Proceedings of the 2011 IEEE
International Conference on Robotics and Biomimetics
December 7-11, 2011, Phuket, Thailand

Towards a Biologically Inspired Open Loop Controller for Dynamic Biped Locomotion

Jonathan Spitz, Yizhar Or and Miriam Zacksenhouse

Abstract— **Biped robots are usually controlled using a quasi-static approach. In contrast, dynamic walking robots display more natural gaits which make them faster and more energy efficient. Applying these concepts, a biomimetic open-loop controller based on a Central Pattern Generator (CPG) is designed and applied to a Compass-Gait walker. We use numerical simulations to investigate the resulting gaits and verify stability. The robustness of the system is tested by applying disturbances to the surface slope. The stability of the obtained limit cycles is demonstrated through the system's Poincaré Map.**

I. INTRODUCTION

Central Pattern Generators (CPGs) have been shown to control several types of rhythmic tasks in nature [11]. A wide range of activities such as walking, swimming and chewing have been shown to rely on an underlying CPG. In quadrupeds, for example, different locomotion gaits are generated by CPGs located in their spinal cord. These CPGs are composed of several coupled neurons which coordinate the activation of the different muscles in all legs [2], [15]. A variety of gaits can be generated using different coupling weights between said neurons, e.g., walk, trot, gallop, etc. Several biomimetic robots can be found in literature that use these controllers to generate locomotion: hexapods [16], quadrupeds [3], [12] and bipeds [1], [10], [18]. CPGs have also been successfully used to control a Yo-Yo playing robot [6], [7].

CPGs are specially suited to control walking bipeds since they take advantage of the system's inherent dynamic properties. Biped robots are usually designed and controlled using a quasi-static approach, moving the body's center of mass from one leg to the other before taking the next step, i.e., the robot moves from one equilibrium posture to the other. This usually results in a slow, quirky walk. In contrast, a dynamic walking robot never reaches static equilibrium but constantly prevents falling down by taking another step. Dynamic walking gaits take advantage of the robot's inertia to keep the robot moving forward, usually much faster than a quasi-static gait. Dynamic gaits are also very efficient energetically, making them perfect candidates for autonomous robots [4].

Biped models have been studied with great detail, especially the gait of Passive Dynamic Walkers [4], [13]. These systems display stable gaits when walking down slopes, without any actuation. In recent years there has been a growing interest in the field, generating several works concerning the control of passive dynamics walker with actuators [1], [5], [10], [18]. The addition of actuators allows the robot to walk also on a level plane. One of the simplest biped walkers is the Compass-Gait Biped, composed of two rigid legs and a hip mass (Cf. Fig. 1).

The primary purpose of this paper is to implement a CPG based controller and demonstrate its ability to generate stable walking gaits when coupled to a Compass-Gait Biped. The CPG generates periodic signals that activate the robot's muscles, effectively generating torque on each leg. We focused on strongly coupled neural oscillators as the basic components of our CPG [7], [14].

The Compass-Gait model and the controller are described in section II. Numerical analysis to demonstrate the gait stability is presented in section III. The results of our numerical simulations are presented in section IV. We conclude in section V.

II. MODELLING AND CONTROL CONFIGURATION

A. Compass-Gait Biped Model

Several different approaches to modelling the dynamics of a planar compass biped can be found in literature [8], [9]. One of the simplest is the 2-DOF model comprised of two legs joined at the hip [4] shown in Fig. 1. Each leg is rigid, has a mass m, moment of inertia I and length L. Both legs are attached to a hip mass m_h and allowed to rotate around it. This model has several variations of its own, e.g.: choice of system's coordinates, different distribution of mass on the legs or consideration of the ground slope (affecting the system's coordinates). At any given moment, only one leg is in contact with the ground, except when impact occurs. Thus, the robot's step cycle can be divided into two phases: a continuous single-support phase and a discrete double-support transition phase.

B. Continuous Phase

During the single support phase the stance leg is assumed to be attached to the ground through a rotation joint, i.e., the ground's normal force is always positive and there is no slip. This assumption is confirmed in our numerical simulations. Hence, the robot's dynamics are modelled by 2-DOF equations:

$$M(q)\ddot{q} + N(q,\dot{q})\dot{q} + G(q) = Eu \qquad q = [\theta_1, \theta_2[^T \quad (1)$$

This work was supported by the Arlene & Arnold Goldstein Center at the Technion's Autonomous Systems Program, under grant number 2012506.

Jonathan Spitz and Miriam Zacksenhouse are with the Sensory-Motor Integration Laboratory, Faculty of Mechanical Engineering, Technion Israel's Institute of Technology, Haifa 32000, Israel (e-mail: spitz@technion.ac.il, mermz@technion.ac.il)

Yizhar Or is with the Laboratory of Bio Dynamics and Mechanics of Locomotion, Faculty of Mechanical Engineering, Technion Israel's Institute of Technology, Haifa 32000, Israel (e-mail: izi@technion.ac.il)

978-1-4577-2138-0/11/$26.00 © 2011 IEEE

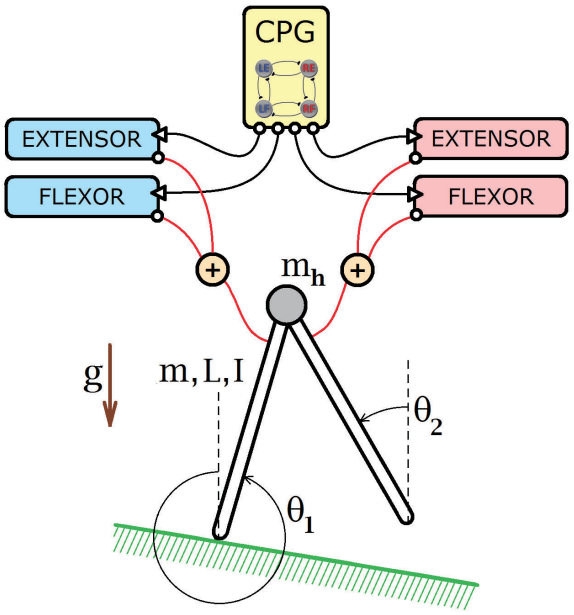

Fig. 1. Compass-gait biped model and control scheme. The model is composed of two rigid legs of mass m, length L and moment of inertia I and a hip mass m_h. The CPG generates command signals at specific intervals. Those signals activate the different muscles (LE,LF,RE,RF), applying a torque of predetermined strength and duration.

where θ_1 is the stance leg's orientation with respect to the vertical and θ_2 is the swing leg's orientation with respect to the vertical, as seen in Fig. 1. The matrices are given by

$$M(q) = \begin{bmatrix} \frac{5}{4}mL^2 + m_hL^2 + I & -\frac{mL^2}{2}c_{12} \\ -\frac{mL^2}{2}c_{12} & \frac{1}{4}mL^2 + I \end{bmatrix} \quad (2)$$

$$N(q,\dot{q}) = \begin{bmatrix} d_1 & -\frac{mL^2}{2}s_{12}\dot{\theta}_2 \\ -\frac{mL^2}{2}s_{12}\dot{\theta}_1 & d_2 \end{bmatrix} \quad (3)$$

$$G(q) = \begin{bmatrix} -\left(\frac{3}{2}m + m_h\right)Lg\sin(\theta_1) \\ \frac{1}{2}mLg\sin(\theta_2) \end{bmatrix} \quad (4)$$

where $s_{12} = \sin(\theta_1 - \theta_2)$, $c_{12} = \cos(\theta_1 - \theta_2)$ and d_1, d_2 are damping coefficients for the stance and swing leg, respectively.

Every biped model possessing rigid legs encounters the same problem of the swing leg "scraping" the ground in mid-swing. Several modelling and physical solutions to this problem can be found in literature, e.g. shortening the leg during swing or ignoring the impact when the angle between both legs is small. For our research we slightly shorten the robot's swing leg at predefined times specified by the controller.

C. Discrete Transition Phase

The transition of support between both legs occurs when the swing leg impacts the ground, i.e., when

$$y_{swing} = Surf\left(x_{swing}\right) \quad (5)$$

where $Surf(x)$ is the function that describes the terrain. Upon impact the swing leg becomes attached to the ground, i.e., it does not bounce back and there is no slip. The relation between the stance foot and swing foot position is given by:

$$\begin{cases} x_{swing} = x_{stance} - L\sin(\theta_1) + L\sin(\theta_2) \\ y_{swing} = y_{stance} + L\cos(\theta_1) - L\cos(\theta_2) \end{cases} \quad (6)$$

The transition equations can be obtained by applying the conservation of angular momentum law about the new stance foot for the whole system and about the hip for the new swing leg [4]. These equations can be written in matrix form:

$$Q\left(q^+\right)\dot{q}^+ = P\left(q^-\right)\dot{q}^- \quad (7)$$

where $\dot{q}^+$ and $\dot{q}^-$ are the system's angular velocities after impact and before impact, respectively. Impact is assumed to occur instantaneously, thus q remains the same right before and after impact. The transition equation is then:

$$\dot{q}^+ = H\left(q^-\right)\dot{q}^- = Q^{-1}\left(q^-\right)P\left(q^-\right)\dot{q}^- \quad (8)$$

where:

$$P(q^-) = \begin{bmatrix} 1 - \frac{4I}{mL^2} & 0 \\ -\left(3 + 4\frac{m_h}{m}\right)c_{12} - \frac{4I}{mL^2} & 1 - \frac{4I}{mL^2} \end{bmatrix} \quad (9)$$

$$Q(q^-) = \begin{bmatrix} 2c_{12} & -1 - \frac{4I}{mL^2} \\ 2c_{12} - 5 - 4\frac{m_h}{m} - \frac{4I}{mL^2} & 2c_{12} - 1 - \frac{4I}{mL^2} \end{bmatrix} \quad (10)$$

D. Central Pattern Generator configuration

The controller presented here generates locomotion patterns by using coupled neurons. This kind of controllers can be seen in many places in nature, e.g. the swimming motion of the Lamprey fish has been shown to arise from the coupling of neurons in its spinal cord [15].

The CPG used for this Biped Walker utilizes four pulse coupled neurons - a pair for each leg, to generate locomotion patterns. Each pair of neurons activates antagonist muscles (flexor/extensor) of one leg. Different patterns or gaits arise from different coupling weights between the neurons. For example, if the diagonal weights are set to zero (Cf. Fig. 2) and the remaining weights are all equal, then the diagonal neurons will fire in phase and adjacent neurons will fire 180 degrees out of phase (Cf. Fig. 3). This ability makes the CPG very versatile, allowing the same CPG to generate different gaits by changing the coupling weights.

The open loop controller described here generates a stable walking pattern when applied to the 2-DOF compass biped model. The robot exhibits a dynamic walking gait on a plane and is able to maintain orbital stability even for small slopes (See Section IV - Results).

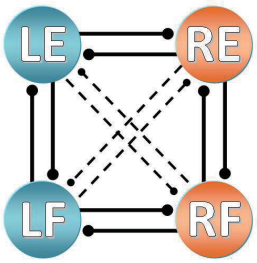

Fig. 2. Central Pattern Generator composed of four coupled neurons. Each neuron controls the activation of a certain muscle: LE Left extensor, LF Left flexor, RE Right extensor, RF Right flexor. Here a general coupling scheme is shown.

E. Oscillatory units

The CPG's neurons are modelled as simple Leaky Integrate and Fire (LIF) oscillators. The dynamics for each neuron are given by a first order ODE. Each neuron is connected to its neighbours via pulse coupling. The strength of each coupling varies from neighbour to neighbour. The change in membrane potential due to pulse coupling is assumed to be instantaneous. The dynamics for the whole controller is a set of four differential equations:

$$\tau_m \dot{V}_i = V_{SS} - V_i + \epsilon \sum_{n=1}^{4} W_{in} \delta\left(t - t_n\right) \tag{11}$$

where V_i is the i'th neuron membrane potential, V_{SS} is the steady state potential that the membrane is trying to reach, τ_m is the membrane's time constant, ϵ is the base coupling strength, W is the coupling weight matrix (with zero diagonal) and t_n is the time of activation of the n'th neuron.

The solution to the set of equations in (11) is given by:

$$V_i(t) = V_{SS} - \Delta V e^{-\frac{t-t_i}{\tau_m}} + \epsilon \sum_{n=1}^{4} W_{in} U(t-t_n)\, U(t_n - t_i)\, e^{-\frac{t-t_n}{\tau_m}} \tag{12}$$

where $\Delta V = V_{SS} - V_{reset}$.

When the membrane potential reaches $V_{threshold}$, the neuron fires and its potential is reset to V_{reset}. The effect of the coupling is also reset after the neuron fires so that coupling terms are only added for $t \geq t_n \geq t_i$, as implied by the combination of step functions U. The CPG's period can be expressed as a function of ΔV, τ_m, ϵ and W.

The firing pattern of the neurons is shown in Fig. 3.

F. Controller Output

The oscillating neurons composing the CPG generate periodic activation pulses at defined intervals. These pulses initiate a predefined muscle activation pattern, i.e., whenever a neuron fires the corresponding extensor or flexor muscle is activated. The muscle activation pattern is defined by the duration and amplitude of the torque applied. Specifically, the activation pattern for the extensor muscles is a short, strong pulse that propels the swing leg forward whereas the activation pattern for the flexor muscles is a long, weaker pulse that pushes the robot forward, as seen in Fig. 4. This activation pattern was purposely designed to take advantage of the inertial dynamics of the robot by applying an initial torque to the swing leg and allowing it to swing freely thereafter.

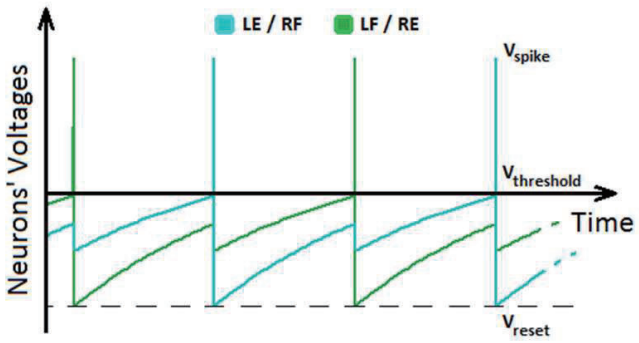

Fig. 3. Neurons firing pattern arising from the CPG. This specific coupling scheme has no diagonal weights (Cf. Fig. 2), resulting in opposite neurons firing together.

III. SIMULATIONS AND NUMERICAL ANALYSIS

We developed several numerical simulations in order to test the designed controller. The simulations were run on MATLAB.

A. Simulation setup

The continuous phase, transition phase and controller equations are numerically integrated to simulate the robot's motion. The continuous equations are used to simulate the swing phase up until the leg hits the ground. Upon impact, the simulation is stopped and the impact (transition) equations are used to obtain the initial conditions for the next step-cycle. The numerical experiment runs until: (a) the robot falls or (b) the initial conditions at the beginning of each step converge to steady state.

Three numerical experiments were performed using different terrains: a flat surface, a terrain with unbounded increasing slope and finally, a terrain with bounded increasing slope. The purpose of the first experiment was to obtain a set of

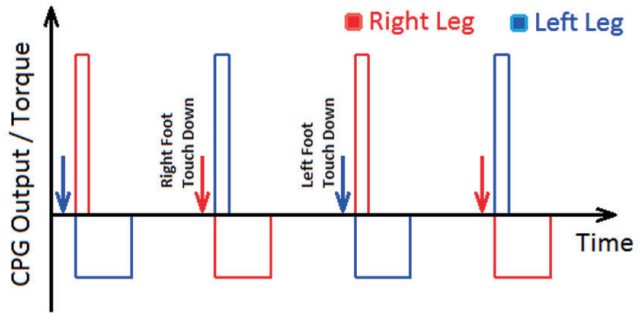

Fig. 4. Muscle activation pattern arising from the CPG. Right after foot touchdown, the stance leg's flexor muscle is activated together with the swing leg's extensor muscle.

initial conditions, robot parameters and controller parameters that generate a stable walking gait on the horizontal plane.

For our second experiment, the terrain starts flat and then the slope is gradually increased (either uphill or downhill). The rate of change in slope is constant, so the surface can be seen as a parabola. The purpose of this numerical analysis was to test the robustness of the implemented controller. This experiment allowed us to test robots with different control parameters until failure, i.e. until the slope was steep enough to make the walking orbits unstable.

A third simulation was then built to demonstrate the stability of the walking gaits seen in our second experiment. The terrain was modified to gradually increase its slope, as in the previous experiment, but only up to a set maximum slope. The slope is then kept constant. The simulation runs until the initial conditions at the beginning of each step reach steady state. The results from this simulation are used to generate the system's Poincaré Map.

B. Poincaré Map

To describe the dynamics of both the robot and the CPG, we define a fifth-dimensional state-space:

$$X(t) = \begin{bmatrix} \theta_1 & \dot{\theta}_1 & \theta_2 & \dot{\theta}_2 & \varphi \end{bmatrix}^T \tag{13}$$

where φ is the CPG's phase.

In order to demonstrate the stability of the limit-cycles obtained through numerical simulations we generate the Poincaré Map [17]. We define the Poincaré Section S, as the system's state right after impact. The Poincaré Map is the return map of an orbit in the state-space from one intersection with S to the next. This is noted as:

$$X^+_{k+1} = P\left(X^+_k\right) \tag{14}$$

where X^+ is the state right after impact.

A geometric condition exists between the angles of both legs right after impact. Thus, there are four parameters that define a point on the Poincaré Section S: the aperture between the legs, their angular velocities and the CPG's phase. A fixed point on S satisfies:

$$X^* = P\left(X^*\right) \tag{15}$$

We assume that the steady state initial conditions obtained in the third experiment for different slopes are fixed points in the Poincaré Map. In order to verify these assumptions, small perturbations are made to the initial conditions and then the system is numerically integrated until the next return to S. The result is noted as:

$$\begin{aligned} X^* + v_1 &= P(X^* + v_0) \\ &= P(X^*) + DP(X^*)\, v_0 + O\left(|v_0|^2\right) \end{aligned} \tag{16}$$

where $DP(X)$ is the linearized Poincaré Map matrix, also called the Jacobian Matrix of the Poincaré Map.

We specifically perturb the system along its coordinates' directions, i.e. $v_0^{(k)} = \epsilon e_k$ is a perturbation in the k'th coordinate direction. Then, the resulting linearized first return map is [17]:

$$DP(X^*) = \epsilon^{-1} \begin{bmatrix} v_1^{(1)} & v_1^{(2)} & \cdots & v_1^{(k)} \end{bmatrix} \tag{17}$$

The walking gaits are stable if all the eigenvalues of the Jacobian matrix $DP(X)$ are inside the unit circle in the complex plane, i.e. all the eigenvalues have a magnitude smaller than 1.

IV. RESULTS

In this section we present the results from our numerical analysis. The limit cycle obtained in Numerical Experiment 1 is shown in Fig. 6. Then, the results from Numerical Experiment 2 are illustrated using stick-figure diagrams (Cf. Fig. 7). The robot is shown at intervals of 1 second while walking (a) uphill or (b) downhill. For each experiment, the maximal slope shown represents the terrain's slope at the last step that the robot took before falling. It should be noted that this is not necessarily the same slope where the walking gait became unstable.

It can be clearly seen that the robot was much better at walking downhill than uphill using the set controller. This is due to the fact that when walking downhill, potential

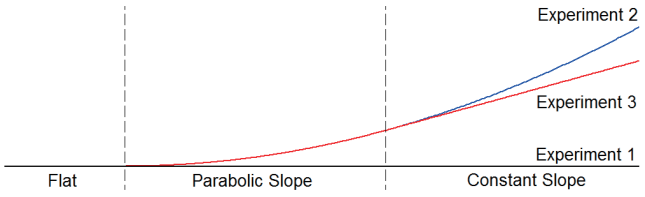

Fig. 5. Experimental Walking Surface. Experiment 1: a flat surface was used. Experiment 2: the surface starts flat and then the slope is gradually increased until the robot loses orbital stability. Experiment 3: starts with a flat surface, then the slope is gradually increased up to a set maximal slope after which the slope remains constant.

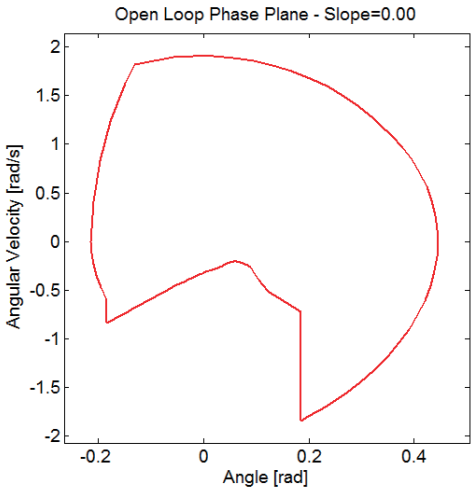

Fig. 6. Limit-cycle of the open-loop controlled robot walking on a flat surface. The orbit shown represents one step cycle for the right leg (swing phase + stance phase)

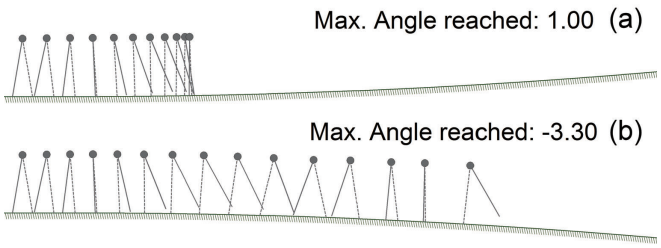

Fig. 7. Stick-figure diagrams showing the results of Experiment 2, which tested the system's robustness when walking on inclined surfaces, either (a) uphill or (b) downhill. The robot is drawn every 1 second and the maximum angles shown represent the surface's slope at the point where the robot failed.

energy is transformed into kinetic energy whereas when walking uphill the opposite occurs. In addition, the step length increases as the downhill slope becomes steeper and decreases almost to zero as the uphill slope becomes steeper. Numerical Experiment 2 demonstrated that our proposed controller is able to walk not only on a flat surface but on a range of slopes around zero.

The steady state initial conditions of stable orbits, at the time of intersection with the Poincaré Section S are shown in Fig. 8 for different slopes. A few examples of the limit-cycles obtained in Numerical Experiment 3 are shown in Fig. 9. Upon reaching a certain slope, the symmetrical gait is lost via a bifurcation and we obtain a period-doubling walking gait. A second bifurcation occurs at an even steeper slope. This can be seen in Fig. 9(a) and 9(b).

The results from Numerical Experiment 3 show that the gaits generated by our controller are stable within the given range of slopes, since all eigenvalues of the Poincaré Map lie within the unit disc (Cf. Fig. 10). We have also demonstrated that our controller is able to produce stable gaits not only on a flat surface but also on a small set of slopes.

V. CONCLUSIONS

The main objective of the presented research was to design a bio-inspired controller that generates a stable walking gait. The proposed controller, based on a CPG, produces rhythmic signals that drive the muscles of the Compass-Gait model used. The muscles' activation is synchronized by the CPG whereas the strength and duration of activation is defined separately. Through numerical simulations we have shown that the robot is able to walk uphill as well as downhill. We showed that the gaits generated by the robot for slopes between $-0.9°$ and $+0.7°$ are stable.

It should be noted that the CPG based controller is extremely flexible, e.g. before arriving at the configuration shown here, a stable walking gait was achieved with a different activation pattern (by changing the coupling weight matrix).

We studied the effect of different slopes on the robot's dynamics. In future work we intend to analyze how modifying the CPG's intrinsic period, the strength or duration of muscle activation affect the robot's gait: step length, walking speed and robustness.

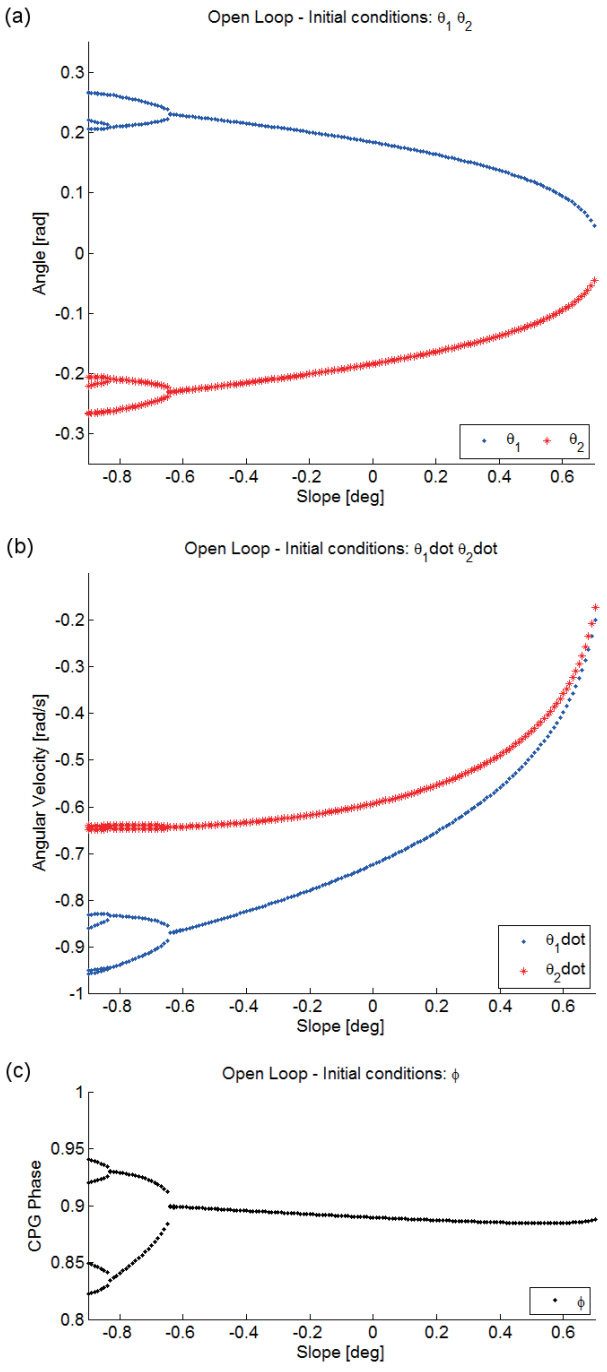

Fig. 8. Steady state initial conditions of stable orbits, at the time of intersection with the Poincaré Section are shown as functions of the terrain's slope.

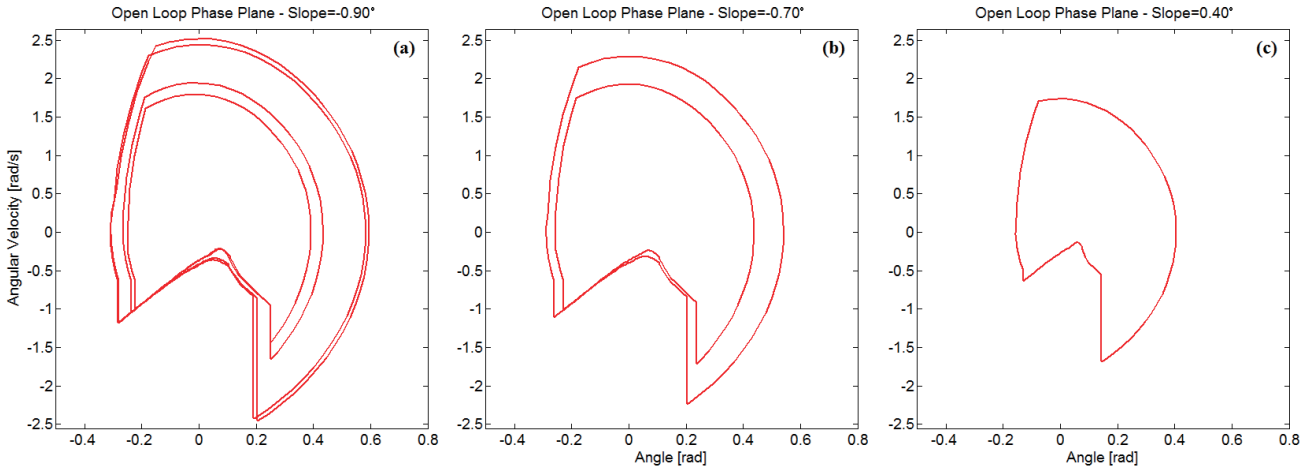

Fig. 9. Limit-cycles obtained in Experiment 3 for different slopes: When walking uphill, the limit-cycle shrinks as the slope becomes steeper (c). A period-doubling bifurcation occurs for downhill slopes above $-0.47°$ (b). A second period-doubling bifurcation appears above $-0.82°$ (a).

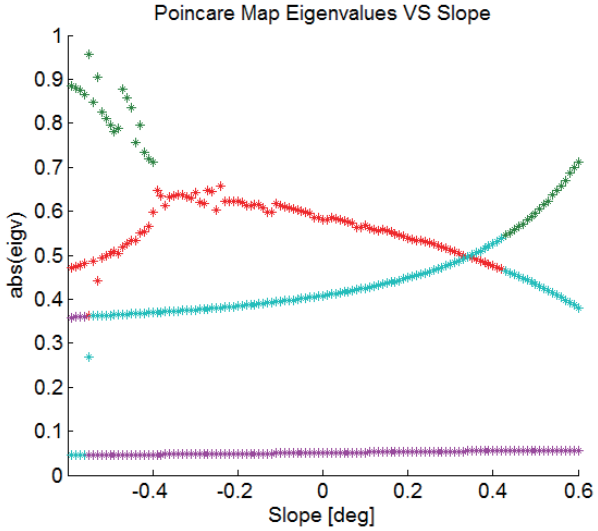

Fig. 10. Absolute value of the Poincaré Map eigenvalues. The system is shown to be stable within the given range of slopes since all eigenvalues lie within the unit disc

VI. ACKNOWLEDGMENTS

The work presented here has been supported by the Arlene & Arnold Goldstein Center at the Technion's Autonomous Systems Program, under grant number 2012506.

REFERENCES

[1] S. Aoi and K. Tsuchiya, "Stability analysis of a simple walking model driven by an oscillator with a phase reset using sensory feedback", *IEEE Transactions on Robotics*, 22(2), pp. 391-397, April 2006.

[2] F. Delcomyn, "Walking robots and the central peripheral control of locomotion in insects", *Autonomous Robots*, pp. 259-270, November 1999.

[3] Y. Fukuoka, H. Kimura and A. H. Cohen, "Adaptive dynamic walking of a quadruped robot on irregular terrain based on biological concepts", *The International Journal of Robotics Research*, 22(3-4), pp. 187-202, March-April 2003.

[4] A. Goswami, B. Thuilot and B. Espiau, "A study of the passive gait of a compass-like biped robot: symmetry and chaos", *The International Journal of Robotics Research*, 17(12), pp. 1282-1301, December 1998.

[5] A. Goswami, B. Espiau and A. Keramane, "Limit cycles in a passive compass gait biped and passivity-mimicking control laws", *Autonomous Robots*, Vol. 4, pp. 273-286, July 1997.

[6] J. Hui-Liang and M. Zacksenhouse, "Oscillatory neural networks for robotic yo-yo control", *IEEE Transactions on Neural Networks*, 14(2), pp. 317-325, March 2003.

[7] J. Hui-Liang and M. Zacksenhouse, "Necessary condition for simple oscillatory neural control of robotic yoyo", *in Proc. IJCNN '02*, Vol. 2, pp. 1427-1432, 2002.

[8] Y. Hurmuzlu, F. Génot and B. Brogliato, "Modeling, stability and control of biped robots - a general framework", *Automatica*, 40(10), pp. 1647-1664, October 2004.

[9] Y. Hurmuzlu and D. B. Marghitu, "Rigid body collisions of planar kinematic chains with multiple contact points", *The International Journal of Robotics Research*, 13(1), pp. 82-92, February 1994.

[10] F. Iida and R. Tedrake, "Minimalistic control of biped walking in rough terrain", *Autonomous Robots*, to be published.

[11] A. J. Ijspeert, "Central pattern generators for locomotion control in animals and robots: A review", *Neural Networks*, 21(4), pp. 642-653, May 2008.

[12] H. Kimura, Y. Fukuoka and A. H. Cohen, "Adaptive dynamic walking of a quadruped robot on natural ground based on biological concepts", *International Journal of Robotics Research*, 26(5), pp. 475-490, May 2007.

[13] T. McGeer, "Passive dynamic walking", *The International Journal of Robotics Research*, 9(2), pp. 62-82, April 1990.

[14] R. E. Mirollo and S. H. Strogratz, "Synchronization of pulse-coupled biological oscillators", *SIAM Journal on Applied Mathematics*, 50(6), pp. 1645-1662, December 1990.

[15] R. H. Rand, A. H. Cohen and P. J. Holmes, "Systems of coupled oscillators as models of central pattern generators", *in Neural Control of Rhythmic Movements in Vertebrates*, John Wiley & Sons, pp. 333-367, 1988.

[16] U. Saranli, M. Buehler and D. E. Koditschek, "RHex A simple and highly mobile hexapod robot", *International Journal of Robotics Research*, 20(7), pp. 616-631, July 2001.

[17] S. H. Strogatz, *Nonlinear dynamics and chaos: with applications to physics, biology, chemistry and engineering*, Reading, Massachusetts: Perseus Books, 1994, pp. 278-284.

[18] G. Taga, Y. Yamaguchi and H. Shimizu, "Self-organized control of bipedal locomotion by neural oscillators in unpredictable environment", *Biological Cybernetics*, 65(3), pp. 147-159, July 1991.