear programming and collocation," *J. Guid., Control, Dyn.*, vol. 10, no. 4, pp. 338–342, 1987.

[41] O. Von Stryk, *Numerical Solution of Optimal Control Problems by Direct Collocation*. New York, NY, USA: Springer-Verlag, 1993.

[42] A. Hereid, E. A. Cousineau, C. M. Hubicki, and A. D. Ames, "3D dynamic walking with underactuated humanoid robots: A direct collocation framework for optimizing hybrid zero dynamics," in *Proc. IEEE Int. Conf. Robot. Autom*, 2016, 1447–1454.

[43] A. D. Ames, R. Vasudevan, and R. Bajcsy, "Human-data based cost of bipedal robotic walking," in *Proc. 14th Int. Conf. Hybrid Syst., Comput. Control.*, 2011, pp. 153–162.

[44] J. W. Grizzle, C. Chevallereau, R. W. Sinnet, and A. D. Ames, "Models, feedback control, and open problems of 3D bipedal robotic walking," *Automatica*, vol. 50, no. 8, pp. 1955–1988, 2014.

[45] Y. Hurmuzlu, F. Génot, and B. Brogliato, "Modeling, stability and control of biped robots—A general framework," *Automatica*, vol. 40, no. 10, pp. 1647–1664, 2004.

[46] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL, USA: CRC Press, 1994.

[47] C. Glocker and F. Pfeiffer, "Dynamical systems with unilateral contacts," *Nonlinear Dyn.*, vol. 3, no. 4, pp. 245–259, 1992.

[48] Y. Hurmuzlu and D. B. Marghitu, "Rigid body collisions of planar kinematic chains with multiple contact points," *Int. J. Robot. Res.*, vol. 13, no. 1, pp. 82–92, 1994.

[49] A. D. Ames, "Human-inspired control of bipedal walking robots," *IEEE Trans. Automat. Control*, vol. 59, no. 5, pp. 1115–1130, May 2014.

[50] H. Zhao, A. Hereid, W.-L. Ma, and A. D. Ames, "Multi-contact bipedal robotic locomotion," *Robotica*, vol. 35, pp. 1072–1106, 2015.

[51] H. H. Zhao, W. L. Ma, A. D. Ames, and M. B. Zeagler, "Human-inspired multi-contact locomotion with AMBER2," in *Proc. ACM/IEEE Int. Conf., Cyber-Phys. Syst.*, Apr. 2014, pp. 199–210.

[52] A. D. Ames, "First steps toward automatically generating bipedal robotic walking from human data," in *Proc. Robot Motion Control*, New York, NY, USA: Springer-Verlag, 2011, pp. 89–116.

[53] B. Morris and J. W. Grizzle, "Hybrid invariant manifolds in systems with impulse effects with application to periodic locomotion in bipedal robots," *IEEE Trans. Automat. Control*, vol. 54, no. 8, pp. 1751–1764, Aug. 2009.

[54] A. D. Ames, E. A. Cousineau, and M. J. Powell, "Dynamically stable bipedal robotic walking with NAO via human-inspired hybrid zero dynamics," in *Proc. 15th ACM Int. Conf. Hybrid Syst., Comput. Control*, 2012, pp. 135–144.

[55] A. Hereid, Christian M. Hubicki, E. A. Cousineau, J. W. Hurst, and A. D. Ames, "Hybrid zero dynamics based multiple shooting optimization with applications to robotic walking," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2015, pp. 5734–5740.

[56] J. Stoer and R. Bulirsch, *Introduction to Numerical Analysis* (Texts in Applied Mathematics), 3rd ed. New York, NY, USA: Springer, 2002.

[57] C. De Boor, C. De Boor, E.-U. Mathématicien, C. De Boor, and C. De Boor, *A Practical Guide to Splines*, vol. 27. New York, NY, USA: Springer-Verlag, 1978.

[58] J. R. Gilbert, C. Moler, and R. Schreiber, "Sparse matrices in MATLAB: Design and implementation," *SIAM J. Matrix Anal. Appl.*, vol. 13, no. 1, pp. 333–356, 1992.

[59] R. Jafari, L. L. Flynn, A. Hellum, and R. Mukherjee, "Energy-conserving gaits for point-foot planar bipeds: A five-DOF case study," in *Proc. ASME Dyn. Syst. Control Conf.*, 2013, Paper DSCC2013-3723.

[60] W. L. Ma, H. H. Zhao, S. Kolathaya, and A. D. Ames, "Human-inspired walking via unified PD and impedance control," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2014, pp. 5088–5094.

[61] A. Wächter and T. L. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Math. Program.*, vol. 106, no. 1, pp. 25–57, 2005.

[62] "DURUS 3D flat-footed walking gait," 2016. [Online]. Available: https://youtu.be/zpWmKQzexSQ

[63] E. Cousineau and A. D. Ames, "Realizing underactuated bipedal walking with torque controllers via the ideal model resolved motion method," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2015, pp. 5747–5753.

[64] S. H. Collins and A. Ruina, "A bipedal walking robot with efficient and human-like gait," in *Proc. IEEE Int. Conf. Robot. Autom.*, Apr. 2005, pp. 1983–1988.

[65] "DURUS 2D heel-toe walking gait," 2016. [Online] Available: https://youtu.be/X0eVwGCHVJU

[66] M. J. Powell, A. Hereid, and A. D. Ames, "Speed regulation in 3D robotic walking through motion transitions between human-inspired partial hybrid zero dynamics," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2013, pp. 4803–4810.

[67] N. Dantam, A. Hereid, A. Ames, and M. Stilman, "Correct software synthesis for stable speed-controlled robotic walking," in *Proc. 9th Robot., Sci. Syst. Conf.*, 2013, pp. 24–28. [Online]. Available: http://hdl.handle.net/1853/48196

[68] A. Mazumdar *et al.*, "Using parallel stiffness to achieve improved locomotive efficiency with the Sandia STEPPR robot," in *Proc. IEEE Conf. Robot. Autom.*, 2015, pp. 835–841.

[69] "Dynamics walking on DURUS (a.k.a. PROXI) at the 2015 DRC finals." [Online]. Available: https://youtu.be/a-R4H8-8074

[70] "DURUS walks like a human," 2016. [Online] Available: https://www.youtube.com/watch?v=1fC7b2LjVW4

[71] J. P. Reher, A. Hereid, S. Kolathaya, C. M. Hubicki, and A. D. Ames, "Algorithmic foundations of realizing multi-contact locomotion on the humanoid robot DURUS," in *Proc. Int. Workshop Algorithmic Found. Robot.*, 2016.

Ayonga Hereid (M'16) received the Ph.D. degree in mechanical engineering from Georgia Institute of Technology, Atlanta, GA, USA, in 2016.

He is currently a Postdoctoral Research Fellow with the EECS Department, University of Michigan, Ann Arbor, MI, USA. His research interests include the intersection of nonlinear control and optimization theory, with a particular focus on developing elegant and principled control solutions for complex robotic systems, including bipedal robots and exoskeletons.

Dr. Hereid was the recipient of the Best Student Paper Award in 2014 from the ACM International Conference on Hybrid System: Computation and Control and was nominated as the Best Conference Paper Award Finalists in 2016 at the IEEE International Conference on Robotics and Automation.

Christian M. Hubicki (M'15) received the Ph.D. degree in robotics/mechanical engineering from Oregon State University, Corvallis, OR, USA, in 2015.

He was a Postdoctoral Research Fellow in mechanical engineering with Georgia Institute of Technology from 2015 to 2016. He is currently a postdoctoral fellow with the School of Physics, Georgia Institute of Technology, Atlanta, GA, USA. His research interests include bipedal robotics, nonlinear optimization, nonlinear control, biomechanics, and soft-matter physics. Much of his work to date has focused on fundamental questions of design and control in bipedal robotic locomotion on unstructured terrain. His overall research portfolio has striven to connect robotics with the basic sciences through a broadly interdisciplinary research approach.

Eric A. Cousineau (M'16) received the B.S. and M.S. degrees in mechanical engineering from Texas A&M University, College Station, TX, USA, working on Valkyrie and DURUS as part of Aaron Ames's Bipedal Locomotion group, in 2011 and 2014.

He is currently a Software Engineer with Toyota Research Institute, helping to develop Drake, a software framework for planning, control, and analysis of nonlinear dynamical systems. His current research interests include the implementation of perception and control algorithms for design, testing, and analysis.

Aaron D. Ames (SM'17) received the B.S. degree in mechanical engineering and the B.A. degree in mathematics from University of St. Thomas, St. Paul, MN, USA, in 2001, and the M.A. degree in mathematics and the Ph.D. degree in electrical engineering and computer sciences from University of California (UC), Berkeley, Berkeley, CA, USA, in 2006.

He is the Bren Professor of mechanical and civil engineering and control and dynamical systems with Caltech, Pasadena, CA, USA. Prior to joining Caltech in 2017, he was an Associate Professor with Georgia Tech in the Woodruff School of Mechanical Engineering and the School of Electrical and Computer Engineering. He was a Postdoctoral Scholar with Control and Dynamical Systems, Caltech, from 2006 to 2008, and began his faculty career at Texas A&M University in 2008. His research interests includes the areas of robotics, nonlinear control and hybrid systems, with a special focus on applications to bipedal robotic walking—both formally and through experimental validation. His lab designs, builds and tests novel bipedal robots, humanoids and prostheses with the goal of achieving human-like bipedal robotic locomotion and translating these capabilities to robotic assistive devices.

Prof. Ames was the recipient of the 2005 Leon O. Chua Award for achievement in nonlinear science at UC Berkeley, the 2006 Bernard Friedman Memorial Prize in applied mathematics, and the NSF CAREER award in 2010 and the 2015 Donald P. Eckman Award.