3D Dynamic Walking on Stepping Stones with Control Barrier Functions

Quan Nguyen[1], Ayonga Hereid[2], Jessy W. Grizzle[3], Aaron D. Ames[2], Koushil Sreenath[1]

***Abstract*— 3D dynamical walking subject to precise footstep placements is crucial for navigating real world terrain with discrete footholds. We present a novel methodology that combines control Lyapunov functions—to achieve periodic walking—and control Barrier functions—to enforce strict constraints on step length and step width—unified in a single optimization-based controller. We numerically validate our proposed method by demonstrating dynamic 3D walking at 0.6 m/s on DURUS, a 23 degree-of-freedom underactuated humanoid robot.**

I. Introduction

The primary advantage of legged locomotion on robotic systems is the ability to traverse terrain not accessible by wheeled devices—this can be canonically represented by terrain with discrete footholds such as "stepping stones." Yet current approaches to handling this terrain type use simplistic methods—both at the level of models and control of walking robots—to plan footstep locations and quasi-statically drive a fully-actuated humanoid robot to achieve the desired foot placements. The overarching goal of this work is to create a formal framework that will enable bipedal humanoid robots to achieve dynamic and rapid locomotion over a randomly placed set of stepping stones.

Footstep placement for fully-actuated legged robots essentially rely on quasi-static walking using the ZMP criterion that requires slow walking speeds and small steps [13], [7]. Impressive results in footstep planning and placements in obstacle filled environments with vision-based sensing was carried out in [6]. However, these ZMP-based methods impose strict restrictions on the walking gaits as they rely on kinematics of quasi-static motions or simple dynamical models such as the linear inverted pendulum with massless legs, see [8], [18]. Moreover, these methods typically require full-actuation, cannot handle compliance well, and are not applicable for dynamic walking with faster walking gaits.

This paper presents initial results on precise footstep placement for 3D dynamic walking. The proposed method is based on feedback control of the full nonlinear and underactuated hybrid dynamic model of bipedal robots to achieve periodic dynamic walking gaits with formal stability guarantees that enforce the safety-critical foot placement constraints. The main contribution of this paper is a novel safety-critical control strategy that can guarantee precise footstep placement for 3D dynamic walking of a high-dimensional bipedal robot. We achieve this by combining (a) control Lyapunov function based quadratic programs (CLF-QPs) [9] for enforcing virtual constraints, that represent a nominal gait, while simultaneously respecting the saturation limits of the actuators, and (b) control Barrier function based quadratic programs (CBF-QPs) [4] to guarantee state dependent safety constraints.

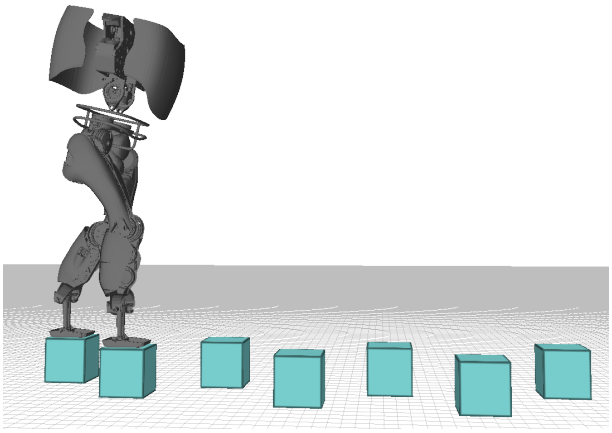

Fig. 1: The problem of dynamically walking in 3D over a randomly generated set of discrete footholds. Simulation video: `https://youtu.be/yUSTraDn9-U`.

The goal of this paper is to relax the tracking behavior of the nominal gait by enforcing a set of state-dependent safety constraints, governed by control Barrier functions, that guide the swing foot trajectory to the discrete footholds. Our method enables dealing with a large range of desired foothold separations with precise placement of footsteps on small footholds. This requires simultaneously guaranteeing precise step length and step width constraints at foot contact. This work builds off our recent work on precise footstep placement for planar (2D) walking [14]. In comparison to our prior work, this paper makes the following additional contributions:

- We consider 3D dynamic bipedal walking in contrast to planar walking.
- We extend the applicability of our method to a 23 degree-of-freedom bipedal system (up from 7 degree-of-freedom model in planar walking) establishing scalability of the proposed method.
- We consider additional degrees of underactuation in the form of compliant feet.

[1]Dept. of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213.

[2]Dept. of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332.

[3]Dept. of Electrical Engineering and Computer Science, Univ. of Michigan, Ann Arbor, MI 48109.

This work is supported by NSF grants IIS-1526515, IIS-1526519, and IIS-1525006.

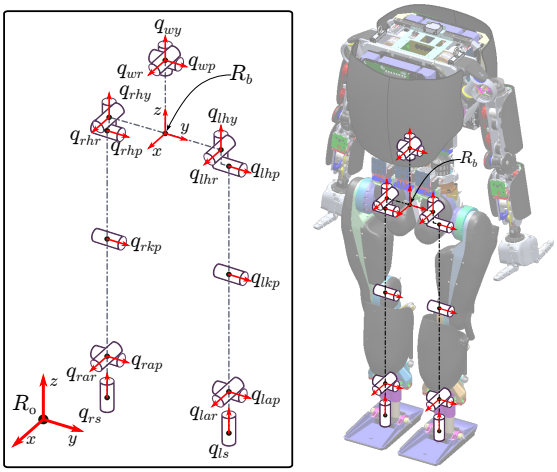

Fig. 2: The Humanoid DURUS is a 23 degree-of-freedom system, which break down into 15 actuated joints, 2 passive springs at the feet, and a 6 degree-of-freedom floating base. The coordinates of DURUS are illustrated with the red arrows representing the positive rotation (or translation) axis of the robot joints.

- In addition to step length constraints in planar walking, we simultaneously address both step length and step width constraints.

The rest of the paper is organized as follows. Section II presents the hybrid dynamical model of DURUS, a 3D humanoid robot. Section III revisits control Lyapunov function-based quadratic programs (CLF-QPs). Section IV presents exponential control Barrier functions (ECBFs) for enforcing safety constraints. Section V presents the proposed ECBF-CLF-QP based feedback controller for enforcing precise footstep placement for 3D dynamic walking. Section VI presents numerical validation of the controller on DURUS. Finally, Section VII provides concluding remarks.

II. Hybrid System Model

In this section, we briefly discuss the two-domain hybrid system model for the flat foot walking of underactuated 3D robot, DURUS. Hybrid zero dynamics control framework is also concisely introduced as a way to synthesize walking gait for the robot.

A. DURUS Model

DURUS is an underactuated humanoid robot with 15 actuated joints and two passive springs (see Fig. 2), designed and built by SRI International for the study of high efficiency multi-domain bipedal locomotion [12], [17]. The two passive linear springs are attached to the end of each ankle joint, such that they are rigidly perpendicular to the foot, and are designed for reducing energy loss and mitigating mechanical shocks at impacts.

In this paper, the generalized floating-base coodinates, $q = [p_b, \phi_b, q_r]^T \in \mathcal{Q} = \mathbb{R}^3 \times SO(3) \times \mathcal{Q}_r$, of the robot is used to model this high-dimensional humanoid, where $p_b \in \mathbb{R}^3$ is the Cartesian position and $\phi_b \in SO(3)$ is the orientation of the body base frame R_b–which is attached to the center of the pelvis link–with respect to the world frame, and $q_r \in \mathcal{Q}_r$ is the 17-dimensional joint coordinates of DURUS, as shown in Fig. 2 (see [17] for more details.)

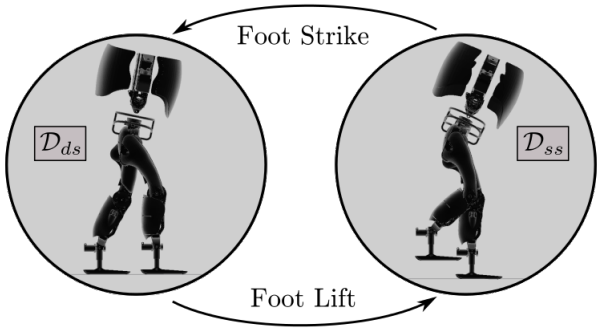

Fig. 3: The directed cycle structure of the multi-domain hybrid system model for flat-foot humanoid walking.

B. Hybrid System Model for Bipedal Walking

Due to the existence of both continuous and discrete dynamics, bipedal locomotion is naturally modeled as a hybrid control system [11]. The existence of passive springs leads to two continuous domain behaviors in the case of flat-foot walking: a *double support* domain, where both feet are on the ground, and a *single support* domain, where the non-stance foot is above the ground while only the stance foot stays on the ground, as shown in Fig. 3. Therefore, the multi-domain hybrid control system of DURUS walking is defined as a *tuple* [1]:

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

where $\Gamma = \{V, E\}$ is a directed cycle with vertices $V = \{ds, ss\}$, where ds represents the double support domain and ss represents the single support domain, and the edges $E = \{ds \to ss, ss \to ds\}$, as depicted in Fig. 3. See [12] for the detailed definition of (1).

The domain breakdown is determined by the changes in the contact condition with the environment. Often we use holonomic constraints, $\eta_v(q)$, to model the foot contact with the ground [10]. Given the mass, inertia, length and the center of mass position of each link of the robot, the affine control system $\{f_v, g_v\}$ of a continuous domain determined by the Lagrangian dynamics of the multi-link rigid body system and contact holonomic constraints with the environment is given by

$$\dot{x} = f_v(x) + g_v(x)u_v, \tag{2}$$

where $x = (q, \dot{q}) \in T\mathcal{Q}$ is the system state.

The transition from the double support to single support domain occurs when the normal ground force on the non-stance foot crosses zero. Hence, the reset map for this edge is an identity map, i.e., $\Delta_{ds \to ss} = \mathbf{I}$. On the other hand, the reset map from the single support to double support incorporates the impact dynamics when the non-stance foot hits the ground, during which the joint velocities undergo

discrete changes due to the introduction of new contact constraints. Given the pre-impact states $(q^-, \dot{q}^-)$, the post impact states $(q^+, \dot{q}^+) = \Delta_{ss\to ds}(q^-, \dot{q}^-)$ are determined by assuming a perfectly plastic impact of the rigid body model [12].

C. Hybrid Zero Dynamics (HZD) Control Framework

Given the hybrid control system model, we now present the *hybrid zero dynamics* framework in which virtual constraints are employed as a method to synthesize feedback control to render stable walking of DURUS.

Virtual Constraints. Any admissible state-based feedback controller that has been applied to the control system, *FG*, yields a closed-loop hybrid system [1]. This can be done by defining a set of virtual constraints–also referred as *outputs*, which is the difference between actual and desired outputs– and applying feedback controllers to drive them to zero [19].

In this paper, we use the same set of virtual constraints defined in [12]. In particular, the linearized forward hip velocity is chosen as the relative degree one output $y^a_{1,v} = \delta\dot{p}_{hip}(q, \dot{q})$ to regulate the forward velocity of the robot during both domains, i.e., $v \in \{ds, ss\}$.

We define that the desired velocity is parameterized by a constant v_d, i.e., $y_{1,v}(q, \dot{q}, v_d) = y^a_{1,v}(q, \dot{q}) - v_d$. Desired relative degree two outputs $y^d_{2,v}(\tau(q), \alpha_v)$ represented by 7^{th} order Bézier polynomials [19] parameterized by a set of parameters α_v with $v \in \{ds, ss\}$. The virtual constraints on $\mathcal{D}_v$ then can be defined as:

$$y_{2,v}(q, \alpha_v) = y^a_{2,v}(q) - y^d_{2,v}(\tau(q), \alpha_v), \tag{3}$$

where $\tau(q)$ is a monotonic state-based parameterization of time, defined as $\tau(q) = \frac{\delta p_{hip}(q) - \delta p_{hip}(q^+)}{\delta p_{hip}(q^-) - \delta p_{hip}(q^+)} \in [0, 1]$, where δp_{hip} is the linearized hip position. In particular, the desired outputs of the stance and non-stance foot orientations are set to be zero respectively to keep the feet being flat throughout the step.

To drive the virtual constraints $y_v = (y_{1,v}, y_{2,v}) \to 0$ exponentially for each $v \in \{ds, ss\}$, we utilize the feedback linearization control law

$$u^\varepsilon_v = -\mathcal{A}_v^{-1}\left((\mathcal{L}^2_f)_v + \mu^\varepsilon_v\right), \tag{4}$$

where $\mathcal{A}_v = [\mathcal{L}_{g_v} y_{1,v}(q, \dot{q}); \mathcal{L}_{g_v}\mathcal{L}_{f_v} y_{2,v}(q)]$ is the decoupling matrix, and $(\mathcal{L}^2_f)_v = [0; \mathcal{L}_{f_v}\mathcal{L}_{f_v} y_{2,v}(q)]$, with $\mathcal{L}$ being the Lie derivative. With the given control law, we have the output dynamics $(\dot{y}_{1,v}, \ddot{y}_{2,v}) = -\mu^\varepsilon_v$, where μ^ε_v can be chosen so that the outputs converge to zero exponentially at a rate of $\varepsilon > 0$.

Partial Hybrid Zero Dynamics. Moreover, the control law in (4) renders the *full zero dynamics surface* exponentially stable and invariant over both continuous domains. Due to the impact of the non-stance foot, however, the invariance of the full zero dynamics surface is no longer guaranteed. Particularly, it would be impossible to enforce the relative degree one output to be invariant through impact due to the changes in joint velocities. Therefore, here we only enforce the *partial hybrid zero dynamics* for the two-domain walking of DURUS; see [1]. Specifically, consider the surface:

$$\mathcal{PZ}_v = \{(q, \dot{q}) \in \mathcal{D}_v : y_{2,v}(q) = 0, \dot{y}_{2,v}(q, \dot{q}) = 0\},$$

with $v \in \{ds, ss\}$. Therefore, the goal of designing a periodic and dynamic walking gait is to find a set of parameters $\alpha = \{v_d, \alpha_{ds}, \alpha_{ss}\}$ that ensures there exists a periodic orbit for the system in (1) and the partial zero dynamics surface is invariant through impact. The process of finding α can be formulated as a nonlinear constrained optimization problem subject to the multi-domain hybrid system model and HZD control framework [12].

III. Control Lyapunov Functions Revisited

For each domain $v \in \{ds, ss\}$, one may choose μ_v in (4) in a way with which the resulting output dynamics is stable. Letting $\eta_v = (y_{1,v}, y_{2,v}, \dot{y}_{2,v})$, the linear output dynamics can be written as

$$\dot{\eta}_v = \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & I \\ 0 & 0 \end{bmatrix}}_{F_v} \eta_v + \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & I \end{bmatrix}}_{G_v} \mu_v. \tag{5}$$

Then in the context of this control system, we consider the continuous time algebraic Riccati equations (CARE):

$$F_v^T P_v + P_v F_v - P_v G_v G_v^T P_v + Q_v = 0, \tag{6}$$

for $Q_v = Q_v^T > 0$ with solution $P_v = P_v^T > 0$. One can use P_v to construct a RES-CLF that can be used to exponentially stabilize the output dynamics at a user defined rate of $\frac{1}{\varepsilon}$ (see [2], [3]). In particular, define

$$V^\varepsilon_v(\eta_v) = \eta_v^T \underbrace{I^\varepsilon P_v I^\varepsilon}_{P^\varepsilon_v} \eta_v, \qquad \text{with} \quad I^\varepsilon = \text{diag}(\frac{1}{\varepsilon} I, I), \tag{7}$$

wherein it follows that:

$$\dot{V}^\varepsilon_v(\eta_v) = \mathcal{L}_{F_v} V^\varepsilon_v(\eta_v) + \mathcal{L}_{G_v} V^\varepsilon_v(\eta_v)\mu_v,$$

with

$$\begin{aligned} \mathcal{L}_{F_v} V^\varepsilon_v(\eta_v) &= \eta_v^T (F_v^T P^\varepsilon_v + P^\varepsilon_v F_v)\eta_v, \\ \mathcal{L}_{G_v} V^\varepsilon_v(\eta_v) &= 2\eta_v^T P^\varepsilon_v G_v. \end{aligned} \tag{8}$$

With the goal of exponentially stabilizing the η_v to zero, we wish to find μ_v such that,

$$\mathcal{L}_{F_v} V^\varepsilon_v(\eta_v) + \mathcal{L}_{G_v} V^\varepsilon_v(\eta_v)\mu_v \le -\frac{\lambda}{\varepsilon} V^\varepsilon_v(\eta_v),$$

for some $\lambda > 0$. In particular, it allows for specific feedback controllers, e.g., the min-norm controller, which can be stated as the closed form solution to a quadratic program (QP). See [9], [5], [3] for the further information.

Recalling that $\mathcal{A}_v u_v = -(\mathcal{L}^2_f)_v + \mu_v$, it follows that:

$$\mu_v^T \mu_v = u_v^T \mathcal{A}_v^T \mathcal{A}_v u_v + 2(\mathcal{L}^2_f)_v^T \mathcal{A}_v u_v + (\mathcal{L}^2_f)_v^T (\mathcal{L}^2_f)_v,$$

which allows for reformulating the QP problem in terms of u_v instead of μ_v, so that additional constraints on torques

or reaction forces can be directly implemented in the formulation. To achieve an optimal control law, we can relax the CLF constraints and penalize this relaxation. In particular, we consider the following modified CLF-based QP in terms of u_v and a relaxation factor δ_v:

$$u_v^*(x) = \underset{(u_v,\delta_v)}{\operatorname{argmin}} \quad u_v^T \mathcal{A}_v^T \mathcal{A}_v u_v + 2(\mathcal{L}_f^2)_v^T \mathcal{A}_v u_v + p_v \delta_v^2 \quad (9)$$

$$\text{s.t} \qquad \tilde{A}_v^{\text{CLF}}(q,\dot{q}) u_v \leq \tilde{b}_v^{\text{CLF}}(q,\dot{q}) + \delta_v \quad \text{(CLF)}$$

where,

$$\begin{aligned} \tilde{A}_v^{\text{CLF}}(q,\dot{q}) :=& \mathcal{L}_{G_v} V_v^{\varepsilon}(q,\dot{q}) \mathcal{A}_v(q,\dot{q}), \\ \tilde{b}_v^{\text{CLF}}(q,\dot{q}) :=& -\frac{\lambda}{\varepsilon} V_v^{\varepsilon}(q,\dot{q}) - \mathcal{L}_{F_v} V_v^{\varepsilon}(q,\dot{q}) \\ & - \mathcal{L}_{G_v} V_v^{\varepsilon}(q,\dot{q}) (\mathcal{L}_f^2)_v, \end{aligned} \quad (10)$$

and $p_v > 0$ is a large positive constant that penalizes violations of the CLF constraint. Note that we use the fact that η_v is a function of the system states $(q,\dot{q})$, so the constraints can be expressed in the term of system states.

The end result of solving this QP is the optimal control law that guarantees exponential convergence of the control objective $(y_{1,v}, y_{2,v}) \to 0$ if $\delta_v \equiv 0$. In the case of sufficiently small δ_v, we still achieve exponential convergence of the outputs, which motivates the minimization of δ_v in the cost of QP.

IV. Control Barrier Functions

Having seen control Lyapunov function based controllers for achieving stable periodic walking, the goal of this section is to introduce control Barrier functions as means to enforce strict safety constraints. We will also introduce exponential control Barrier functions and then incorporate the control Barrier function constraints into the control Lyapunov function based controller introduced earlier.

A. Control Barrier Function

Consider a control affine system:

$$\dot{x} = f(x) + g(x)u, \quad (11)$$

with f and g locally Lipschitz, state variable $x \in \mathbb{R}^n$ and control input $u \in U \subset \mathbb{R}^n$. The goal is to design a controller to keep the state x in the set

$$\mathcal{C} = \{x \in \mathbb{R}^n : h(x) \geq 0\}, \quad (12)$$

where $h : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function. Then, a function $B : \mathcal{C} \to \mathbb{R}$ is a Control Barrier Function (CBF), [4], if there exists class $\mathcal{K}$ function α_1 and α_2 such that, for all $x \in Int(\mathcal{C}) = \{x \in \mathbb{R}^n : h(x) > 0\}$,

$$\frac{1}{\alpha_1(h(x))} \leq B(x) \leq \frac{1}{\alpha_2(h(x))}, \quad (13)$$

$$\dot{B}(x,u) = L_f B(x) + L_g B(x) u \leq \frac{\gamma}{B(x)}. \quad (14)$$

From [4], the important properties of the CBF condition in (14) is that if there exists a control Barrier function, $B : \mathcal{C} \to \mathbb{R}$, then $\mathcal{C}$ is forward invariant, or in other words, if $x(0) = x_0 \in \mathcal{C}$, i.e., $h(x_0) \geq 0$, then $x = x(t) \in \mathcal{C}, \forall t$, i.e., $h(x(t)) \geq 0, \forall t$.

B. Exponential Control Barrier Function

The above formulation of control Barrier functions required relative-degree one $h(x)$. In order to systematically design safety-critical controllers for higher order relative degree constraints, we will use "Exponential Control Barrier Functions" (ECBFs) [15].

With application to precise footstep placement, our constraints will be position based, $h(q) \geq 0$, which has relative degree 2. For this problem, we can design an Exponential CBF as follows:

$$B(q,\dot{q}) = \dot{h}(q,\dot{q}) + \beta h(q), \quad (15)$$

and the exponential CBF condition is simply:

$$\dot{B}(q,\dot{q},u) + \gamma B(q,\dot{q}) \geq 0, \quad (16)$$

where $\beta > 0, \gamma > 0$. Enforcing (16) will then enforce $B(q,\dot{q}) \geq 0$. Moreover, we also note that by plugging the ECBF (15) into the condition (16), we have,

$$(\frac{d}{dt} + \beta) \circ (\frac{d}{dt} + \gamma) \circ h(q) \geq 0. \quad (17)$$

Thus, β, γ play the role of pole locations for the constraint dynamics $\dot{h}(q,\dot{q},u)$.

C. Combination of CBF and CLF-QP

Consider the exponential control Barrier candidate function (15), then we can incorporate the condition (16) into the Quadratic Program (9) as follows:

$$u_v^*(x) = \underset{(u_v,\delta_v)}{\operatorname{argmin}} \quad u_v^T \mathcal{A}_v^T \mathcal{A}_v u_v + 2(\mathcal{L}_f^2)_v^T \mathcal{A}_v u_v + p_v \delta_v^2 \quad (18)$$

$$\begin{aligned} \text{s.t} \quad & \tilde{A}_v^{\text{CLF}}(q,\dot{q}) u_v \leq \tilde{b}_v^{\text{CLF}}(q,\dot{q}) + \delta_v && \text{(CLF)} \\ & A_v^{CBF}(q,\dot{q}) u_v \leq b_v^{CBF}(q,\dot{q}) && \text{(CBF)} \\ & u_{min} \leq u \leq u_{max} && \text{(TS)} \end{aligned}$$

where,

$$\begin{aligned} A_v^{CBF}(q,\dot{q}) :=& -\mathcal{L}_{G_v} B_v(q,\dot{q}), \\ b_v^{CBF}(q,\dot{q}) :=& \mathcal{L}_{F_v} B_v(q,\dot{q}) + \gamma B_v(q,\dot{q}). \end{aligned} \quad (19)$$

Because the CBF condition needs to be strictly satisfied, the same relaxation on CLF cannot be applied. Under certain aggressive cases the QP could be infeasible due to input saturation. In order to handle this issue, we will present a method to enlarge the feasible domain of the CBF by giving the QP freedom on optimizing γ without violating the constraint.

D. CBF with Time-Varying Gamma

From the exponential CBF condition,

$$\dot{B}(x,u) + \gamma B(x) \geq 0, \quad (20)$$

we have for all $\gamma \geq 0$,

$$B(t) \geq B(0) e^{-\gamma t} \geq 0. \quad (21)$$

This property is also satisfied with time varying $\gamma = \gamma(t)$. In particular, we assume $\gamma(t) \in (0, \gamma_{max}], \forall t$. Therefore:

$$\dot{B}(x,u) \geq -\gamma(t)B(x) \geq -\gamma_{max}B(x), \tag{22}$$

$$\implies B(x(t)) \geq B(0)e^{-\gamma_{max}t} \geq 0. \tag{23}$$

We then have, CBF-CLF-QP controller with time-varying γ:

$$u_v^*(x) = \underset{(u_v,\delta_v,\delta_\gamma)}{\operatorname{argmin}} \; u_v^T \mathcal{A}_v^T \mathcal{A}_v u_v + 2(\mathcal{L}_f^2)_v^T \mathcal{A}_v u_v + p_v\delta_v^2 + p_\gamma\delta_\gamma^2 \tag{24}$$

$$\text{s.t} \quad \tilde{A}_v^{\text{CLF}}(q,\dot{q})u_v \leq \tilde{b}_v^{\text{CLF}}(q,\dot{q}) + \delta_v \quad \text{(CLF)}$$

$$A_v^{CBF}(q,\dot{q})u_v \leq \tilde{b}_v^{CBF}(q,\dot{q}) \quad \text{(CBF)}$$

$$\gamma_d - \delta_\gamma \geq 0$$

$$u_{min} \leq u \leq u_{max} \quad \text{(TS)}$$

with

$$\tilde{b}_v^{CBF}(q,\dot{q}) := \mathcal{L}_{F_v}B_v(q,\dot{q}) + (\gamma_d - \delta_\gamma)B_v(q,\dot{q}). \tag{25}$$

Basically, the QP will optimize the parameter $\gamma(t) = \gamma_d - \delta_\gamma(t) \geq 0$ to increase the feasibility of the QP and we minimize $p_\gamma\delta_\gamma^2$ since we want the optimized parameter γ to be closed to the designed γ_d.

Having presented control Barrier function based quadratic programs, we will now formulate our controller to achieve dynamic walking subject to achieving precise footstep placements on discrete footholds.

V. 3D Stepping Stones

To enforce precise foot placements on the stepping stones, we need to enforce strict constraints on the step length and step width at swing foot touchdown (see Fig. 4). To achieve this, we will first develop CBFs that enforce strict step length constraints, followed by CBFs that enforce strict step width constraints. We will finally put these together to achieve simultaneous step length and step width constraints to achieve 3D walking on the stepping stones.

A. Stepping constraints for step length

If we want to force the robot to step onto a specific position (see Fig.5), we need to guarantee that the step length when the robot swing foot hits the ground is bounded within a given range $[l_{min}, l_{max}]$. Let $h_f(q)$ be the height of the swing foot to the ground and $l_f(q)$ be the distance between the stance and swing feet, as defined in Fig. 4. We define the step length at impact as,

$$l_s := l_f(q)|_{h_f(q)=0, \dot{h}_f(q,\dot{q})<0}. \tag{26}$$

The discrete foothold constraint to be enforced then becomes,

$$l_{min} \leq l_s \leq l_{max}. \tag{27}$$

In order to guarantee this final-time constraint, we formulate a new state-dependent constraint that when enforced during the entire step, will satisfy the discrete foothold constraint (27) at impact. This constraint arises through the geometric construction in Fig. 5. In particular, if we can guarantee the trajectory of the swing foot, F, to be bounded between the domain of the two circles O_1 and O_2 (with radii R_1, R_2),

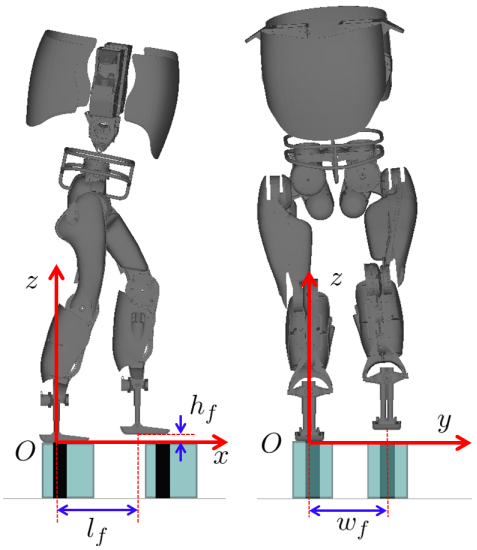

Fig. 4: DURUS swing foot coordinates: The swing foot position in 3D is defined by the step length, step width, and step height coordinates (l_f, w_f, h_f) along the x, y, z axes respectively.

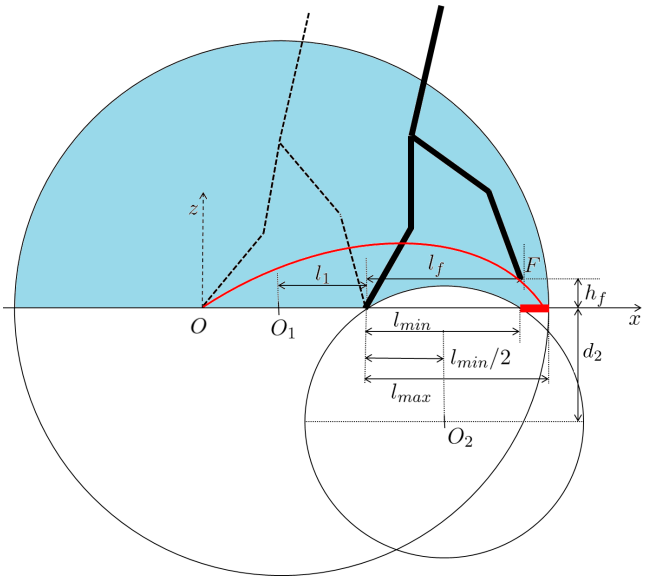

Fig. 5: Geometric explanation of the stepping stone foot placement constraint with only step length constraint.

then the step length at impact is guaranteed to be within $[l_{min}, l_{max}]$. Thus, a sufficient condition to enforce constraint (27) is to enforce the constraints:

$$O_1F \leq R_1, \quad \text{and} \quad O_2F \geq R_2, \tag{28}$$

where,

$$O_1F = \sqrt{(l_1 + l_f)^2 + h_f^2},$$
$$R_1 = l_1 + l_{max},$$
$$O_2F = \sqrt{(d_2 + h_f)^2 + (l_f - \frac{l_{min}}{2})^2},$$
$$R_2 = \sqrt{d_2^2 + (\frac{l_{min}}{2})^2}, \tag{29}$$

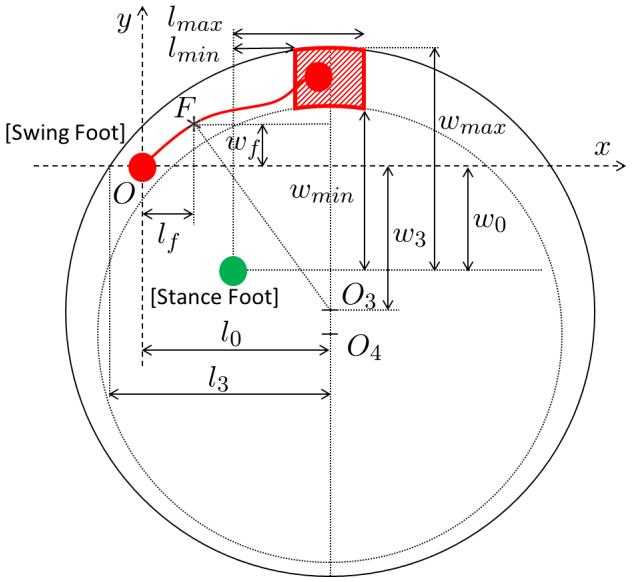

Fig. 6: Geometric explanation of the stepping stone foot placement constraint with only step width constraint.

with l_1 being the distance between the stance foot and O_1, as shown in Fig. 5. When the swing foot hits the ground at the end of the step, $h_f = 0, \dot{h}_f < 0$, and the step length is l_s. Substituting this into (29), the constraints (28) thus becomes,

$$\begin{aligned} \sqrt{(l_1 + l_s)^2} &\leq l_1 + l_{max}, \\ \sqrt{d_2^2 + (l_s - \frac{l_{min}}{2})^2} &\geq \sqrt{d_2^2 + (\frac{l_{min}}{2})^2}. \end{aligned} \tag{30}$$

This is equivalent to (27) since the constants l_1, l_s, l_{max} are positive. Thus, we need to enforce the position constraints,

$$\begin{aligned} h_1(q) &= R_1 - O_1F \\ &= l_1 + l_{max} - \sqrt{(l_1 + l_f(q))^2 + h_f(q)^2} \geq 0, \\ h_2(q) &= O_2F - R_2 \\ &= \sqrt{(d_2 + h_f)^2 + (l_f - \frac{l_{min}}{2})^2} - \sqrt{d_2^2 + (\frac{l_{min}}{2})^2} \geq 0. \end{aligned} \tag{31}$$

These position constraints can be enforced through the following control Barrier candidate functions:

$$\begin{aligned} B_1(q, \dot{q}) &= \gamma_b h_1(q) + \dot{h}_1(q, \dot{q}) \geq 0, \\ B_1(q, \dot{q}) &= \gamma_b h_2(q) + \dot{h}_2(q, \dot{q}) \geq 0. \end{aligned} \tag{32}$$

We now can apply the CBF-CLF-QP based controller presented in Section IV to enforce these Barrier constraints, resulting in the footstep placement constraint $l_s \in [l_{min}, l_{max}]$.

B. Stepping constraints for step width

A sufficient condition to guarantee the step width $w_f \leq w_{max}$, is to maintain the swing foot position to be inside the circle O_3 (see Fig.6), i.e.,

$$O_3F \leq R_3, \tag{33}$$

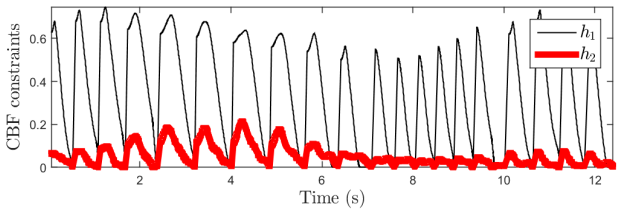

Fig. 7: CBF constraints for Case 1 (changing step length only).

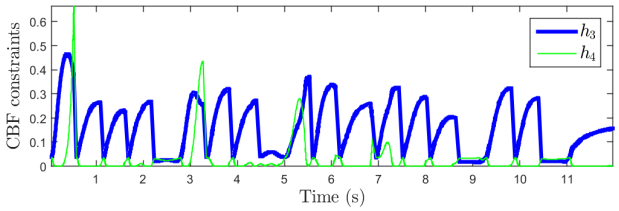

Fig. 8: CBF constraints for Case 2 (changing step width only).

where,

$$O_3F = \sqrt{(l_0 - l_f)^2 + (d_3 + w_f)^2}. \tag{34}$$

The circle O_3 is defined so that it is tangent to the maximum boundary of desired step width and containing the initial swing foot position (O). Therefore the radius and center of the circle O_3 can be derived from l_3 and w_{max} as follows:

$$\begin{aligned} R_3^2 &= l_3^2 + w_3^2 = l_3^2 + (R_3 - w_{max} + w_0)^2, \\ \implies R_3 &= \frac{l_3^2 + (w_{max} - w_0)^2}{2(w_{max} - w_0)}, \end{aligned} \tag{35}$$

where w_0 is the step width of the previous step. The same principle can be applied to enforce the minimum step width $w_f \geq w_{min}$ by requiring $O_4F \geq R_4$. These constraints can be enforced through a CBF controller from Section IV by defining the following position constraints,

$$\begin{aligned} h_3(q) &= R_3 - O_3F \geq 0, \\ h_4(q) &= O_4F - R_4 \geq 0, \end{aligned} \tag{36}$$

along with Barriers B_3, B_4 to enforce them. Thus, the step width will satisfy the desired foothold $w_s \in [w_{min}, w_{max}]$.

C. Stepping constraints for step length and step width

In order to control the footstep placement with step length constraint ($l_s \in [l_{min}, l_{max}]$) and step width constraint ($w_s \in [w_{min}, w_{max}]$) for the robot, we can apply the CBF-CLF-QP based controller with the four constraints $\begin{bmatrix} h_1(q) & h_2(q) & h_3(q) & h_4(q) \end{bmatrix} \geq 0$ as defined in (31), (36).

In particular, enforcing the step length and step width constraints in (31), (36) respectively, will guarantee the swing foot position to impact the ground inside the red shaded region depicted in Fig. 6.

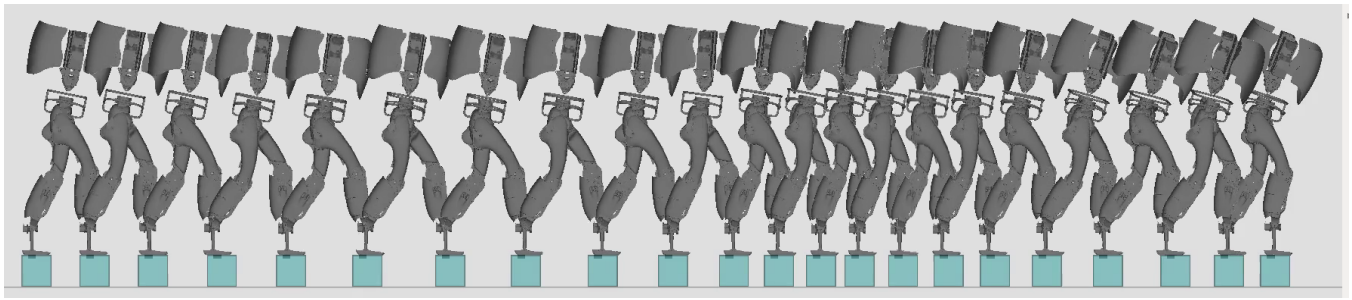

Fig. 14: Snapshots of simulation results (Case 1: Changing step length only). Simulation video: `https://youtu.be/yUSTraDn9-U`.

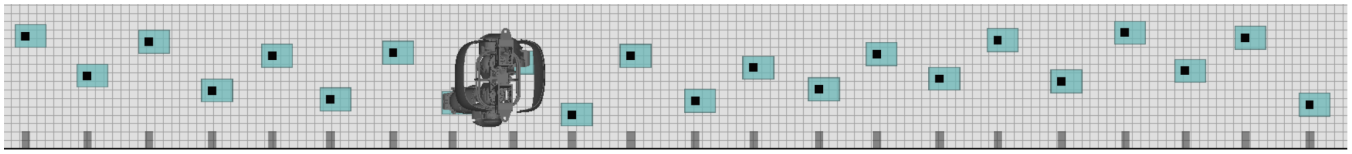

Fig. 15: Snapshots of simulation results (Case 2: Changing step width only). The small vertical bars at the bottom of the figure illustrate that the step length doesn't change.

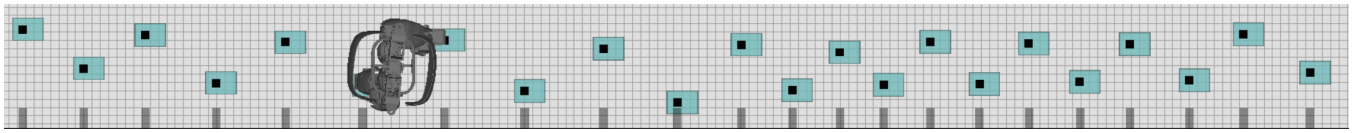

Fig. 16: Snapshots of simulation results (Case 3: Changing step length and step width) From the small vertical bars at the bottom of the figure, it's clear that the step length also changes in addition to step width.

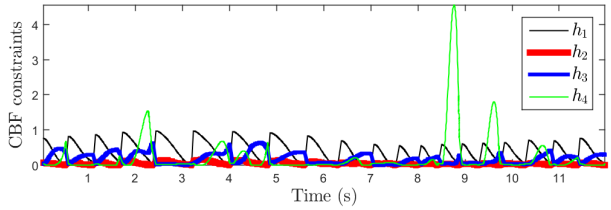

Fig. 9: CBF constraints for Case 3 (changing step length and width)

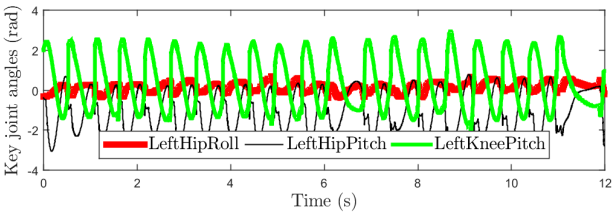

Fig. 11: Key joints angles of left leg for Case 3 (changing step length and width).

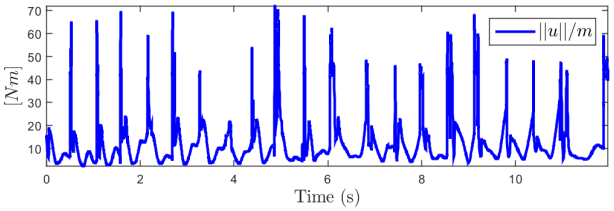

Fig. 10: Norm torque for Case 3 (changing step length and width).

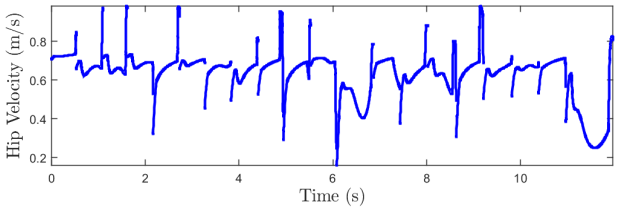

Fig. 12: Hip Velocity for Case 3 (changing step length and width).

VI. Numerical Validation

Having developed the control framework based on control Lyapunov and control Barrier functions, we will now numerically validate the proposed controller on DURUS. We will consider three different simulation cases:

- Case 1: Changing step length only: $l_s \in [l_{min}, l_{max}]$,
- Case 2: Changing step width only: $w_s \in [w_{min}, w_{max}]$,
- Case 3: Changing both step length and width,

where l_{min}, l_{max} and w_{min}, w_{max} were picked with an offset of $\pm 2.5cm$ about the desired footstep location l_d, w_d.

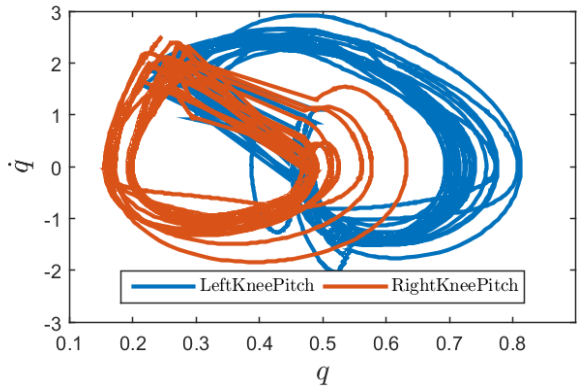

Fig. 13: Phase portrait of left and right knee pitch for Case 3 (changing step length and width).

To be more specific, we have:

$$l_{min} = l_d - 0.025,\ l_{max} = l_d + 0.025,$$
$$w_{min} = w_d - 0.025,\ w_{max} = w_d + 0.025. \quad (37)$$

The controller is simulated with l_d and w_d chosen randomly at each step. While the nominal walking gait has step length $36.6cm$ and step width $23.3cm$, our proposed controller was able to address a large range of desired footholds with the following specific values:

- Case 1: Changing step length only: $l_d \in [22cm, 50cm]$ ($\pm 39\%$ of the nominal step length.);
- Case 2: Changing step width only: $w_d \in [12cm, 33cm]$ (-48% to 43% of the nominal step width);
- Case 3: Changing both step length and width: $l_d \in [24cm, 47cm]$ (-33% to 31% of the nominal step length), $w_d \in [20cm, 33cm]$ (-13% to 43% of the nominal step width).

Beyond these step length and step width values, the initial condition for the subsequent step is too far from the nominal walking gait and the controller is unable to track.

Figures 7,8,9 clearly show that the CBF constraints on footstep placement are strictly satisfied for all three simulation cases. Figures 14,15,16 illustrate snapshots of the simulation with footstep constraints for the three cases respectively. For Case 3 (changing both step length and width), Figures 11, 12, 13, illustrate the key joint angles, hip velocity and phase portrait respectively.

VII. Conclusion

We have presented a control framework that combines control Lyapunov functions and control Barrier functions in a quadratic program which is solved point-wise in time. Control Barrier functions are constructed to enforce step length and step width constraints at impact. The resulting controller thus achieves dynamic 3D walking while enforcing strict constraints on step length and step width at impact, resulting in dynamic walking over stepping stones. Numerical validations of the proposed method on DURUS, a 23 degree-of-freedom humanoid robot, has been carried out. This has resulted in the controller being able to simultaneously handle random step length variations that are between -33% to 31% of the nominal step length and random step width variations that are between -13% to 43% of the nominal step width. Preliminary results on safety constraints with model uncertainty are addressed through the formulation of robust control barrier functions [16].

References

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

[2] A. D. Ames, K. Galloway, and J. W. Grizzle, "Control Lyapunov functions and hybrid zero dynamics," in *IEEE International Conference on Decision and Control*, Dec 2012, pp. 6837–6842.

[3] A. D. Ames, K. Galloway, K. Sreenath, and J. W. Grizzle, "Rapidly exponentially stabilizing control lyapunov functions and hybrid zero dynamics," *IEEE Transactions on Automatic Control*, vol. 59, no. 4, pp. 876–891, April 2014.

[4] A. D. Ames, J. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *IEEE Conference on Decision and Control*, 2014.

[5] A. D. Ames and M. J. Powell, "Towards the unification of locomotion and manipulation through control lyapunov functions and quadratic programs," in *Control of Cyber-Physical Systems*. Springer, 2013, pp. 219–240.

[6] J. Chestnutt, J. Kuffner, K. Nishiwaki, and S. Kagami, "Planning biped navigation strategies in complex environments," in *IEEE International Conference on Humanoid Robotics*, 2003.

[7] J. Chestnutt, M. Lau, G. Cheung, J. Kuffner, J. Hodgins, and T. Kanade, "Footstep planning for the honda asimo humanoid," *Proceedings of the 2005 IEEE International Conference on Robotics and Automation, ICRA* ., pp. 629 – 634, 2005.

[8] R. Desai and H. Geyer, "Robust swing leg placement under large disturbances," *Proceedings of the IEEE International Conference on Robotics and Biomimetics*, pp. 265–270, 2012.

[9] K. Galloway, K. Sreenath, A. D. Ames, and J. W. Grizzle, "Torque saturation in bipedal robotic walking through control lyapunov function based quadratic programs," *IEEE Access*, vol. PP, no. 99, p. 1, April 2015.

[10] 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.

[11] J. Guckenheimer and S. Johnson, "Planar hybrid systems," in *Hybrid Systems II*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 1995, vol. 999, pp. 202–225.

[12] 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 *IEEE International Conference on Robotics and Automation*, 2016.

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

[14] Q. Nguyen and K. Sreenath, "Safety-critical control for dynamical bipedal walking with precise footstep placement," in *The IFAC Conference on Analysis and Design of Hybrid Systems*, 2015.

[15] ——, "Exponential control barrier functions for enforcing high relative-degree safety-critical constraints," in *American Control Conference*, 2016, pp. 322–328.

[16] ——, "Optimal robust control for constrained nonlinear hybrid systems with application to bipedal locomotion," in *American Control Conference*, 2016, pp. 4807–4813.

[17] 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 *IEEE International Conference on Robotics and Automation*, 2016.

[18] M. Rutschmann, B. Satzinger, M. Byl, and K. Byl, "Nonlinear model predictive control for rough-terrain robot hopping," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1859–1864, 2012.

[19] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback control of dynamic bipedal robot locomotion*. CRC press Boca Raton, 2007.