This article has been accepted for publication in IEEE Robotics and Automation Letters. This is the author's version which has not been fully edited and content may change prior to final publication.

Highly Maneuverable Humanoid Running via 3D SLIP+Foot Dynamics

Sait Sovukluk, Johannes Englsberger, and Christian Ott

DOI: 10.1109/LRA.2023.3342668

Copyright Notice

© 2023 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Highly Maneuverable Humanoid Running via 3D SLIP+Foot Dynamics

Sait Sovukluk, Johannes Englsberger, and Christian Ott

***Abstract*—Spring Loaded Inverted Pendulum is a template dynamics used to model the steady-state running of humans and animals ranging from cockroaches to horses. This study extends the conventional 3D SLIP model with a foot and an active controller to also model transitioning from stationary to high-speed running and vice versa. It also compares behavioral differences between the conventional deadbeat-controlled 3D SLIP and actively controlled 3D SLIP with a foot, especially during trajectory transitioning. Focusing on humanoid robots, the objective is to enhance the system's trajectory switching and the disturbance rejection performance while keeping the trajectories implementable and forces feasible for the whole body dynamics. The results are verified on a humanoid robot Kangaroo through simulations in MuJoCo.**

***Index Terms*—Humanoid and Bipedal Locomotion, Whole-Body Motion Planning and Control**

I. INTRODUCTION

ACCUMULATED biomechanical data showed that the steady-state running of humans and animals ranging from cockroaches to horses could be modeled via a template model called spring-loaded inverted pendulum (SLIP) [1], [2]. Seyfarth et al. [3] studied the stability characteristics of planar SLIP dynamics, and they discovered a J-shaped dependency in the adjustment of the angle of attack to leg stiffness. Similar findings triggered the implementation of apex-to-apex deadbeat controllers to maintain stability via policies such as changing touchdown angle and leg stiffness of the nonlinear hybrid SLIP dynamics [4]–[7].

Even though SLIP dynamics seems to be very successful in modeling the steady-state running of humans, very few studies address the implementation of this model on humanoid robots [8]–[10]. Wensing et al. [8] achieved impressive running performance via mapping 3D-SLIP dynamics to a human model through a whole-body controller (WBC). They first search for periodic 3D SLIP trajectories consistent with experimental human data regarding stance and aerial timings. Then they develop a deadbeat controller using first-order Taylor series approximation to maintain stability. Finally, these center of mass (CoM) trajectories are commanded to a robotic human model through a whole-body controller while keeping the torso upright and employing proper leg and arm swinging. Consistent with the nature of the SLIP model, they examine only the steady-state behavior and change the desired running velocity very slowly.

Another study related to this paper is impulse control of a planar point foot SLIP dynamics by Koepl et al. [11] to reject disturbances caused by changing surface conditions. They discuss preserving the stability around the desired periodic running trajectory and active adaptation to changing surface conditions. Inspired by impulse control, Dadashzadeh et al. [12] proposed an active feedback linearization control for stance dynamics of the planar point foot SLIP model. They heuristically select touchdown angle and, assuming pinned ground contact, do not discuss the feasibility of the additional control forces. Englsberger et al. [13] developed a SLIP-like, analytical running controller that can break periodicity and run over stepping stones. Their method suffers from parameter selection requirements since it requires manual selections of critical parameters such as touchdown height, stance time, and aerial time. Additionally, depending on selected parameters, kinematic limits such as leg length can be exceeded in the template model, so mapping this trajectory to a whole-body model may become impossible. Similarly, some recent studies on 3D-ZMP-based running [14], [15] again rely on the user's stance and flight time selections. On the other hand, the SLIP optimization provides these parameters implicitly, and its stability does not rely on the parameter selection. This is most crucial when the robot switches between different types of running frequently, for instance, while running over unknown terrain or stepping stones. In this case, the user has to select parameters for each different type of running.

Besides 2D SLIP, very few studies address the control of 3D SLIP models for frequent trajectory changes. Without focusing on any robotic implementation, [16] presents a deadbeat controller for the 3-D SLIP model that can cope with unknown ground height variations of up to 30% of the leg length. Yet, they assume energy conservation and operate on constant energy levels.

Even though the SLIP dynamics models steady-state running well, it performs poorly when transitioning between different running velocities and initiating running from a stationary position mainly for three reasons. First, the deadbeat controllers are local linearizations of periodic trajectories, and

Manuscript received: July, 26, 2023; Revised October, 24, 2023; Accepted November, 29, 2023.

This paper was recommended for publication by Editor Abderrahmane Kheddar upon evaluation of the Associate Editor and Reviewers' comments. This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. 819358).

Sait Sovukluk is with the Automation and Control Institute (ACIN), TU Wien, 1040 Vienna, Austria (e-mail: sovukluk@acin.tuwien.ac.at)

Johannes Englsberger is with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Weßling, Germany (e-mail: johannes.englsberger@dlr.de).

Christian Ott is with the Automation and Control Institute (ACIN), TU Wien, 1040 Vienna, Austria, and also with the Institute of Robotics and Mechatronics, German Aerospace Center (DLR), 82234 Weßling, Germany (e-mail: christian.ott@tuwien.ac.at).

Digital Object Identifier (DOI): see top of this page.

they are only valid when the apex states are close enough to their periodic values. Second, since SLIP is a point-foot model adapted to model steady-state running, it does not include sufficient capabilities to initiate movements when the robot is stationary. On the contrary to humans, to initiate a forward velocity, it has to jump, place the leg behind and bounce forward. Third, due to the nature of deadbeat controllers, the system reacts to disturbances and transitioning commands only when it reaches the apex state.

Focusing on humanoid robot running, this study addresses the aforementioned issues by introducing a foot and an active control policy to the conventional deadbeat-controlled 3D-SLIP model. Preserving the core ideas of periodic trajectory generation and the deadbeat controller, the introduced foot structure enables the robot to actively change the direction and magnitude of ground reaction force, enhancing SLIP dynamics' convergence and transitioning characteristics. Thanks to the active control, rejecting the disturbances and transitioning to other trajectories can be initiated in the stance phase. As a result, the apex error is reduced, and the deadbeat controller works more efficiently. Additionally, since this approach does not require a constant energy level assumption for its control, the robot can accelerate or decelerate while keeping the apex height and all other parameters constant. When initiated at the beginning of a stance phase, the proposed control structure allows convergence from $0m/s$ to $2m/s$ in the next stance. On the other hand, the conventional apex-to-apex deadbeat controlled system fails to maintain stability since the states are too far away from the desired trajectory.

II. Background

This section collects SLIP preliminaries from the existing literature.

A. SLIP Dynamics

Defining $\boldsymbol{p}_c \in \mathbb{R}^3$ to be the position of CoM, the stance dynamics of the point foot 3D SLIP (see Fig. 1) is:

$$m\ddot{\boldsymbol{p}}_c = k(r_0 - ||\boldsymbol{r}||)\hat{\boldsymbol{r}} + m\boldsymbol{g}, \tag{1}$$

where m, k, r_0, and $\boldsymbol{g}$ are the total mass, spring constant, leg rest length, and gravitational vector. Additionally, $\boldsymbol{r} = \boldsymbol{p}_c - \boldsymbol{p}_f$ represents the leg vector, i.e., a vector from the foot point to the point mass (see Fig. 1) and $\hat{\boldsymbol{r}}$ is its unit vector. Through the use of virtual hip, the relation between the point mass and the foot locations is a function of θ_1 and θ_2, pitch and yaw angles of the virtual leg, respectively. The foot point $\boldsymbol{p}_f$ is a function of virtual leg angles, hip distance, and virtual leg length:

$$\boldsymbol{p}_f = \boldsymbol{p}_c + \begin{bmatrix} 0 \\ \sigma y_h \\ 0 \end{bmatrix} + l_h \begin{bmatrix} \sin\theta_1 \cos\theta_2 \\ \sigma \sin\theta_2 \\ -\cos\theta_1 \cos\theta_2 \end{bmatrix}, \tag{2}$$

where σ is a sign multiplier and changes between $+1$ and -1 depending on which leg is the stance leg. On the other hand, during the stance phase, $\boldsymbol{p}_f$ is fixed, and the virtual leg angles θ_1 and θ_2 are driven implicitly by the point mass dynamics. A touchdown and liftoff conditions separate the stance and flight dynamics such that the overall dynamics Σ is given by:

$$\Sigma : \begin{cases} \ddot{\boldsymbol{p}}_c = \dfrac{k}{m}(r_0 - ||\boldsymbol{r}||)\hat{\boldsymbol{r}} + \boldsymbol{g} & ||\boldsymbol{r}|| < r_0 \\ \ddot{\boldsymbol{p}}_c = \boldsymbol{g} & ||\boldsymbol{r}|| \geq r_0 \end{cases}. \tag{3}$$

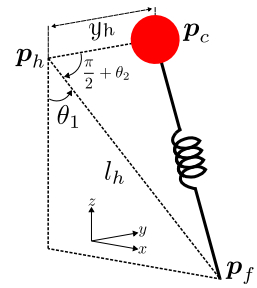

Fig. 1. 3D SLIP model representation. In the sketch, y_h separates the robot's center of mass and hip positions. The virtual leg has a length of l_h and connects the hip with the foot. The touchdown angle is determined by θ_1 and θ_2. Since the virtual leg is offset by y_h, if θ_2 is zero, the foot is in the hip sagittal (parasagittal) plane. Otherwise, e.g., $\theta_2 \neq 0$, the lateral foot placement point is wider or narrower than the virtual hip point y_h.

B. SLIP Periodic Trajectory Search and Deadbeat Control

Studies in [7] and [17] showed that for a constant energy level of a planar SLIP model, there exists a unique relationship between some parameters. They discovered that, for example, for a specific spring constant, running height, and running velocity, there exists a unique touchdown angle that results in a periodic limit cycle. The remaining one is always unique among these four parameters. Consequently, if θ is the touchdown angle of a planar SLIP model, this parameter is a function of running height h, spring constant k, and running velocity v, i.e., $\theta(h, k, v)$. Selecting $\boldsymbol{\theta} = [\theta_1, \theta_2]^\top$ and $\boldsymbol{v} = [v_x, v_y]^\top$ for the 3D SLIP, one can rewrite the touchdown angle function as $\boldsymbol{\theta}(h, k, \boldsymbol{v})$.

For an apex state definition of $\boldsymbol{x} = [v_x, v_y, h]^\top$ and input definition of $\boldsymbol{u} = [\theta_1, \theta_2, k_1, k_2]^\top$, a mapping function from one apex state to the next one can be defined as $\boldsymbol{f} : (\boldsymbol{x}_k, \boldsymbol{u}) \rightarrow \boldsymbol{x}_{k+1}$. Here the spring definition in input vector $\boldsymbol{u}$ is divided into k_1 and k_2, compression and decompression spring constants, to enable the controller to increase or decrease the total mechanical energy of the system for control purposes [8]. If the motion is periodic, these two spring constants are equal.

Accordingly, for a selection of h, k, and $\boldsymbol{v}$, the periodic trajectory search problem for 3D SLIP can be formulated as a nonlinear least-squares problem:

$$\min_{\boldsymbol{\theta}} ||\boldsymbol{x}_k - \boldsymbol{E}\boldsymbol{x}_{k+1}||, \tag{4}$$

where $E = \mathrm{diag}(1, -1, 1)$ is placed for sign changes due to the leg switching between two subsequent apex states. Similarly, since it is not straightforward to pick a proper value for v_y, for a selection of h, k, v_x, and θ_2[1], the nonlinear least-squares problem can be reformulated as:

$$\min_{\theta_1, v_y} ||\boldsymbol{x}_k - \boldsymbol{E}\boldsymbol{x}_{k+1}||. \tag{5}$$

Following the control system idea introduced in [8], to reject the disturbances, a first-order Taylor Series approximation can be used to implement apex-to-apex deadbeat controller.

[1]During periodic trajectory optimization, θ_2 is usually selected to be zero, i.e., the foot is in the hip sagittal (parasagittal) plane. In case of a disturbance, this value is modified by the Deadbeat controller to maintain stability.

Assuming $(\boldsymbol{x}^*, \boldsymbol{u}^*)$ results in a periodic trajectory, the first order Taylor Series approximation leads to:

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{f}(\boldsymbol{x}_k^* + \boldsymbol{\Delta x}, \boldsymbol{u}^* + \boldsymbol{\Delta u}) \\ &\approx \boldsymbol{E}\boldsymbol{x}_k^* + \boldsymbol{J}_x \boldsymbol{\Delta x} + \boldsymbol{J}_u \boldsymbol{\Delta u}, \end{aligned} \tag{6}$$

where $\boldsymbol{J}_x = \partial \boldsymbol{f} / \partial \boldsymbol{x} \in \mathbb{R}^{3\times 3}$ and $\boldsymbol{J}_u = \partial \boldsymbol{f} / \partial \boldsymbol{u} \in \mathbb{R}^{3\times 4}$ are Jacobians of the return map evaluated at $(\boldsymbol{x}^*, \boldsymbol{u}^*)$. Then, deviations from periodic apex states can be rejected by changing the input of the system:

$$\boldsymbol{J}_u \boldsymbol{\Delta u} = -\boldsymbol{J}_x \boldsymbol{\Delta x}. \tag{7}$$

Selecting the compression and decompression spring constant deltas to be equal in amount but with a different sign, i.e., $\Delta k_1 = -\Delta k_2$, one can solve (7) and obtain a gain matrix $\boldsymbol{K} \in \mathbb{R}^{4\times 3}$ that offsets the periodic input parameters of the system to reject the state disturbances:

$$\boldsymbol{u} = \boldsymbol{u}^* + \boldsymbol{K} \underbrace{(\boldsymbol{x} - \boldsymbol{x}^*)}_{\boldsymbol{\Delta x}}, \tag{8}$$

where

$$\boldsymbol{K} = -\boldsymbol{A}(\boldsymbol{J}_u \boldsymbol{A})^{-1} \boldsymbol{J}_x \text{ and } \boldsymbol{A} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1 \end{bmatrix}.$$

III. Active SLIP Control

This section points out some drawbacks of the conventional apex-to-apex deadbeat controlled 3D SLIP model and addresses these drawbacks with an extended control policy.

- Even though the SLIP model is very successful in modeling the running of animals ranging from cockroaches to horses [1], [2], the discussions are usually limited to steady-state behaviors and constant energy levels. Consequently, the relevant control designs focus on stabilizing these steady-state behaviors.
- Since the deadbeat control is a local linearization of periodic trajectories, it produces unknown behaviors when apex state errors are not small enough. These errors might result from disturbances or changes in velocity commands.
- The conventional apex-to-apex deadbeat control policy is a discrete-time controller and manipulates the system parameters only once in every step. Hence, if there is a disturbance, the system waits until the next apex state to take action.
- Since the 3D SLIP is a point foot model, it cannot take advantage of changing the center of pressure location. This drawback prevents the system from initiating a velocity during the stance phase. A SLIP model that bounces stationarily cannot initiate a forward velocity during this phase. Instead, the system waits to jump, places the foot backward in the air, and bounces forward.

A. Control Force

To address these drawbacks, we introduce a foot structure and a related active control system to the conventional 3D SLIP model (see Fig. 3). Instead of limiting the CoP (center of pressure) to be at the same point all the time, the proposed control method allows it to change within the foot to enhance the system's control, convergence, and transitioning performance. In this regard, an additional control force is added to the system dynamics. The following subsections will introduce some rules by which we will ensure feasibility of the resulting control action.

The new stance dynamics with the additional control force $\boldsymbol{f} \in \mathbb{R}^3$ is given by:

$$m\ddot{\boldsymbol{p}}_c = \underbrace{k(r_0 - ||\boldsymbol{r}||)\hat{\boldsymbol{r}}}_{\boldsymbol{f}_s} + m\boldsymbol{g} + \boldsymbol{f}. \tag{9}$$

The active control action aims to achieve convergence to the unperturbed SLIP trajectory during the stance phase. As the touchdown happens, the foot $\boldsymbol{p}_f = \boldsymbol{p}_{td}$ becomes a fixed touchdown point on the ground. In the moment of touchdown, substituting $\boldsymbol{p}_f$ into (2) and solving for $\boldsymbol{p}_{c,d}$ with the periodic values of virtual leg angles θ_1 and θ_2 results in an initial condition $\boldsymbol{p}_{c,d}$ for the related periodic trajectory (see Fig. 2). Starting from the initial condition where both systems have the same touchdown point, integrating the undisturbed periodic SLIP dynamics, one can get desired position, velocity, and acceleration evolution of the CoM dynamics. The difference between the actual and desired evolution forms the error definition $\boldsymbol{e} = \boldsymbol{p}_{c,d} - \boldsymbol{p}_c$.

Ideally, when there is no constraint, selecting $\boldsymbol{f} = \boldsymbol{f}_d$, where,

$$\boldsymbol{f}_d = m\ddot{\boldsymbol{p}}_{c,d} - k(r_0 - ||\boldsymbol{r}||)\hat{\boldsymbol{r}} - m\boldsymbol{g} + \boldsymbol{K}_D\dot{\boldsymbol{e}} + \boldsymbol{K}_P\boldsymbol{e}, \tag{10}$$

results in the following exponentially stable closed-loop dynamics:

$$m\ddot{\boldsymbol{e}} + \boldsymbol{K}_D\dot{\boldsymbol{e}} + \boldsymbol{K}_P\boldsymbol{e} = 0. \tag{11}$$

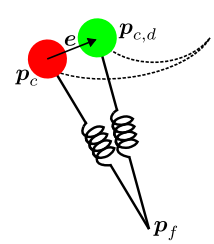

Fig. 2. The error definition of the control system where the green SLIP represents the desired undisturbed periodic state evolution and the red one is the actual evolution. Sharing the same foot point $\boldsymbol{p}_f$, the red system converges to the green system during the stance phase.

B. Feasibility Constraints

The combination of floating base dynamics with limited foot region implies some constraints on the applicable control force. These constraints are related to positive ground reaction force, foot region, and friction cone.

1) Positive ground-reaction force constraint: During the stance phase, the foot must remain in contact with the ground. Suppose the total ground reaction force, i.e., $\boldsymbol{f}_t = \boldsymbol{f}_d + \boldsymbol{f}_s$, happens to be less than zero in the z direction. In that case, the control force $\boldsymbol{f}_d$ has to be rescaled to ensure contact with the ground. Consequently, the condition becomes $f_{t,z} > 0$.

2) Foot boundary constraint: The total ground reaction force must stay within the foot boundaries to prevent foot tilting. In case of violation, the total force should be projected to the feasible area.

A combination of four triangular surfaces constitutes a feasible region for $\boldsymbol{f}_t$ (See Fig. 3). Assume $\boldsymbol{V} = \begin{bmatrix}\hat{\boldsymbol{v}}^{\boldsymbol{p}_c}_{\boldsymbol{p}_{f1}} & \hat{\boldsymbol{v}}^{\boldsymbol{p}_c}_{\boldsymbol{p}_{f2}} & \hat{\boldsymbol{v}}^{\boldsymbol{p}_c}_{\boldsymbol{p}_{f3}} & \hat{\boldsymbol{v}}^{\boldsymbol{p}_c}_{\boldsymbol{p}_{f4}}\end{bmatrix} \in \mathbb{R}^{3\times 4}$ represents the combination of all corner unit vectors where $\boldsymbol{v}^{(*)}_{(\cdot)}$ represents vector from point $(\cdot)$ to point $(*)$. A linear combination of the corner vectors would represent all feasible forces within the foot region. For a set of scalar multipliers $\boldsymbol{\alpha} \in \mathbb{R}^4$, if the cost function of the minimization problem

$$\min_{\boldsymbol{\alpha}} \ ||\boldsymbol{f}_t - \boldsymbol{V}\boldsymbol{\alpha}|| \tag{12}$$

converges to zero, then the total force is within the polyhedral limits. Otherwise, the minimization results in the closest projection of f_t onto the polyhedral limits.

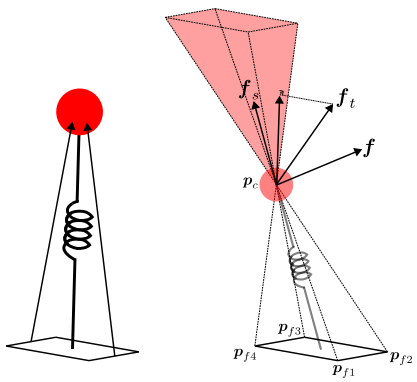

Fig. 3. A sketch of 3D SLIP model with foot. The left sketch shows possible new CoP locations to enhance the system's performance. The right sketch represents force projections in case the total ground reaction force $\boldsymbol{f}_t = \boldsymbol{f}_s + \boldsymbol{f}$ is outside of the feasible polyhedral force cone.

3) Friction constraint: Imposing $|f_{t,x}|/f_{t,z} \leq \mu$ and $|f_{t,y}|/f_{t,z} \leq \mu$ on f_t, where $\mu \in \mathbb{R}$ is the friction constant, results in the total force that is within the friction cone.

C. Overall Control Force Formulation

Overall SLIP control force formulation (13) combines all the feasibility constraints we define in the previous subsection. The cost function reconstructs the desired total force and looks for the closest solution within the polyhedral region. The positive ground reaction force and friction cone constraints are handled by (13a) and (13b).

$$\min_{\boldsymbol{\alpha}} \ ||\boldsymbol{f}_t - \boldsymbol{V}\boldsymbol{\alpha}|| \tag{13}$$

Such that:

$$\boldsymbol{\alpha} > 0 \tag{13a}$$

$$\frac{|(\boldsymbol{V}\boldsymbol{\alpha})_x|}{(\boldsymbol{V}\boldsymbol{\alpha})_z} \leq \mu, \quad \frac{|(\boldsymbol{V}\boldsymbol{\alpha})_y|}{(\boldsymbol{V}\boldsymbol{\alpha})_z} \leq \mu \tag{13b}$$

The allowable control force in (9) that would result in the allowable total force $\boldsymbol{f}_{t,\text{new}} = \boldsymbol{V}\boldsymbol{\alpha}$ becomes:

$$\boldsymbol{f} = \boldsymbol{f}_{t,\text{new}} - \boldsymbol{f}_s. \tag{14}$$

Note that since the total and control forces always target the CoM, it is assumed that the ground reaction force can be achieved without changing centroidal angular momentum.

IV. Discussion of Stabilization Strategy

Since the search for a periodic trajectory and design of the deadbeat control are performed using the conventional point foot 3D SLIP model, the plane foot has no additional effect on the system once it is on a periodic trajectory. In case of a disturbance or trajectory switching, if $\boldsymbol{f}_d$ is always feasible, then (11) holds. Moreover, if the stance time is long enough, the system converges to a new trajectory or rejects disturbances in a single stance phase. On the other hand, assume $\boldsymbol{f}_d$ is not feasible and the stance time is not long enough to converge, only a portion of the error gets canceled, and the rest of the error is carried into the next step. Reducing a portion of the error before the next apex increases the validity/viability of the deadbeat controller.

Since the ground reaction forces originate around the touchdown point, the system's stability is closely related to the foot placement point. A proper foot placement point for the next step would also increase the convergence rate. Assume the system is faster than the desired running velocity. Placing the foot ahead of its periodic position enables the controller to apply more force to decelerate than it would from the periodic foot placement point due to feasibility constraints and leg vector direction — the same hold for acceleration. This heuristic, manually-tuned foot placement control is well known since [18] and is widely employed in the literature.

On the other hand, the deadbeat control by Wensing et al. [8] in (8) provides the same behavior without any heuristic parameter selection. The control method indicates that if the apex error of the system is small enough, a change in the angle of attack and spring stiffness exists such that the error is rejected in the next apex. As a result, combining the deadbeat control with the additional control force introduced in (9) indicates that if the system states are not far away from the linearization point, the disturbances will be rejected in a single step. Since the additional control force actively rejects disturbances during the stance time, it reduces, if not completely rejects, the total error until the next apex such that the linearized deadbeat controller provides more accurate results. Consequently, we combine the deadbeat control with our active force control in this study. This combination spends stance time more efficiently, and changes foot placement points to increase the convergence rate.

V. Implementation on Humanoid Robots

A. Floating Base Model

The general tree structured floating base robot dynamics for n configuration variables $\boldsymbol{q} \in \mathcal{Q}$ is expressed as

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) = \boldsymbol{B}\boldsymbol{\tau} + \boldsymbol{J}_c^\top(\boldsymbol{q})\boldsymbol{f}_c + \boldsymbol{J}_p^\top \boldsymbol{f}_p, \tag{15}$$

where $\boldsymbol{M}$, $\boldsymbol{C}$, $\boldsymbol{g}$, $\boldsymbol{B}$, $\boldsymbol{\tau}$, $\boldsymbol{J}_c$, $\boldsymbol{f}_c$, $\boldsymbol{J}_p$, and $\boldsymbol{f}_p$ represent inertia matrix, Coriolis matrix, gravity vector, input mapping matrix, actuation efforts, contact Jacobians, contact forces, closed chain constraint Jacobians, closed chain constraint forces respectively. In case of closed linkages, the configuration variables are constituted by active and passive joints $\boldsymbol{q} = [\boldsymbol{q}^a \ \boldsymbol{q}^p]^\top$, and the passive joints of each closed-chain are constrained to move together with their corresponding

actively driven counterparts. In this case, the system dynamics is combined with holonomic constraints. We briefly discuss the holonomic constraints in control formulation in the following subsections. The study [19] discusses it in detail.

B. Task Formulation via Inverse Dynamics Control (IDC)

For each task space Jacobian $\boldsymbol{J}_i$, the task space velocity can be represented as:

$$\dot{\boldsymbol{x}}_i = \boldsymbol{J}_i \dot{\boldsymbol{q}}. \tag{16}$$

Similarly, the time derivative of (16) yields the task acceleration:

$$\ddot{\boldsymbol{x}}_i = \boldsymbol{J}_i \ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}_i \dot{\boldsymbol{q}}. \tag{17}$$

If a controller manages the task acceleration $\ddot{\boldsymbol{x}}_i$ to be equal to:

$$\ddot{\boldsymbol{x}}_{i,d} = \ddot{\boldsymbol{x}}_{i,ref} + \boldsymbol{K}_{D,i}(\dot{\boldsymbol{x}}_{i,ref} - \dot{\boldsymbol{x}}_i) + \boldsymbol{K}_{P,i}(\boldsymbol{x}_{i,ref} - \boldsymbol{x}_i), \tag{18}$$

where $\boldsymbol{K}_{D,i}$ and $\boldsymbol{K}_{P,i}$ are positive definite gain matrices, for an error definition of $\boldsymbol{e}_i = \boldsymbol{x}_{i,d} - \boldsymbol{x}_i$, the closed-loop error dynamics for the particular task turns out to be exponentially stable:

$$\ddot{\boldsymbol{e}}_i + \boldsymbol{K}_{D,i}\dot{\boldsymbol{e}}_i + \boldsymbol{K}_P \boldsymbol{e}_i = 0. \tag{19}$$

C. Closed Kinematics Constraint Formulation

In case the robot contains closed kinematic linkages, as in case of Kangaroo, since most dynamics libraries consider the system an open kinematic chain, one should solve for the constraint forces that enforce corresponding endpoints to move together. These constraint forces will drive the open chain ends such that they follow their counterpart link, and the chain is closed. The task acceleration (17) of both ends should be equal. Assuming $\boldsymbol{J}_{k,1}$ and $\boldsymbol{J}_{k,2}$ to be Jacobian mappings of both endpoints of the kth closed chain, the constraint equation becomes

$$\underbrace{(\boldsymbol{J}_{k,1} - \boldsymbol{J}_{k,2})}_{\boldsymbol{J}_{p,k}} \ddot{\boldsymbol{q}} + \underbrace{(\dot{\boldsymbol{J}}_{k,1} - \dot{\boldsymbol{J}}_{k,2})}_{\dot{\boldsymbol{J}}_{p,k}} \dot{\boldsymbol{q}} = 0. \tag{20}$$

A more detailed discussion, including state estimation for the passive joints, is covered in [19].

D. Whole Body Control Formulation

The desired SLIP trajectories can be mapped to a humanoid robot via an optimization-based whole-body controller. The study in [19] discusses task, closed kinematic chain constraint, and other constraint formulations in detail for Kangaroo. Selecting the parameters to optimize for as $[\ddot{\boldsymbol{q}}; \boldsymbol{\tau}; \boldsymbol{f}_c; \boldsymbol{f}_p]$, the combination of all task controllers (18) into a quadratic problem constitutes the cost function of the optimization problem:

$$\min_{\ddot{\boldsymbol{q}}, \boldsymbol{\tau}, \boldsymbol{f}_c, \boldsymbol{f}_p} \sum_i (\ddot{\boldsymbol{x}}_{i,d} - \ddot{\boldsymbol{x}}_i)^\top \boldsymbol{W}_i (\ddot{\boldsymbol{x}}_{i,d} - \ddot{\boldsymbol{x}}_i) \tag{21}$$

Such that:

$$\boldsymbol{M}\ddot{\boldsymbol{q}} + \boldsymbol{C}\dot{\boldsymbol{q}} + \boldsymbol{g} = \boldsymbol{B}\boldsymbol{\tau} + \boldsymbol{J}_c^\top \boldsymbol{f}_c + \boldsymbol{J}_p^\top \boldsymbol{f}_p \tag{21a}$$

$$\boldsymbol{J}_c \ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}_c \dot{\boldsymbol{q}} = 0 \tag{21b}$$

$$|f_{x,l}| \leq \frac{\mu f_{z,l}}{\sqrt{2}}, \ |f_{y,l}| \leq \frac{\mu f_{z,l}}{\sqrt{2}}, \text{ and } f_z \geq 0 \ \forall l \tag{21c}$$

$$\boldsymbol{J}_p \ddot{\boldsymbol{q}} + \dot{\boldsymbol{J}}_p \dot{\boldsymbol{q}} = 0 \tag{21d}$$

$$\boldsymbol{\tau}_{min} \leq \boldsymbol{\tau} \leq \boldsymbol{\tau}_{max} \tag{21e}$$

where $\boldsymbol{\tau}_{min}$ and $\boldsymbol{\tau}_{max}$ represent minimum and maximum limits of input torques and forces. Additionally, (21c) represents the friction pyramid constraint for each foot corner. Since the cost function (21) is constituted by a combination of different tasks and their corresponding weight matrices $\boldsymbol{W}_i$, the task errors with higher weights are prioritized. The interaction between the whole body controller and SLIP dynamics is represented in Fig. 4. Note that there is no fixed timing between the stance and flight phases. The transition from the stance to flight (or vice versa) is entirely state-based, and even though the ideal SLIP model switches to the aerial phase, the actuated SLIP model continues to converge to it as long as there is contact with the ground, i.e., $||\boldsymbol{r}|| \leq r_0$.

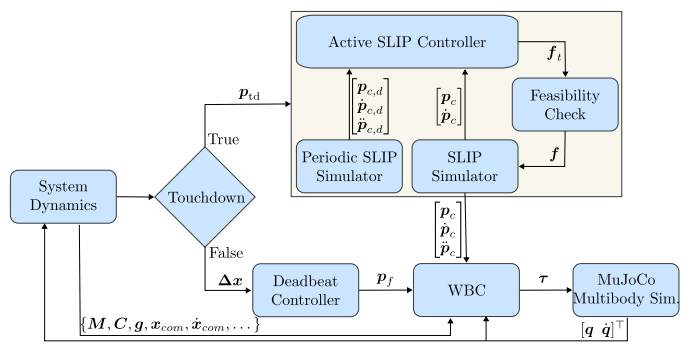

Fig. 4. The proposed control system diagram. Based on the foot touchdown position of the robot, the active SLIP control generates the desired CoM trajectory during the stance phase. On the other hand, based on the deviations in the apex states, the deadbeat controller supplies the new angle of attack target to the WBC throughout the aerial phase.

E. Parameter Selection and List of Tasks

Weight, stiffness, and damping parameters are tuned based on observations. List of tasks, constraints, weights, and controller parameter selections for the stance and flight phases can be found in Table I. During the flight phase, the contact constraints are deactivated, and the contact forces are constrained to zero. The values represent repeating elements of diagonal matrices, and ξ represents the damping ratio. This study employs OSQP [20] to solve the optimization problem in (21).

VI. Swing Leg Trajectory and Torso Orientation

One of the important aspects of humanoid robot locomotion is the torso and swing leg evolution during running. Since the 3D SLIP dynamics does not fully cover the robot dynamics, the movement of any limb may disturb the desired CoM evolution. Pontzer et al. in [21] suggest that the trunk and shoulders act primarily as elastic linkages between the pelvis, shoulder girdle, and arms and the upper body movement is primarily powered by lower body movement. Consequently, the upper body movement is inherently self-tuned.

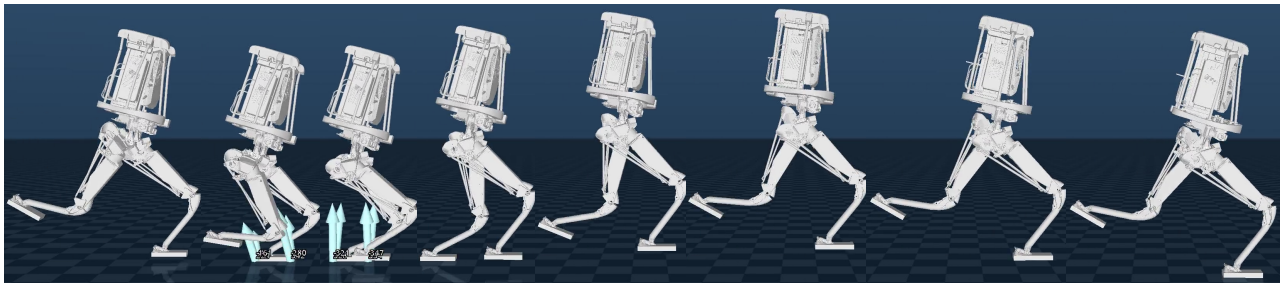

Fig. 5. Kangaroo robot running with $2m/s$ forward velocity. For more details, see the supplemental video.

TABLE I
LIST OF TASKS, CONSTRAINTS, WEIGHTS, AND CONTROLLER PARAMETER SELECTIONS FOR THE STANCE AND FLIGHT PHASES.

Stance		$\boldsymbol{W}$	$\boldsymbol{K}_P$	ξ
	CoM trajectory	10	100	1.0
	Torso orientation	2	25	1.0
	Swing foot trajectory	5	500	1.0
	Swing foot orientation	5	1000	1.0
	Constraints	(21a), (21b), (21c) (21d), (21e)		
Flight		$\boldsymbol{W}$	$\boldsymbol{K}_P$	ξ
	Upcoming stance foot position	10	1000	1.0
	Upcoming stance foot orientation	10	1000	1.0
	Swing foot position	0.05	500	1.0
	Swing foot orientation	10	1000	1.0
	Torso orientation	2	50	1.0
	Constraints	(21a), (21d), (21e) $\boldsymbol{f}_c = 0$		

The swing leg can compensate for the effect of the upcoming stance leg such that the torso stays upright during the aerial phase. As the arms act as passive mass dampers, which reduce torso and head rotation [21], this is also relevant and very important for robots with light or missing arms, such as Kangaroo. Consequently, a torso orientation task for the aerial phase can be seen in Table I along with a low-gain and low-priority fixed position and zero velocity task w.r.t. CoM for the trailing swing leg. The torso orientation task during the flight phase utilizes the swing foot as a balancing limb and moves it so that the torso stays vertical. On the other hand, the swing foot has a certain reachable region in space. We command a certain foot position task against the torso orientation task. Since it is a low-gain task, it would allow the torso orientation task to drive the leg. As the foot translates far away, the pullback force of the foot position task increases to prevent the system from reaching joint limits. During stance, since there is contact with the ground, the robot then swings the leg forward.

VII. SIMULATION RESULTS

The simulation consists of two parts: comparison of SLIP models and implementation on Kangaroo. Aligned with the structure of Kangaroo robot, parameters are selected as $m \approx 41kg$ and $y_h = 15cm$. By selection, the virtual leg length $l_h = 80cm$, spring constant $k = 10kN/m$, jumping height $10cm$, and lateral leg opening $\theta_2 = 0rad$. Aligned with the foot size of the robot, the SLIP foot is selected as $20cm$ in length and $8cm$ in width. Using the periodic trajectory search problem (5), three periodic trajectories are found for $0m/s$, $1m/s$, and $2m/s$ running speeds and along with their corresponding deadbeat control gains $\boldsymbol{K}_{0m/s}$, $\boldsymbol{K}_{1m/s}$, and $\boldsymbol{K}_{2m/s}$ by (7). Finally, MuJoCo simulator [22] is used for multibody simulations, accounting for impact forces and closed-kinematic chains.

A. Point Mass Simulations

Since the trajectories are generated using the point foot SLIP model, the steady-state response of each system is identical. As Fig. 6 shows, the differences appear during transitioning. When a velocity change is commanded from $0m/s$ to $2m/s$, the model with active control immediately recognizes the change in trajectory and applies additional control force to approach the desired trajectory. Once it enters the aerial phase, it already carries around $1m/s$ velocity. Hence less error is fed into the deadbeat controller. During the next stance, it fully converges into the desired trajectory. Note that since the system model is in the frontal plane during stationary jumping, without a leg and changed CoP location, initiating a force in the forward direction would be impossible during the stance phase.

On the other hand, the deadbeat controlled point foot model does not contain any policy to take advantage of the stance phase. It waits to jump and feeds a $2m/s$ velocity error into the deadbeat controller. Since the states are far from the periodic trajectory, the output of the deadbeat controller does not produce stable behavior. Instead, with special care, we don't use $\boldsymbol{K}_{2m/s}$, but $\boldsymbol{K}_{1m/s}$ for the first apex as it would be the average of current and expected velocities. In this case, after showing overshoots and drastic changes in the CoM height h and lateral velocity v_y, the conventional model seems to be converged to the desired velocity in the third stance. The drastic changes are caused by aggressive stiffness changes and excessive touchdown angles (θ_1 and θ_2) produced by the deadbeat controller. These drastic changes make the trajectory hard to follow due to multi-body effects and require high power inputs. Additionally, touchdown angles that are too horizontal cause friction cone-related issues in whole-body dynamics and reduce the trackability of the trajectory since friction is constrained in WBC formulation. The figure depicts a similar pattern for transitioning from $2m/s$ running to stationary jogging.

The ability to actively change the direction and magnitude of ground reaction forces is shown in Fig. 7. Even if the desired

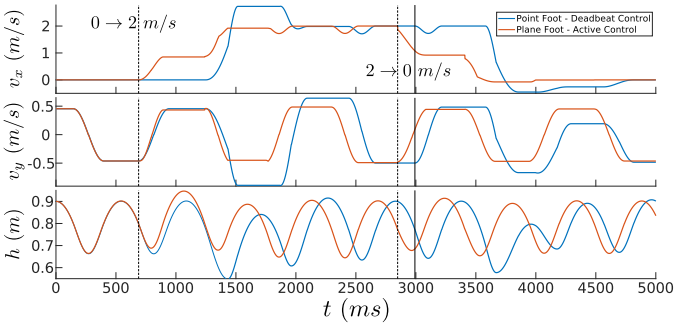

Fig. 6. State evolution comparison of actively controlled plane foot SLIP and deadbeat controlled point foot SLIP. Vertical dashed lines represent the touchdown moments at which an aggressive velocity change is commanded. Due to the phase difference after the first transition, the next transition is commanded at different times to coincide with respective touchdown moments.

trajectory changes during the stance phase, the active force controller initiates a force in the respective direction while keeping the CoP inside the foot boundaries. After convergence, since the control force is zero, the system dynamics are identical to the conventional 3D SLIP model. As a result, the ground reaction forces follow the exact pattern that a conventional 3D SLIP model would. Since the forces are constrained by foot, and the additional control force is constructed around the spring force, the ground reaction forces preserve their smoothness and start from zero. Sign changes in the y axis represent leg changes.

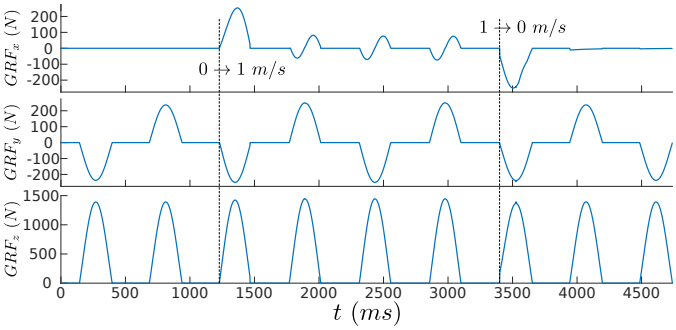

Fig. 7. Ground reaction forces of actively controlled SLIP model. As the desired running velocity changes, the active controller breaks the periodicity to converge into the desired trajectory. New trajectories are commanded right after the touchdowns.

B. Whole Body Simulations

The whole body control formulation shows robust tracking for the desired CoM trajectories in Fig. 8. SLIP is a point mass model; hence it does not cover the multibody effects, such as swing leg and torso effects, and the impact dynamics. Consequently, momentarily velocity jumps are observed in the figure at the beginning of every step. The figure also depicts that, thanks to the extended SLIP model, the humanoid robot can initiate running from the stance phase. The ground reaction force directions right after the velocity change command during the stance phase are shown in Fig. 9.

Observing both systems' ground reaction force trajectories is another way to examine the match between the whole

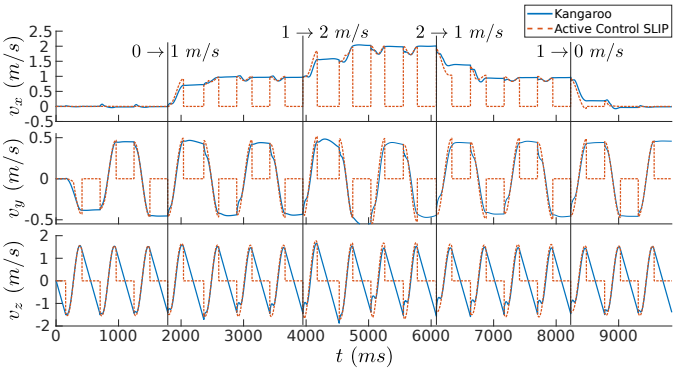

Fig. 8. CoM evolution of Kangaroo robot following SLIP trajectories with aggressive velocity changes. The dashed lines represent SLIP trajectories, which are only activated during the stance phase. Vertical dashed lines represent SLIP activation and deactivation moments. When SLIP is not activated, zero values are plotted.

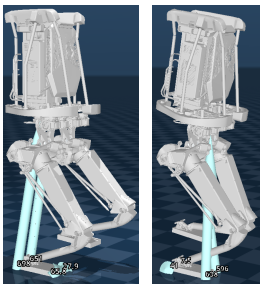

Fig. 9. Ground reaction force directions at the beginning of transitioning from stationary to running (left) and running to stationary (right). Trajectory switching commands are sent right after the touchdown moment.

body and SLIP dynamics. Again, as shown in Fig. 10, the touchdown impacts appear as impulsive force jumps in the system dynamics. Right after the impact, the forces converge back into the SLIP forces. The multibody effects on the CoM dynamics are again observable, and torso orientation corrections appear as force jumps during the stance phase in the x direction even when the running velocity is zero.

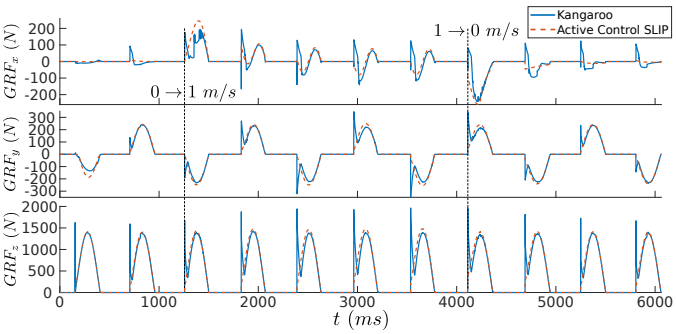

Fig. 10. Resultant ground reaction forces of the whole body model and SLIP dynamics. Initially, the robot is released from the apex height with zero velocity.

One of the important challenges of running is estimating the states accurately during the aerial phase. We randomly change the ground truth information at each step to show our model is robust against poor state estimations. As a result, the transition from flight to stance, along with consecutive contact constraints (see Table I), is applied earlier or later than the actual landing. The simulation in Fig. 11 shows the system's response when the contact is randomly activated either $5cm$ above the ground or $50ms$ later than the actual moment while

Kangaroo runs at $1m/s$. As the wrong estimations disturb the system continuously, the robot preserves its stability and continues running. The illustration of the behavior is shown in the supplemental video.

C. Comments on Real-Time Capability

Excluding MuJoCo's processing time, the overall control system outputs torque commands in less than one millisecond with our average daily usage computer with AMD Ryzen 7 5800X CPU.

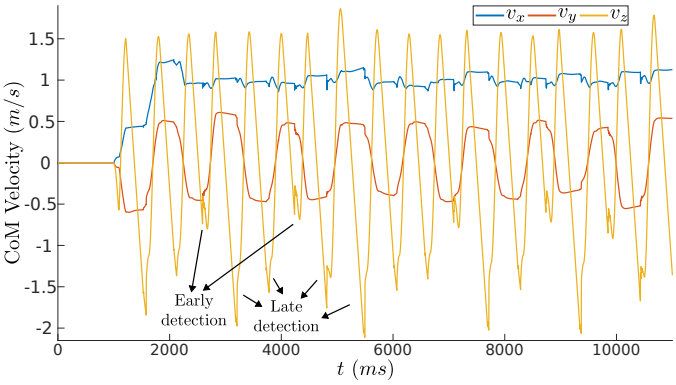

Fig. 11. The system's CoM velocity response when the robot starts from double supported complete stop and accelerates to $1m/s$ steady running. The contact is randomly activated either $5cm$ above the ground or $50ms$ later than the actual moment at each touchdown.

VIII. Conclusion

Focusing on enhancing the trajectory-switching and disturbance rejection capabilities of the conventional 3D SLIP model, this study proposed an actively controlled SLIP with a foot model without any constant energy level assumptions. Addressing the simplicity of the conventional SLIP model, the proposed control method utilizes the limbs of the humanoid robot better. It generates realistic behaviors and enables the system to initialize velocity from the stance phase, which is impossible with the conventional point foot 3D SLIP model, as it actively varies the force directions and magnitudes of CoM during the stance phase. We showed that even though the SLIP model does not account for impacts and multibody effects, precise tracking of these trajectories is possible with a proper task selection for the torso and swing leg, even with a robot that cannot take advantage of arm swings for stabilization and angular momentum dissipation.

A. Future Work

Thanks to the trajectory-switching capability of the proposed control model, the authors plan to extend this study to model running over obstacles and through stepping stones. We believe that with the help of a trajectory library and a proper trajectory-switching policy, precise foot placement control can be achieved to run over obstacles and stepping stones.

References

[1] R. J. Full and D. E. Koditschek, "Templates and anchors: neuromechanical hypotheses of legged locomotion on land," *Journal of experimental biology*, vol. 202, no. 23, pp. 3325–3332, 1999.

[2] P. Holmes, R. J. Full, D. Koditschek, and J. Guckenheimer, "The dynamics of legged locomotion: Models, analyses, and challenges," *SIAM review*, vol. 48, no. 2, pp. 207–304, 2006.

[3] A. Seyfarth, H. Geyer, M. Günther, and R. Blickhan, "A movement criterion for running," *Journal of Biomechanics*, vol. 35, no. 5, pp. 649–655, 2002.

[4] M. M. Ankarali and U. Saranli, "Stride-to-stride energy regulation for robust self-stability of a torque-actuated dissipative spring-mass hopper," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 20, no. 3, p. 033121, 09 2010.

[5] G. Secer and U. Saranli, "Control of planar spring–mass running through virtual tuning of radial leg damping," *IEEE Transactions on Robotics*, vol. 34, no. 5, pp. 1370–1383, 2018.

[6] U. Saranli, W. Schwind, and D. Koditschek, "Toward the control of a multi-jointed, monoped runner," in *Proceedings. 1998 IEEE International Conference on Robotics and Automation (Cat. No.98CH36146)*, vol. 3, 1998, pp. 2676–2682 vol.3.

[7] H. Geyer, "Simple models of legged locomotion based on compliant limb behavior," Ph.D. dissertation, 2005.

[8] P. M. Wensing and D. E. Orin, "High-speed humanoid running through control with a 3d-slip model," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 5134–5140.

[9] ——, "3d-slip steering for high-speed humanoid turns," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 4008–4013.

[10] ——, "Development of high-span running long jumps for humanoids," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 222–227.

[11] D. Koepl and J. Hurst, "Impulse control for planar spring-mass running," *Journal of Intelligent & Robotic Systems*, vol. 74, pp. 589–603, 2014.

[12] B. Dadashzadeh, H. R. Vejdani, and J. Hurst, "From template to anchor: A novel control strategy for spring-mass running of bipedal robots," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 2566–2571.

[13] J. Englsberger, P. Kozłowski, C. Ott, and A. Albu-Schäffer, "Biologically inspired deadbeat control for running: From human analysis to humanoid control and back," *IEEE Transactions on Robotics*, vol. 32, no. 4, pp. 854–867, 2016.

[14] Z. He and K. Yamamoto, "Humanoid running based on 3d cog-zmp model and resolved centroidal viscoelasticity control," in *2022 IEEE-RAS 21st International Conference on Humanoid Robots (Humanoids)*, 2022, pp. 645–650.

[15] T. Sugihara, K. Imanishi, T. Yamamoto, and S. Caron, "3d biped locomotion control including seamless transition between walking and running via 3d zmp manipulation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 6258–6263.

[16] A. Wu and H. Geyer, "The 3-d spring–mass model reveals a time-based deadbeat control for highly robust running and steering in uncertain environments," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1114–1124, 2013.

[17] M. Ernst, H. Geyer, and R. Blickhan, "Spring-legged locomotion on uneven ground: a control approach to keep the running speed constant," in *Mobile Robotics: Solutions and Challenges*. World Scientific, 2010, pp. 639–644.

[18] M. H. Raibert, *Legged robots that balance*. MIT press, 1986.

[19] S. Sovukluk, J. Englsberger, and C. Ott, "Whole body control formulation for humanoid robots with closed/parallel kinematic chains: Kangaroo case study," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2023, to appear.

[20] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd, "OSQP: an operator splitting solver for quadratic programs," *Mathematical Programming Computation*, vol. 12, no. 4, pp. 637–672, 2020.

[21] H. Pontzer, J. H. Holloway 4th, D. A. Raichlen, and D. E. Lieberman, "Control and function of arm swing in human walking and running," *Journal of Experimental Biology*, vol. 212, no. 4, pp. 523–534, 2009.

[22] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2012, pp. 5026–5033.