

# An in silico central pattern generator: silicon oscillator, coupling, entrainment, and physical computation

M. Anthony Lewis<sup>1</sup>, Ralph Etienne-Cummings<sup>2</sup>, Mitra J. Hartmann<sup>3</sup>, Zi Rong Xu<sup>5</sup>, Avis H. Cohen<sup>4</sup>

<sup>1</sup>Iguana Robotics, Inc., P.O. Box 628, Mahomet, IL 61853, USA (e-mail: tlewis@iguana-robotics.com)

<sup>2</sup>Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218, USA  
(e-mail: etienne@ece.jhu.edu)

<sup>3</sup>Jet Propulsion Laboratory, 303-300, California Institute of Technology, 4800 Oak Grove Drive, Pasadena, CA 91109, USA  
(e-mail: hartmann@brain.jpl.nasa.gov)

<sup>4</sup>Department of Biology, Neuroscience and Cognitive Science, University of Maryland, College Park MD 20742, USA  
(e-mail: ac61@umail.umd.edu)

<sup>5</sup>Department of Electrical and Computer Engineering, Johns Hopkins University, Baltimore, MD 21218, USA

Received: 31 October 2001 / Accepted in revised form: 17 September 2002

**Abstract.** In biological systems, the task of computing a gait trajectory is shared between the biomechanical and nervous systems. We take the perspective that both of these seemingly different computations are examples of physical computation. Here we describe the progress that has been made toward building a minimal biped system that illustrates this idea. We embed a significant portion of the computation in physical devices, such as capacitors and transistors, to underline the potential power of emphasizing the understanding of physical computation. We describe results in the exploitation of physical computation by (1) using a passive knee to assist in dynamics computation, (2) using an oscillator to drive a monopod mechanism based on the passive knee, (3) using sensory entrainment to coordinate the mechanics with the neural oscillator, (4) coupling two such systems together mechanically at the hip and computationally via the resulting two oscillators to create a biped mechanism, and (5) demonstrating the resulting gait generation in the biped mechanism.

## 1 Introduction

Locomotion is a fundamental activity in both animals and robots. Recent work in biology and robotics has clearly shown that the computation of the gait trajectory during locomotion is shared between the biomechanical and nervous systems (Taga et al. 1991; Taga 1995a,b; Pratt and Pratt 1998; Kimura et al. 1999; Pratt and Pratt 1999; Dickinson et al. 2000; Kimura et al. 2001a,b). In this paper, we demonstrate a silicon-circuit embodiment of a central pattern generator (CPG) locomotor controller for a robotic monopod and biped. In both monopodal and bipedal systems, the passive dynamics contribute substantially to the computation of the

overall leg trajectories. By explicitly sharing the computation of gait trajectory between a CPG, instantiated in silicon, and the mechanics, instantiated in robotic hardware, we have created a robot with a very low-complexity control system.

When fully developed, this chip could be embedded in a robot such as Tekken (Kimura et al. 2001a,b) to create a remarkably elegant and compact robot control system. This technology may also have alternative uses in the rehabilitation of patients with various neurological deficits, since the circuit could be adapted for correction, augmentation, and generation of locomotor patterns in such patients.

### 1.1 Background

Locomotion in animals is strongly periodic. The animal's nervous system modulates its locomotor behavior in response to altered sensory input and the volition of the animal. It is generally agreed that the neural architecture necessary to support the control of locomotion is structured as follows: (1) Neural circuits in the spinal cord called the CPG are capable of generating the neural oscillations necessary for locomotion. (2) The CPG interacts strongly with sensory feedback from the muscles and joints. (3) Descending modulatory influences from the brain produce adaptive locomotor movements based on volition and exteroceptive senses, e.g., vision (Grillner 1981; Armstrong 1986; Cohen and Boothe 1999).

The CPG is composed of intrinsic oscillators embedded in a distributed system of circuits (Grillner 1981). These circuits enforce constraints among themselves, which gives rise to phenomena such as patterns of phase relationships between oscillators. This allows flexibility in generating the necessary phase relationship for a variety of gaits. This is evident in quadrupeds that can walk, trot, canter, gallop, pronk, and pace, presumably by modifying the connections between the intrinsic oscillators to generate the required movement

patterns. A computational model of biological nervous systems must have a similarly distributed architecture.

The mechanics also “compute” a great deal of the walking movement, as has been shown by Ruina and colleagues in their work on passive walking bipeds (Garcia et al. 1998a,b; Collins et al. 2001) and by Pratt and Pratt, in the use of lower-leg-segment swing in an actively controlled robot (Pratt and Pratt 1998, 1999). By physical computation, researchers understand that the physics of devices (whether Newtonian mechanics or device physics of circuits) can be a substrate for computation. Exploiting the physics of computation may lead to vastly more efficient designs. This is a common thread in the work presented here as well.

The contribution of this current work is to demonstrate the minimal system that satisfies the principles above (i.e., distributed CPG, computation using passive dynamics) that can control a bipedal walking mechanism. The work gives compelling support for the hypothesis that systems designed based on such principles can be realized using minimal computational resources.

This study does not address balance and postural control. To our knowledge, integration of the CPG with balance and postural control in a real bipedal machine has not been addressed. We plan to reserve this topic for future work, but we are confident that it will be compatible with the current approach.

### 1.2 Comparison to previous CPG chip work

CPG chips and circuits have been created before. For example, Still reports on a VLSI (very large scale integration) implementation pattern generator used to drive a small robot (Still 2000; Still et al. 2000). This circuit captured some of the basic ideas of a CPG, but it did not incorporate a motor neuron output stage, and the system did not provide for adaptation via sensory input. However, the work did demonstrate control of a rudimentary walking machine.

The work of DeWeerth and colleagues (Patel et al. 1998) captures the neural dynamics on a much more detailed level than has been achieved here. However, there are great difficulties in applying such a system to the control of a robot. Primarily, parameter sensitivity makes such circuits difficult to tune. To address this issue, DeWeerth and collaborators have implemented neurons that self-adapt their firing rate (Simoni and DeWeerth 1998). The adaptation, however, is independent of external inputs from sensors. While detailed neural models are difficult to work with in silicon, we will undoubtedly learn a great deal from these efforts in the future.

Ryckebusch and colleagues (1994) created a VLSI CPG chip based on observations in the thoracic circuits controlling locomotion in locusts. The resulting VLSI chip was used as a fast simulation tool to explore understanding of the biological system. Their system did not use feedback from sensors, nor was it connected to a robotic system. However, their objective – of modeling a particular biological circuit – was different than the objective described in this paper.

Our work differs from the previous work in several respects. First, we allow adaptation based on sensory input. Adaptation is shown as a phase resetting of the CPG based on certain sensory triggers. Second, we make use of integrate-and-fire neurons for the output motor neurons. Our abstraction is at a higher level than other reported work (Patel et al. 1998; Simoni and DeWeerth 1998). We believe that by using a higher level of abstraction we will be able to more easily implement on-chip learning. In systems based on numerous inter-related parameters, it is not apparent how learning at the level of behavior can be coupled to low-level-parameter changes.

### 1.3 Current approach

The long-range objective of the current work is to develop a CPG chip that can control a biped or quadruped robot. This chip would adapt based on sensory feedback as well as on input from higher centers. Physical computation – both in the mechanics and in the control system – will be maximally exploited. In the initial work, described here, we develop a minimal system that incorporates the major principles of biped locomotion, with the exception of the incorporation of higher-level input. The key components we have identified are:

- (1) Creating a monopod where the lower limb segment “computes” a significant portion of the gait trajectory. Below we will begin by analyzing the dynamics of the lower-limb segment and understanding the dynamics of this system through mathematical analysis.
- (2) Driving this monopod with a single oscillator.
- (3) Using entrainment to close the loop between the mechanical system and the neural oscillator. We give results from an analysis of the role of sensory feedback in entraining the oscillator and we use geometric methods to visualize this effect. We also show the results of “lesion” experiments in which we demonstrate the effect of reversibly removing sensory feedback to the oscillator. We can now use this single leg to build a biped.
- (4) Together the two oscillators for each half monopod. Coupling of oscillators is a nontrivial problem. First, we give mathematical results of coupling using pulse trains and determine that for our system multispike coupling gives superior performance over single-spike coupling. This was a major result of the analysis. Now we demonstrate that the effect can be seen in the geometry of the map function describing the convergence of the two oscillators. Finally, we perform experiments on the chip to empirically determine the map function of the chip. We find that the model gives an excellent prediction of the chip behavior.
- (5) Demonstrating the complete biped. Here we record the overall gait trajectories to show that the gait is smooth and resembles a natural gait.

## 2 Experimental apparatus

### 2.1 Robot mechanism

*Mechanical elements.* Two robotic mechanisms were used in the experiments described here. The first was a mechanism with a single leg. The leg consisted of a small (14 cm high, foot to hip), two-degree-of-freedom (DOF) mechanism. The upper joint was driven by a hobby-type servomotor Futaba MDL 9303. For all experiments described only the hip joint was driven, and the “knee” remained completely passive. The knee rotated on a low-friction ball-bearing joint and was prevented from going into hyperextension by a hard mechanical stop at zero degrees relative to the thigh. The passive knee is a critical element of the robotic setup, as we exploit the dynamics of the knee to compute the lower-limb trajectories.

The leg was suspended by a harness that served to constrain the mechanism’s hip rotational axis, as well as mediolateral (side to side) movements. The monopod could move forward and up and down to a small extent.

In the bipedal device used in the walking experiments, two legs were attached to a “hip” assembly that in turn was attached to a rotating boom. For the running experiments, the boom was also used; in some cases a linear sliding mechanism was used. In all cases, the hip was restricted rotationally. Because of this constraint, the effect of one leg on the other was primarily through translational movements (up/down and forward/backward). The bipedal setup is shown in Fig. 1. The pitch angle of the hip, specified by the harness, was found to be critical during running (see Sect. 2 in the Appendix B). If the harness were not set up with a correct angle, the system would not run well. The harness also limited how far the leg could fall. This limitation allowed the robot to start running from a standstill. Without this



**Fig. 1.** RunningMan Biped on treadmill. Mechanism is about 14 cm tall. Hips are actuated. Knees are passive. The foot is a round pad

limitation, the robot would probably not have been able to raise itself into a running posture on its own.

*Sensors.* Each robotic leg had three sensors on it. Two custom-built inductive LVDT-type (linear variable differential transformer) sensors monitored the position of the knee and hip joints. LVDT sensors were used because they introduce minimal friction and have infinite resolution. Having minimal friction minimized interference with the natural dynamics of the passive knee. A miniature load-cell sensor (Sensotec Model 13) on the foot was used to monitor ground forces. The units of the load cell are uncalibrated in all figures.

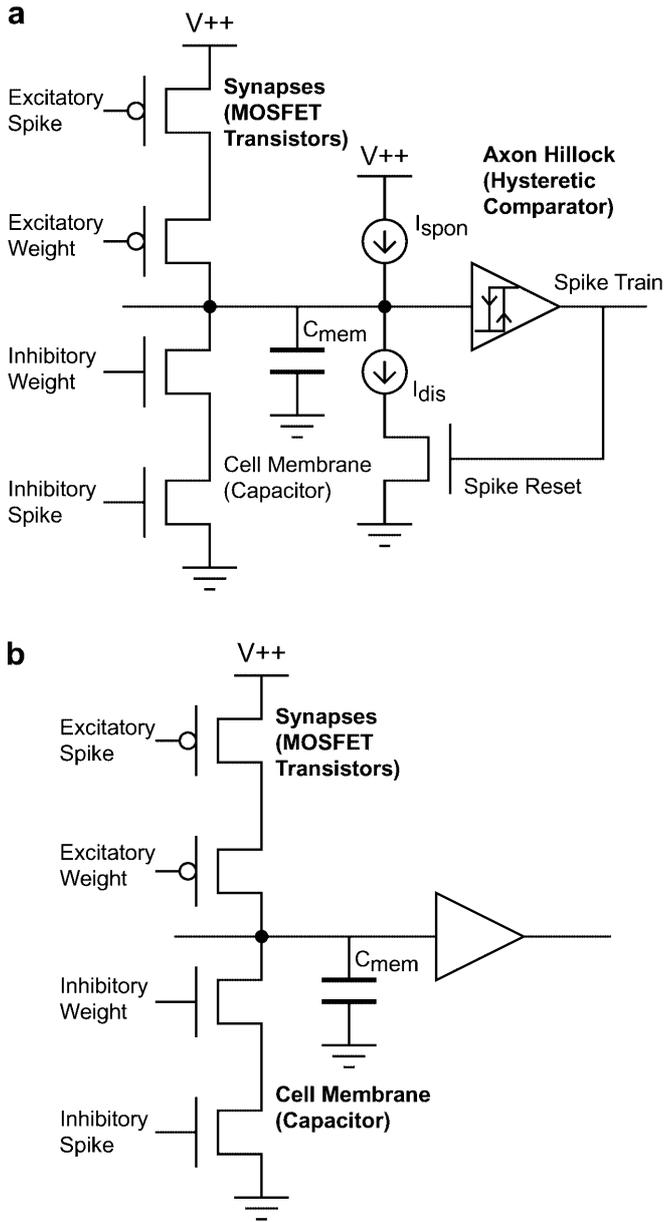
*Sensor signal processing.* The signal from the LVDT sensor at the hip monotonically increased as a sigmoid function of hip angle over the swing range used in our experiments. The output of the sensor was sampled digitally and interval-coded. Two intervals were selected as representing the extreme front or back movement of the hip. When these extremes were reached, the corresponding interval became active. When the leg swung forward past the point designated as the front extreme, the front interval went active. When the leg swung backward past the point designated as the back extreme, the back interval went active. These intervals thus served as binary signals that indicated the extremes of hip movement and were thus used to entrain or adapt the CPG chip. The analog values of the sensors indicating the joint angle were used for analysis but were not fed back to the controller.

*Walking and running surfaces.* In all experiments with the monopod, the running surface consisted of a rotating drum that was free to rotate under the contact forces of the leg. In the walking biped experiments, a flat, carpeted surface was used. The running surface for the legs consisted of a powered conveyor belt that served as a treadmill (Dormer Corp., model 2100M). A custom-built speed controller allowed the treadmill to establish and maintain a given rate.

### 2.2 The CPG chip

A CPG chip was fabricated in silicon using a 1.2-micron CMOS process. The chip contains electronic analogs of a biological neuron: inhibitory and excitatory “synapses,” “cell membrane,” “axon hillock,” and “axon.” In addition, the chip contains dynamic analog memories that can be used with synapses to modulate weights or to modulate the membrane conductance. Using these components, nonlinear oscillators, integrate-and-fire spiking neurons, and graded response neurons can be constructed (Fig. 2). In addition, in some cases the hysteretic comparator was replaced by a linear buffer circuit to obtain a graded response. The resulting circuit was used as part of adaptive circuits for the motor neurons.

Two *integrate-and-fire* neurons, based on the neuron drawn schematically in Fig. 2a, are included on the chip.



**Fig. 2.** Schematic of neurons implemented on the CPG chip. **a** Pacemaker and integrate-and-fire neural elements. **b** Interneuron with linear amplification

The integrate-and-fire neuron can also be made to act as a pacemaker envelope neuron.

We model the integrate-and-fire neuron of the chip using the following set of nonlinear differential equations:

$$C_i^{\text{mem}} \frac{dV_i^{\text{mem}}}{dt} = I_{\text{spont}} - S_i^T I_{\text{dis}} + I_0 \left( \sum_j W_{ij}^+ S_j^T - \sum_j W_{ij}^- S_j^T \right) \quad (1)$$

$$S_i^{T+\varepsilon} = \begin{cases} 1 & \text{if } (S_i^T = 1 \wedge V_i^{\text{mem}} > V_t^-) \vee (V_i^{\text{mem}} > V_t^+) \\ 0 & \text{if } (S_i^T = 0 \wedge V_i^{\text{mem}} < V_t^+) \vee (V_i^{\text{mem}} < V_t^-) \end{cases} \quad (2)$$

where  $C_i^{\text{mem}}$  is the “membrane” capacitance and  $S_i^{T+\varepsilon}$  is the state of the hysteresis comparator as well as the neuron output (0 or 1) at time  $T + \varepsilon$ . The currents  $I_{\text{dis}}$  and  $I_{\text{spont}}$  are discharge and spontaneous charging rates, respectively. The current  $I_0$  is the nominal synaptic currents from incoming excitatory and inhibitory spikes;  $S_j^T \in [0, 1]$  represents the state of the weighting factors applied to the incoming spikes, and  $V_i^{\text{mem}}$  is the “membrane” voltage.  $W_{ij}^-$ ,  $W_{ij}^+$  are inhibitory and excitatory weight factors, respectively.

*Relationship of CPG circuit to biological neurons.* The neuron is a lumped parameter membrane model of a biological neuron. A single voltage,  $V_i^{\text{mem}}$ , represents the state of the membrane voltage at the dendrites and cell body. The state,  $S_i^T$ , represents the state of the axon (firing or not firing).

The synapse strength consists of a “weight” that is represented as a transistor that controls the maximum current flow onto the membrane capacitor. A constant, applied voltage sets the current flow, or “weight.” In principle, this voltage could be placed in an analog memory on board the chip, but we did not do that here. This weight determines the quanta of charge placed on the cell membrane with each incoming spike. In contrast to biological neurons, this “quanta” of charge is relatively independent of the cell-membrane voltage.

The axon hillock, which in biological neurons is the site of axon spike generation (Kandel et al. 1991), is represented as a hysteresis comparator and a “spike reset” transistor, representing a voltage-gated ion channel. In contrast to a biological neuron, we use “hysteresis” and a “spike reset” gate as a rough substitute for a time-inactivation mechanism (Hille 1984). The hysteresis comparator and “spike reset” transistor work in the following fashion: the hysteresis comparator has upper  $V_t^+$  and lower  $V_t^-$  thresholds, both of which are set at design time and cannot be changed after fabrication. When the membrane potential rises to exceed the  $V_t^+$ , the output of the comparator goes high. The output will stay high until the membrane potential falls below  $V_t^-$ . The output of the comparator is fed back to a gate at the input to the comparator (“spike reset”). This resets the membrane potential by draining charge from  $C_{\text{mem}}$ . Because of the hysteresis, the draining continues until the membrane potential is below  $V_t^-$ . Since the magnitude of the discharge current  $I_{\text{dis}} > I_{\text{spont}}$  by design, the net charge is drained from  $C_{\text{mem}}$ . This organization introduces a second state variable needed to create an oscillatory system. This two-state system is in the spirit of the neuronal models of FitzHugh (1961) and Nagumo et al. (1962).

*Creating spiking and envelope pacemaker neurons.* Note that in this circuit, the interspike interval is determined by the magnitude of  $I_{\text{spont}}$ . By setting  $I_{\text{dis}} \gg I_{\text{spont}}$  we obtained a spiking output. If  $I_{\text{dis}} \sim 2 \times I_{\text{spont}}$ , we obtain a “pacemaker” output with roughly symmetric high and low periods. By varying the ratio of the  $I_{\text{spont}}$  and  $I_{\text{dis}}$  currents, the neuron can behave as an integrate-and-fire neuron or as a pacemaker (oscillator) neuron. In the case

of the pacemaker, although the output is square-wave and nonspiking, it can be considered the time-averaged output of a spike train.

*Interneuron.* By using a linear buffer in place of the hysteretic comparator and eliminating the discharge path, we can create a graded response neuron that does not spike.

### 2.3 Enhancement to pacemaker neuron

In some experiments reported below, we used an enhanced pacemaker neuron. A small PIC<sup>TM</sup> (MicroChip 16C877) processor was used to generate a brief burst of pulses upon transition from the low to the high state. As will be discussed below, the chip was originally designed for continuous coupling between neurons to achieve the alternating burst pattern seen in motor neurons across the midline. However, we found it necessary to use pulse coupling for these neurons (cf. Sect. 3.4 for further explanation of this problem and its solution). This change resulted in a shortage of silicon neurons. As a consequence, we used a small processor to generate the bursting pattern needed for pulse coupling. We feel that this change did not materially affect our results, as this processor can be replaced in future work with a neuronal equivalent.

### 2.4 Interface to robot

A PIC<sup>TM</sup> (MicroChip 16C877) processor was used to convert the pulses from the motor neurons to a pulse-width-modulated signal for the motors. The PIC implemented a pure integrator. This integrated signal was then set as the position for the motor, which has its own position-control circuit on board. A position command is given to the motor, and the motor is driven toward the desired set point. We plan to use force-level control of the hip in future iterations.

The neurons of the CPG chip were interfaced to a servomotor using a rudimentary muscle model, and the muscle dynamics were simulated as a low-pass filter to smooth the output of the spiking neurons. This was followed by an integrator, implemented in software, to convert the position signal to a position command needed by the servomotor. A bias was intentionally introduced into the chip to cause an asymmetry in the backward and forward swing of the leg. This bias might be typical of uncompensated parameters in a chip.

An oscillator frequency was selected by hand to be approximately 2–3 Hz. This frequency would excite the mechanical structure of the robot and cause the leg to run on a rotating drum. At lower frequencies, for example 1 Hz, the foot would drag on the treadmill when in swing phase. If the frequencies became too high, the amplitude of the leg swing decreased significantly due to the limited bandwidth of the hip actuator. We assume that if the actuator had sufficient bandwidth, the leg

shank would not lock in place at the end of the swing phase.

## 3 Analysis, simulation and empirical results

### 3.1 Analysis of passive knee

The passive knee can be modeled with the following equations:

$$\ddot{q} = \frac{(-\sin(q) \cdot \ddot{y} - \cos(q) \cdot \ddot{x} - \sin(q) \cdot g)}{l} - \lambda \cdot \dot{q} \quad (3)$$

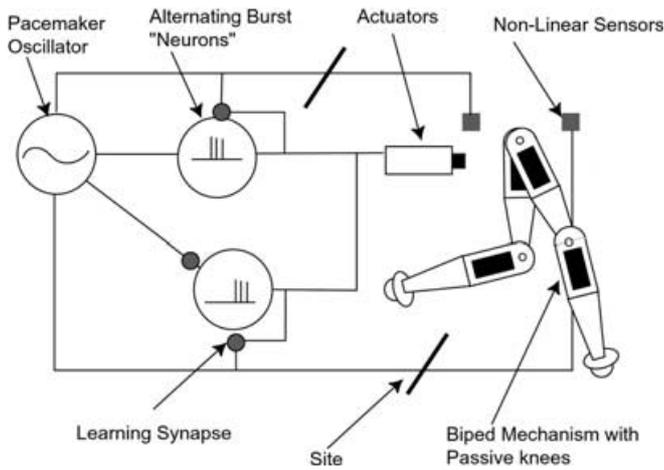
$$q < \alpha \quad (4)$$

where  $[\ddot{x}, \ddot{y}]$  is the knee acceleration,  $q$  is the shank (the lower segment below the knee) angle,  $\lambda$  is velocity dampening,  $l$  is the shank center of mass relative to the knee,  $g$  is the gravitational constant, and  $\alpha$  is the hip angle. By inspection, the only way to influence knee trajectory is by accelerations at the knee cap. In the absence of knee acceleration, the shank behaves as an inverted pendulum. The most significant way to influence the frequency of the knee, and hence the running frequency of the entire system, is to control the velocity of the knee joint at the stance-swing boundary. Through simulation analysis, (see Appendix B) we found that the range over which the system could run could be slightly extended by controlling the energy of the lower limb at takeoff. But in all cases, the frequency of the swing phase could not be significantly changed if running was to be stable. This accords well with the observation that in animals the stance phase is altered in the transition from walking to running. It is not typically altered in the swing phase, which remains constant. We anticipate that very little control would be needed in the knee during normal walking and gait transition. Appendix B gives the complete analysis of the passive knee via simulation results.

### 3.2 The complete monopod circuit

Figure 3 shows the overall schematic for the circuitry to control the monopod leg and a picture of the biped mechanism. It is important to note that the circuitry must be duplicated twice to control the biped shown. For each leg, a pacemaker neuron drives two bursting (spiking) neurons. In our system, we used two outputs from the pacemaker: an inverted and a noninverted output. When active, these signals inhibit the motor neuron on which they synapse. This arrangement does not allow both motor neurons to become active at one time (i.e., no cocontraction is possible).

The spiking neuron receives self-limiting feedback. This feedback limits the firing rate of the neuron. Another circuit, built from an interneuron, supplies a bias to *increase* the firing rate of the neuron, acting in opposition to the self-limiting feedback. This bias is reduced as the leg reaches the limits of travel. The



**Fig. 3.** Interface schematic for RunningMan Biped. The CPG chip is used to construct pacemaker and bursting neurons. The outputs of the bursting neurons are summed. A conversion circuit (not shown) converts the burst patterns to motor commands for the biped mechanism. The *filled circles* represent inhibitory synapses

circuit ensures that the stride length has the correct width.

A feedback path goes from the nonlinear sensor to the pacemaker cells. This feedback pathway resets the pacemaker and allows the movement of the leg to *entrain* or synchronize the pacemaker oscillator.

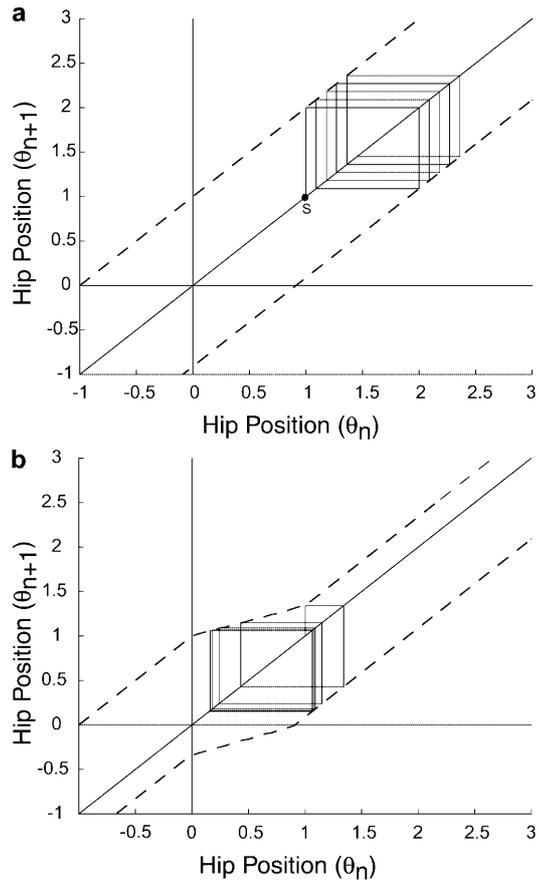
### 3.3 Effect of sensory feedback

#### 3.3.1 General analysis of the effects of sensory feedback.

As mentioned in the introduction, sensory feedback is critical in adapting a walking machine, either animal or robotic, to the terrain. We performed a mathematical analysis of the effect of sensory feedback on the oscillator using map functions and staircase diagrams (Hale and Koçak 1991).

Figure 4 considers the case of connecting a single oscillator to a leg of a monopod robot. The upper and lower dashed lines in Fig. 4a and b are the “map functions.” These functions are created by solving the differential equations (Eqs. 1–4 of Sect. 2.2) for hip position at every half cycle of movement. The hip position at every half cycle is then geometrically visible by connecting the main diagonal with the two map functions (see caption for details). Note that the choice for the hip position to range between  $\sim -1$  and  $\sim 3$  represents only arbitrary units.

Figure 4a clearly shows that in the absence of sensory feedback, the hip position moves forward unboundedly. In the real robot, this would mean a rapid deterioration of the gait. In contrast, the addition of sensory feedback to entrain the CPG had the effect of forming a restriction or “bottleneck” on the geometric surface (Fig. 5b). This forced the leg to oscillate around a particular sensory region, ensuring a stable gait. In contrast, in the absence of sensory feedback, the leg continuously moves forward, as will be described in Sect. 3.3.3; we were able to confirm the results of this analysis through lesion experiments in the real monopodal system. The staircase



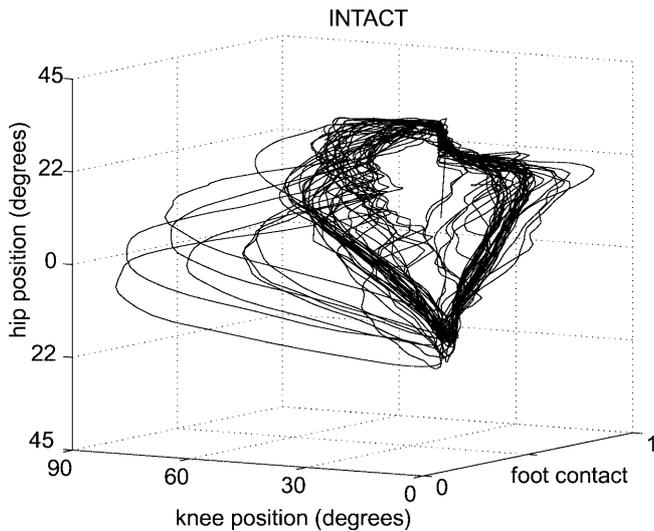
**Fig. 4.** The effect of sensory feedback is to alter the map function describing the nominal position of the limb at each cycle. To compute hip position and drift using these diagrams: (1) Choose a starting point on the diagonal line (e.g., point ‘S’ in the figure). This is the initial starting position of the hip. (2) Now move along a vertical line to the upper dashed line, then horizontally to the diagonal line. This is the position of the hip after one half cycle. (3) Repeat the process, this time drawing the vertical line *down* to the *lower* dashed line. In a perfectly balanced or symmetric system, the distance between the dashed lines and the diagonal solid line are equal. In an unbalanced or perturbed system, these lines have unequal distances. **a** In the absence of sensory feedback, the lower dashed line is slightly closer to the diagonal than the upper dashed line. The position of the limb gradually drifts. **b** In this case, we add sensory feedback acting in a phase-dependent manner. The effect is to create a bottleneck. This bottleneck “traps” the trajectory of the limb, holding it in a stable position. Lesioning (Sect. 3.2.3) transforms the geometry of the system from **b** to that of **a**

analysis presented above exactly confirmed the results of running in the real robot.

Although the details of this analysis are complex and highly specific to this chip, the staircase method for visualizing the stability of the walking/running system is effective.

#### 3.3.2 Monoped “Running” with a passive knee: sensory feedback intact.

In this experimental setup, the CPG circuit drove the actuator in the hip joint, and the monopod was suspended above a rotating drum. Data were collected from three sensors: foot pressure, knee, and hip. Hip-sensor feedback from the hip was used as feedback to the CPG. The foremost result was that the



**Fig. 5.** Hip-, knee- and foot-contact phase diagram. Most of the trajectory is in a tight bundle, while the outlying trajectories represent perturbations

circuit adapts such that the passive knee joint has the correct dynamics to enable running.

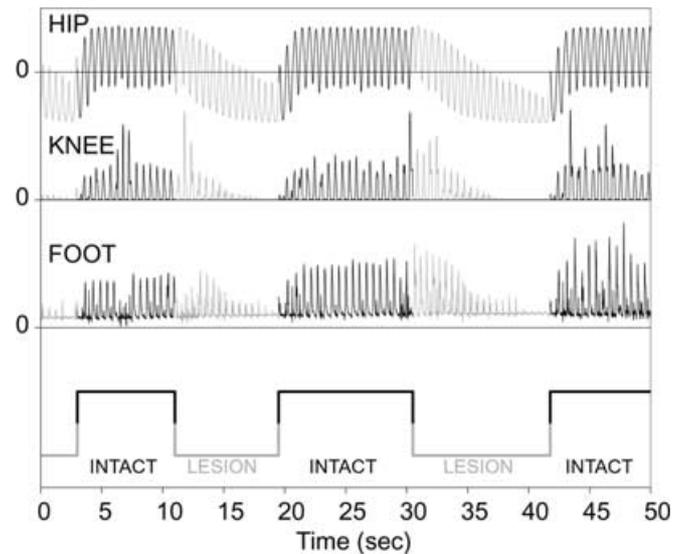
Figure 5 shows a phase plot of the hip and knee position and foot contact force. The bulk of the trajectory describes a tight “spinning top” shape, while the few outlying trajectories are caused by external disturbances. After a disturbance the trajectory quickly returned to its nominal orbit, implying that the system was stable.

**3.3.3 Monoped running with a passive knee: sensory feedback lesioning.** We next established that sensory feedback was critical to entrain the pacemaker neuron and thereby ensure a centered stride, by performing selective sensory feedback “lesions.” As described in Sect. 2.4, we first added a chip bias so that there were significant asymmetries in the forward and backward swing of the limb. This accounts for the asymmetric map functions of Fig. 4a. Figure 6 shows the effect of lesioning sensory feedback on the position of the hip and knee joints and on the tactile input to the foot. When feedback is intact, the circuit adjusts for the chip-induced asymmetry, but when feedback is lesioned, the leg drifts backwards. The leg returns to a stable gait only after sensory input is restored.

We can relate these results to Fig. 4. The intact case corresponds to Fig. 4b. The lesion case corresponds to Fig. 4a. The effect of lesioning the robot is to transform the geometry of the map function from a bottle neck configuration to two parallel lines. These parallel lines are sensitive to the slightest imbalance in the system. The caption for Fig. 4 describes how to compute the position of the hip through time.

### 3.4 Creating a minimal coupled CPG

The most basic property of the spinal CPG is coordination across the midline, which requires neurons to be



**Fig. 6.** The effects of lesioning sensory feedback. When the feedback is lesioned (time 11–19 s and 31–42 s), the hip drives backward significantly. As it does, the foot begins to lose contact with the ground, and the knee stops moving. When the lesion is reversed at 19 and 42 s, the regularity of the gait is restored

bidirectionally coupled. In our first approach to bipedal coupling, we directly coupled two pacemaker neurons together, with the output of one being fed directly to the membrane capacitor of its complement. We called this continuous coupling. We found that continuous coupling resulted in patterns of oscillation that were very unstable, since one neuron would disrupt the behavior of the other.

In biological systems, however, coupling across the midline in vertebrates is accomplished using *spiking* neurons with decaying spike rates after the initial onset of coupling (i.e., biological neurons use *pulse coupling* with rate adaptation). This is quite different from the continuously coupled nonspiking pacemaker neurons in our system.

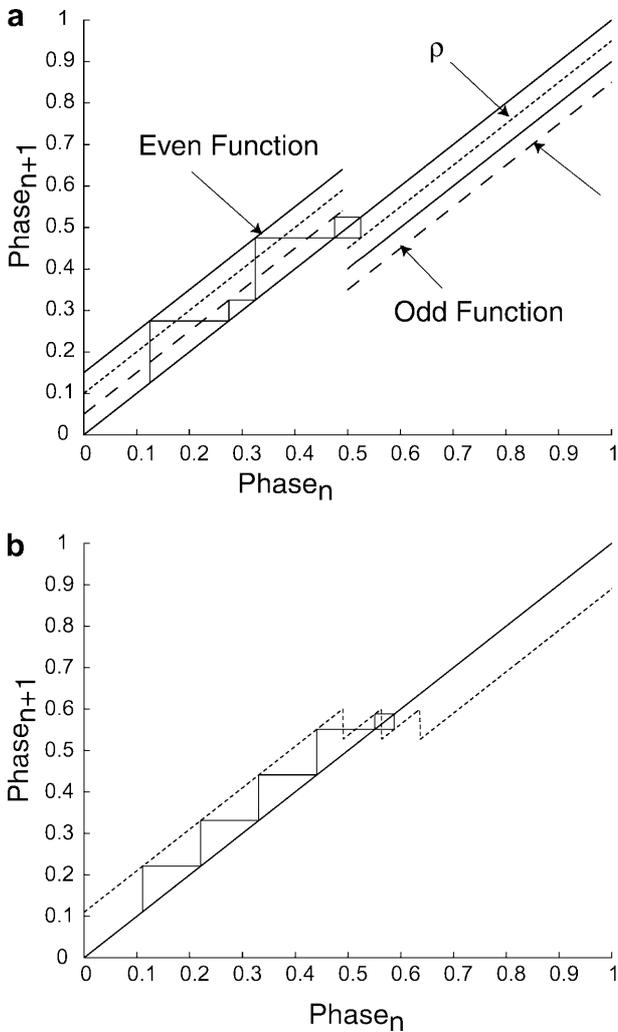
We therefore tried a second approach in which we used spiking neurons to couple the two oscillators. This approach revealed a significant advantage of using spiking neurons in our system. The CPG oscillators were particularly well behaved, and very stable, when constructed with spiking neurons to achieve *pulse coupling*. Specifically, we found that the use of spiking neurons improved the convergence of the phase locking.

It is generally agreed that the primary benefit of spiking neurons is their tolerance to noise in the axon (transmission line) and their ability to transmit over large distances. In our case, however, we discovered another advantage of using spiking neurons: the system is amenable to discrete analysis using map functions to describe their dynamics. The analysis and simulation details are presented in Appendix A, and we summarize only the major results in Sects. 3.4.1 and 3.4.2.

**3.4.1 Geometric analysis of pulse coupled neurons.** As in the case of sensory feedback, we used geometrical

analysis to understand bidirectional pulse coupling between two neurons. In this case, the map functions (Hale and Koçak 1991), describe the phase change of the output of a spiking neuron due to pulse perturbations from a second spiking neuron. As shown in Fig. 7, the phase difference between the two oscillators does not converge to an asymptotically stable fixed point, but rather, convergence is bounded to a certain region. That is, the phase difference will be bounded, but it is not guaranteed to be zero. The details of this analysis can be found in Appendix A.

We validated this mathematical analysis of spike coupling by determining the real map function in the CPG chip. As will be shown in Sect. 3.4.2, the map function predicted by the analysis and the map function empirically determined from the chip accord extremely well.



**Fig. 7.** Map functions of bidirectional coupled neurons. **a** Single pulse case, different periods. The derivative at the fixed point is infinity. Thus, with a single pulse it is not possible to achieve asymptotic convergence with this neuron. **b** Multipulse case, equal periods. The geometry begins to alter such that the line passing through the fixed point is horizontal. Although the derivative is still infinity over a small region, this region is much smaller

**3.4.2 Empirical determination of map function for multipulse neuronal coupling.** One of the major results of the mathematical analysis derived in Appendix B is that multiple spikes will result in better coupling characteristics for our system. However, a key remaining question is whether the mathematical model is a valid description of the CPG chip.

To answer this question, we performed experiments on the chip to determine the geometry of coupling. We conducted an investigation in the case of a master oscillator driving a slave with either a single pulse or two pulses for coupling.

For the driving oscillator, we used a very slow master oscillator that had a period that was not an integer multiple of the slave frequency. We then recorded the state of the neuron output,  $S$ , for an extended period of time (approximately 1–2 min).

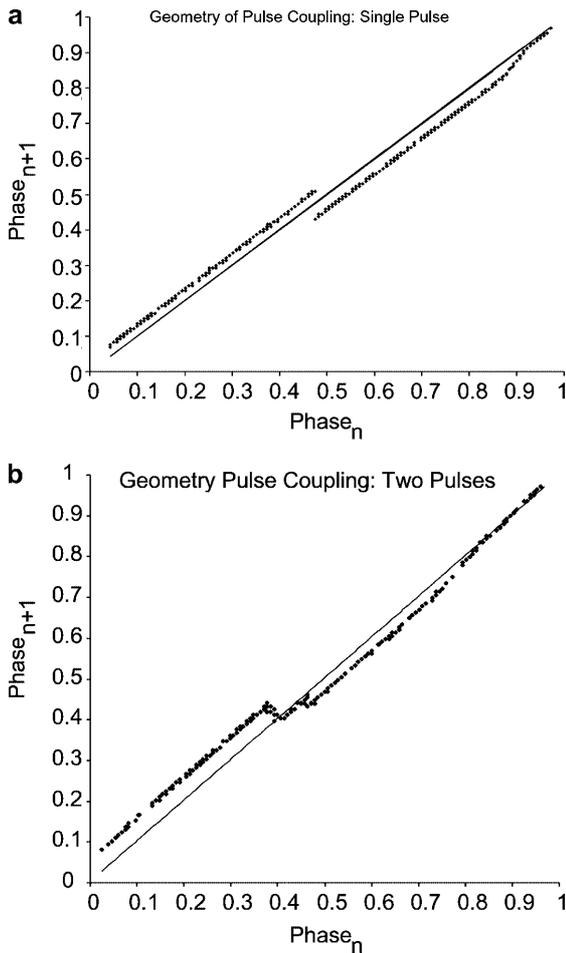
Custom-built software extracted the period of the pacemaker when there were no perturbations (control case) and before and after a spike (or spikes) from a second neuron perturbed the pacemaker. Figure 8a shows the geometry of coupling based on a perturbation with a single pulse. Figure 8b shows the geometry of coupling based on perturbation with double spikes. On these graphs, a fixed point is indicated by a map function crossing the diagonal line.

In Fig. 8a, the slope becomes vertical at the fixed point, manifesting itself as discontinuity when the phase is about 0.46. On these graphs, a fixed point is indicated by a map function crossing the diagonal line. In the case of two pulses, the slope flattens out dramatically. This indicates a region of asymptotic convergence for the oscillators. Comparing the result to those predicted by the model (see Figs. A1a and A3a), we find excellent accord between the predictions of our model of the CPG chip and the actual data collected. These results strongly support the mathematical analysis presented in the Appendix.

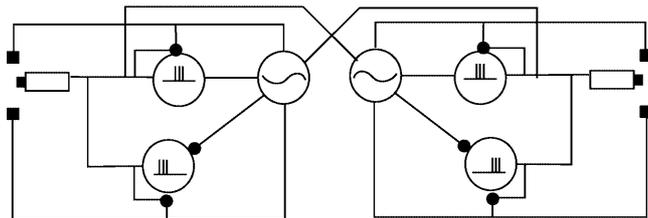
**3.4.3 Coupling the CPG to the runningman robot.** We next used the pulse-coupled pacemakers to control the running biped. Figure 9 shows the circuit for the complete CPG for the biped robot. The pacemaker circuit drives motor neurons that interface the minimal CPG to the robot, and the phase of the CPG and the firing rate of the motor neurons are adjusted based on sensory feedback from the leg. Because RunningMan has passive knees, the trajectory of the lower limb segment is implicitly “computed” as a physical computation.

### 3.5 Bipedal running

The biped mechanism was suspended above the treadmill using the harness, and the CPG was then started. A digital video camera placed perpendicular to the treadmill recorded the movement of the legs. To facilitate data reduction, LED markers were placed on the robot. The approximate positions of the markers were: center



**Fig. 8.** Results of empirical experiment supporting the CPG model and analysis of coupling. **a** Single-spike coupling case. **b** Two-spike coupling case

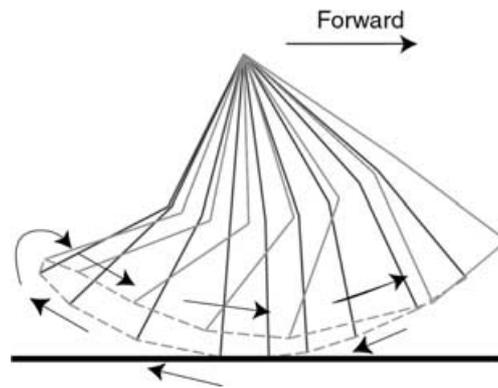


**Fig. 9.** Circuit for controlling the biped robot. The two halves of the control system are coupled together using pulse coupling (multiple spikes)

of hip rotation, center of knee rotation, and middle of foot.

Data analysis was done offline. Custom software was written to track the  $x$ - $y$  pixel position of each marker for each frame and produce a table of results.

A second software program plotted the movement of the legs. Figure 10 shows a single stride for the right leg. The dark line shows the leg positions as it moves backward. The foot hits the ground, and the knee locks. When the leg changes direction (the leg is now drawn in light gray), the knee breaks, and the leg swings forward.



**Fig. 10.** The trajectory of one leg of the biped. See text for explanation

When the leg reaches the end of travel, it hits an end stop.

The light gray dashed line shows the trajectory of the foot through time. The trajectory appears smooth and somewhat natural. Notice that the frame of reference of this video is the hip of the robot.

Notice also that the leg appears to swing forward more than it does backward. Empirically, if the leg is rotated more forward, the limb collapses under the robot. In practice, the lower limb would need to have a mechanism to keep it locked in place. Pratt uses a locking torque, applied at the knee, to accomplish this (Pratt and Pratt 1999).

It is important to notice that if the knee had been actuated, then all the positional and dynamical relationships between the hip and knee actuators of both limbs would have to have been computed and imposed by the control system. This would have required a complicated controller, which is difficult to implement in software and more difficult in hardware. A major result of this work is that the “physical computation” performed by the passive knees simplifies the biped control system immensely. Furthermore, the minimal controller and the natural dynamics of the passive knees accomplish very realistic (biologically speaking) running action.

Experiments were also performed on the ground where the robot was able to walk quickly around a circle. AVI clips showing the running biped, monopod, and walking biped are available at <http://www.iguana-robotics.com>.

## 4 Conclusions

In this work we presented analysis, simulation, and empirical results in the use of a CPG chip to control a monopod and a biped mechanism. We demonstrated how, beginning with capacitors and transistors, we built three different neural elements. We also showed how these elements could be coupled together to produce a simple circuit sufficient for creating a basic pattern of movement in the hip joint of the mechanism. Through analysis using geometric methods and simulation, we

showed how pulse coupling could very efficiently couple together basic neural elements to create a simple CPG. We showed how sensory feedback influenced the geometry of the system's limit cycle.

In addition, we used a passive knee and showed how this leg segment could participate in the computation of the overall trajectory of the mechanism.

We conclude that with a reduced system of a few transistors and capacitors (not counting the PIC<sup>TM</sup> processor, which could be replaced easily with a neuronal equivalent in future work) and using insights from biology, it is possible to generate basic movement patterns in a mechanical system that bear a remarkable overall resemblance to their biological counterparts.

The major challenge ahead is to incorporate balance control and supraspinal control. The minor challenges include the creation of a new chip with sufficient neurons to implement all aspects of the design completely on a self-contained chip and to create a new mechanism with a locking knee for stability and a foot and ankle. Finally, after the evolution of the chip is complete, it may be able to control its biological counterparts, since it has been designed from the beginning to be compatible with the principles of biological systems.

*Acknowledgements.* The authors acknowledge support of Grant No. N00014-99-0984 from ONR to Lewis & Etienne-Cummings, NSF Career Grant #9896362 to Etienne-Cummings and NIH grant MH44809 to Cohen. MAL thanks Suzanne Still for formative discussions on the possibility of building a silicon CPG chip. The authors would also like to thank Francesco Tenore for work in constructing the experimental setup as well for as useful comments on this manuscript. Many thanks go to Kaijen Hsiao and Chris Milne for their suggestions and comments on the manuscript.

## Appendix A. Analysis of pulse coupling

Here we describe the use of geometric analysis to understand pulse coupling between our silicon neurons.

Consider two envelope pacemaker oscillators with similar frequencies and one-way spike coupling. By "one-way" we mean that one oscillator will be the "master," trying to entrain the "slave" oscillator. By "spike coupling" we mean that on each cycle, the master oscillator will send a certain number of spikes (perhaps only one) to the slave oscillator, instead of sending a continuous entraining pulse, as would be the case in "continuous coupling." Let us derive an equation describing the change in phase of the slave oscillator when the master sends a single spike to the slave oscillator. The slave oscillator has equations governing its behavior:

$$C_i^{\text{mem}} \frac{dV_i^{\text{mem}}}{dt} = I_{\text{spon}} + \Lambda \quad (\text{A1})$$

$$C_i^{\text{mem}} \frac{dV_i^{\text{mem}}}{dt} = I_{\text{spon}} - I_{\text{dis}} + \Lambda \quad (\text{A2})$$

for the charge and discharge cycle, respectively, where  $\Lambda$  is the spike-coupling function, considered to be an impulse function that adds charge  $\rho$  to the driven

oscillator. The effect of a single pulse on either the charge or the discharge cycle is:

$$V^+ - V^- = \frac{\rho}{C_i^{\text{mem}}} \quad (\text{A3})$$

That is, the effect of a pulse is to raise the voltage by an amount proportional to the charge in the pulse and inversely proportional to the capacitor size. Here, the  $-$  and  $+$  superscript indicate times just before and just after the receipt of the impulse.

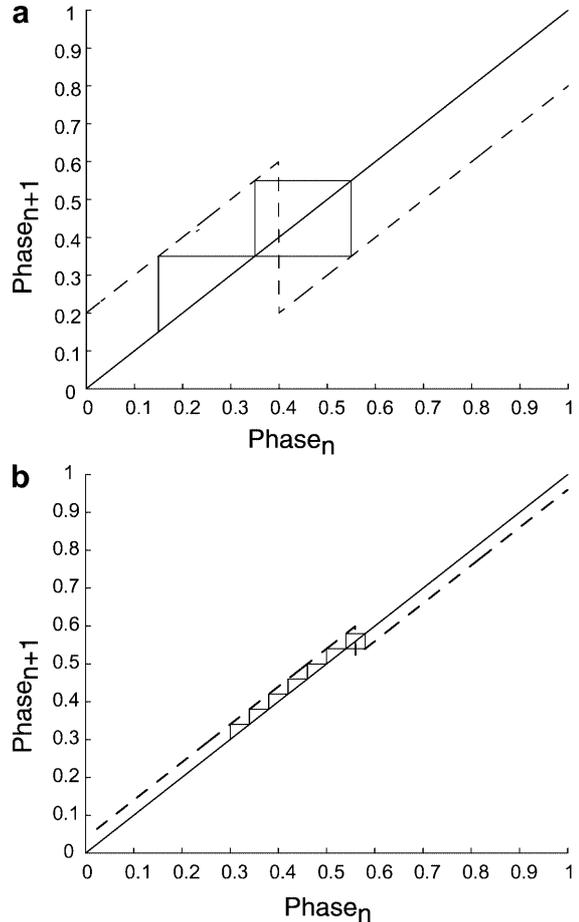
What is the effect of such a pulse on the phase of the slave oscillator? Let the difference between the low and high threshold of the oscillator circuit be  $\Delta V$ . If we assume a 50% duty cycle, then we quickly find that:

$$\Delta\theta = \frac{\rho}{2 \cdot \Delta V \cdot C_i^{\text{mem}}} = \frac{(V^+ - V^-)}{2 \cdot \Delta V} \quad (\text{A4})$$

which is the equation we sought.

For example, if  $\Delta V = 0.1$ , and the pulse changes the voltage by 0.02, then the phase change will be 0.1.

Figure A1a shows a stairstep diagram for two same-frequency oscillators coupled using single pulses. We see a rapid convergence to a fixed point. Note that the final position is not unique; rather, it is determined by the initial conditions of the oscillators. Even so, the final



**Fig. A1a,b.** Geometry of the convergence of two oscillators of the same frequency. **a** Strong coupling; **b** Weaker coupling

solution is confined to a small region, as shown in Fig. A1b. If the coupling strength is reduced, the two regions bounding the staircase are squeezed together (see Fig. A1b).

### A.1 Coupling with different periods

We now analyze the more general case in which the oscillators' periods (and therefore frequencies) are different. Consider two oscillators with periods related as:  $T_1 = T_2 + \Delta\theta$ ,  $\Delta\theta > 0$ . Oscillator 2, with the time constant  $T_2$ , is the faster oscillator since its period is shorter. Let oscillator 1 be the master oscillator.

The map function that describes the phase advance of the slave oscillator due to the coupling of the master is shown in Fig. A2. Basically, the introduction of  $\Delta\theta$  shifts the graph vertically. In this figure,  $\Delta\theta = 0.05$  indicating a slightly faster slave and  $\rho/C = 0.1$ .

Notice that we converge but experience an interesting coupling phenomenon. We get a large *jump forward* in phase followed by *three small steps backward* (for these particular parameters). From this diagram, it is easy to infer that if the periods are very different, we will not be able to entrain the slave oscillator to the master. In that case, the trajectory of the phase of oscillator will move above the diagonal. At that point, entrainment becomes impossible.

This phenomenon of dithering during locking is fairly common. In simulations of the oscillators coupled together, we often saw "jittering" in the coupling.

The case of bidirectional coupling is no more difficult. It is very simple if we assume that pulses do not arrive at exactly the same time. Under those conditions, we can alternate back and forth, allowing each oscillator to be the master and slave in turn.

By overlapping the two resulting map functions, we can visualize bidirectionally coupled neurons. By denoting the first oscillator's map function as "odd" and the second as "even" we realize the combined graph in Fig. 7a.

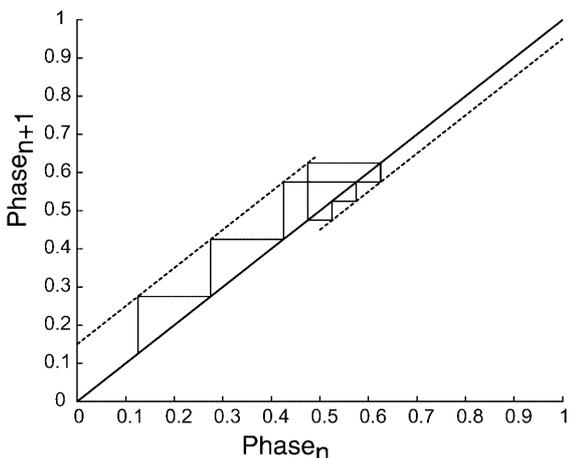


Fig. A2. Entrainment of two oscillators with different frequencies

### A.2 Multipulse coupling

The more interesting case is when we consider multiple pulses. There are three degrees of freedom in specifying two pulses. The first degree of freedom is the strength of the pulse. For the sake of comparison with the single-pulse case, we will set the strength equal to  $\rho^n = \rho/n$ , where  $\rho$  is the charge of a pulse in the case of one pulse, and  $n$  is the number of pulses. A second degree of freedom is the relative phase of the second pulse with respect to the first pulse. We denote this relative phase difference as  $\Delta\theta^R$ . The third degree of freedom is the absolute phase of the first pulse. This third degree of freedom can be set equal to zero without loss of generality. The only degree of freedom that is not specified is  $\Delta\theta^R$ , the relative phase of the first vs the second oscillator. To find this value, there are two cases to consider:

Case 1.  $\theta < 0.5 - (2\Delta\theta + \Delta\theta^R)$  or  $\theta > 0.5 - 2\Delta\theta$

In this case, the two pulses act like a single pulse because both pulses influence *either* the rising capacitor voltage *or* the falling capacitor voltage. Their effect is indistinguishable from a single pulse of weight  $2^*\rho$ .

Case 2.  $\theta > 0.5 - (2\Delta\theta + \Delta\theta^R)$  and  $\theta < 0.5 - 2\Delta\theta$

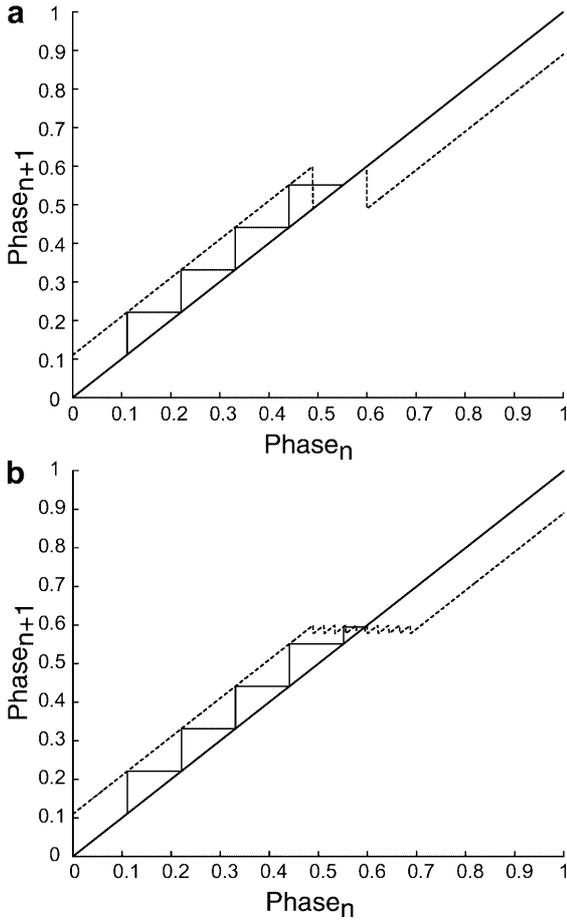
In this case, the effects of the two pulses cancel each other out since one acts on the rising capacitor voltage and the other acts on the falling capacitor voltage. For example, the first pulse advances the phase. Then, the second pulse retards the phase by an equal amount, as the two pulses have the same weighting. Thus, if we were to plot a staircase diagram for this canceling case, we would find a region where there was no net change in the phase. This would appear as a line segment on the main diagonal.

If three pulses occur, then we have the following three cases: (1) All three pulses could occur either before or after  $\theta = 0.5$ . In this case, all three pulses act either to advance or retard the phase and produce results identical to a single pulse. (2) One pulse advances the phase while the other two retard the phase. (3) Two pulses advance the phase while the third pulse retards the phase.

For the case of one pulse, we had two distinct regions in the staircase diagram. If we have two pulses, we have three distinct regions (Fig. A3a). If we have three pulses we have four distinct regions. By inference, if we have  $n$  spikes, we will have  $n+1$  distinct regions in the corresponding staircase diagram. This is illustrated in Fig. A3, which shows the geometry of coupling given two spikes (Fig. A3a) and ten spikes (Fig. A3b).

The point at which the map function crosses the diagonal line is a fixed point (see Fig. 7a). By Theorem 3.8 in Hale and Koçak (1991), if 'x' is a fixed point,  $f(x)$  a map function, and  $|f'(x)| < 1$ , then point 'x' is an asymptotically stable fixed point. If  $|f'(x)| > 1$ , the point is unstable. In the case of a single-pulse coupling, we found that  $|f'(x)| = \infty$  and is therefore not asymptotically stable.

When we tried multiple spikes, we found, interestingly, that  $|f'(x)|$  converged to zero as more spikes were



**Fig. A3a,b.** Effect of multiple pulses on coupling. **a** Two pulses; **b** 10 pulses. Notice that the convergence in the region of the fixed point improves considerably

added. This implies asymptotic stability. Thus, we showed that multispikes coupling is better than single-spike coupling in our system.

It should be noted that the following factors would also improve the stability: a single pulse with extended spike duration is exactly equivalent to multiple single spikes occurring on the same side of  $\theta = 0 \dots 0.5$ . A second stabilizing factor would be a strong nonlinearity near the threshold of the neuron such that it became *less* sensitive to firing. This would be the case in a biological synapse where the current flow has a voltage-dependent synapse. Simulation results (not published) show that adding voltage dependency improves coupling characteristics.

## Appendix B: Passive knee analysis

### B.1 Shank Dynamics

We make the following assumptions when modeling the robot: (1) During stance phase, the robot leg is locked. (2) If one leg is in stance, the other is in swing, except during an instantaneous moment of dual support phase. (3) The hip is held rotationally fixed. (4) The significant

dynamic element is the shank (lower portion of the leg) of the swinging leg. (5) The criterion for a good “swing” is that the shank hits, or nearly hits, the knee lock in swing phase the first time the hip of the corresponding leg has near-zero velocity (i.e., the point where it is beginning its return stroke). (6) The leg moves backward the same distance that it moves forward.

If 1 through 4 are true, the position of the knee of the leg in *swing* is determined by its kinematics. In particular, if the hip angle  $\alpha$  is known, the knee position of the swing leg is determined. This assumption will simplify the analysis considerably.

The dynamics of the shank can be found by using the Lagrange method (Asada and Slotine 1986). We model the shank as a point mass concentrated at a length  $l_3$  from the knee.

We write down an equation for the position of this point mass, differentiate, and give the equations for kinetic and potential energy of the leg:

$$x = x_{\text{knee}} + l_3 \sin(q(t)) \quad (\text{B1})$$

$$y = y_{\text{knee}} - l_3 \cos(q(t)) \quad (\text{B2})$$

$$\dot{x} = \dot{x}_{\text{knee}} + l_3 \cos(q(t))\dot{q}(t) \quad (\text{B3})$$

$$\dot{y} = \dot{y}_{\text{knee}} + l_3 \sin(q(t))\dot{q}(t) \quad (\text{B4})$$

$$KE = \frac{m}{2} \cdot ((\dot{x})^2 + (\dot{y})^2) \quad (\text{B5})$$

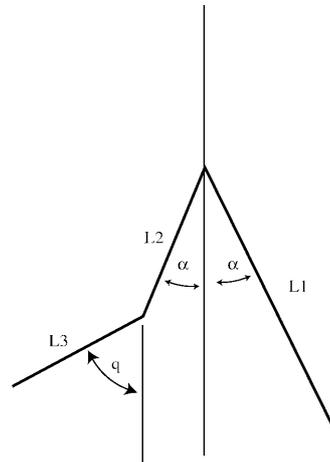
$$PE = y_{\text{knee}} - \cos(q(t)) \cdot l_3 \cdot g \cdot m \quad (\text{B6})$$

where  $g$  is the gravitational constant,  $l_3$  the distance to the center of mass of the shank, and  $m$  the mass of the shank.

The Lagrangian is:

$$L = KE - PE \quad (\text{B7})$$

The dynamics are simply:



**Fig. B1.** Kinematic arrangement of the biped robot. The leading leg is in stance phase. The trailing leg is in swing phase

$$\tau = \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}} \right) - \left( \frac{\partial L}{\partial q} \right) \quad (\text{B8})$$

where  $\tau$  is the joint torque. Since the link is passive,  $\tau \equiv 0$ .

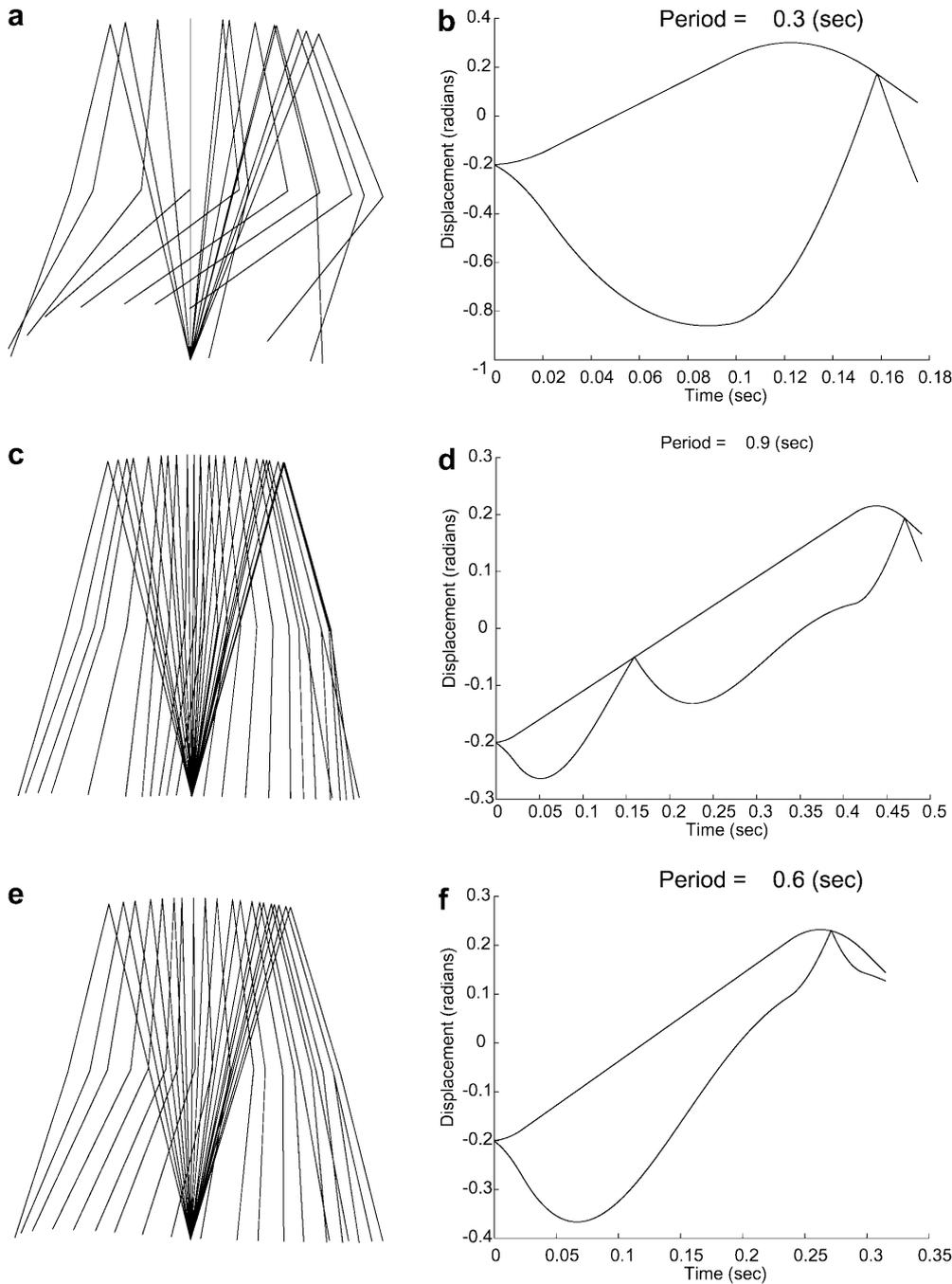
The knee has a lock to prevent it from hyperextending. This could be simulated as an elastic collision, in which momentum and kinetic energy are conserved. However, we have chosen to model the collision as a very strong spring and damper that becomes active when the leg locks. Thus, the leg will hit the end-of-joint lock and then rebound elastically.

Plugging Eqs. B1–7 into B8 and simplifying we have:

$$\ddot{q}(t) = (-\sin(q(t)) \cdot \ddot{y}_{\text{knee}} - \cos(q(t)) \cdot \ddot{x}_{\text{knee}} - \sin(q(t)) \cdot g) / l_3 - \beta \cdot r - \lambda \cdot \dot{q}(t) \quad (\text{B9})$$

where the  $\beta \cdot r$  term is the strong repelling spring simulating the knee lock,  $r$  the distance past the lock, and  $\lambda \cdot \dot{q}(t)$  a damping element included to simulate minor frictional effects.

Examining this equation, we see that in the absence of knee acceleration and knee lock, the leg acts as damped inverted pendulum, as would be expected. As we shall see shortly, in our model, knee accelerations happen principally at the joint limits. Therefore, for most of the time, the system behaves like an inverted pendulum.



**Fig. B2a–f.** Trajectories and joint angles of the leg for various parameter conditions. **(a and b)** Typical trajectory of the leg when the parameters are set to a low frequency. In **a**, the leg is moving too fast, and thus the knee never locks to late. ( $v_{\text{max}} = 3.0$ ), while in **c** the leg is moving too slowly and the knee locks too soon. In **e**, the parameters are set to a “good” combination, and the knee locks at exactly the correct point in time when the hip changes direction. **b, d, and f** joint angles (hip and knee) are shown for the cases **a, c, and e** above. The *top trace* in each graph is the hip joint angle while the *lower trace* is the knee angle

We can also conclude that the frequency and period of the leg are given as:

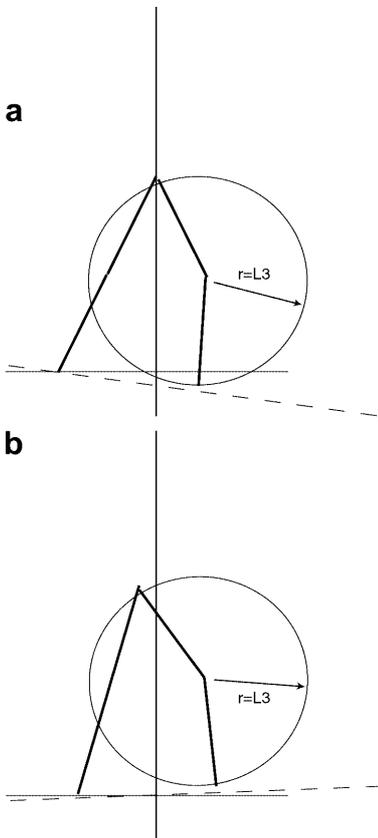
$$\omega = \sqrt{\frac{g}{l}} . \quad (\text{B10})$$

$$t = \sqrt{\frac{l}{g}} \cdot 2\pi \quad (\text{B11})$$

Using rough measures from the robot, and assuming the center of mass is about 2.54 cm below the knee, an estimate for the frequency is about 3 Hz.

Figure B2 shows simulation results. Figure B2a, c, and e are stick figures of the walking trajectory. Figure B2b, d, and f are joint trajectories. Touching upper and lower trajectories indicates that the shank has hit its joint lock and is fully extended. The hip follows a roughly symmetric, triangular trajectory. Figure B2b, d, and f show the rising half of this triangular trajectory as the hip moves forward. Ideally, the knee should lock around the apogee of the swing (i.e., at zero velocity when the hip reverses direction).

In Fig. B2a and b the lower segment hits the limit prematurely. In Fig. B2c and d the hip swings too quickly. In Fig. B2e and f the lower and upper limb are perfectly synchronized so that the knee trajectory



**Fig. B3a,b.** Kinematic scuff model. **a** symmetric gait. Notice that as the shank swings forward, it scuffs the ground. A solution is to only walk on a surface sloped downward. **b** If the robot is tilted backward (asymmetric gait), then the biped can walk on a surface that is level

intercepts the hip trajectory at the hip trajectories' apogee at approximately 0.26 s.

### B.2 Effect of initial conditions

The initial conditions can have a significant effect on the behavior of the system as well. We noticed in empirical experiments that the foot accelerated backward as it unloaded. This is undoubtedly due to the conversion of strain energy to kinetic energy as well as some rotational energy (the entire leg is rotating), as there are no other sources of energy in the shank. This initial condition of the swing phase can significantly affect the timing of the leg. In simulation experiments, we found that by varying the initial leg velocity we could slow down the period of the system by about 30% for reasonable initial velocities.

The overall posture is important as well. Given our model, it is worth noting that a perfectly symmetric gait on a perfectly level surface will cause the swing leg to “scuff” the floor as it swings forward. See Fig. B3 for a typical case.

Notice that, as shown in Fig. B3, this system is very sensitive to the endpoints (which determine the symmetry of the gait) and the slope of the surface that the system walks on. In even the best case, the actual range of slopes over which we can walk, for a given set of endpoints, is rather small.

In the results on the biped (Fig. 10), we note that the robot appears to be “leaning backward” a bit as it runs. This is not surprising given this analysis of the geometry.

To achieve a more robust gait, it is necessary to consider either actuation of the knee and/or modulation of the hip so as to introduce accelerations mid-stride. These accelerations can be generated based on the dynamics equations above.

Equation B9 indicates that if the joint is accelerated *backward*, and the foot is lifted off the ground if the leg is decelerated, the shank accelerates fast toward the knee. Thus, we can, in principle, shape the knee trajectory by actuating the hip to achieve a greater ground clearance. *This requires descending control in synchrony with the limb.*

The analysis of the effect of supraspinal input would require an entire paper in itself.

### References

- Armstrong DM (1986) Supraspinal contributions to the initiation and control of locomotion in the cat. *Progr Neurobiol* 26: 273–361
- Asada H, Slotine JJE (1986) *Robot analysis and control*. Wiley, New York
- Cohen AH, Boothe DL (1999) Sensorimotor interactions during locomotion: principles derived from biological systems. *Autonomous Robots (Special Issue on Biomorphical Robots)* 7(3): 225–238
- Collins SH, Wisse M, Ruina A (2001) A three-dimensional passive-dynamic walking robot with two legs and knees. *Int J Robot Res* 20(7): 607–615

- Dickinson MH, Farley CT, Full RJ, Koehl MA, Kram R, Lehman S (2000) How animals move: an integrative view. *Science* 288(5463): 100–106
- FitzHugh R (1961) Impulses and physiological states in models of nerve membrane. *Biophys J* 1: 445–466
- Garcia M, Chatterjee A, Ruina A, Coleman M (1998a) The simplest walking model: stability, complexity, and scaling. *ASME J Biomech Eng* 120(2): 281–288
- Garcia M, Ruina A, Chatterjee A, Colemann M (1998b) The simplest walking model: stability, complexity, and scaling. *ASME J Biomech Eng* 120(2): 281–288
- Grillner S (1981) Control of locomotion in bipeds, tetrapods, and fish. In: Brooks VB (ed) *Handbook of physiology: the nervous system*, vol 2. American Physiological Society, Waverly Press, pp 1179–1236
- Hale J, Koçak K (1991) *Dynamics and Bifurcations*. Springer, New York
- Hille B (1984) *Ionic channels of excitable membranes*. Sinauer, Sunderland
- Kandel ER, Schwartz JH, Jessell TM (1991) *Principles of neural science*. Elsevier, New York
- Kimura H, Akiyama S, Sakurama K (1999) Realization of dynamic walking and running of the quadruped using neural oscillator. *Autonomous Robots* 7(3)
- Kimura H, Fukuoka Y, Hada Y, Takase K (2001a) 3D adaptive dynamic walking of a quadruped robot by using neural system model. In: *Proceedings of the 4th international Conference on climbing and walking robots (CLAWAR2001)*, Karlsruhe
- Kimura H, Fukuoka Y, Konaga K, Hada Y, Takase K (2001b) Towards 3D adaptive dynamic walking of a quadruped robot on irregular terrain by using neural system model. In: *IEEE&RSJ IROS 2001*, Hawaii
- Nagumo J, Arimoto S, Yoshizawa S (1962) An active pulse transmission line simulating a nerve axon. *Proc IRE* 50: 2061–2070
- Patel G, Holleman J, DeWeerth S (1998) Analog VLSI model of intersegmental coordination with nearest-neighbor coupling. In: Jordan M, Kearns M, Solla S (eds) *Advances in neural information processing*, vol 10. Morgan-Kaufman, Cambridge, MA, pp 719–725
- Pratt J, Pratt G (1998) Exploiting natural dynamics in the control of a planar bipedal walking robot. In: 36th annual Allerton conference on communication, control, and computing, Monticello, IL
- Pratt J, Pratt G (1999) Exploiting natural dynamics in the control of a 3D bipedal walking simulation. In: *Proceedings of the 2nd international conference on climbing and walking robots (CLAWAR99)*, Portsmouth, UK
- Ryckebusch S, Wehr M, Laurent G (1994) Distinct rhythmic locomotory patterns can be generated by a simple adaptive neural circuit: biology, simulation and vlsi implementation. *J Comp Neurosci* 1: 339–358
- Simoni MF, DeWeerth SP (1998) Adaptation in an aVLSI model of a neuron. In: 1998 IEEE international symposium on circuits and systems
- Still S (2000) Walking gait control for four-legged robots – bio-inspired technology for robots. Dept. of Physics, Swiss Federal Institute of Technology
- Still S, Schölkopf B, Hepp K, Douglas R (2000) Four-legged walking gait control using a neuromorphic chip interfaced to a support vector learning algorithm. *Advances in neural information processing systems*. MIT Press, Cambridge, MA
- Taga G (1995a) A model of the neuro-musculo-skeletal system for human locomotion. I. Emergence of basic gait. *Biol Cybern* 73: 97–111
- Taga G (1995b) A model of the neuro-musculo-skeletal system for human locomotion. II. Real-time adaptability under various constraints. *Biol Cybern* 73: 113–121
- Taga G, Yamaguchi Y, Shimizu H (1991) Self-organized control of bipedal locomotion by neural oscillators in unpredictable environment. *Biol Cybern* 65: 147–159