Central Pattern Generator with Inertial Feedback for Stable Locomotion and Climbing in Unstructured Terrain

Guillaume Sartoretti[1], Samuel Shaw[2], Katie Lam[1], Naixin Fan[1], Matthew Travers[1], and Howie Choset[1]

***Abstract*— Inspired by the locomotor nervous system of vertebrates, central pattern generator (CPG) models can be used to design gaits for articulated robots, such as crawling, swimming or legged robots. Incorporating sensory feedback for gait adaptation in these models can improve the locomotive performance of such robots in challenging terrain. However, many CPG models to date have been developed exclusively for open-loop gait generation for traversing level terrain. In this paper, we present a novel approach for incorporating inertial feedback into the CPG framework for the control of body posture during legged locomotion on steep, unstructured terrain. That is, we adapt the limit cycle of each leg of the robot with time to simultaneously produce locomotion and body posture control. We experimentally validate our approach on a hexapod robot, locomoting in a variety of steep, challenging terrains (grass, rocky slide, stairs). We show how our approach can be used to level the robot's body, allowing it to locomote at a relatively constant speed, even as terrain steepness and complexity prevents the use of an open-loop control strategy.**

I. INTRODUCTION

In this work, we focus on the problem of controlling the body posture of legged robots, extending their locomotive ability to steep, unstructured terrain (shown in Fig. 1). Previous works have looked at legged locomotion on complex terrain, but have generally focused on more engineered solutions, such as precise foothold planning [1] when a terrain map is available, or reactive posture correction separately from locomotion [2], [3], [4]. In this work, we present a new method to directly incorporate inertial feedback into the bio-inspired central pattern generator (CPG) framework, in order to implement reactive posture correction directly as part of the locomotion.

Nature provides an effective solution to the complex problem of coordinated control in legged locomotion. Legged animals of varying sophistication subconsciously navigate extreme terrain with ease. In animals, CPGs are neural networks located in the spinal cord that produce signals to drive the rhythmic motions required for locomotion [5], [6]. Although CPGs do not rely on sensory inputs to produce locomotive outputs, these outputs are adapted based on proprioceptively-sensed environmental information [7], [8]. That is, sensory input is used to adapt the gait produced by the CPG in real time by inhibiting or extenuating certain gait characteristics.

[1]G. Sartoretti, K. Lam, N. Fan, M. Travers, and H. Choset are with the Robotics Institute at Carnegie Mellon University, Pittsburgh, PA 15213, USA. {gsartore,mtravers,klam1,nfan}@andrew.cmu.edu, choset@cs.cmu.edu

[2]S. Shaw is with Tufts University, Medford, MA 02155. samuel.shaw@tufts.edu



Fig. 1: Hexapod robot climbing down a rocky slope. The mounted camera is only used for point-of-view video recording.

In the context of robotics, CPGs are modeled as a system of coupled oscillators that controls joint motion. Such models have been widely used as a tool for gait generation and decentralized control of legged robots. For legged robots on level ground, CPGs have been used for open-loop gait generation to great success [9]. A handful of recent works have implemented closed-loop CPG implementations based on various sensory feedback [10], [11], [12], [13], [14]. However, CPG-based modelling still generally lacks generic methodologies for integrating sensory feedback to adapt the locomotion [13], [15].

In this paper, we present a method for adapting CPG parameters based on inertial feedback to obtain stable locomotion in unstructured terrain. Specifically, based on the robot's body position, orientation and height, our approach adapts the limit cycles of the coupled oscillators in the joint space to ensure stable locomotion. We demonstrate how, by levelling its body using inertial feedback as described, a legged robot's center of gravity (COG) is positioned more centrally within its support polygon, allowing a legged robot to traverse unstructured environments and climb up or down steep terrain without any *a priori* knowledge or other types of sensory feedback, such as vision. Additionally, in cases where a robot is equipped with a vision system, levelled-body locomotion can help gather clear visual feedback for autonomous operations or for user controlling the robot via remote camera feed. By relying on the CPG framework, our approach can be easily adapted to different legged robots and

with different gaits through simple parameter changes.

This paper is structured as follows: Section II provides background on the CPG framework and describes our CPG model. Section III describes how the CPG's limit cycle can be adapted to achieve stability based on inertial feedback. In Section IV, we detail our specific implementation and experimental setup, and validate our model experimentally on a hexapod robot. Section V discusses the experimental results, while Section VI summarizes our work and offers suggestions for future work.

II. CPG Modeling

CPGs are usually expressed as a set of coupled oscillators, i.e., a set of coupled ordinary differential equations. In the joint space of a n-legged robot, let $x(t) = [x_1(t), ..., x_n(t)]$ represent the angles of the shoulder joints of each leg in the axial plane, and $y(t) = [y_1(t), ..., y_n(t)]$ represent the angle of the shoulder joints in the sagittal plane.

In this work, we model the nominal CPG to which inertial feedback is added on [13], where the base CPG model reads:

$$\begin{cases} \dot{x_i}(t) &= -\omega \cdot y_i(t) + \gamma \left(\mu^2 - \sqrt{x_i(t)^2 + y_i(t)^2}\right) \cdot x_i(t) \\ \dot{y_i}(t) &= +\omega \cdot x_i(t) + \gamma \left(\mu^2 - \sqrt{x_i(t)^2 + y_i(t)^2}\right) \cdot y_i(t) \\ &\quad +(\lambda \sum_j K_{ij} y_j(t)), \end{cases} \tag{1}$$

where μ defines the radius of the circular limit cycle, ω defines the angular frequency of oscillation, γ defines the forcing to the limit cycle, λ defines the coupling strength, and K, the coupling matrix [13], defines the gait by setting the phase relationship between legs.

In CPGs, the limit cycle defines a closed path in the joint space, which ultimately influences the trajectory of the robots' end-effectors (feet/arms). In this work, we modify Eq.(1) to consider a limit cycle defined by the superellipse [16], [17], which reads:

$$H(x, y) = \left|\frac{x}{a}\right|^d + \left|\frac{y}{b}\right|^d, \tag{2}$$

where a and b define the major and minor axes of the limit ellipse (i.e., the maximum magnitude attained by x and y, respectively) and d defines the curvature. Specifically, $d = 2$ provides an elliptic limit cycle, whereas $d > 2$ provides a more rectangular limit cycle with sharper corners as d increases (an example comparing $d = 2$ and $d = 4$ is shown in Fig. 2). We believe that such a rectangular-looking limit cycle seems better suited for walking in unstructured environment. In general, the choice of d should be guided by the type of terrain the robot will evolve in: $d = 2$ seems allows quick and stable locomotion on flat ground, whereas $d = 4$ seems better suited for rough or inclined terrain. By inducing a clean, vertical trajectory of the end effector when stepping on/off the ground, this cycle limits the chances of the robot's feet getting caught by rocks/debris on the ground. In general, this family of limit cycles allows for independent control of step length, height, and shape through the selection of a, b, and d, respectively.

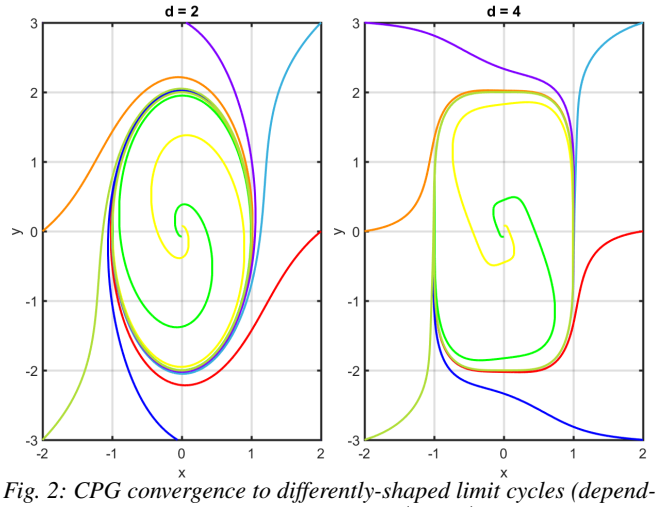


Fig. 2: CPG convergence to differently-shaped limit cycles (depending on d) from different initial positions (x_0, y_0) (different colors). The limit cycle reads Eq.(2) with $a = 1$, $b = 2$, $\gamma = 2$, and $\omega = 5$.

We adapt the CPG model shown in Eq.(1) to employ a general limit cycle $H(x, y)$ by relying on our previous work [18], and obtain:

$$\begin{cases} \dot{x_i}(t) &= -\omega \cdot \partial H_{y_i} + \gamma\Big(1 - H(x_i(t), y_i(t))\Big) \cdot \partial H_{x_i} \\ \dot{y_i}(t) &= +\omega \cdot \partial H_{x_i} + \gamma\Big(1 - H(x_i(t), y_i(t))\Big) \cdot \partial H_{y_i} \\ &\quad +(\lambda \sum_j K_{ij} y_j(t)), \end{cases} \tag{3}$$

with $\partial H_\zeta = \frac{\partial H}{\partial \zeta}(x_i(t), y_i(t))$. When H does not describe a circular limit cycle ($d > 2$), ω does not directly translate to the oscillation frequency, but is linearly linked to the gait period. Therefore, in the implementation, ω controls the forward velocity of the robot.

In order to extend the CPG in Eq.(3) by incorporating inertial feedback for body control, we consider the offsets c_x and c_y as the center of the limit cycle by defining:

$$H_c(x, y) = \left|\frac{x - c_x}{a}\right|^n + \left|\frac{y - c_y}{b}\right|^n. \tag{4}$$

By using $H_c(x, y)$ with leg-specific offsets $c_{x,i}$ and $c_{y,i}$, we achieve the following modified CPG equations:

$$\begin{cases} \dot{x_i}(t) &= -\omega \cdot \partial H_{y_i} + \gamma\Big(1 - H_{c_i}(x_i(t), y_i(t))\Big) \cdot \partial H_{x_i} \\ \dot{y_i}(t) &= +\omega \cdot \partial H_{x_i} + \gamma\Big(1 - H_{c_i}(x_i(t), y_i(t))\Big) \cdot \partial H_{y_i} \\ &\quad +(\lambda \sum_j K_{ij} (y_j(t) - c_{y,j}), \end{cases} \tag{5}$$

with $\partial H_\zeta = \frac{\partial H_{c_i}}{\partial \zeta}(x_i(t), y_i(t))$.

In this work, the offsets $c_{x,i}, c_{y,i}$ introduced in Eq.(5) will be used to achieve body control during the locomotion of the robot while preserving the step heights and forward speed. More specifically, the $c_{y,i}$'s are adapted at each time step to achieve body posture control by updating the vertical angle of the robot's shoulders, thus causing each leg to operate in a lower or higher range relative to the body. The $c_{x,i}$'s are only set initially to adjust the spread of the legs by selecting the angle of the robot's shoulders in the axial plane. In either case, note that the limit cycles are only translated in the joint space of the associated leg, meaning that the limit cycles' shape and scale are left unchanged.

III. Inertial Posture Control

In this section, we present our approach to body posture control, which adapts the limit cycle of each leg at each time step, based on inertial feedback. We first present our method for computing the end-effector error, such that when corrected the body will have achieved a desired orientation. We then detail our approach to adapting the vertical CPG offsets from the end-effector error.

A. Computing CPG Offsets Using end-effector Error

Let $P \in SO(3)$ be the orientation of the body in the world frame, which can be obtained from inertial sensors on the robot. Let $R \in SO(3)$, the world-frame rotation which positions the robot's body to a desired target roll-pitch orientation given by $T \in SO(3)$ in its body frame (in order to preserve its current heading).

Without limiting the generality of this derivation, we assume that the robot's heading is collinear with the y-axis of its body frame. We first need to align T with the current heading of the robot. To this end, we first look for the normalized, planar heading vector of the robot in the world's XY plane, which reads:

$$z_y = \frac{\begin{pmatrix}-i & 1 & 0\end{pmatrix} \cdot \left(P \cdot \begin{pmatrix}0 & 1 & 0\end{pmatrix}^T\right)}{\left\| \begin{pmatrix}-i & 1 & 0\end{pmatrix} \cdot \left(P \cdot \begin{pmatrix}0 & 1 & 0\end{pmatrix}^T\right) \right\|_2} = e^{i\,\theta_z} \tag{6}$$

From the yaw angle θ_z of z_y in the world frame, we can finally express the $SO(3)$ transformation, R, which brings the robot's body from its current pose, P, to its target pose in the world frame, as

$$R = \overbrace{R_z(\theta_z) \cdot T}^{\text{target pose in world frame}} \cdot P^{-1} = R_z(\theta_z) \cdot T \cdot P^T. \tag{7}$$

with $R_z(\theta_z) = \begin{pmatrix}\cos(\theta_z) & -\sin(\theta_z) & 0\\ \sin(\theta_z) & \cos(\theta_z) & 0\\ 0 & 0 & 1\end{pmatrix} \in SO(3)$. In particular, we are interested in keeping the robot's body level, i.e., $T = \mathbb{I}_3 \in SO(3)$ (the identity matrix), we finally simply obtain $R = R_z(\theta_z) \cdot P^T$.

Relying on the robot's forward kinematics, we obtain the end-effector positions $r \in \mathbb{R}^{3\times n}$ in the robot's body frame:

$$r = \begin{bmatrix} x_1 & \cdots & x_n \\ y_1 & \cdots & y_n \\ z_1 & \cdots & z_n \end{bmatrix}. \tag{8}$$

The corrected end-effector positions in the body frame, $\tilde{r}$, are then obtained by applying the rotation R to their value in the world frame, to achieve the desired body orientation (see Figure 3), before translating back into the body frame:

$$\tilde{r} = P^T \cdot R \cdot P \cdot r. \tag{9}$$

The z-components of $\tilde{r}$ are then adjusted to ensure that the body height, defined as the vertical distance between the geometric center of the body and the estimated ground plane, remains constant. We define e to be the error in end-effector position of each leg in the body frame, which reads

$$e = \tilde{r} - r. \tag{10}$$

Using the well-known kinematics equation,

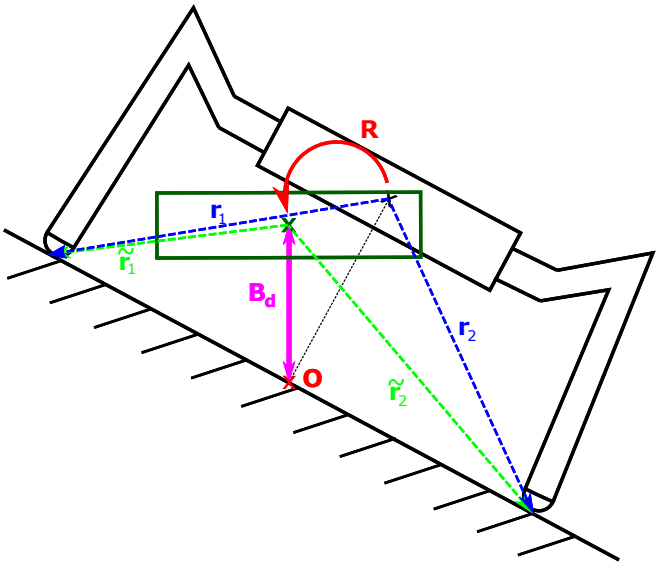


Fig. 3: Illustration of the body/end-effector rotations happening during stabilization. Resting on an inclined ground, the robot is shown in black, with its body as the black rectangle. The rotation $R \in SO(3)$, applied around the rotation point O (ground projection of the robot's center of gravity), enables the body to reach the correct orientation $T = \mathbb{I}_3$ and body height B_d (green rectangle). To achieve this rotation, the end effector positions r_i ($1 \leq i \leq 6$), calculated in the body frame, need to be corrected to $\tilde{r}_i$.

$$\dot{r} = J\dot{\theta}, \tag{11}$$

expanded as follows,

$$\begin{bmatrix} \dot{x_1} & \cdots & \dot{x_n} \\ \dot{y_1} & \cdots & \dot{y_n} \\ \dot{z_1} & \cdots & \dot{z_n} \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial \theta_1} & \cdots & \frac{\partial x}{\partial \theta_m} \\ \frac{\partial y}{\partial \theta_1} & \cdots & \frac{\partial y}{\partial \theta_m} \\ \frac{\partial z}{\partial \theta_1} & \cdots & \frac{\partial z}{\partial \theta_m} \end{bmatrix} \cdot \begin{bmatrix} \dot{\theta_{1,1}} \cdots \dot{\theta_{1,n}} \\ \vdots \quad\quad \vdots \\ \dot{\theta_{m,1}} \cdots \dot{\theta_{m,n}} \end{bmatrix}, \tag{12}$$

where $\dot{r}$ is the end-effector velocity in cartesian space, J the linear Jacobian, $\dot{\theta}$ the angular speed of the joints, n the number of legs present on the robot, and m the number of joints composing each leg. Approximating Eq.(12) for small Δt, we can estimate the error in joint angles $\Delta\theta_e \in \mathbb{R}^{m\times n}$:

$$\frac{e}{\Delta t} \approx J \cdot \frac{\Delta\theta_e}{\Delta t} \iff \Delta\theta_e \approx J^{\dagger} e \tag{13}$$

where e is the error in end-effector position given by Eq.(10), and $J^{\dagger}$ is the Moore-Penrose pseudoinverse.

Note that, in Eq.(10), the end-effector positions are rotated in the body frame. Therefore, the body's position in the world frame will be translated, as the legs are re-arranged. This correction is truly beneficial: on steep terrain, as the body is levelled, it will also be translated closer to the slope, and naturally be re-positioned over the support polygon created by the legs. This posture control mechanism, crucial when climbing steep terrain, helps stabilize the robot's locomotion (as demonstrated in Section V).

B. Adaptation of CPG Parameters for Body Posture Control

Eq.(13) allows us to translate the current end-effector error $e(t)$ into instantaneous joint offsets $\Delta\theta_e(t)$. In particular, the first two rows of $\Delta\theta_e(t)$ are of interest, as they directly influence c_x and c_y respectively.



Fig. 4: Hexapod leg joint configuration.

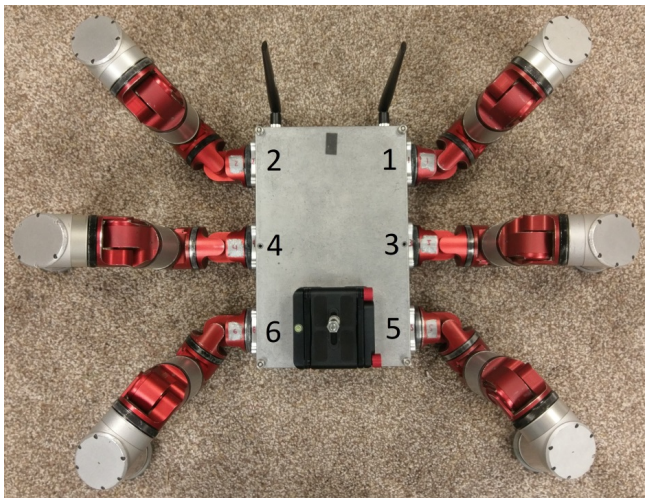


Fig. 5: Hexapod leg numbering convention.

Since the proposed body control aims at keeping the robot's heading constant in the world frame, it follows that the first row of $\Delta\theta_e(t)$ is composed of zeros (no change in c_x). That is, the body rotation R in Eq.(7) is in the null-space of the most proximal joints of each leg. However, by integrating the second component of $\Delta\theta_e(t)$ with time, we can adapt $c_y(t)$ with time from its initial value c_{y_0}:

$$c_y(t) = c_{y_0} + \int_0^t \Delta\theta_e(t)\ dt. \tag{14}$$

We additionally incorporate $\Delta\theta_{e_{2,i}}(t)$ (the second row of $\Delta\theta_e$ from Eq.(13)), the instantaneous vertical increment in $c_{y,i}$, into the CPG Eq.(5), which finally reads:

$$\begin{cases} \dot{x_i}(t) &= -\omega \cdot \partial H_{y_i} + \gamma\Big(1 - H_{c_i(t)}(x_i(t), y_i(t))\Big) \cdot \partial H_{x_i} \\ \dot{y_i}(t) &= +\omega \cdot \partial H_{x_i} + \gamma\Big(1 - H_{c_i(t)}(x_i(t), y_i(t))\Big) \cdot \partial H_{y_i} \\ &\quad +(\lambda \sum_j K_{ij}(y_j(t) - c_{y,j}) + \Delta\theta_{e_{2,i}}(t). \end{cases} \tag{15}$$

with $\partial H_\zeta = \frac{\partial H_{c_i(t)}}{\partial \zeta}(x_i(t), y_i(t))$.

IV. Hardware Experiments

In this section, we begin by describing the specifications of the hexapod robot used for our experimental validation. Then, we detail how we initialize the CPG parameters, and how we define joint angles for each leg based on the output of the CPG model. Finally, we present experimental results that compare open-loop to stabilized locomotion.

A. Robot Specifications

We use a modular hexapod robot as our experimental platform [19]. Each leg of the hexapod, shown in Fig. 5, consists of three modular joints: a proximal joint aligned with the yaw axis, and intermediate and distal joints aligned with the roll axis. The two most proximal joints act as the shoulder of the robot.

The robot's body is a rectangular prism with length, width, and height dimensions of 27cm, 17cm, and 7cm, respectively. Legs may extend from the body approximately 44cm horizontally or 32cm vertically.

The robot is blind, meaning that no on-board vision system is used to close the loop; on-board sensing is provided by the joint modules themselves, each containing an inertial measurement unit (IMU) and encoders [20].

B. CPG Implementation

Since a hexapod robot is used, an alternating tripod gait is chosen for its prevalence as a locomotive gait in insects, and for its static stability [21]; three legs always remain on the ground, forming a constant, large support polygon. In this sense, the stability of the robot in unstructured terrain relies only on the integrity of environment footholds. To implement this gait within the CPG framework, we use the following coupling matrix in Eq.(15)

$$K = \begin{bmatrix} 0 & -1 & -1 & 1 & 1 & -1 \\ -1 & 0 & 1 & -1 & -1 & 1 \\ -1 & 1 & 0 & -1 & -1 & 1 \\ 1 & -1 & -1 & 0 & 1 & -1 \\ 1 & -1 & -1 & 1 & 0 & -1 \\ -1 & 1 & 1 & -1 & -1 & 0 \end{bmatrix}. \tag{16}$$

Given the selected tripod gait, we define the estimated ground plane based on the positions of the lowest tripod, which is given by either $r_{3,(1,4,5)}$ or $r_{3,(2,3,6)}$ in Eq.(8). This allows us to measure the body height of the robot and calculate e in Eq.(10).

We initialize constant offsets, c_{x_0} and c_{y_0} on the center of the CPG limit cycle on a leg-by-leg basis:

$$\begin{aligned} c_{x_0} &= \begin{bmatrix} \frac{\pi}{4} & \frac{\pi}{4} & 0 & 0 & -\frac{\pi}{4} & -\frac{\pi}{4} \end{bmatrix} \\ c_{y_0} &= \begin{bmatrix} \frac{\pi}{16} & \frac{\pi}{16} & \frac{\pi}{16} & \frac{\pi}{16} & \frac{\pi}{16} & \frac{\pi}{16} \end{bmatrix} \end{aligned}. \tag{17}$$

It should be noted in the proposed approach, $c_x(t) = c_{x_0}$ throughout the run. That is, since c_x defines the spread of the legs, adapting it with time would decouple the legs' phases and may destabilize the gait.

The remaining CPG parameters are initialized as follows:

$$\gamma = 40, \quad \lambda = \tfrac{1}{4}, \quad a = \tfrac{\pi}{18}, \quad b = \tfrac{\pi}{6} \quad . \tag{18}$$

Note that since the values of x and y are used directly to assign joint angles to the proximal and intermediate joints, a and b are in radians.

Joint angles are set based on the CPG outputs as follows:

$$\begin{cases} \theta_{1,i} = x_i \\ \theta_{2,i} = \max(y_i, c_{y,i}) \\ \theta_{3,i} = f(\theta_{1,i}, \theta_{2,i}) \end{cases} . \quad (19)$$

As shown in Eq.(19), the x-outputs from the CPG determine $\theta_{1,i}$, the angles of the proximal joints. Similarly, the y-outputs describe $\theta_{2,i}$, the angles of the intermediate joints, with one exception: we assume that $y_i < c_{y,i}$ defines the set of legs that are on the ground, and so, we set $\theta_{2,i} = c_{y,i}$ to ensure a constant body height throughout the stance phase (limit-cycle clipping, as shown in the setting of $\theta_{2,i}$ in Eq.(19)). For implementation purposes, we make sure that the joint angle correction does not diverge (due to noise in pose estimation from onboard inertial sensors that may build up with time), by adding a small dissipation to the origin:

$$\Delta\theta_e = \Delta\theta_e - \alpha \cdot \theta_e, \quad (20)$$

where $\alpha \in \mathbb{R}$ governs the dissipative term.

The closed-form inverse kinematics function $f(\theta_{1,i}, \theta_{2,i})$ is used to determine $\theta_{3,i}$, the angle of the distal joint. This ensures that, as the proximal and intermediate joints move, the end-effector remains at a constant, desired distance from the body. Given this architecture, we allow the intermediate joint to control the body orientation, while the distal joint ensures an ideal, linear end-effector trajectory aligned with the robot's heading. In this sense, we distribute the different stability/locomotion constraints among the degrees-of-freedoms of each leg, to ensure that stability and locomotion do not interact adversarially. This constraint distribution also reduces the computational complexity, by allowing us to directly use the outputs of the CPG (x and y) for the two shoulder joints, while only solving an IK problem for the distal joint (closed-form solution in our case).

C. Experimental Results

We rely on the average forward locomotive speed of the robot, as the metric to measure the efficacy of the proposed stabilization method. Our goal is to keep the robot's body level (i.e., $T = \mathbb{I}_3$ in Eq.(7)) while locomoting in different terrain (planar or unstructured) of varying steepness. Locomotive speed comparison between the stabilized and the open-loop controllers were conducted on a twelve-degree slope, featuring randomly-placed and -oriented rocks that varied in elevation from approximately 5cm to 20cm. A twelve-degree slope provided a suitable comparison between methods, as its steepness is appropriate for examining both stability and locomotive speed (i.e., it is neither so steep that the open-loop controller will be unable to progress nor so shallow that body stability is no longer as essential).

An overhead camera was used to capture the robot's movement up the slope; feature-based tracking was used in OpenCV to record the position of the robot with time,

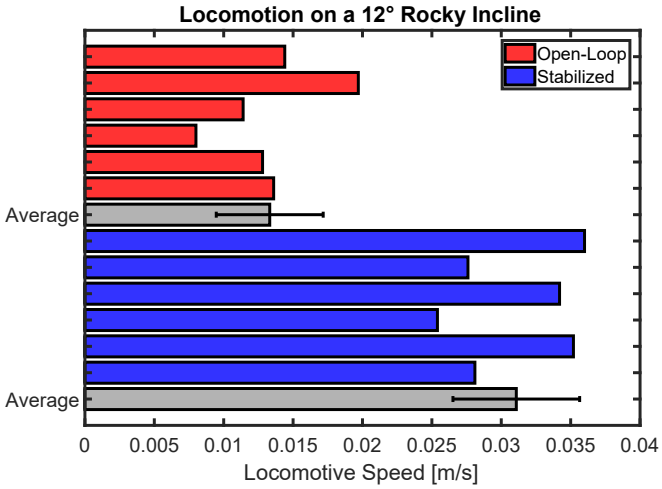


Fig. 6: Performance of stabilized trials (blue) and open-loop trials (red); average values are shown in gray with overlaid standard deviations. Locomotive speed greatly improves when robot body orientation is controlled.

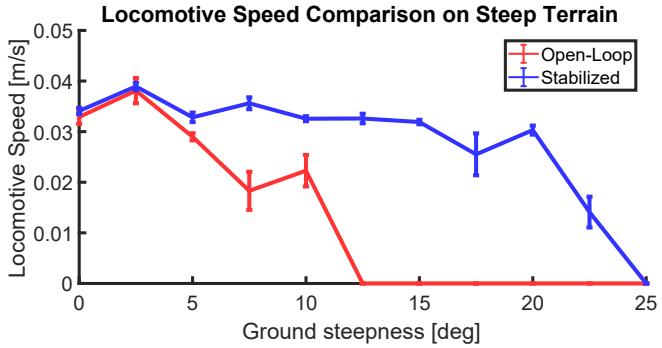


Fig. 7: Comparison of the average forward locomotive speed on flat terrains of different steepness, under both controllers (Stabilized vs. Open-Loop). Each data point is averaged from 5 experiments. Notice how the Open-loop controller's speed decreases as the steepness of the terrain increases, whereas the stabilized controller allows the robot to locomote at a constant speed until its breaking point, where the frictional force is no longer great enough to support locomotion (for this terrain, 25 degrees).

and determine its average forward speed, and thus overall locomotive performance.

Figs. 6 and 7 present our experimental results. Specifically, Fig. 6 documents the average speed and orientation error of the hexapod during each of the twelve trials on a twelve-degree, rocky incline. Six trials serve as the control group, lacking CPG-incorporated sensory feedback (red); another six employ the described inertial stabilization strategy that aims at levelling the robot's body (blue). Additionally, the mean speeds and standard deviations for both the control and stabilized groups are shown (gray). Fig. 7 compares the forward velocity of the robot in trials with and without CPG-incorporated inertial stabilization on planar slopes (no rocks) of varying constant steepness.

As previously indicated, choosing a limit cycle with $d = 4$ allows the end-effector to lift vertically (as opposed to an elliptic cycle which couples forward and vertical movement), limiting the chances of the robot's feet getting trapped in deep crevices within the terrain. Therefore, in unstructured



Fig. 8: Sequence of frames of the hexapod robot locomoting through various environments. Using our approach, the robot is able to climb steep grass (around 20 degrees), climb down a similarly-steep rocky slope, and climb up stairs. Full video: `https://goo.gl/awwYfT`

environments, we employ a limit cycle with $d = 4$ (Fig. 6); on planar terrains, we use a limit cycle with $d = 2$ (Fig. 7). Videos of the trials, as well as additional videos of stabilized climbing in outdoor environments (rock slides, stairs, and sloped grass) are available online at `https://goo.gl/w9JybR`. Fig. 8 presents a series of frames showing the progression of the robot through such environments.

V. Discussion

Inertial-based body control for levelling the robot's body improves locomotive performance in two ways. First, as shown by Fig. 6, for a given navigable terrain, locomotive speed greatly increases compared to that produced by an open-loop gait. Additionally, as shown by Fig. 7 inertial-based body control for stabilization enables locomotion on significantly more extreme slopes, which an open-loop locomotive model would be unable to navigate – either due to end-effector slippage or tipping.

It is important to point out the obvious trade-off between locomotion speed and stability; as speed increases, stability inherently diminishes. Locomotion speed is directly proportional to the angular speed parameter ω in the CPG Eq.(15). The speed at which significant degradation in stability occurs largely depends both on the roughness and integrity of the terrain. In the trials shown by Fig. 6, we prioritized stability over climbing speed (small ω).

Locomotive performance increases for several reasons when a body stabilization strategy is employed. First, levelling the body of the robot reduces load on the lateral, proximal joints that are responsible for propelling the body forward. Specifically, levelling the robot's body places the axes of rotation of these joints in parallel with the gravitational force, meaning that they need not overcome a component of the gravitational force. This allows the robot to maintain a relatively constant locomotive speed regardless of the slope or complexity of the terrain presented.

Second, levelling the body naturally positions the body's COG more centrally within the support polygon formed by grounded legs as shown in Fig. 9. In addition to preventing the robot from tipping over, this positioning of the COG distributes the weight of the body evenly between the legs. An even weight distribution ensures that the normal force exerted on the end-effector of each grounded leg remains significant, minimizing end-effector slip. End-effector slip is problematic for a variety of reasons, mainly because it introduces rapid changes in body position which can drastically disrupt stable locomotion. In particular, uneven

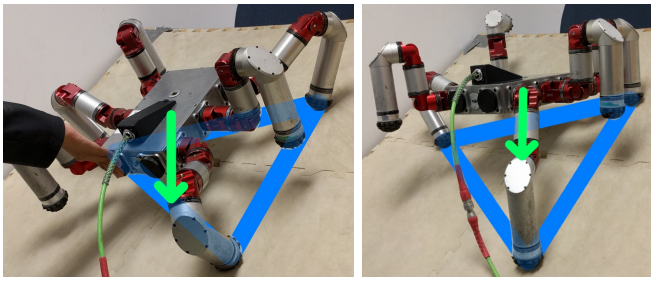


Fig. 9: Comparison of the robot's COG position relative to its support polygon in cases where the body is levelled via body control (right) and no body control is applied (left). Note how, when no body control is applied, the robot's COG is outside its support polygon and, thus preventing it from climbing by tipping backwards.

end-effector slip – when the end-effectors on one side of the robot have better ground contact – can reorient the robot. In this case, redirecting the robot can prove difficult, especially if slippage persists.

Third, this body orientation prevents the robot from tipping while stepping onto or off of an obstacle; with a body stabilization strategy, if the robot is able to step onto an object, it will be able to stabilize. This is especially important when the terrain is unstructured. While the slope may appear navigable, the local slope in cases where the terrain is unstructured can greatly exceed the general trend, due to crevices or holes as well as other vertical obstacles.

Finally, it is crucial to realize that with stabilization the hexapod traversed unstructured terrain of up to 30 degrees (locally sometimes much steeper) in field testing – greatly out-performing what is achievable with an open-loop control strategy. The trials presented in Fig. 7 provide a comparison between the open-loop and stabilized control strategies for a given planar surface, and in no way indicate the maximum slopes navigable. When using the open-loop controller, the robot's front end-effectors do not support weights great enough to produce a frictional force necessary for locomotion once the slope exceeds $12.5°$, while the robot even begins to tip at approximately $15°$. In contrast, controlling the body posture ensures that the body's weight remains well distributed across all the legs. On slopes greater than $20.0°$ slipping still occurs under the stabilized controller, given the end-effector and planar-surface materials used in the trials. However, provided that the requested body orientation can be achieved within the workspace of the legs, the robot will never tip when using the stabilized controller.

We believe that a planar surface provides a relatively-consistent test environment, which ensures a fair comparison

between the open-loop and stabilized trials. However, the shortcoming of a planar surface is that it lacks the footholds present in unstructured terrain that can eliminate end-effector slip. Therefore, even though the COG may remain well within the support polygon on an extreme slope, the frictional force between the end-effector and planar surface is often not sufficient for locomotion. Nevertheless, the results in Fig. 7 demonstrate that keeping the COG well within the support polygon is vital for navigating steep terrain.

VI. Conclusion

In this paper, we present a method for stable locomotion in unstructured terrain that integrates inertial feedback for body posture control directly within the CPG framework. We show that by extending the limit cycle of the CPG to the superellipse family, we can generate different limit cycle shapes solely by selecting a single parameter, to produce a variety of end-effector trajectories that are suitable for various environmental conditions. In this CPG model, we detail how to devise joint offsets from inertial feedback and highlight the beneficial adaptability of the resulting inertial-based CPG controller for stable locomotion and climbing in steep, unstructured terrain.

We implement our approach on a high-DOF hexapod robot. Experimental results demonstrate how body posture control improves locomotion by centering the robot's body inside its support polygon, and by smoothing out sudden perturbations (e.g., end-effector slips). In trials on steep, rocky slopes, we show how the proposed body posture control beneficially impacts locomotive speed and allows a robot to traverse steeper and more challenging terrain, where open-loop locomotion otherwise fails.

In order to improve locomotive ability, future work will explore the addition of an onboard vision system attached to the body, and the use of visual feedback directly into the CPG model. Doing so would allow some of the remaining CPG parameters (e.g., limit cycle dimensions) to be adapted directly through recognizing the size and orientation of obstacles in the path of the robot. Additionally, future work will present a comparison between the presented approach and other state-of-the-art methods for stabilized locomotion in complex terrains, such as precise foothold planning and reactive posture correction approaches.

ACKNOWLEDGMENTS

Samuel Shaw would like to thank the National Science Foundation for funding his work on this project as part of the 2017 Carnegie Mellon University Robotics Institute Summer Scholar (RISS) program. Detailed comments from anonymous referees strongly contributed to the presentation and quality of this paper.

References

[1] D. Belter, P. Łabęcki, and P. Skrzypczyński, "Map-based adaptive foothold planning for unstructured terrain walking," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 5256–5261.

[2] I. Roditis, T. Nitsos, A. Porichis, P. Chatzakos, G. Bertos, K. Lika, and E. Papadopoulos, "Maintaining static stability and continuous motion in rough terrain hexapod locomotion without terrain mapping," in *2016 24th Mediterranean Conference on Control and Automation (MED)*, June 2016, pp. 545–550.

[3] M. Bjelonic, N. Kottege, and P. Beckerle, "Proprioceptive control of an over-actuated hexapod robot in unstructured terrain," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 2042–2049.

[4] A. Roennau, G. Heppner, M. Nowicki, J. M. Zöllner, and R. Dillmann, "Reactive posture behaviors for stable legged locomotion over steep inclines and large obstacles," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 4888–4894.

[5] 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.

[6] S. Rossignol, R. Dubuc, and J.-P. Gossard, "Dynamic sensorimotor interactions in locomotion," *Physiological reviews*, vol. 86, no. 1, pp. 89–154, 2006.

[7] M. MacKay-Lyons, "Central pattern generation of locomotion: a review of the evidence," *Physical therapy*, vol. 82, no. 1, pp. 69–83, 2002.

[8] M. R. Dimitrijevic, Y. Gerasimenko, and M. M. Pinter, "Evidence for a spinal central pattern generator in humans," *Annals of the New York Academy of Sciences*, vol. 860, no. 1, pp. 360–376, 1998.

[9] A. Sproewitz, L. Kuechler, A. Tuleu, M. Ajallooeian, R. Möckel, A. Ijspeert *et al.*, "Oncilla robot: a light-weight bio-inspired quadruped robot for fast locomotion in rough terrain," in *5th International Symposium on Adaptive Motion of Animals and Machines*, no. EPFL-CONF-182313, 2011.

[10] M. Ajallooeian, S. Pouya, A. Sproewitz, and A. J. Ijspeert, "Central pattern generators augmented with virtual model control for quadruped rough terrain locomotion," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3321–3328.

[11] V. Barasuol, J. Buchli, C. Semini, M. Frigerio, E. R. De Pieri, and D. G. Caldwell, "A reactive controller framework for quadrupedal locomotion on challenging terrain," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 2554–2561.

[12] K. Seo, S.-J. Chung, and J.-J. E. Slotine, "Cpg-based control of a turtle-like underwater vehicle," *Autonomous Robots*, vol. 28, no. 3, pp. 247–269, 2010.

[13] L. Righetti and A. J. Ijspeert, "Pattern generators with sensory feedback for the control of quadruped locomotion," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*. IEEE, 2008, pp. 819–824.

[14] H. Kimura, Y. Fukuoka, and A. H. Cohen, "Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts," *The International Journal of Robotics Research*, vol. 26, no. 5, pp. 475–490, 2007.

[15] A. J. Ijspeert, "Central pattern generators for locomotion control in animals and robots: a review," *Neural networks*, vol. 21, no. 4, pp. 642–653, 2008.

[16] G. Loria, "Spezielle algebraische und transzendente ebene kurven: Theorie und geschichte. bd. 1, die algebraischen kurven," *BG Teubners Sammlung von Lehrbüchern auf dem Gebiete der Mathematischen Wissenschaften mit Einschluss ihrer Anwendungen*, 1910.

[17] N. T. Gridgeman, "Lamé ovals," *The Mathematical Gazette*, vol. 54, no. 387, pp. 31–37, 1970.

[18] G. Sartoretti, M.-O. Hongler, M. Elias de Oliveira, and F. Mondada, "Decentralized self-selection of swarm trajectories: From dynamical system theory to robotic implementation," *Swarm Intelligence*, vol. vol. 8, no. no. 4, pp. 329–351, 2014.

[19] M. J. Travers, J. Whitman, P. Schiebel, D. I. Goldman, and H. Choset, "Shape-based compliance in locomotion." in *Robotics: Science and Systems*, 2016.

[20] D. Rollinson, S. Ford, B. Brown, and H. Choset, "Design and modeling of a series elastic element for snake robots," in *Proceedings of ASME Dynamic Systems and Control Conference (DSCC)*, 2013, pp. V001T08A002–V001T08A002.

[21] P. Ramdya, R. Thandiackal, R. Cherney, T. Asselborn, R. Benton, A. J. Ijspeert, and D. Floreano, "Climbing favours the tripod gait over alternative faster insect gaits," *Nature communications*, vol. 8, 2017.