Bipedal Robotic Running on Stochastic Discrete Terrain

Ayush Agrawal and Koushil Sreenath

***Abstract*— To navigate over discrete terrain with large displacements in the stepping locations, bipedal robots have to be able to perform agile and dynamic maneuvers such as jumping or running while also satisfying strict constraints on foot placement and ground contact forces. In this paper, we analyze the problem of bipedal running over stochastically varying discrete terrain with large changes in step lengths. Specifically, our method is based on designing a library of running gaits that are two-step-periodic. We illustrate the capabilities of the proposed controller through numerical simulations of a five link underactuated robot RABBIT, running over discrete terrain with step lengths that vary between 0.6m and 1.2m. This is about 1.5 times the robot's leg lengths and twice the step length that could have been achieved by walking.**

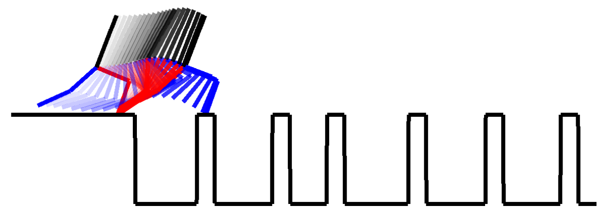

Fig. 1: Illustration of the stepping stones problem. The goal of the feedback control design is for the bipedal robot to traverse over a set of discrete terrain with wide gaps. Such a terrain can only be traversed by performing agile maneuvers like running or jumping.

I. Introduction

Legged robots have the promise of being able to serve in applications such as in space and urban exploration, as personal robots in homes and in search and rescue operations. It is their inherent morphology and mechanical structure that renders them as potentially superior candidates over their wheeled counterparts. A key task in such applications is the ability to locomote over discrete footholds such as in unstructured environments like wooded paths or over a flight of stairs in indoor environments. This, however, introduces several challenges and constraints including (a) Strict constraint on foot placement, (b) Friction constraints and (c) Input constraints. Violation of any of these constraints will render the system unstable.

A. Related Work

The problem of robotic legged bipedal walking over discrete terrain has been studied in the past, with a wide variety of techniques being used. Early methods for footstep planning, such as in [11], relied on simple models like the 3D inverted pendulum and cart-table models to generate walking patterns to walk over randomly generated stepping stones. In [20], the authors present a method based on the concept of capture points to control a bipedal robot to walk over discrete steps. The controller regulates the center-of-pressure on the stance foot so as to 'guide' the capture point to the desired stepping location. More recently, in [23], the authors present a centroidal momentum based controller for a high-dimensional robot ATLAS, to walk over partial footholds including line and point contact surfaces. In [4], a mixed integer quadratically constrained quadratic program is presented for footstep planning of a humanoid robot to walk on uneven terrain with obstacles. In [17], the authors present a method based on Control Barrier Functions (CBFs) to design feedback controllers for high dimensional 3D robots to walk over stochastically generated discrete steps with changing step heights and step lengths. In [14], the authors propose a method that combines a one-step-periodic gait library approach with a CBF based feedback controller that significantly improved the performance as in [18]. In [6], the authors propose a bio-inspired controller based on Central Pattern Generators (CPGs) to achieve step length and step height modulations over a wide range for bipedal walking.

With regards to bipedal robotic running, while numerous studies have been carried out, there have been limited studies on agile locomotion, like running, of legged systems over discrete terrain. One of the earliest works on running over rough terrain was by Hodgins and Raibert [9]. The authors presented three intuitive control designs for regulating the step length for a bipedal robot that could run over rough terrain including stairs with changing step heights. In [19], the authors propose a reinforcement learning based approach to control a physics-based legged character to navigate over terrain with wide gaps, steps and obstacles. The authors are able to translate their method to a wide variety of high-dimensional characters including a 21-link planar dog and a 7-link planar biped. In [5], the authors propose an intuitive dead-beat control strategy based on a point-mass model for running over 3D stepping stones. The authors perform numerical simulations on a point-mass model with massless legs for running over 3D stepping stones. In [7], the authors present a neuromuscular controller for bipedal running.

In our most recent work [15], [16], we presented a method to design a feedback controller for an underactuated legged robot to walk over discrete terrain with stochastically varying step heights and step lengths. The method relied on pre-

Ayush Agrawal and Koushil Sreenath are with Department of Mechanical Engineering, University of California at Berkeley {ayush.agrawal, koushils}@berkeley.edu

This work was supported in party by NSF Grant IIS-1834557.

computing a library of a small number of walking gaits (through an offline nonlinear program) that were two-step-periodic and parametrized by the step lengths and step heights in the first and second steps. The controller then performed a bilinear interpolation between the different gaits based on the current and desired step lengths/heights during run-time. By switching among a set of two-step-periodic gaits, the controller was able to achieve aperiodic walking, with precise footstep placement over randomly generated discrete terrain. The control method was successfully implemented on an underactuated bipedal robot, ATRIAS, over a wide range of complex terrain.

B. Problem Statement and Approach

In this paper, we study the problem of bipedal robotic running over randomly varying discrete terrain with large changes in step lengths (Figure 1). In particular, we develop a model-based feedback controller for an underactuated legged system that does not have knowledge of the entire terrain ahead of time but only the position of the next stepping location is known. By only using a one-step preview of the upcoming stepping location, the method renders itself capable of being combined with vision sensors (like cameras and LiDAR) to estimate the stepping locations.

Motivated by the success of the 'two-step-periodic' gait library approach, we extend this approach to the case of bipedal running. A primary advantage of the proposed method is that, unlike most other methods that rely on simplifications of the system dynamics, it considers the full nonlinear hybrid dynamics of the system, both, during offline gait generation and during the control phase. Given the current state-of-the-art in trajectory optimization for nonlinear hybrid systems and computational power, another advantage of our method is that it is easy to implement on a physical system. A key reason for the success of the two-step-periodic gait approach in [16] was it allowed for smooth transitions between walking gaits at different step-lengths/heights, thereby inherently preventing violations in friction cone and unilateral ground reaction force constraints and by using only a small number of gaits in the gait library. The method, therefore, seems promising to use in richer locomotion behaviors such as running and jumping.

The problem of robotic running over discrete footholds, however, places some additional challenges which stems from the loss of control authority of the angular momentum during flight phases (i.e. when the robot is completely in the air). Additional challenges also arise due to the increased number of possible hybrid modes of the system. These are potential reasons for why the applicability of the 'two-step-periodic' gait library approach might not be straightforward. In particular, we show that by reasoning about the dynamics and by the proper choice of output variables to be controlled, the two-step-periodic gait library approach can be extended to the case of running as well.

C. Contributions

The key contributions of this paper are as follows:

1) We extend the two-step-periodic gait library approach, initially presented in [15], [16] to develop a control strategy for bipedal robotic running over discrete footholds that follows strict constraints on the states and control inputs of the system;
2) By doing so, we are able to expand the capabilities of the robot to traverse over wider gaps in the discrete footholds that was not possible with only walking;
3) We show that, by a proper selection of output variables to be controlled, we can significantly reduce the error in the foot placement locations while running.

D. Organization

The rest of the paper is organized as follows: In Section II, we present the hybrid model for running, followed by the trajectory optimization and control design method in Section III. Finally, in Section IV, we present results from numerical simulation of a five link underactuated bipedal robot.

II. Dynamical Model for Running

In this section, we present a brief overview of the dynamical model of a planar five-link two legged robot for running behaviors. These models will be used for generating optimal trajectories as well as for control synthesis. The specific robot under consideration is RABBIT. More details about the dynamical model can be found in [2].

Figure 2 illustrates a schematic diagram of the robot. The configuration variables, defined as $q := \begin{bmatrix} p^x_{hip} & p^z_{hip} & q_T & q_{1R} & q_{2R} & q_{1L} & q_{2L} \end{bmatrix}^T$ include the world frame position of the hip $\begin{bmatrix} p^x_{hip} & p^z_{hip} \end{bmatrix}^T$, world frame orientation of the torso q_T and the relative joint angles of the thigh q_1 and shin q_2 links. The subscripts L and R refer to the left and right links respectively. For future reference, we define the actuated joints $q_a := \begin{bmatrix} q_{1R} & q_{2R} & q_{1L} & q_{2L} \end{bmatrix}$ The equations of motion are derived using the method of Lagrange and have the form during stance,

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = Bu + J^T F, \tag{1}$$

where $u \in \mathbb{R}^4$ are the control inputs that actuates each of the joints q_a, $F \in \mathbb{R}^2$ are the ground reaction forces at the stance foot and $J := \frac{\partial p_{st}}{\partial q} \in \mathbb{R}^{2\times 7}$ is the Jacobian of the stance foot position p_{st}. We note that during flight, since there are no external contact forces acting on the robot, $F \equiv 0$.

Like walking, the dynamics for running motions is also hybrid. Specifically, running comprises of alternating phases of single-support Σ_s and flight Σ_f phases as illustrated in Figure 3. The hybrid dynamics is written as

$$\begin{aligned} \Sigma_s &: \begin{cases} \dot{x} &= f_s(x) + g_s(x)u, \ (x,u) \notin \mathcal{S}_{s\to f} \\ x^+ &= \Delta_{s\to f}(x^-), \ (x,u) \in \mathcal{S}_{s\to f} \end{cases} \\ \Sigma_f &: \begin{cases} \dot{x} &= f_f(x) + g_f(x)u, \ x \notin \mathcal{S}_{f\to s} \\ x^+ &= \Delta_{f\to s}(x^-), \ x \in \mathcal{S}_{f\to s} \end{cases} \end{aligned} \tag{2}$$

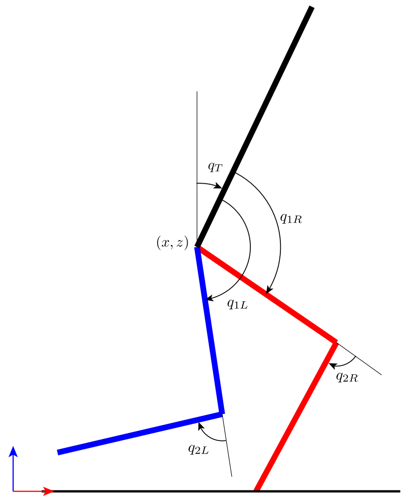

Fig. 2: Generalized coordinates of the bipedal robot RABBIT.

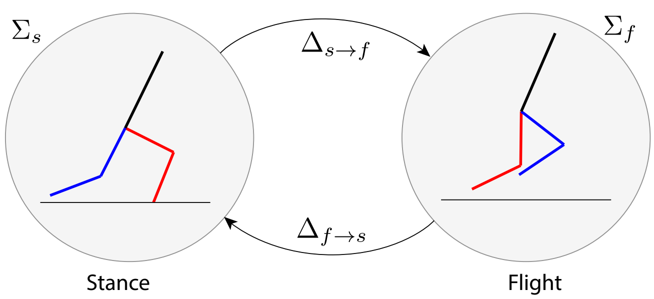

Fig. 3: Illustration of the domains in running.

Here $\mathcal{S}_{s\to f} := \{(x,u) \mid F^z(x,u) = 0\}$ is the switching surface corresponding to the transition between stance and flight domains and is defined as the set of states and control inputs such that the vertical ground reaction force $F^z(x,u)$ is zero (this corresponds to the case when the stance foot lifts off from the ground). Similarly, we define the switching surface $\mathcal{S}_{f\to s} := \{x \mid p^z_{sw}(x) = 0\}$ corresponding to the transition between flight to stance as the set of states such that the vertical component of the position of the swing foot is zero.

In addition, $\Delta_{s\to f} = \mathcal{I}$ is the identity operator and $\Delta_{f\to s}$ is obtained from rigid impact dynamics. The vector fields $f(x)$ and $g(x)$ are obtained for stance and flight phases using Lagrange's equations of motion (1).

In the next sections, we present our control approach based on the 'two-step-periodic' gait library.

III. Hybrid Zero Dynamics Based Control

In this section, we briefly present the Hybrid Zero Dynamics (HZD) framework [22], [21] which uses the dynamical model presented in Section II to generate periodic gaits and design feedback controllers for running. The HZD method begins by selecting a set of outputs y_a for the hybrid dynamical system as in (2). Driving these outputs to a set of desired quantities y_d define how the various links of the robot move. The HZD controller then implements an Input-Output (IO) Linearizing controller to drive the outputs y_a to y_d.

A. Output Selection

In this section we present our choice of outputs y_a (also known as virtual constraints). We note that several valid choices for the outputs exist. As in [15], a candidate for y_a is the set of actuated joint angles q_a. However, since we are interested in achieving precise step-lengths, which is achieved through the flight phase, we choose the horizontal velocity of the center of mass as one of the outputs during the stance phase. This choice of output is motivated by the fact that the horizontal distance achieved during flight is dependent on the exit-velocity of the stance phase (velocity of the center of mass at the instant before entering flight phase). We also note that this is a *relative degree 1 output* [10] [1]. The complete set of outputs during the stance phase is chosen as

$$y^s_a = \begin{bmatrix} v^x_{com} \\ q^{st}_1 \\ q^{sw}_1 \\ q^{sw}_2 \end{bmatrix}. \tag{3}$$

During flight, we choose the following outputs to be controlled

$$y^f_a := \begin{bmatrix} q_T \\ q^{st}_2 \\ q^{sw}_1 \\ q^{sw}_2 \end{bmatrix} \tag{4}$$

The superscripts st and sw in y^s denote stance and swing legs respectively, and $st = L/R$ (and $sw = R/L$), depending on whether the Left or Right Leg is in stance respectively. In y^f, the superscripts st and sw denote the stance and swing legs in the preceding stance phase.

The desired outputs $y_d := y_d(\tau^p, \alpha^p)$ are parametrized by Bézier splines, where α^p are the coefficients of the Bézier spline and τ^p is a phase variable that monotonically increases from 0 to 1 and $p \in \{s, f\}$. Specifically, we will find the Bézier parameters α^p and phase variable parameters through a Nonlinear Program such that enforcing the outputs y_a to the desired outputs $y_d(\tau^p, \alpha^p)$ through a feedback controller will result in a *two-step-periodic* solution for the hybrid model in (2). This is schematically illustrated in Figure 4. We note that, there are several choices for the phase variable τ^p. In particular, we choose the normalized absolute stance leg-angle q^{st}_{LA} as the phase variable during stance and normalized time during flight,

$$\tau^s := \frac{q^{st}_{LA} - q^{st}_{LA,max}}{q^{st}_{LA,max} - q^{st}_{LA,max}}, \tag{5}$$

$$\tau^f := \frac{t - t_{min}}{t_{max} - t_{min}}, \tag{6}$$

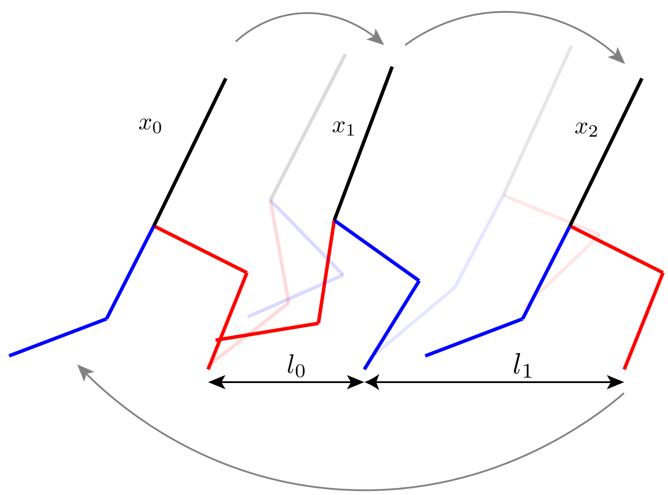

Fig. 4: Illustration of two-step-periodic gait design. We optimize over two running steps with constraints on the step lengths l_0 and l_1 in the first and second running steps respectively. The periodicity constraint enforces the states of the robot at the end of the second step x_2 to return to the states at the beginning of the first step x_0.

with q^{st}_{LA} defined as

$$q^{st}_{LA} := q_T + q^{st}_1 + \frac{q^{st}_2}{2} - \frac{\pi}{2}; \tag{7}$$

$q^{st}_{LA,max}, q^{st}_{LA,min}, t_{min}, t_{max}$ are constants to be determined. For future reference, we collect the constant parameters used in the gait phase variable definitions above into the following vectors,

$$\theta^s := \begin{bmatrix} q^{st}_{LA,max} \\ q^{st}_{LA,min} \end{bmatrix}, \tag{8}$$

$$\theta^f := \begin{bmatrix} t_{max} \\ t_{min} \end{bmatrix}. \tag{9}$$

B. Two-Step-Periodic Gait Design

Having presented the hybrid dynamical model for running and the choice of outputs y_a to be controlled, we now present a method to find the parameters α^s, α^f, θ^s and θ^f, such that the resulting gaits are two-step-periodic.

The two-step-periodic gait design involves obtaining gait parameters such that the post-impact states of the system after two running steps return to the initial states at the start of the first step. The gaits are parametrized by the step-lengths in the first and second second running step l_0 and l_1 respectively and we define the set of parameters as

$$\mathcal{P}\left(l_0, l_1\right) := \{\alpha^s\left(l_0, l_1\right) \alpha^f\left(l_0, l_1\right), \theta^s\left(l_0, l_1\right), \theta^f\left(l_0, l_1\right)\}. \tag{10}$$

Subsequently, we find parameters $\mathcal{P}\left(l_0, l_1\right)$ for $(l_0, l_1) \in L \times L$ to build a library of gaits,

$$\mathcal{G} := \{\mathcal{P}\left(l_0, l_1\right) \mid (l_0, l_1) \in L \times L\}, \tag{11}$$

Motor Torque	$\lvert u\rvert \leq 10\,\mathrm{N\,m}$
Friction Cone	$\left\lvert \frac{F^h_{st}}{F^v_{st}} \right\rvert \leq 0.6$
Vertical Ground Reaction Force during stance	$F^v_{st} \geq 0\,\mathrm{N}$
Swing Foot Clearance during stance	$h_f\rvert \geq 0.05\,\mathrm{m}$

TABLE I: Optimization constraints

where L is a predefined set of step lengths. Specifically, we choose $L = \{0.6, 0.8, 1.0, 1.2\}$, with a total of 16 gaits in the gait library.

The problem of obtaining the gait library $\mathcal{G}$ is cast as a nonlinear program with the objective function taken as the integral of squared torques over step length:

$$J = \int_0^T ||u(t)||_2^2 \, dt. \tag{12}$$

and constraints for the optimization are formulated as in Table I.

In addition to the above constraints, we also need to guarantee the periodicity of the gait through the *periodicty constraints*:

1) The initial state at start of the first stance phase is given by $x = x_0^+$with corresponding (initial) step length l_0.
2) *Transition constraints between stance and flight:* The state at the end of the first stance phase is equal to the state at the beginning of the first flight phase (corresponding to the step length l_0).
3) *Step Length constraint:* The step-length constraint is enforced as the difference between the position of the stance foot at the beginning of the flight phase and the position of the swing foot at the end of the flight phase being equal to the desired step-length l_0.
4) The state at the end of the first flight phase (before impact) is $x = x_1^-$ with (resulting) step length l_0.
5) Impact constraints at the end of the first flight-phase are enforced as $x_1^+ = \Delta(x_1^-)$.
6) The initial state at start of the second stance phase is given by $x = x_1^+$ with corresponding (initial) step length of l_1.
7) *Transition between Stance and Flight phase:* The constraint is enforced as in 2 between the second stance and flight phase.
8) The state at the end of the second flight phase (before impact) is $x = x_2^-$ with (resulting) step length of l_2.
9) Impact constraints at the end of the second step are enforced as $x_2^+ = \Delta(x_2^-)$.
10) Periodic constraints are then enforced as $x_2^+ = x_0^+$, resulting in $l_2 = l_0$.

The generation of the two-step-periodic running gaits using direct collocation with the specifications mentioned above, involves discretization of each phase in time by a specified number of nodes N,

$$0 = t_0 < t_1 < t_2 < \cdots < t_N = T, \tag{13}$$

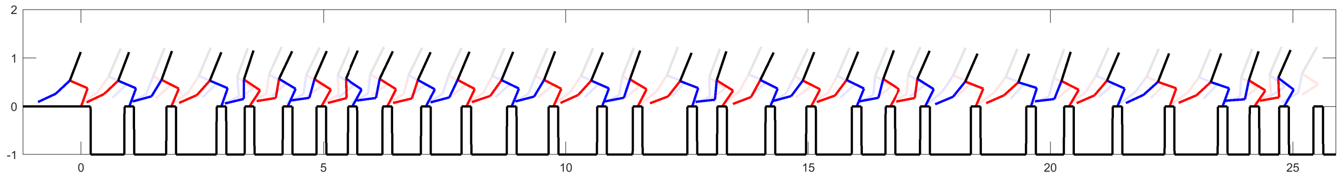

Fig. 5: Snapshots of the robot running over the discrete footholds using the control method presented in this paper.

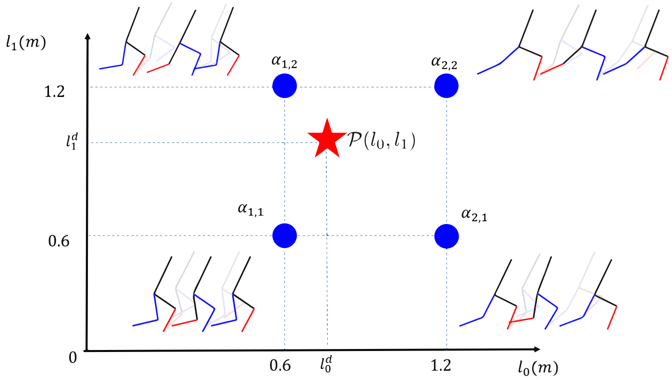

Fig. 6: Illustration of gait interpolation. The desired gait parameters are obtained based on the desired step length of the previous step l_0^d and the desired step length of the current step l_1^d. Points marked by blue circles denote gait parameters in the gait library. Red star denotes the gait parameters based on the desired step lengths and obtained using bilinear interpolation of the existing gaits in the gait library.

where T represents the time to impact. In particular, we use $N = 10$ for each phase and use the method of Direct collocation to solve the following trajectory optimization problem,

$$\begin{aligned} J = \min_{u(t)} \quad & \int_0^T ||u(t)||_2^2 \, dt \qquad (14) \\ \text{st.} \quad & x(t) = \int_0^t f(x(t)) + g(x(t))u(t)dt \\ & c(x(t), u(t)) \leq 0, \quad 0 \leq t \leq T. \end{aligned}$$

Here, $c(x(t), u(t))$ represent the physical constraints described in Table I as well as *periodicity constraints*. The desired gait parameters $\mathcal{P}(l_0, l_1)$ can be extracted from the optimal state trajectories through a simple Bézier curve fit. We use the open-source optimization and simulation toolbox `FROST` [8] to perform the above optimization. We refer the reader to [12] for more details on the specifics of the trajectory optimization scheme. We then generate the gait library $\mathcal{G}$ by obtaining the parameters $\mathcal{P}(l_0, l_1)$ for different values of $(l_0, l_1) \in L \times L$ through the NLP described above.

C. Control Design

We use an input-output linearizing controller as in [22], [21]. We first define the outputs to be regulated to zero as the difference between the actual and desired quantities y_a and y_d as:

$$y^p := y_a^p - y_d^p, \; p \in \{s, f\}. \qquad (15)$$

We further differentiate between the relative degree one and relative degree two outputs during stance as

$$y_1^s := \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} (y_a^s - y_d^s), \qquad (16)$$

$$y_2^s := \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} (y_a^s - y_d^s). \qquad (17)$$

The IO Linearizing controller is then given by,

$$u = (A^p)^{-1} (-B^p(x, \alpha^p, \theta^p) + v^p), \; p \in \{s, f\}, \qquad (18)$$

where A^p is the decoupling matrix and is defined as

$$A^s := \begin{bmatrix} L_g y_1^s \\ L_g L_f y_2^s \end{bmatrix} \qquad (19)$$

$$A^f := L_g L_f y^f \qquad (20)$$

and B^p is defined as

$$B^s := \begin{bmatrix} L_f y_1^s \\ L_f^2 y_2^s \end{bmatrix} \qquad (21)$$

$$B^f := L_f^2 y^f. \qquad (22)$$

Here, $L_f y$ and $L_g y$ denote the Lie-derivatives of the output y with respect to the vector fields f and g. Further, v is a feedback term, which could be, for example, a linear feedback controller

$$v^s := \begin{bmatrix} -K_1 y_1^s \\ -K_2 y_2^s - K_3 \dot{y}_2^s \end{bmatrix} \qquad (23)$$

$$v^f := -K_4 y^f - K_5 \dot{y}^f, \qquad (24)$$

where $K_i > 0, i = 1, 2, ...5$ are appropriate gain matrices.

The specific values of the parameters α^p and θ^p depend on the desired step-lengths of the preceding step l_0^d and current step l_1^d. The superscript d represents a desired quantity. We restrict the desired step lengths l_1^d to be within the range of L. This is schematically illustrated in Figure 6 We use bilinear interpolation of the gait parameters $\mathcal{P}$ (10) as in [15] to compute the gait parameters $\mathcal{P}(l_0, l_1)$ corresponding to the step lengths l_0^d and l_1^d.

Remark: 1. In our method, we perform a linear interpolation of the Bézier parameters that parametrize the periodic gaits rather than the time trajectories of the states. The interpolated gait is therefore also a smooth Bézier curve. The main motivation behind doing so is since the desired step length is different in every step, planning for each of these

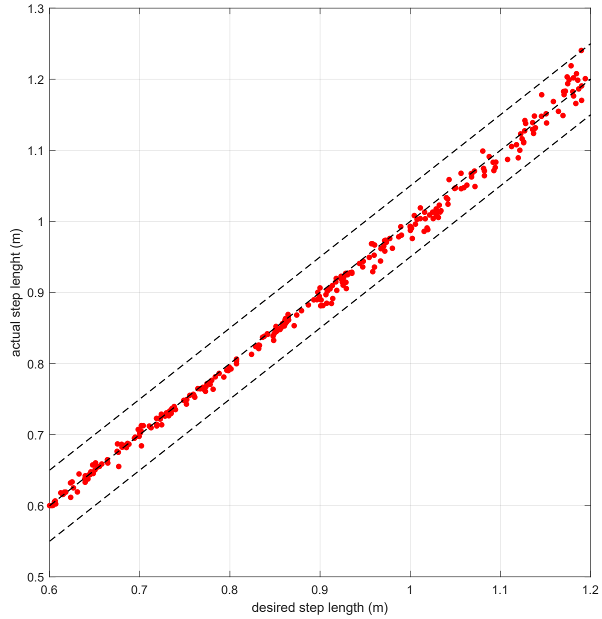

Fig. 7: Resulting location of foot steps from 300 steps, over 10 experiments (30 steps per experiment). Outer dashed lines indicate a $5\mathrm{cm}$ deviation from the desired step length. Red dots denote the actual value of the step length obtained from simulation.

transients would cause an explosion of planned trajectories. Instead, we plan for periodic gaits - specifically a two-step periodic gait to build a library of gaits $\mathcal{G}$ as in (11). During implementation, depending on the current step length l_0 and the desired step length l_1 of the next step, we select the four closest gaits (in terms of step length) from $\mathcal{G}$ and perform a bilinear interpolation (See Figure 6) resulting in $\mathcal{P}\left(l_0, l_1\right)$. Specifically, we only use the first step of the interpolated two-step periodic gait and then switch to another interpolated gait at the end of the first step. This makes the transients smoother (as opposed to large jumps in the desired outputs which generally causes a violation of unilateral constraints such as friction constraints and input constraints) as the exit state at the end of the first step is close to the entry state of the periodic gait used for the second step.

Remark: 2*.* The proposed method performs a *linear* interpolation as opposed to a nonlinear interpolation. There are certainly several ways to represent this nonlinear model. For example, in [3], the authors propose Support Vector Machines (SVMs) and neural network model to interpolate between the different gaits.

IV. Numerical Validation

In this section, we present our numerical results from simulation of 5-link underactuated robot RABBIT. We performed multiple simulations with randomly varying desired step lengths. The desired step lengths were sampled from a uniform distribution between $0.6\mathrm{m}$ and $1.2\mathrm{m}$ which is about twice the desired step length reported in [18] and about 1.5 times the robot's leg length.. Figure 5 presents snapshots of

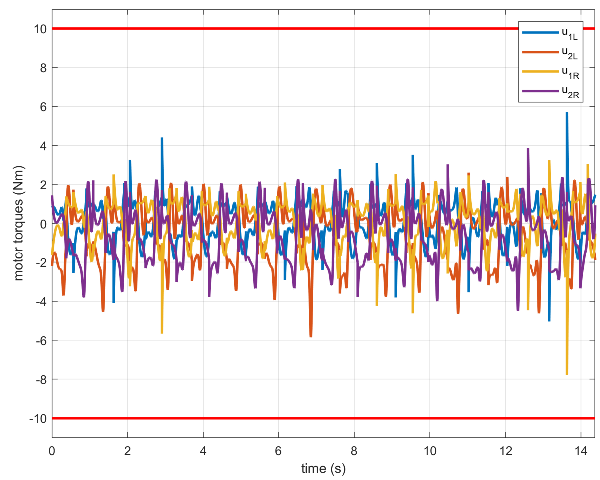

Fig. 8: Motor Torques from one of the simulations. The top and bottom red lines indicate the maximum and minimum allowable control inputs.

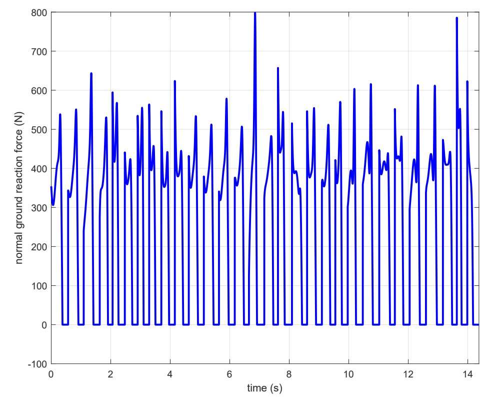

Fig. 9: Vertical Ground Reaction Forces from one of the simulations.

the robot running over one realization of the discrete terrain. Figure 7 illustrates the foot step locations along with the desired stepping locations from ten such simulations with 30 steps per simulation. We note that the average step-length error increases as the desired step-length increases.

In all our simulations, the foot placement was accurate to $\pm 5\mathrm{cm}$ of the desired step lengths while all other constraints such as input limits (Figure 8) and unilateral vertical ground reaction force (Figure 9) constraints were met. The average running velocity was $1.8\mathrm{m/s}$.

Remark: 3*.* As mentioned in Section III-A, a candidate choice for the outputs during the stance phase are the actuated joint angles q_a (all outputs have relative degree two). However, the results we obtained using these outputs were very different from those obtained using the outputs defined in (3) with one relative degree one output. In particular, we observe a significantly poor performance in the placement of footsteps with a maximum error of $39\mathrm{cm}$. We attribute the success of the outputs in (3) to the fact that the horizontal displacement of the center of mass depends solely on its exit velocity during stance. Regulating the exit velocity directly

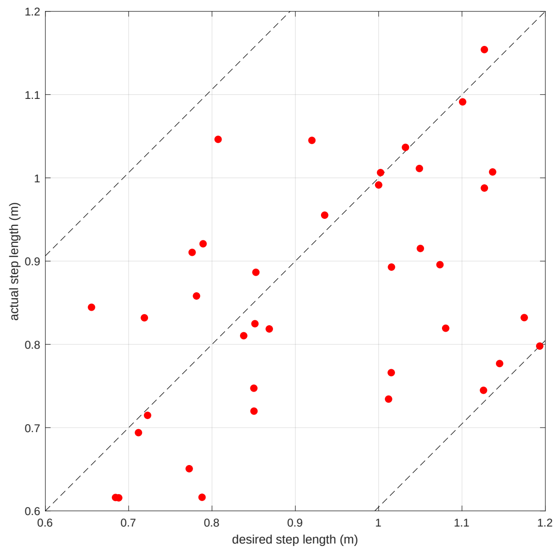

Fig. 10: Resulting location of foot steps from 50 steps from a single simulation using q_a (vector relative degree 2) as the outputs during stance phase. Outer dashed lines indicate a $39\mathrm{cm}$ deviation from the desired step length. Red dots denote the actual value of the step length obtained from simulation.

during stance will potentially lead to an accurate step-length at the end of the flight phase and hence smaller errors in step length. Figure 10 illustrates the performance of the controller using the actuated joints q_a as the outputs.

V. Conclusion

In conclusion, we have presented a control strategy for bipedal robotic running over stochastically varying discrete terrain and potentially increased the range of step lengths to twice that could have been achieved only by walking. The controller maintains the stability of the robot while respecting critical safety constraints such as constraints on foot placement and ground reaction forces. With proper choice of the outputs to be controlled, the resulting step length from simulation is accurate to $5\mathrm{cm}$ of the desired step length. While we are yet to formally prove any theoretical guarantees on switching between the different periodic gaits, we provide some intuitive explanation about the success of the method as in Remark 1. A potential direction to address this are the stability conditions for switching controllers between different exponentially stable periodic orbits as provided in [13]. Moreover, the method here is simple to implement and requires a small number of two-step periodic gaits to run over a terrain with a wide range of step lengths.

References

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

[2] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. Westervelt, C. C. de Wit, and J. Grizzle, "Rabbit: A testbed for advanced control theory," *IEEE Control Systems Magazine*, vol. 23, no. 5, pp. 57–79, 2003.

[3] X. Da, R. Hartley, and J. W. Grizzle, "Supervised learning for stabilizing underactuated bipedal robot locomotion, with outdoor experiments on the wave field," in *2017 IEEE International Conference on Robotics and Automation*, 2017, pp. 3476–3483.

[4] R. Deits and R. Tedrake, "Footstep planning on uneven terrain with mixed-integer convex optimization," in *IEEE-RAS International Conference on Humanoid Robots*, 2014, pp. 279–286.

[5] J. Englsberger, P. Kozłowski, and C. Ott, "Biologically inspired deadbeat control for running on 3d stepping stones," in *IEEE-RAS 15th International Conference on Humanoid Robots*, 2015, pp. 1067–1074.

[6] P. Greiner, N. Van Der Noot, A. J. Ljspeert, and R. Rousse, "Continuous modulation of step height and length in bipedal walking, combining reflexes and a central pattern generator," in *IEEE International Conference on Biomedical Robotics and Biomechatronics*, 2018, pp. 342–349.

[7] M. Harding, N. Van der Noot, B. Somers, R. Ronsse, and A. J. Ijspeert, "Augmented neuromuscular gait controller enables real-time tracking of bipedal running speed," in *IEEE International Conference on Biomedical Robotics and Biomechatronics*, 2018, pp. 364–371.

[8] A. Hereid and A. D. Ames, "Frost: Fast robot optimization and simulation toolkit," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2017, pp. 719–726.

[9] J. K. Hodgins and M. Raibert, "Adjusting step length for rough terrain locomotion," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 289–298, 1991.

[10] Z. Huihua, S. N. Yadukumar, and A. D. Ames, "Bipedal robotic running with partial hybrid zero dynamics and human-inspired optimization," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 1821–1827.

[11] 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," in *IEEE International Conference on Robotics and Automation*, vol. 3, 2003, pp. 1620–1626.

[12] W.-L. Ma, A. Hereid, C. M. Hubicki, and A. D. Ames, "Efficient hzd gait generation for three-dimensional underactuated humanoid running," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2016, pp. 5819–5825.

[13] M. S. Motahar, S. Veer, and I. Poulakakis, "Composing limit cycles for motion planning of 3d bipedal walkers," in *IEEE Conference on Decision and Control*, 2016, pp. 6368–6374.

[14] Q. Nguyen, X. Da, J. Grizzle, and K. Sreenath, "Dynamic walking on stepping stones with gait library and control barrier," in *Workshop on Algorithimic Foundations of Robotics*, 2016.

[15] Q. Nguyen, A. Agrawal, X. Da, W. C. Martin, H. Geyer, J. W. Grizzle, and K. Sreenath, "Dynamic walking on randomly-varying discrete terrain with one-step preview," in *Robotics: Science and Systems (RSS)*, 2017.

[16] Q. Nguyen, A. Agrawal, W. Martin, H. Geyer, and K. Sreenath, "Dynamic bipedal locomotion over stochastic discrete terrain," *The International Journal of Robotics Research*, vol. 37, no. 13-14, pp. 1537–1553, 2018.

[17] Q. Nguyen, A. Hereid, J. W. Grizzle, A. D. Ames, and K. Sreenath, "3d dynamic walking on stepping stones with control barrier functions," in *IEEE Conference on Decision and Control*, 2016, pp. 827–834.

[18] 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*, vol. 48, no. 27, 2015, pp. 147–154.

[19] X. B. Peng, G. Berseth, and M. Van de Panne, "Dynamic terrain traversal skills using reinforcement learning," *ACM Transactions on Graphics*, vol. 34, no. 4, p. 80, 2015.

[20] J. E. Pratt and R. Tedrake, "Velocity-based stability margins for fast bipedal walking," in *Fast Motions in Biomechanics and Robotics*. Springer, 2006, pp. 299–324.

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

[22] E. R. Westervelt, J. W. Grizzle, and D. E. Koditschek, "Hybrid zero dynamics of planar biped walkers," *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 42–56, 2003.

[23] G. Wiedebach, S. Bertrand, T. Wu, L. Fiorio, S. McCrory, R. Griffin, F. Nori, and J. Pratt, "Walking on partial footholds including line contacts with the humanoid robot atlas," in *IEEE-RAS International Conference on Humanoid Robots*, 2016, pp. 1312–1319.