2024 American Control Conference (ACC)
July 8-12, 2024. Toronto, Canada

Safe Whole-Body Task Space Control for Humanoid Robots

Victor C. Paredes[1] and Ayonga Hereid[1]

Abstract— **Complex robotic systems require whole-body controllers to handle contact interactions, handle closed kinematic chains, and track task-space control objectives. However, for many applications, safety-critical controllers are essential to steer away from undesired robot configurations and prevent unsafe behaviors. A prime example is legged robotics, where we can have tasks such as balance control, regulation of torso orientation, and, most importantly, walking. As the coordination of multi-body systems is non-trivial, following a combination of those tasks might lead to configurations that are deemed dangerous, such as stepping on its support foot during walking, leaning the torso excessively, or producing excessive centroidal momentum, resulting in non-human-like walking. To address these challenges, we propose a formulation of an inverse dynamics control enhanced with control barrier functions that allow general higher-order relative degree safe sets for robotic systems with numerous degrees of freedom. Our approach utilizes a quadratic program that respects closed kinematic chains, minimizes the control objectives, and imposes desired constraints on the Zero Moment Point, friction cone, and torque. More importantly, it also ensures the forward invariance of a general user-defined high Relative-Degree safety set. We demonstrate the effectiveness of our method by applying it to the 3D biped robot Digit, both in simulation and with hardware experiments.**

I. Introduction

Humanoid robots have emerged as a highly promising platform for performing complex tasks in human-centered environments due to their anthropomorphic structure. With dedicated legs and arms, these robots are well-equipped to walk and manipulate objects simultaneously. However, effectively coordinating the movements of legs and arms safely and stably is a challenging task. The dynamic coupling between these components makes independent control prone to instability and subpar performance. Therefore, developing a holistic controller that can safely coordinate the entire body is necessary to accomplish these tasks while respecting the robot's dynamics. Safety considerations are of utmost importance when deploying complex robots in real-world scenarios. Even if a desired task can be controlled, not taking into account safety measures might generate unsatisfactory performance. For instance, a non-safe task-space controller may successfully track a desired swing foot pose of a humanoid robot. However, it may fail to check if any leg joints are approaching mechanical limits or the robot is at risk of self-collision. By incorporating a safety layer, the controller explicitly verifies and enforces control solutions prioritizing safety.

*This work was supported in part by the National Science Foundation under grant FRR-21441568.

[1]Mechanical and Aerospace Engineering, Ohio State University, Columbus, OH, USA. (paredescauna.1, hereid.1)@osu.edu.

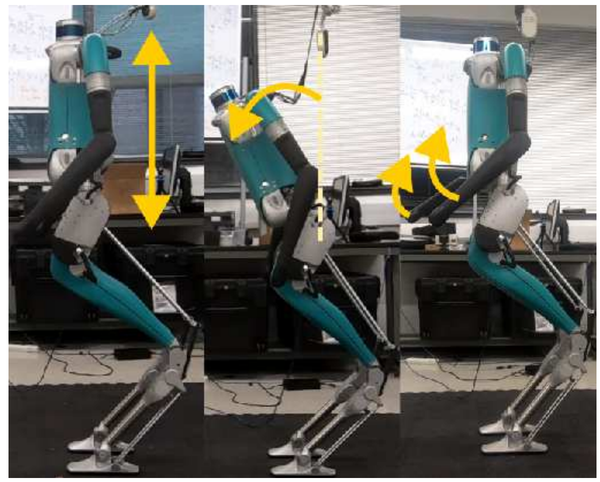

Fig. 1. The humanoid robot Digit performs different tasks involving CoM motion, torso orientation, and arm movements using our proposed safe whole body task space control.

Safety-critical control systems have been extensively studied using barrier certificates [1], [2], defined by state-dependent sets described by a function that must remain positive. A control algorithm can ensure this safe set remains invariant by appropriately restricting the controller action. There has been active research using the concept of Control Barrier Functions (CBFs) within the context of Lyapunov Theory [3] and optimization [4]–[6] to express safety certificates. Notably, the formulation of High-Order Control Barrier Function [6] (HOCBF) and Exponential Control Barrier Functions [5] (ECBF) has been a significant advancement, as it allows safety constraints to have arbitrarily high relative degrees, making it less restrictive for real-world applications as it enables to go beyond geometrical constraints [7], [8]. Note that ECBFs are a particular case of ECBFs; moreover, both formulations are suitable for use in an optimization-based controller.

Task-space control has been extensively studied using both model-based inverse dynamics [9] and model-free inverse kinematics approaches [10]. Inverse dynamics offers the advantage of considering model constraints such as contact constraints, friction cone, zero moment point, and torque limits. However, controlling bipedal systems presents challenges due to their intrinsic under-actuation and floating base. Nakanishi et al. [11] proposed a closed-form control solution that estimates contact forces to obtain the constrained dynamics and the Jacobian projection of the task space objectives. An improved version of this controller [12] utilizes orthogonal decomposition to work in a reduced dimensional space and avoids the need for estimating contact forces. However, these formulations do not explicitly incorporate contact wrenches, limiting their ability to introduce additional relevant constraints such as the zero moment point or friction.

To effectively address these constraints, it is crucial to explicitly consider the contact wrenches in an inverse dynamics controller that leverages the dynamics of user-defined general task outputs. Herzog et al. [13] proposed a quadratic programming (QP) formulation that incorporates the robot dynamics and treats the contact wrenches as decision variables. This QP-based approach offers advantages in reducing the complexity of matrix operations and enabling the handling of multiple constraints. Building upon this foundation, Reher et al. [14] introduced a similar QP optimization structure to construct a Control Lyapunov Function (CLF) that respects constraints such as the zero moment point (ZMP), contact, and friction cone. Instead of using the robot torques as the decision variable, it used a decision variable that utilized acceleration, torque, and constraint wrenches, providing enhanced control capabilities over these variables and avoiding solving the constrained dynamics explicitly. However, the inverse dynamics formulations in these works do not consider safety. The work of [8] provides a formulation of whole-body control with a Control Barrier Function (CBF) designed explicitly for position-based objectives. On the other hand, Nguyen et al. [15], [16] presented a QP-based controller incorporating an ECBF. This formulation extends the CBF by considering general state-space-based safety sets with arbitrary relative degrees. They employed a CLF-based controller that explicitly constructs safety certificates in both cases. Their results show applications to constrain footstep placements on stepping stones and limit the velocity to avoid collisions on cruise control. However, their optimization formulation utilized only motor torques as decision variables, resulting in increased numerical complexity due to the inversion of the mass matrix. Nevertheless, the formulation does not consider friction or ZMP constraints important for more realistic implementations. To overcome this challenge, we extend the work of Reher et al. [14] by reformulating the barrier functions as an acceleration-based certificate instead of a torque-based certificate. The safety certificates can be based on ECBF or more generally HOCBF; however, ECBFs are considered for simplicity. This novel formulation avoids explicit dependence on torques and effectively alleviates the numerical cost associated with the QP formulation.

The main contribution of this paper is the development of a novel Quadratic-Programming (QP)-based safe inverse dynamics controller that offers several key advantages:

1) **Avoidance of mass matrix inversions:** Building upon previous work on inverse dynamic formulations, we leverage a numerically efficient program that explicitly considers joint accelerations, torques, and wrenches as decision variables. This approach eliminates the need for computationally expensive mass matrix inversions.
2) **Handling of kinematics constraints:** Our QP-based inverse dynamics formulation enables the straightforward incorporation of closed-loop kinematics and other essential constraints, such as contacts, zero moment point (ZMP), and the friction cone. This flexibility allows for more accurate and realistic modeling of the robot's behavior.
3) **Enforcement of safety through acceleration-based exponential control barrier functions (A-ECBFs):** To ensure the safety of the system, we construct an exponential control barrier function that guarantees the invariant behavior of a predefined safe set. By formulating this safety certificate as an inequality constraint dependent solely on joint accelerations, we exploit the inherent structure of the controller formulation.
4) **Application of safe control actions to a 3D humanoid robot:** We demonstrate the performance and effectiveness of our controller through extensive simulation and hardware experiments on a 3D humanoid robot. These experiments validate the controller's ability to achieve desired tasks while maintaining safety.

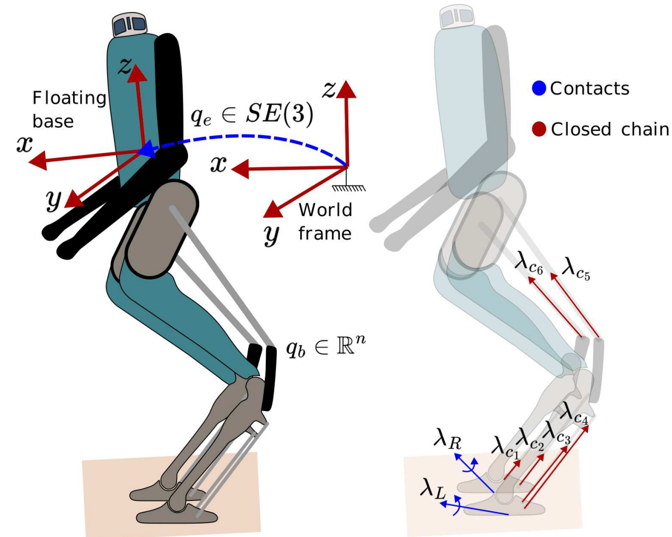

Fig. 2. A humanoid robot with floating base is described by its internal body coordinates q_b and its floating coordinates q_e as shown in the left. Furthermore, this robot experiences contact wrenches and forces due to the closed chain mechanisms as seen in the right.

The remainder of the paper is organized as follows. Section II presents the mathematical modeling of humanoid robots with floating base coordinates, contacts, closed-loop mechanisms, and critical dynamics constraints. In Section III, we present a task-space inverse dynamics control algorithm that is expressed as a quadratic program, followed by formulating an Acceleration-based Exponential Control Barrier Function that can be naturally included in the inverse dynamics formulation. In Section IV, we showcase the effectiveness of our whole body controller and its safety enforcement with various tasks for the 3D bipedal robot, Digit (Fig. 1). This section provides empirical evidence of the controller's performance and demonstrates its ability to handle complex tasks.

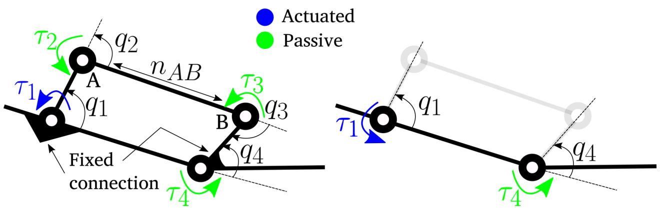

Fig. 3. The closed loop kinematics on the right can be expressed as the open kinematic chain on the left with a holonomic constraint that enforces n_{AB} constant. This constraint will relate the actuated torque τ_1 to the passive joint torque τ_4.

II. Humanoid Dynamics with Constraints

The kinematics of humanoid robots can be described by a floating frame fixed to a base point, introducing respective floating coordinates $q_e \in SE(3)$, as shown in Fig. 2, and the body coordinates describe the relative motion of its joints. For a robot with n_b joints, we represent body coordinates as $q_b \in \mathbb{R}^{n_b}$. The configuration space Q of a legged robot with a floating base, such as Digit as shown in Fig. 2, then can be represented by $q = [q_e^\top, q_b^\top]^\top \in Q = \mathbb{R}^n$ with $n = n_b + 6$ being the total degrees of freedom of the robot. The dynamics of the multi-body system can be described by the Euler Lagrangian equations of motion [17]:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = Bu + J(q)^\top \lambda, \tag{1}$$

where $M(q) \in \mathbb{R}^{n\times n}$, $C(q,\dot{q}) \in \mathbb{R}^{n\times n}$, and $G(q) \in \mathbb{R}^n$ are inertia matrix, Coriolis matrix, and gravity vector, respectively, $B \in \mathbb{R}^{n\times m}$ is the torque distribution matrix that maps the torque of the m actuators $u \in U \subset \mathbb{R}^m$, $\lambda \in \mathbb{R}^{n_h}$ is the collection of constraint wrenches or external forces, and $J(q) \in \mathbb{R}^{n_h \times n}$ its respective Jacobian matrix. Constraint wrenches are due to kinematic closed-chains, or contacts, as illustrated in Fig. 2 and discussed below.

A. Closed kinematic chain constraints

A closed kinematic chain, such as the four-bar linkage shown in Fig. 3, is popular in legged robot designs. However, many existing controllers do not explicitly address constraints associated with closed-chain kinematics [18]. To model a closed kinematic chain without resolving the constrained dynamics, we can virtually disconnect each closing link and use their lengths given by $n_k(q) \in \mathbb{R}, \forall k \in \Omega_{chain}$ (e.g., n_{AB} in Fig. 3) to construct appropriate holonomic constraints that enforce $\ddot{n}_k(q,\dot{q},\ddot{q}) = 0, \forall k \in \Omega_{chain}$ [19]. The inertial effects of the connecting rods are typically neglected as they have a much smaller mass than other links. The collection of such constraints generates,

$$J_{chain}(q)\ddot{q} + \dot{J}_{chain}(q,\dot{q})\dot{q} = 0 \tag{2}$$

where, $J_{chain}(q)$ is the jacobian of the collection of the closed kinematic constraints.

B. Contact constraints

The contact constraints are included whenever the robot must preserve contact with a point or a surface. While such ground contacts are unilateral, they can be modeled as holonomic constraints with additional inequality constraints describing the limitations imposed by friction and the direction of the normal force. Similarly to the closed kinematic constraints, for a set of contacts $c \in \Omega_{cont}$, and their respective pose $n_c(q) \in \mathbb{R}^6$, we enforce its invariance through:

$$J_{cont}(q)\ddot{q} + \dot{J}_{cont}(q,\dot{q})\dot{q} = 0 \tag{3}$$

Additionally, to impose friction constraints and avoid slipping we use the pyramidal friction cone approximation [17]. For each contact wrench we can decompose it into forces and moments as $\lambda^c = \left[\lambda^c_{fx}, \lambda^c_{fy}, \lambda^c_{fz}, \lambda^c_{mx}, \lambda^c_{my}, \lambda^c_{mz}\right]^T$ indexed by $c \in \Omega_{cont}$. We impose the following constraints that can be expressed in linear form,

$$|\lambda^c_{fx}| \le \frac{\mu}{\sqrt{2}} \lambda^c_{fz}, \tag{4}$$

$$|\lambda^c_{fy}| \le \frac{\mu}{\sqrt{2}} \lambda^c_{fz}, \tag{5}$$

$$\lambda^c_{fz} > 0, \tag{6}$$

by incorporating a soft-finger contact type:

$$|\lambda^c_{mz}| \le \gamma \lambda^c_{fz}, \tag{7}$$

where γ is the torsional friction coefficient and λ^c_{mz} is the rotational moment component of a wrench along the z-axis of the contact frame.

Additional Zero Moment Point (ZMP) constraints should be enforced to prevent tipping over edges [17], [20] during fully actuated operation, i.e., when the support ankle is not passive. The typical contact cases are single and double support configurations, as seen in Fig. 4. For a series of contacts, we need to project each wrench into a unique frame, for instance, the world frame with an associated global wrench λ^w. We can achieve this by applying the adjoint transformation (represented by Ad_g) to every single contact wrench to obtain the equivalent contact wrench expressed in the world frame:

$$\lambda^w = \sum_{i=1}^{N_c} Ad^T_{g_{c\to w}}(q)\lambda^c_i = \mathcal{A}(q)\lambda^c \tag{8}$$

where $g_{c\to w}$ is the homogeneous transformation matrix of the world frame w.r.t. the contact frame and $c \in \Omega_{cont}$.

The ZMP is computed as $p^x_{zmp} = \lambda^w_{my}/\lambda^w_{fz}$, $p^y_{zmp} = -\lambda^w_{mx}/\lambda^w_{fz}$ and must be inside the support polygon $\mathcal{P}$ defined by the contact geometry, as shown in Fig. 4, i.e. $\begin{bmatrix} p^x_{zmp} & p^y_{zmp} \end{bmatrix} \in \mathcal{P}$. In the simplest case $\mathcal{P}$ can be represented as a rectangle and the ZMP constraints can be written as:

$$L_{min}\lambda^w_{fz} \le \lambda^w_{my} \le L_{max}\lambda^w_{fz} \tag{9}$$

$$W_{max}\lambda^w_{fz} \le \lambda^w_{mx} \le W_{min}\lambda^w_{fz} \tag{10}$$

Since the global wrench λ^w is linearly related to the contact wrenches λ^c as described by (8), the constraints in (9) and (10) are also linear in λ.

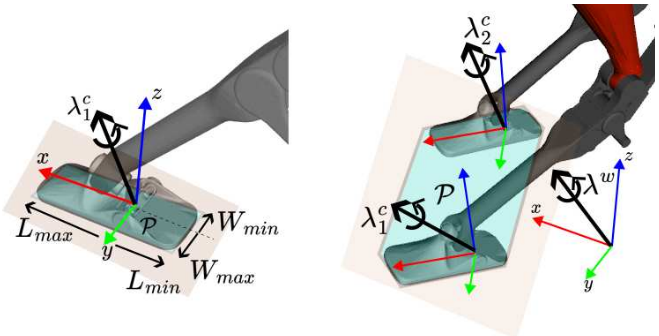

Fig. 4. Typical cases: One contact point on the left and two on the right; in both cases, the wrenches are projected into λ^w for ZMP computations.

III. Safe Task Space Whole Body Control via CBF-QP

This section presents an inverse dynamics controller based on quadratic programming (QP) that incorporates holonomic constraints arising from contacts and closed chain mechanisms. This approach avoids the need for matrix inversions and the calculation of constrained dynamics, providing numerical benefits. Subsequently, we introduce our paper's main contribution: applying a safety certificate as an Exponential Control Barrier Function (ECBF) to ensure the invariance of a user-defined safe set. Specifically, we formulate an acceleration-based ECBF compatible with the whole-body inverse dynamics control, as shown in Fig. 5.

A. Task Space Inverse dynamics

We consider task space outputs as general positions and orientations of user-defined frames. These outputs can specify the robot's desired behavior, such as the center of mass (CoM) position and orientation and the end-effector poses of the robot's arms and legs. For simplicity, we present an output that is of relative degree two, given as:

$$y(t,q) := y^a(q) - y^d(t), \tag{11}$$

where, $y^a(q), y^d(t) \in \mathbb{R}^m$ are the actual and desired outputs respectively. Since the task considers positions and orientations, the outputs are relative degree two. In consequence,

$$\ddot{y}(t,q,\dot{q},\ddot{q}) = \dot{J}_y(q,\dot{q})\dot{q} + J_y(q)\ddot{q} - \ddot{y}^d(t) \tag{12}$$

where $J_y(q) \in \mathbb{R}^{m\times n}$ is the jacobian of $y^a(q)$. Moreover, a relative degree two output will exhibit dependency in the control input u; however, to avoid solving the dynamics, we can leave it in terms of the joint acceleration $\ddot{q}$ and implicitly use the robot dynamics to relate u.

We formulate the inverse dynamics problem as a quadratic program using the decision variable $X := [\ddot{q}^T, u^T, \lambda^T]^T$. Hence, the implicit constrained robot dynamics in (1) - (3) can be rewritten as:

$$\underbrace{\begin{bmatrix} M(q) & -B & -J(q)^T \end{bmatrix}}_{D_{eq}(q)} X = -C(q,\dot{q})\dot{q} - G(q), \tag{13}$$

$$\underbrace{\begin{bmatrix} J(q) & 0 & 0 \end{bmatrix}}_{C_{eq}(q)} X = -\dot{J}(q,\dot{q})\dot{q}, \tag{14}$$

where $J(q) = \begin{bmatrix} J_{chain}(q) \\ J_{cont}(q) \end{bmatrix}$. To obtain stable output dynamics, we enforce a linear output dynamics $\ddot{y}(t,q,\dot{q},\ddot{q}) = -K_p y(t,q) - K_d \dot{y}(t,q,\dot{q})$, which holds true whenever,

$$J_y(q)\ddot{q} + \dot{J}_y(q,\dot{q})\dot{q} = \underbrace{-K_p y(t,q) - K_d \dot{y}(t,q,\dot{q}) + \ddot{y}^d(t)}_{y^*} \tag{15}$$

where, $K_p, K_d \in \mathbb{R}^{m\times m}$ are positive definite gain matrices. We formulate the QP-based inverse dynamics controller with the optimal decision variable X^* as,

$$\begin{aligned} X^* = \underset{X\in\mathcal{X}}{\operatorname{argmin}} & ||J_y(q)\ddot{q} + \dot{J}_y(q,\dot{q})\dot{q} - y^*||^2 + X^T\Gamma X \\ \text{s.t} \quad & D_{eq}(q)X = -C(q,\dot{q}) - G(q) \\ & C_{eq}(q)X = -J(q)\dot{q} \\ & A(q)X \in \mathcal{X} \end{aligned} \tag{16}$$

where $\mathcal{X}$ captures the domain constraints such as torque limits and also friction cone, and ZMP constraints expressed on (4) - (7), (9) and (10). We use the regularization term $X^T\Gamma X$, with $\Gamma > 0$ with a weight represented by a diagonal matrix, to avoid high-frequency changes in the optimization variables.

B. Exponential control barrier functions

To enhance the capabilities of the inverse dynamics control, we incorporate safety through the use of Exponential Control Barrier Functions (ECBF) [5], which are a particular case of the High-Order Control Barrier Function (HOCBF) [6]. They ensure the forward invariance of a user-defined safe set. Moreover, we will utilize an acceleration-based formulation of the ECBF (A-ECBF), which uses the acceleration $\ddot{q}$ to maintain the inverse dynamics approach's numerical benefits by avoiding the explicit analytic solution of the constrained dynamics. For our formulation, we do not require analytically constrained dynamics. However, we include it here to illustrate the standard derivation of the ECBF and highlight our equivalent approach, A-ECBF, that requires no explicit dependency on the control variable u.

1) Review of ECBFs: Consider the state $x = \left[q^T, \dot{q}^T\right]^T \in D \subset TQ$, where D is an open set of admissible states. By solving analytically the constrained dynamics we can obtain the following state-space representation,

$$\dot{x} = f(x) + g(x)u \tag{17}$$

where, $f(x) \in \mathbb{R}^{2n}, g(x) \in \mathbb{R}^{2n\times m}$ and $u \in \mathbb{R}^m$. A safe region can be defined as,

$$\mathcal{C} := \{x \in D : h(x) \geq 0\}, \tag{18}$$

where $h : D \to \mathbb{R}$ is, without loss of generality, assumed to be a smooth function with relative degree r_b, i.e.,

$$h^{(r_b)}(x,u) = L_f^{r_b} h(x) + L_g L_f^{r_b-1} h(x) u. \tag{19}$$

where, $L_f^{r_b} h(x)$, $L_g L_f^{r_b-1} h(x) \in \mathbb{R}$ are the respective Lie derivatives of $h(x)$. Moreover, $L_g L_f^{r_b-1} h(x)$ is non-zero by the definition of the relative degree. We stack the lower derivatives to form the following state,

$$\eta_b(x) = \begin{bmatrix} h(x) \\ h^{(1)}(x) \\ \vdots \\ h^{(r_b-1)}(x) \end{bmatrix} = \begin{bmatrix} h(x) \\ L_f h(x) \\ \vdots \\ L_f^{r_b-1} h(x) \end{bmatrix} \tag{20}$$

and construct a linear representation of the dynamics of $h(x)$ through the mapping $L_f^{r_b} h(x) + L_g L_f^{r_b-1} h(x) u = \nu_b$,

$$\dot{\eta}_b = \underbrace{\begin{bmatrix} \mathbf{0} & I \\ \mathbf{0} & \mathbf{0} \end{bmatrix}}_{F_b} \eta_b + \underbrace{\begin{bmatrix} \mathbf{0} \\ I \end{bmatrix}}_{G_b} \nu_b \tag{21}$$

$$h(x) = \underbrace{\begin{bmatrix} 1 & 0 & 0 & \dots & 0 \end{bmatrix}}_{C_b} \eta_b \tag{22}$$

Next, we need a constraint on ν_b such that $h(x) \geq 0$. Applying the feedback $\nu_b = -K_\alpha \eta_b$, the trajectory becomes $h(x) = C_b e^{(F_b - G_b K_\alpha)t} \eta_b(x_0)$.

By the comparison lemma, setting $\nu_b \geq -K_\alpha \eta_b$ implies $h(x(t)) \geq C_b e^{(F_b - G_b K_\alpha)t} \eta_b(x_0)$.

Definition 1 (**Exponential Control Barrier Function [5]**). *Consider the set $\mathcal{C} \subset D$ defined in (18) as the super-level set of a r_b times continuously differentiable function $h : D \to \mathbb{R}$. Then, h is an exponential control barrier function (ECBF) if there exists $K_\alpha \in \mathbb{R}^{1\times r_b}$ such that,*

$$\sup_{u \in U} \left[L_f^{r_b} h(x) + L_g L_f^{r_b-1} h(x) u \right] \geq -K_\alpha \eta_b(x) \tag{23}$$

$\forall x \in Int(\mathcal{C})$ originates $h(x(t)) \geq C_b e^{(F_b - G_b K_\alpha t)} \eta_b(x_0) \geq 0$ for $h(x_0) \geq 0$.

Such K_α can be determined by considering the feedback $\nu_b = -K_\alpha \eta_b$. Its corresponding closed loop system $\dot{\eta}_b = (F_b - G_b K_\alpha)\eta_b$ has r_b roots that are dependent on the selection of K_α and denoted by $p_b = -[p_1, \dots, p_{r_b}]$. We will construct a family of functions $\mathsf{B}_i : D \to \mathbb{R}$ such that,

$$\mathsf{B}_0(x) = h(x) \tag{24}$$

$$\mathsf{B}_i(x) = \dot{\mathsf{B}}_{i-1}(x) + p_i \mathsf{B}_{i-1}(x), \quad \forall i = 1, \dots, r_b \tag{25}$$

Note that if $\mathsf{B}_i(x) \geq 0$ then $\mathsf{B}_{i-1}(x) \geq 0$ whenever $p_i \geq 0$, $\mathsf{B}_{i-1}(x_0) \geq 0$ and $\mathsf{B}_i(x_0) \geq 0$, where x_0 is the initial state of the system at $t = 0$. This inductive constraint will terminate at $\mathsf{B}_0(x) = h(x)$ which is the barrier function of the safe set $\mathcal{C}$. This is the basis for the next theorem that is proved in [5].

Theorem 1. *[5] Consider the closed loop system $\dot{\eta}_b = (F_b - G_b K_\alpha)\eta_b$. If K_α is designed to have roots $p_b = -[p_1, \dots, p_{r_b}]$ to be,*

$$p_i > 0, \quad p_i \geq -\frac{\dot{\mathsf{B}}_{i-1}(x_0)}{\mathsf{B}_{i-1}(x_0)}, \quad \forall i = 1, \dots, r_b \tag{26}$$

Then, the constraint $\nu_b \geq -K_\alpha \eta_b$ renders $h(x)$ an ECBF.

2) Formulation of A-ECBFs: The ECBF requires to compute the Lie Derivatives of $h(x)$ considering the constrained dynamics in (17) to have a mapping to u. However, to avoid explicitly solving the constrained dynamics, we can directly use the accelerations $\ddot{q}$ that will appear during the partial derivative operation in $h(x)$ without considering the system dynamics. Instead, we expect the quadratic program to implicitly solve the mapping of $\ddot{q}$ to u through the robot dynamics. In other words, there exists an inverse dynamics mapping that relates accelerations and torques such that $u = ID(q, \dot{q}, \ddot{q}) = \bar{M}\ddot{q} + \bar{H}$ for some $\bar{M} \in \mathbb{R}^{n\times n}$ and $\bar{H} \in \mathbb{R}^n$. Therefore can rewrite (19) as

$$h^{(r_b)}(x, \ddot{q}) = L_f^{r_b} h(x) + L_g L_f^{r_b-1} h(x)(\bar{M}\ddot{q} + \bar{H}) \tag{27}$$

$$h^{(r_b)}(x, \ddot{q}) = \mathcal{F}(x) + \mathcal{G}(x)\ddot{q} \tag{28}$$

By choosing the mapping $\mathcal{F}(x) + \mathcal{G}(x)\ddot{q} = \nu_b$ (equivalent to the mapping in the ECBF case), we reach the same linear dynamics shown in (21). Therefore, using this expression, we present an equivalent definition of (23), considering joint accelerations and avoiding the explicit solution of the constrained dynamics needed to compute (17).

Definition 2 (**Acceleration based exponential control barrier function (A-ECBF)**). *The function $h(x)$ is an A-ECBF if there exists K_α such that,*

$$\sup_{\ddot{q} \in \mathbb{R}^n} [\mathcal{F}(x) + \mathcal{G}(x)\ddot{q}] \geq -K_\alpha \eta_b(q, \dot{q}) \tag{29}$$

$\forall x \in Int(\mathcal{C})$ originates $h(x(t)) \geq C_b e^{(F_b - G_b K_\alpha t)} \eta_b(x_0) \geq 0$ for $h(x_0) \geq 0$.

Note that, given previous knowledge of the relative degree of $h(x)$, computing $\mathcal{F}(x)$ and $\mathcal{G}(x)$ is easier to compute than $L_f^{r_b} h(x)$ and $L_g L_f^{r_b-1} h(x)$ in several practical cases. For instance, the CoM height is relative degree two, so to bound $h(x) = p_{CoM}^z(x) - p_{CoM}^{z-min}$ we only need to compute $h^{(2)} = \dot{J}_{CoM}\dot{q} + J_{CoM}\ddot{q}$.

The design of K_α follows the **Theorem 1** rationale. Once we have a suitable value for it, we include the constraint (29) into our inverse dynamics controller (16). In other words, we simply add the A-ECBF certificate as an additional inequality constraint as,

$$\mathcal{F}(x) + \mathcal{G}(x)\ddot{q} \geq -K_\alpha \eta_b \tag{30}$$

This new addition enforces forward invariance of the set $\mathcal{C}$ using our equivalent definition of an A-ECBF in the inverse dynamics formulation (16).

IV. Simulation and Experimental Results

We implement the controller in simulation and the robotic hardware using a unique code structure and controller gains. Our test bed is Digit, a 3D bipedal robot with arms, legs, and

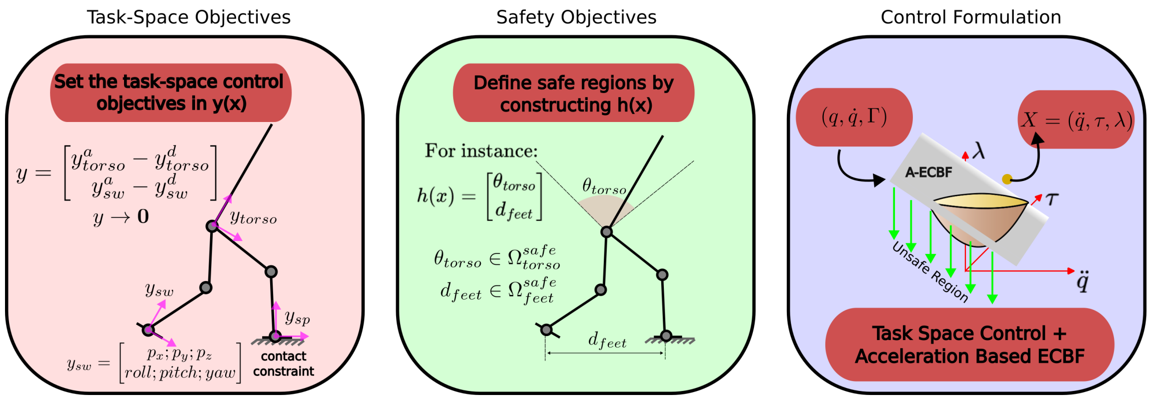

Fig. 5. The safe task-space control framework starts by defining control objectives and safety sets according to the application. The optimization will search a control action in the space of safe solutions.

a torso developed by Agility Robotics. It weighs 45 Kg and has 30 joints with 20 motors. Each leg presents three closed kinematic chains and two spring joints, as seen in Fig. 6.

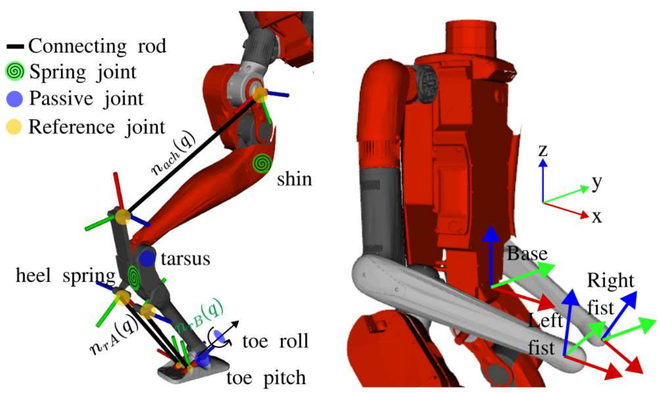

Fig. 6. The figure on the left shows the passive joints and the closed chain loops in the robot legs. The figure on the right shows the fist and the base frames.

We consider the springs as rigid joints due to their high stiffness. Those spring joints will be considered part of the kinematic constraints (kc). The entire kinematic constraints are defined as $n_{kc} = \begin{bmatrix} n_{kc}^L & n_{kc}^R \end{bmatrix}^T$, where L/R stands for left and right, respectively, and $n_{kc}^{L/R} = \begin{bmatrix} n_{ach} & n_{rA} & n_{rB} & q_{shin} & q_{heel-spring} \end{bmatrix}_{L/R}^T$, where, n_{ach} is the length of the achilles rod, n_{rA} and n_{rB} are the lengths of the rods connecting to the ankles while q_{shin} and $q_{heel-spring}$ are the spring joints considered as fixed, as shown in Fig. 6.

We parameterize time by $\tau = t - t_0$, where t_0 indicates the starting time of a step. We show the results in both, plots and video [1], for different whole body motions including walking.

A. Task: Squatting and bowing

In this case we showcase the whole-body controller without safety constraints. During the Double Support domain,

[1]https://youtu.be/vNTIcODR6cI

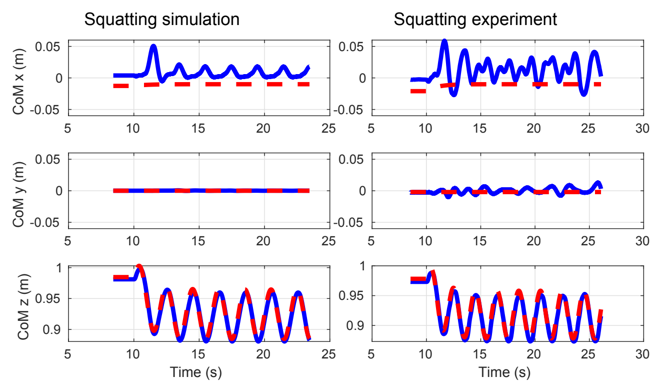

Fig. 7. The squatting motion starts at $t = 10s$ for both simulation and experiment. The blue lines are the actual outputs, and the red dashed lines represent the reference trajectories.

the controlled outputs are:

$$y(q,t) = \begin{bmatrix} p_{CoM}(q) \\ \theta_{torso}(q) \\ q_{arms}(q) \end{bmatrix} - \begin{bmatrix} p_{CoM}^d(t) \\ \theta_{torso}^d(t) \\ q_{arms}^d(t) \end{bmatrix} \tag{31}$$

where, $p_{CoM} \in \mathbb{R}^3$ is the position of the center of mass, $\theta_{torso} \in \mathbb{R}^3$ is the torso orientation expressed in Euler ZYX angles and, $q_{arms} \in \mathbb{R}^8$ are the angular positions of the joints on the left and right arm. Note that during this domain, the robot must keep both feet in contact with the ground, i.e., meet the ZMP and friction constraints.

We test two continuous actions: (1) a squatting motion by specifying a sinusoidal reference to the CoM's height and (2) a bowing motion by commanding the torso pitch to extend and return to its initial pose. The squatting reference is,

$$p_{CoM}^d(t) = \begin{bmatrix} -0.02 \\ 0 \\ 1 - 0.12(1 - e^{-\tau}) + 0.03 sin(\pi\tau) \end{bmatrix} \tag{32}$$

During this motion, we keep the torso orientation straight $\theta_{torso}^d = \mathbf{0}$ and the arms fixed. Fig. 7 shows the controller's performance in simulation and hardware experiments.

Regarding the bowing motion, we set the following reference for the torso pitch, $\theta_{torso}^d(t) =$

$\begin{bmatrix}0 & 0.45\max(3-|\tau-3|,0) & 0\end{bmatrix}^T$ and the other tasks are specified as $p^d_{CoM} = \begin{bmatrix}0 & 0 & 0.95\end{bmatrix}^T$ and the arms fixed. Applying our controller in the hardware results on the behavior observed in Fig. 8.

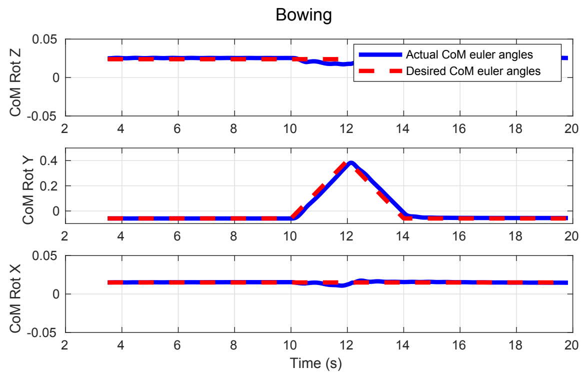

Fig. 8. Tracking the torso orientation while realizing a bowing movement between 10-14 seconds.

B. Task: Arm motion with height limits

We set arm trajectories during a double support domain, keeping the robot's CoM and torso orientation fixed and straight. We will focus on a simple case of the safety-critical feature of the controller by providing an A-ECBF for the first height with respect to the base frame, as seen in Fig. 6. To illustrate the effect of the A-ECBF, we will command both arms, left and right, with the same reference but only equip the left fist with the safety certificate. We consider the safety certificate as,

$$h(q) = -p^L_z(q) - 0.195 \geq 0 \tag{33}$$

where, $p^L_z(q)$ is the z-position of the fist of the left arm w.r.t to the torso base, as seen in Fig. 9. Since $h(x)$ is relative degree two, the A-ECBF constraint takes the form of $J_{p_z}\ddot{q} + \dot{J}_{p_z}\dot{q} \geq -K_\alpha \eta$. Moreover,

$$\dot{\eta} = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\eta + \begin{bmatrix}0\\ 1\end{bmatrix}\ddot{q} \tag{34}$$

where, $J_{p_z} = -\frac{\partial p^L_z}{\partial q}$ and $\eta = [-p^L_z - 0.195, -\dot{p}^L_z]^T$. We design K_α such that the roots of $\dot{\eta} = (F - GK_\alpha)\eta$ follow **Theorem 1**. The references for both left and right arm are $q^d_{arm_L}$ and $q^d_{arm_R}$ respectively, $q^d_{arm_L} = \begin{bmatrix}0 & 0.3\sin(\frac{\pi}{5}\tau) & 0 & 0.2\sin(\frac{\pi}{5}\tau)\end{bmatrix}^T$ and $q^d_{arm_R} = \begin{bmatrix}0 & -0.3\sin(\frac{\pi}{5}\tau) & 0 & -0.2\sin(\frac{\pi}{5}\tau)\end{bmatrix}^T$.

Applying the QP-based controller with the A-ECBF results on the motion shown in Fig. 9 in hardware. We note that the right fist crosses the threshold to reach its desired target, while the left fist avoids it. Fig. 10 shows the tracking of the other objectives.

C. Task: Walking with the A-LIP template model

In the following tasks, we use a single support domain that will enable bipedal walking. We use the Angular Momentum-based Linear Inverted Pendulum (ALIP) model to generate stable walking patterns [21]. In the ALIP, the support ankle

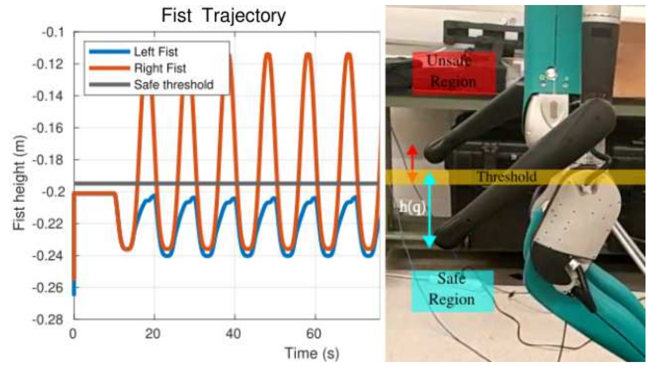

Fig. 9. The trajectory of the left fist and right fist. The left fist is constrained by an A-ECBF that prevents it from crossing the safe threshold, while the right fist is not constrained and crosses it at $t = 30s$ during the hardware experiments.

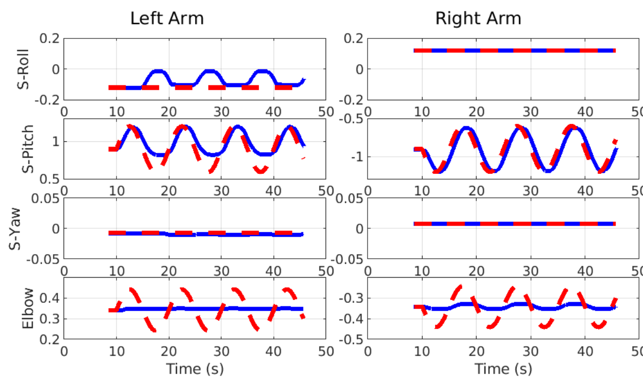

Fig. 10. Arm joint tracking during the hardware experiment. The blue lines represent the actual joint positions, and the red dashed lines show the desired trajectories.

(pitch and roll) is rendered passive to predict the angular momentum at the end of the step. This strategy allows us to plan for stabilizing foot positioning targets (u_x, u_y) and the feet height profile to impact at a specified period $T = 0.35s$. A complete description of the planner can be found at [21], [22]. The swing foot outputs are defined as:

$$p^d_{swing}(\tau) = \begin{bmatrix}x_{sw}(\tau_0) + s(u_x - x_{sw}(\tau_0))\\ y_{sw}(\tau_0) + s(u_y - y_{sw}(\tau_0))\\ 0.08sin(\frac{2\pi\tau}{T})\end{bmatrix} \tag{35}$$

where $s = (1 - e^{-5\frac{\tau}{T}})$ is an smoothing factor. The other ALIP requirements are: the CoM height is constant ($H = 0.9$), the orientation of the torso is kept vertical, and the swing foot remains flat. The controller's application over the outputs produces a stable gait with an average forward speed of 0.2 m/s. The task space tracking for the position of the outputs is summarized in Fig. 11.

D. Task: Collision avoidance

Continuing with the ALIP based gait, we show the effect of an instantaneous ($\delta t = 0.15s$) lateral external force at the torso with magnitude $F_{ext} = -30N$. This produces a collision between the legs, as seen in Fig. 12. However, by providing collision safety through an A-ECBF, we can avoid

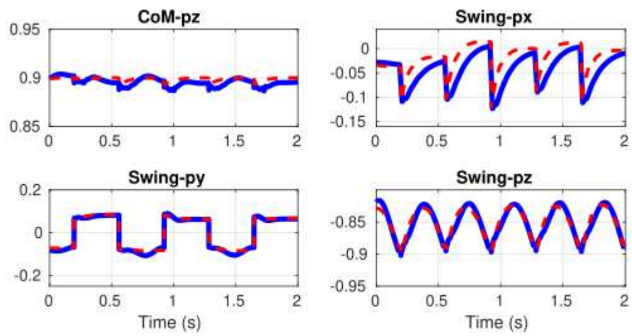

Fig. 11. Task space tracking under the ALIP planner in the Mujoco simulator. We note a good tracking performance during the walking gait, and a similar performance is observed for the orientation tasks.

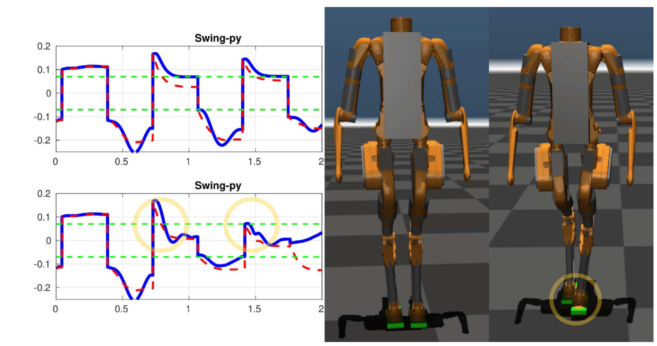

Fig. 12. During the Mujoco simulation, we observe the swing foot tracking in the y-direction. The upper plot has an A-ECBF, while the lower does not. The foot placement reference (red dashed line), crosses the safe region in both plots. However, the A-ECBF prevents the foot from entering it. The yellow circles show the collision events.

this event and recover the balance. The safe set is defined by, $h_{ALIP}(q) = y_{sw}(q) - 0.07 \geq 0$, where, $y_{sw}(q)$ is the y-position of the swing foot w.r.t the support foot. By enforcing the constraint, the swing foot will maintain a distance margin that will avoid collisions. Fig. 12 shows that the reference provided by the ALIP planner crosses the safety region, but the A-ECBF disallows the foot to enter that region.

V. Conclusions

The whole body controller presented realizes stable motions that respect its closed chain kinematic, ZMP, and other physical constraints. We achieved fast squatting and bowing movements that show the controller's capabilities during double support. We also conducted walking experiments with 0.2m/s of speed and a stepping time of $T = 0.35s$ to show fast single support events handling. Furthermore, we showed the formulation of the A-ECBF to provide control safety. This safe controller has the numerical benefit of expressing the dynamics and the constraints separately and avoiding computation of the constrained dynamics. The results of the A-ECBF were applied to both the arms and the legs to show its effectiveness in different scenarios. In general, the results show similar performance between simulation and hardware experiments regarding tracking and safety.

References

[1] S. Prajna, "Barrier certificates for nonlinear model validation," *Automatica*, vol. 42, no. 1, pp. 117–126, 2006.

[2] S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," in *International Workshop on Hybrid Systems: Computation and Control*. Springer, 2004, pp. 477–492.

[3] S. Hsu, X. Xu, and A. D. Ames, "Control barrier function based quadratic programs with application to bipedal robotic walking," in *American Control Conference, ACC 2015, Chicago, IL, USA, July 1-3, 2015*, IEEE. IEEE, 2015, pp. 4542–4548.

[4] Q. Nguyen and K. Sreenath, "Exponential control barrier functions for enforcing high relative-degree safety-critical constraints," in *2016 American Control Conference (ACC)*. IEEE, 2016, pp. 322–328.

[5] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European Control Conference (ECC)*. IEEE, 2019, pp. 3420–3431.

[6] W. Xiao and C. Belta, "High-order control barrier functions," *IEEE Transactions on Automatic Control*, vol. 67, no. 7, pp. 3655–3662, 2021.

[7] A. Del Prete, "Joint position and velocity bounds in discrete-time acceleration/torque control of robot manipulators," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 281–288, 2017.

[8] C. Khazoom, D. Gonzalez-Diaz, Y. Ding, and S. Kim, "Humanoid self-collision avoidance using whole-body control with control barrier functions," in *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*. IEEE, 2022, pp. 558–565.

[9] J. Nakanishi, R. Cory, M. Mistry, J. Peters, and S. Schaal, "Operational space control: A theoretical and empirical comparison," *The International Journal of Robotics Research*, vol. 27, no. 6, pp. 737–757, 2008.

[10] V. Paredes and A. Hereid, "Resolved motion control for 3d underactuated bipedal walking using linear inverted pendulum dynamics and neural adaptation," *arXiv preprint arXiv:2208.01786*, 2022.

[11] J. Nakanishi, M. Mistry, and S. Schaal, "Inverse dynamics control with floating base and constraints," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, 2007, pp. 1942–1947.

[12] M. Mistry, J. Buchli, and S. Schaal, "Inverse dynamics control of floating base systems using orthogonal decomposition," in *2010 IEEE international conference on robotics and automation*. IEEE, 2010, pp. 3406–3412.

[13] A. Herzog, L. Righetti, F. Grimminger, P. Pastor, and S. Schaal, "Balancing experiments on a torque-controlled humanoid with hierarchical inverse dynamics," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, Sept. 2014, pp. 981–988.

[14] J. Reher, C. Kann, and A. D. Ames, "An inverse dynamics approach to control lyapunov functions," in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 2444–2451.

[15] Q. Nguyen, A. Hereid, K. Sreenath, J. W. Grizzle, and A. D. Ames, "3D dynamic walking on stepping stones with control barrier functions," in *Proc. IEEE 55th Conf. Decision and Control (CDC)*. Las Vegas, NV: IEEE, Dec. 2016, pp. 827–834.

[16] Q. Nguyen and K. Sreenath, "Safety-critical control for dynamical bipedal walking with precise footstep placement," *IFAC-PapersOnLine*, vol. 48, no. 27, pp. 147–154, 2015.

[17] 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.

[18] F. L. Moro and L. Sentis, "Whole-body control of humanoid robots," *Humanoid Robotics: A Reference*, Jan. 2018.

[19] J. Reher and A. D. Ames, "Control lyapunov functions for compliant hybrid zero dynamic walking," *arXiv preprint arXiv:2107.04241*, 2021.

[20] M. Vukobratović and B. Borovac, "Zero-moment point: thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, no. 01, pp. 157–173, 2004.

[21] Y. Gong and J. W. Grizzle, "Zero Dynamics, Pendulum Models, and Angular Momentum in Feedback Control of Bipedal Locomotion," *Journal of Dynamic Systems, Measurement, and Control*, vol. 144, no. 12, 10 2022, 121006.

[22] G. Gibson, O. Dosunmu-Ogunbi, Y. Gong, and J. Grizzle, "Terrain-adaptive, alip-based bipedal locomotion controller via model predictive control and virtual constraints," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 6724–6731.