

An Astrocyte-Modulated Neuromorphic Central Pattern Generator for Hexapod Robot Locomotion on Intel's Loihi

Ioannis Polykretis
Computational Brain Lab
Rutgers University

Guangzhi Tang
Computational Brain Lab
Rutgers University

Konstantinos P. Michmizos
Computational Brain Lab
Rutgers University
konstantinos.michmizos@cs.rutgers.edu

ABSTRACT

Locomotion is a crucial challenge for legged robots that is addressed “effortlessly” by biological networks abundant in nature, named central pattern generators (CPG). The multitude of CPG network models that have so far become biomimetic robotic controllers is not applicable to the emerging neuromorphic hardware, depriving mobile robots of a robust walking mechanism that would result in inherently energy-efficient systems. Here, we propose a brain-morphic CPG controller based on a comprehensive spiking neural-astrocytic network that generates two gait patterns for a hexapod robot. Building on the recently identified astrocytic mechanisms for neuromodulation, our proposed CPG architecture is seamlessly integrated into Intel's Loihi neuromorphic chip by leveraging a real-time interaction framework between the chip and the robotic operating system (ROS) environment, that we also propose. Here, we demonstrate that a Loihi-run CPG can be used to control a walking robot with robustness to sensory noise and varying speed profiles. Our results pave the way for scaling this and other approaches towards Loihi-controlled locomotion in autonomous mobile robots.

CCS CONCEPTS

• **Computing methodologies** → **Simulation evaluation**; • **Applied computing** → **Biological networks**; **Systems biology**.

KEYWORDS

Central Pattern Generators, Astrocytes, Hexapod robot, Loihi, Neuromorphic Controller, Locomotion

ACM Reference Format:

Ioannis Polykretis, Guangzhi Tang, and Konstantinos P. Michmizos. 2020. An Astrocyte-Modulated Neuromorphic Central Pattern Generator for Hexapod Robot Locomotion on Intel's Loihi. In *International Conference on Neuromorphic Systems 2020 (ICONS 2020)*, July 28–30, 2020, Oak Ridge, TN, USA. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3407197.3407205>

*This work is supported by Intel's NRC grant award. IP is partially funded by the Onassis foundation scholarship.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ICONS 2020, July 28–30, 2020, Oak Ridge, TN, USA

© 2020 Association for Computing Machinery.

ACM ISBN 978-1-4503-8851-1/20/07...\$15.00

<https://doi.org/10.1145/3407197.3407205>

1 INTRODUCTION

Mobile robots have become successful in various applications including manned and autonomous vehicles [31], rescue missions [11], and planetary or underwater exploration [4, 52]. With wheeled robots being challenged by uneven terrains, an alternative is legged robots that mimic animal locomotion [8, 40]. For controlling their articulated limbs, robots often utilize central pattern generators (CPG), a long-debated control mechanism [6, 25] for a variety of repetitive behaviors that are not restricted to locomotion [21, 22], as it extends from respiration [50] to mastication [26]. Functionally, CPGs are specialized neuronal networks that can generate periodic output in the absence of a periodic input.

Dynamical systems that mimic the oscillatory activity of CPG networks are commonly used to control robotic locomotion as a repetitive behavior [8, 24, 38]. Multiple CPGs are combined to endorse robots with a remarkable set of gait patterns [3, 5], motion speed variation [9, 49] and terrain adaptation [40]. Although such controllers are typically validated based on accuracy, robustness and velocity of the movement, mobile robots deployed in real-world environments have also been increasingly challenged by their power consumption.

Large-scale neuromorphic processors emerged with a central promise to alleviate the energy limitations in mobile robots [12, 19, 28, 41]. Indeed, it was recently demonstrated how brain-dictated spiking neural networks (SNN) developed on Intel's Loihi, can greatly improve power consumption of wheeled robots, while being as accurate as state-of-the-art methods [46, 48]. For legged robots, integrating CPGs into such chips requires a bottom-up rethinking of the conventional algorithms, since the mathematical formalism governing the oscillators is not translatable to non-Von Neumann architectures.

Inspired by the underlying neuronal connectome [5, 27], SNNs are developed for controlling multiple gait patterns [3, 39, 45] and adjusting the speed of mobile robots [8]. This body of prior work spurred a recent interest on neuromorphic implementations of CPGs [16, 23, 33]. However, the application of SNN-controlled CPGs in real-world tasks is staggered by their intrinsic limitations. First, SNN-based CPG controllers mostly operate in open-loop, which inhibits their ability to change their behavior in response to their environment [15, 23] (but see also [44]). Most of them are primed to drive a single pattern and have rather limited abilities in switching between distinct behaviors, especially in response to sensory stimuli. Second, even when they include sensory feedback, its spike-based encoding is prone to noise. This is mitigated by engineered principles that often deviate from biology. For example, some spike-based CPGs encode continuous ranges of joint movement in limited discrete states [23], which decreases motion smoothness and fine

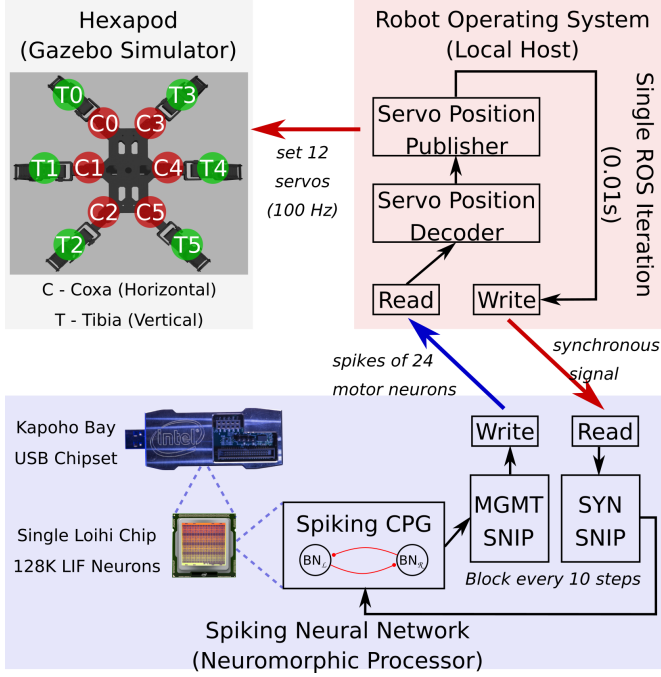


Figure 1: A spiking CPG on Intel’s Loihi neuromorphic processor controlling a hexapod in the Gazebo simulator. Loihi communicated with ROS in real-time using channel interfaces between the local host and the USB chipset. ROS decoded the spikes from 24 motor neurons on Loihi and controlled the positions of 12 servos supporting the hexapod.

limb control. Other solutions artificially impose inter-limb delays that limit flexibility in the robot and interpretability in the model. Interestingly, a universal attribute of all the above approaches is that they regard CPG behavior as an emergent property of exclusively neuronal networks.

Overtuning our assumptions about neural mechanisms, mounting evidence suggests that it is not only neurons that participate in CPGs [1, 7]. Astrocytes, the most abundant type of *non*-neuronal cells, in addition to their effects on synaptic plasticity [13, 35], are now seen as “a primary source for generating neural activity” [32]. Linked to the long suspected role in modulating neural oscillations [17, 34, 37], an astrocytic mechanism that controls CPGs was recently found. Astrocytes are now known to measure the intensity of sensory stimulation and conditionally switch the neuronal firing mode from regular spiking to bursting [30]. With bursting emerging in a periodic fashion, astrocytes may switch neurons to pacemakers which shape patterns of repetitive behavior [27, 30]. Paradoxically, as astrocytes are the underappreciated cells in brain hypotheses, have not been incorporated in possible CPG mechanisms.

In this work, we developed a comprehensive neuronal-astrocytic CPG and implemented it on Loihi, Intel’s neuromorphic chip, to control the locomotion of a hexapod robot. The astrocytic component i) modulated the bursting activity of the CPG neurons, ii) modified the behavior of the network by acting as a switch that turned the CPG on and off, and iii) filtered out the low-frequency input activity associated with sensory noise. Here, we also introduce a multi-spike joint control that results in a fine speed control. To

achieve these results, we also developed the first interaction framework between Loihi and the robot operating system (ROS) that allows real-time control of robots. Overall, supported by the inherent energy-efficiency of neuromorphic approaches, our work aims to become one of the first hints that robotics will soon be, if they are not already are, the sweet spot for neuromorphic computing.

2 METHODS

We took a bottom-up approach to design the CPG controller for the robotic hexapod. At the cellular level, we devised a bursting neuron by expanding the neuron model compartments that Loihi currently supports. We then utilized the neuron’s bursting activity as a pacemaker, to control the flexion/extension of the joints. By connecting a number of such pacemaker neurons together, we created the CPG network, which drove the robot’s walking pattern. To incorporate real-time control of the robotic locomotion, we designed an interactive framework between our Loihi-run CPG and the ROS environment. These steps are presented in more detail below.

2.1 Design of a Bursting Neuron on Loihi

Neurons that exhibited bursting activity were the building block of our spiking CPG network. However, spiking compartments on Loihi only support the leaky integrate-and-fire (LIF) neuronal model without any bursting dynamics, defined by the following equations:

$$u_i(t) = u_i(t-1) \cdot dec_u + \sum_j w_{ij} \cdot s_j \quad (1)$$

$$v_i(t) = v_i(t-1) \cdot dec_v + u_i(t) \quad (2)$$

where t is the Loihi step, u_i is the current, v_i is the voltage, dec_u is the current decay, dec_v is the voltage decay and w_{ij} are the connection weights between the presynaptic compartment j and the postsynaptic compartment i .

We took advantage of Loihi’s fully customizable multi-compartmental neurons to introduce bursting dynamics to our neurons. Specifically, Loihi neurons are defined as binary trees, where either the spikes or the voltage of the child node are passed to the parent node. When a child node passes its voltage to a parent node, the voltage equation of the parent becomes:

$$v_i(t) = v_i(t-1) \cdot dec_v + u_i(t) + v_{ch}(t), \quad (3)$$

where $v_{ch}(t)$ is the voltage of the compartment that is the child node to the parent compartment i . A child node can also pass its spikes to its parent node, which can be used as binary control signals. Another Loihi’s feature that was crucial for our CPG implementation was its support of non-spiking compartments that had no voltage reset. We used those non-spiking compartments to inform their parent nodes about a voltage in a child node exceeding a threshold, without transmitting spikes. This was important to build a bursting neuron on the neuromorphic chip, as described below.

This allowed us to design a conditionally bursting neuron as a multi-compartment neuron on Loihi (Fig. 2A), as described below. The neuron’s input drove two non-spiking compartments: a) the input compartment (Inp) and b) the astrocytic membrane compartment (AM) that both integrated the stimulating input current. The

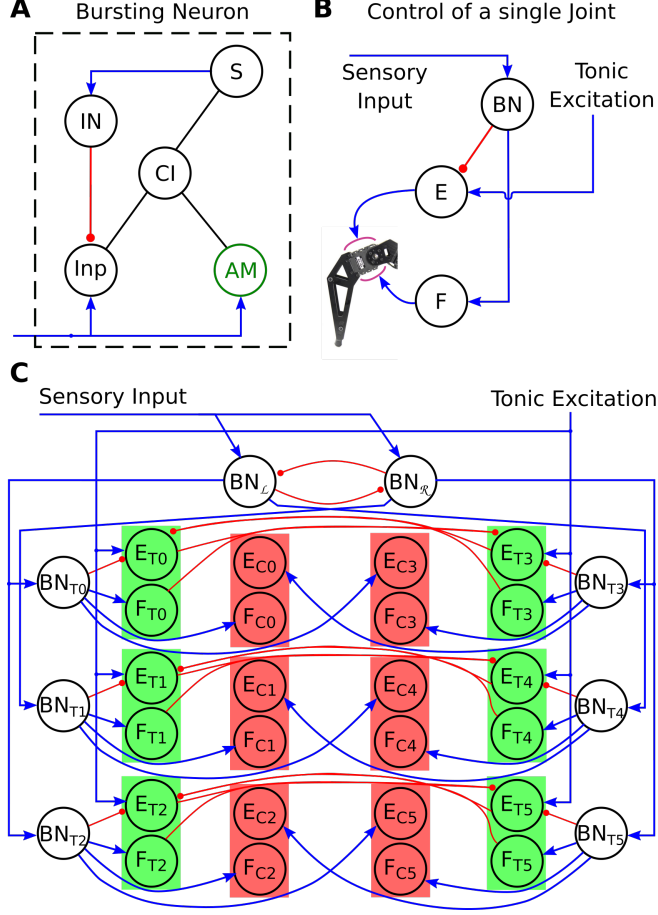


Figure 2: Bottom-up architecture of the neuronal-astrocytic CPG network. A. Compartmental model of an astrocyte-dependent, conditionally bursting neuron on Loihi; B. Network controlling a single tibia joint of the robot; C. CPG architecture for the robot's locomotion. Dots denote connections. Blue arrows with arrowheads represent excitatory connections and red arrows with dotted heads show inhibitory connections.

Conditional Integrator (CI) was Inp's and AM's non-spiking parent node. Inp passed its voltage to CI, while AM passed the information about exceeding its voltage threshold, which is the equivalent of spiking for non-spiking compartments. By using the Loihi's "PASS" operation to process CI's inputs, CI integrated the voltage from Inp only when AM's voltage was above threshold. This allowed the neuron to change its activity based on the sensory input only when the input was prominent enough to depolarize AM above its threshold, introducing a condition for the neural bursting. CI gave the bursting neuron's membrane voltage. As this Loihi compartment was also non-spiking, our neuron model was lacking two crucial mechanisms: (i) spiking bursts and (ii) resetting after the bursts. To address the first issue, we added the somatic compartment (S) as the parent of CI, to receive from CI the information on the voltage exceeding the respective threshold. We used the Loihi's "OR" operation for S to spike whenever CI's voltage was above its

threshold. For addressing the second issue (voltage reset), an inhibitory compartment (IN) was added. When IN fired, it decreased Inp's voltage and, consequently, reset the neuron to its resting state. By modifying the number of spikes from S that were necessary to make the IN compartment spike and reset CI, we obtained a bursting neuron with adjustable burst duration.

2.2 Single Joint Control using Bursting Neurons

We employed the bursting neuron as the building block for controlling each of the robotic joints. Specifically, the periodic bursting was used as a pacemaker for the repetitive flexion/extension of the joints. For a single tibia joint (Fig. 2B), we connected a bursting neuron to a pair of flexor-extensor motoneurons (FMN/EMN). We used the sensory input to drive the bursting neuron and this, in turn, excited the FMN and inhibited the EMN to form the flexion/extension alternation. We drove the EMN with tonic excitation, which normally comes from the higher brain centers [43]. The spikes of the EMN/FMN were then used to drive the servomotors and move the joint. To do so, we decoded each spike of the FMN/EMNs to an increment of the joint's angle, as described by the following equation:

$$\theta \xleftarrow{\text{spike}} \theta + \delta\theta, \quad \delta\theta = \frac{\Delta\theta_{\max}}{T_j}, \quad (4)$$

where θ is the joint's angle, $\delta\theta$ is its increment, $\Delta\theta_{\max}$ is its maximum range, and T_j is the tolerance of the joint to incoming spikes, i.e. the maximum number of spikes of a FMN/EMN that move the joint to its full range.

2.3 Spiking CPG for locomotion

To control the robot's locomotion, we designed a CPG network, consisting of simple joint controllers (Fig. 2C). To achieve a tripod gait, in which at least three of the robot's legs were always in contact with the ground for stability, we grouped the robot's legs into two triplets, the right and the left. The right triplet comprised of the front and hind leg of the robot's right side and the middle leg of the robot's left side. The left triplet consisted of the remaining three legs. Each leg was controlled by two joints (Fig. 1), the tibia joint which lifted and lowered the leg, and the coxa joint which moved the leg back and forth.

First, we needed to generate the alternating movement of the two triplets, referred to as the triplet cycle. For this, we connected two mutually inhibiting bursting neurons (BN_L and BN_R), which fired in an alternating fashion and each of them controlled the movement of one triplet. Hence, the two triplets never moved simultaneously, which stabilized the robot on the ground. Preserving this alternating motion required the full cycle of a single leg (lifting-forward-lowering-backward) to be completed in half the duration of the triplet cycle. For this, we connected BN_L and BN_R to the joints of the respective limbs through an additional layer of bursting neurons, the tibia bursting neurons (BN_{T_i}). We designed the BN_{T_i} to have a bursting frequency that is double the frequency of the triplet cycle. As a result, the BN_{T_i} completed one full leg cycle before BN_L and BN_R alternated.

To mobilize a single leg, we first connected the BN_{T_i} to the tibia FMN/EMN (F_{T_i}/E_{T_i}), as described in section 2.2 (Fig. 2B). This

allowed us to control the flexion/extension of these joints and, consequently the lifting/lowering of the legs. Tonic excitation drove the tibia EMNs (E_{T_i}) to reset the joints after the flexion. The back and forth leg movement was accomplished through the control of the coxa joints of the leg. The forward movement of a leg by its coxa flexion coincided temporally with both, the lifting of the same leg by its tibia flexion, and the backward movement of the contralateral leg by its coxa extension. Hence, we connected the BN_{T_i} to the coxa FMN (F_{C_i}) of the same leg and the coxa EMN of the contralateral leg ($E_{C_{i+3}}$). To prevent the tibia joint extensions from interfering with their opposite leg cycles, even under persistent tonic excitation, we included contralateral inhibitory connections. These connections allowed the tibia FMN/EMN pairs (F_{T_i}/E_{T_i}) to suppress the tibia EMNs on the opposite leg ($E_{T_{i+3}}$).

The spiking activity of the twelve pairs of FMNs/EMNs was decoded to joint angles to control the servomotors, as described in section 2.2.

2.4 Loihi-ROS Interaction for Neuromorphic Implementation

To achieve online robotic control, we developed a faster-than-real-time interaction between the Loihi Kapoho-Bay USB chipset and ROS, and utilized it to control our hexapod robot (Fig. 1). Our SNN CPG controller utilized less than 10% of resources of a single Loihi core (64 compartments and 68 connections). Given that Loihi is an asynchronous system, we expanded our Loihi Astrocytic Module [47] to synchronize the chip with the ROS. This was done by blocking the faster system and forcing it to wait for the slower one. The ROS node iterated 100 times/s to sustain smooth control of the hexapod. For synchronization, T_{epoch} Loihi iterations were executed during a single iteration of the robotic control loop. Each robotic control loop iteration included: i) transferring neuron spikes from Loihi to ROS, ii) decoding the spikes into servo positions (by increasing or decreasing the joint angles by a fixed amount δ until the pre-defined limits - max or min), and iii) setting servo positions through ROS topics. We set T_{epoch} to 10, so that each Loihi iteration represented exactly 1 ms. The “chokepoint” was the communication between Loihi and the local host machine that ran the ROS node. To avoid data transfer in every Loihi iteration, we implemented a Sequential Neural Interacting Process (SNIP) on the x86 LMT chip (within the Loihi chipset) to store the number of spikes for T_{epoch} iterations.

3 RESULTS

To test whether, and to what extent, the expected behavior can emerge out of the CPG network, we observed the kinematics of the robot’s joints and the robot’s motion with respect to the sensory input. Here, we first report the internal dynamics of the bursting neuron, and then we relate this cellular behavior to that of the CPG network in controlling the robot’s gait cycle. We then examine the relation between the speed profile of the robot and the sensory input’s frequency. Finally, we showcase the real-time interaction of Loihi with ROS and our hexapod robot.

3.1 A Bursting Neuron Responding Only to Meaningful Stimuli

The activity within the bursting neuron’s compartments is shown in Fig. 3. In this example, we stimulated the neuron with a 4s Poisson spike train. For the first 500 ms, the firing frequency was kept to its baseline 5Hz, representing sensory noise. Afterwards, the firing frequency increased to 40Hz, representing sensory neuron activity that encodes input stimulus. Using the mechanisms described in section 2.1, Inp transformed its stimulation current to voltage (Fig. 3, bottom-left panel), and AM integrated the input to its voltage (Fig. 3, bottom-right panel). This allowed for the sensory noise to be filtered out, as it was only when the frequency increased to 40 Hz and the voltage of AM exceeded its threshold that CI started integrating the voltage of its child node, Inp, into its own (Fig. 3, middle-right panel). The S compartment received from CI the information about whether CI’s voltage exceeded its threshold. The S compartment continued spiking (Fig. 3, top-right panel) for as long as CI’s voltage was above its threshold. This created the spike burst activity. In turn, the spikes of the S compartment stimulated IN, which integrated them in its voltage. The spikes of IN (Fig. 3, top-left panel) generated inhibitory currents that closed the loop by

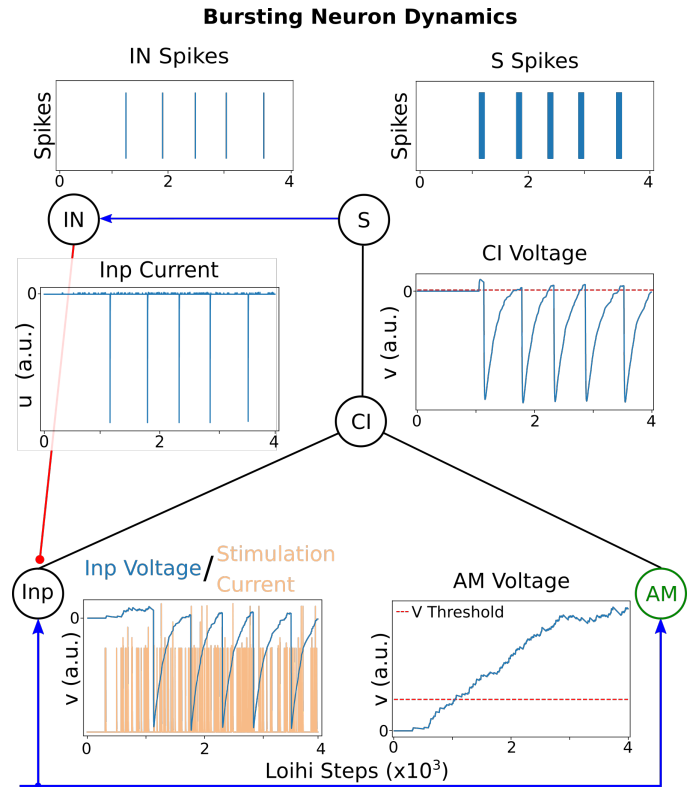


Figure 3: Activity of the bursting neuron’s compartments. Inp voltage is a direct transformation of the input current to voltage. AM is the integral of the input current. CI’s voltage is the bursting neuron’s membrane voltage; when it exceeds the threshold (red dotted) the somatic compartment S spikes. These somatic spikes drive IN to spike and stimulate CI to reset the neuron.

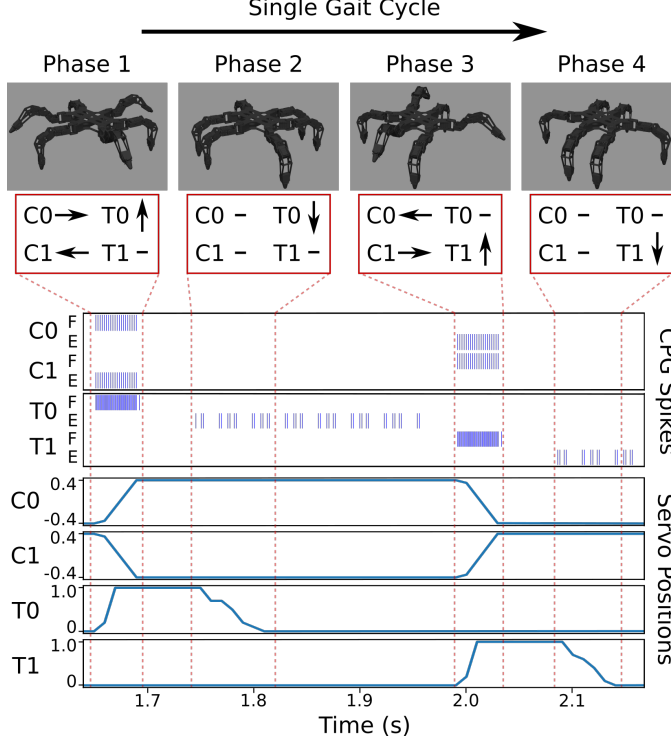


Figure 4: Example of a single hexapod gait cycle controlled by the spiking CPG. Raster plot for motor-neuron spiking activity in the CPG resulted in sequential change of the servos' position, for both flexion and extension.

being fed into Inp (Fig. 3, middle-left panel). This current decreased Inp 's voltage (Fig. 3, bottom-left panel) and reset the neuron to the resting state. The time needed for CI 's voltage to reach the threshold again, after it was reset, was the inactive period between spike burst. With CI integrating the voltage of Inp , which is already an integral of the stimulation current, the voltage of the bursting neuron was a very smooth estimate of the unstable input signal. This allowed for consistent bursting activity, which adapted to the input frequency, but was only slightly affected by the input noise.

3.2 Gait Cycle

To achieve the regular walking pattern of the robot, a single gait cycle (Fig. 4) was repeated, as described below. During the gait cycle, the BN of the tibia (BN_{T0}) drove both the coxa FMN of the same leg (F_{C0}) and the coxa EMN of the opposite leg (E_{C1}). In this way, a leg was lifted by its tibia joint, while its coxa joint moved it forward. Simultaneously, the coxa joint of the opposite leg moved it backwards (Phase 1). Then, the extension of the tibia joint lowered the leg to the ground (Phase 2). This concluded half of the gait cycle and was followed up by the other half (Phases 3 and 4). During this next half, the coxa joints of the legs that had previously moved forward ($C0$) now moved them backwards and vice versa ($C1$). Likewise, the legs that were previously in contact with the ground were now lifted by their respective tibia joints ($T1$).

3.3 Control of the Robot's Speed Using the Input frequency

The dependence of the robot's speed on the input frequency emerged out of our CPG design. We used random Poisson spike trains with a mean firing rate to test the ability of our CPG network to generate a periodic output given a non-periodic input. We chose 4 representative input frequencies to stimulate the CPG network and tracked the robot's speed. Specifically, we first drove our CPG network with a 5Hz Poisson train that simulated sensory noise, as in the case of the single BN (section 3.1). As expected, this baseline sensory activity was insufficient to depolarize the AM compartment of the BN_L and BN_R , and consequently, could not generate any robot motion. We then increased the sensory input frequency beyond the noise level. When the firing activity exceeded the threshold, the AM compartment started getting depolarized and the CPG network generated the motion pattern based on the gait cycle described in section 3.2. In Fig. 5, we show the speed of the walking robot (second panel) in response to its input frequency (top panel) which increased from 5Hz to 40Hz (for $t=[0,20]$ s), 160Hz (for $t=[20,40]$ s) and 320Hz (for $t=[40,60]$ s). This resulted in a smooth increase of the robot's average speed (Fig. 5, second panel; red dotted line). The spiking frequency of the joints' FMNs/EMNs successfully adapted

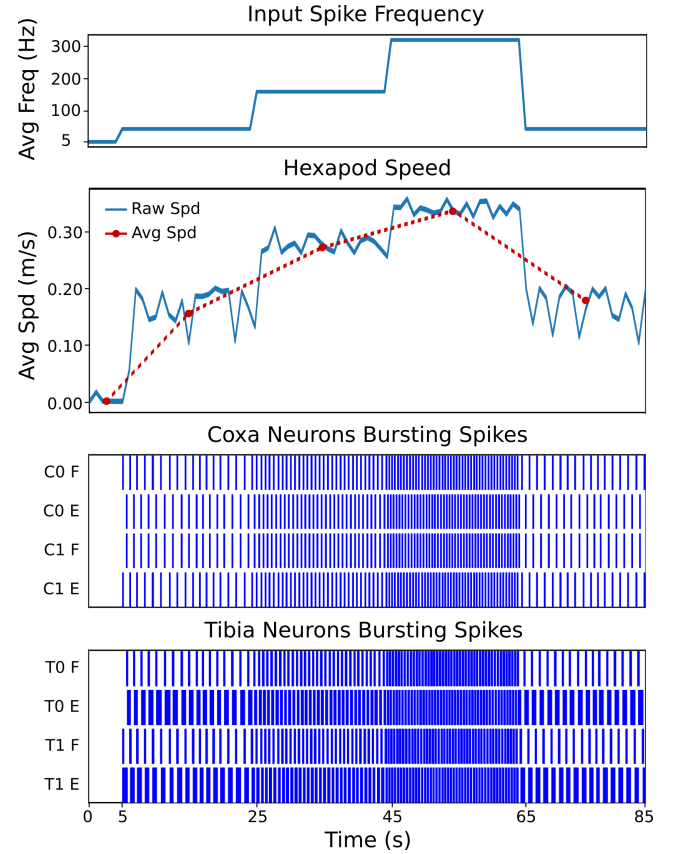


Figure 5: Acceleration and deceleration of the hexapod by changing the input firing activity to the CPG every 20 seconds in a single experiment.

to the input frequency, as shown in the raster plots (Fig. 5, two bottom panels.)

3.4 Speed Profile in Response to Stimulus Frequency

We tracked the robot’s speed as a function of input frequency over the whole functional range of our CPG network. For this, we stimulated the CPG network with 60s Poisson trains, whose frequency increased with a step of 5Hz below 20Hz and with a step of 20Hz beyond that. The network was stimulated 10 times for each input frequency, and we recorded the robot’s average speed for each such trial. Then we computed the mean of these average speeds to obtain the average speed profile, shown in Fig. 6. This plot describes the effective range of the movement speed as a function of the input sensory frequency.

3.5 Loihi-ROS Interaction for a Real-Time Control of Robots

Our framework exhibited a faster-than-real-time interaction between ROS and Loihi. To verify this, we measured the real-time factor (RTF) defined as follows:

$$RTF = \frac{t_{exec}}{t_{wc}} \quad (5)$$

where t_{exec} is the execution time of a single iteration and t_{wc} is the wall-clock time, which for a ROS node iterating at 100 Hz is always equal to 0.01s. When averaged over all the iterations, we obtained an RTF of 1.6, which is far above the minimum requirement of a real-time system (Fig. 7). Consequently, our ROS-Loihi interaction framework can indeed be used to control a real-time hexapod robotic system. In fact, since this framework is not hardware specific, it can be used to control other robotic systems as well.

Moreover, since T_{epoch} Loihi iterations were executed during a single iteration of the robotic control loop, this reduced the communication cost between Loihi and the local host machine by a factor of T_{epoch} .

4 DISCUSSION

In this paper, we presented a comprehensive neuronal-astrocytic CPG network and its integration into Intel’s Loihi neuromorphic processor, to control the locomotion of a hexapod robot. We also proposed a framework for the real-time interaction between the ROS environment and Loihi. This work aims to translate the advantages, including that of simplicity, carried by traditional oscillator-based CPG models [5, 40] into the emerging neuromorphic chips that promise significant energy-efficiency, robustness and versatility in solving real-world robotic problems. Unlike conventional approaches that use dynamical systems as coupled oscillators [5, 8, 24, 40], our proposed neuromorphic control algorithm generated the periodic movement pattern by virtue of the modeled brain cell’s properties, as well as the interaction between neuronal and astrocytic activity.

At the cellular level, we introduced the first conditionally-bursting neuron on a neuromorphic hardware drawing inspiration from

biological findings in insect locomotion [14, 21, 27]. Current neuromorphic chips provide an efficient hardware implementation of the LIF neuron model only, and, therefore, lack an inherent mechanism for emulating the bursting neuronal activity that is present in the brain [27, 30]. While LIF neurons are far from ideal when driving periodic behavior, our proposed CPG model is a perfect fit for a behavior that repeated itself every few hundred milliseconds – a time window that coincides with the behavioral time scale. Such long periodic interchanges between active and inactive periods are primed for tasks where robustness to sensory neuronal noise matters. The proposed bursting neurons encode the oscillatory activity in a bandpass manner (using their inter- and intra-burst bands), which makes them robust to noise present in the full operating spectrum. Our approach also draws from the recent impressive empirical evidence of the neuromodulatory abilities that astrocytes exhibit [30]. Indeed, the astrocyte added a computational layer to the brain-morphic CPG, which was orthogonal to the neuronal layer, and enabled switching of the network’s behavior depending on the input stimulus. The astrocytic component decreased the sensitivity of our network to low-frequency sensory activity, which represents noise in the adopted frequency-encoding regime. In contrast to the fast spiking noise-prone neurons, astrocytes can use their much larger temporal and spatial integration scales [34, 36] to filter out noise transients from meaningful stimuli. We expect this approach to become a significant component in our and others’ efforts towards robust spiking neural networks. In that sense, this work captures the newly identified roles of astrocytes and solidify them as computational primitives applicable to spiking neuronal-astrocytic networks integrated to large-scale neuromorphic chips.

At the network level, the CPG connectome was inspired, and partially constrained, by its biological counterpart [14, 21, 27]. This significantly decreased the size of our proposed network, which consisted of only 64 Loihi compartments (less than 0.5% of the 15K-compartment network in [48] that consumed just 9mW). Although we did not measure the power consumption of our network,

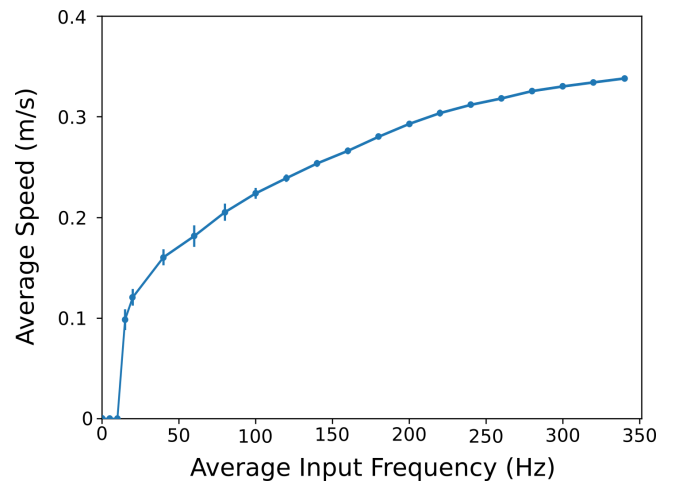


Figure 6: Average speed of the hexapod for different input frequencies. Each data point is the mean of 10 average speed recordings obtained over 60s long experiments. The vertical line shows the speed range within the trials.

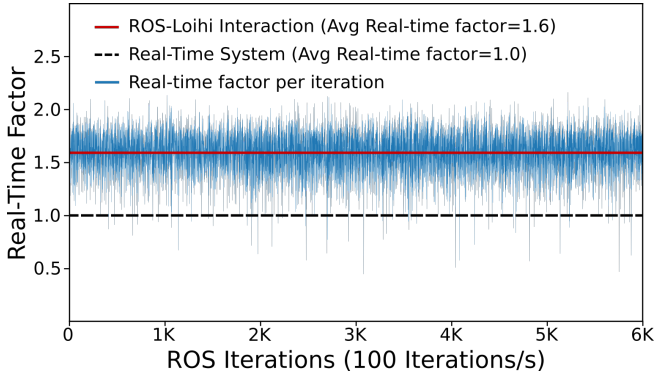


Figure 7: The real-time factor for the spiking CPG system based on time recordings from each ROS iteration.

the asynchronous computation on Loihi that consumes power proportionally to the utilized resources make it reasonable to expect significant power efficiency. The network’s architecture also ensured a range of linear dependence, in which the period of the gait cycle was inversely proportional to the frequency of the sensory input. Contrary to typical neuron models in conventional multi-layer networks that can be optimized to perform complex computational tasks, spiking neuron models have a non-differentiable output (their all-or-none firing) and therefore are incompatible with standard gradient-descent supervised learning methods. In the absence of a strong learning algorithm, the main criticism to neuromorphic solutions is that promising preliminary results [48] cannot share the same scaling abilities with the mainstream deep learning approaches. Adding biological constraints to neuromorphic algorithms, such as the ones we introduced here, removes the need for assuming all-to-all initial connectivity for the trainable network. This may translate to further improvements in training efficiency, as it limits learning to a small number of synaptic connections. Recent applications of gradient-descent alternatives to SNNs [18, 42] are also promising but they inherit the main limitations that the conventional neuronal networks have. For example, learning through backpropagation will basically match the network’s input to its output, much regarding and, thereby, structuring the network as a black-box. Bottom-up neuromorphic approaches in developing neural-controlled robots, such as the one proposed here, can give an additional benefit, by serving as test-beds to inform brain scientists on how the neural system could be structured to function properly. In that sense, this work contributes to a novel direction, where “machine behavior” in general, and robotic function in particular, can emerge naturally from a basic *a-priori* knowledge of its controller’s structure.

The coupling between the robot’s speed and the sensory input frequency is crucial in our CPG design as it may give rise to a number of distinct control strategies that depend on the sensory input. Consider, for example, a visual stimulus being the input to the proposed CPG; The representation of the distance from a desirable target using the input frequency (rate encoding) would lead to a robotic speed that depends on how close the robot is to the perceived target: A large distance would be encoded as high input frequency and cause the robot to move fast towards its target; As the robot approaches the target, the speed decreases. In contrast, encoding

the distance from a dangerous object into the input frequency could force the robot to move fast away from that object, when being in close proximity to it. Therefore, our proposed CPG design allows for multiple parallel perception-action loops, that drive the same spiking network and can modify the robot’s speed in response to a number of sensed changes in a varying environment. It is true that the embodiment of spiking neural networks into robots has been rather sparse and most of the current approaches aim to give a proof of concept [2, 29], rather than a whole-behaving robot. While there is definitely value in studying simplified tasks and basic sensory representations [20], there is an ongoing need to propose new architectures capable of naturally handling richer, noisier and more complex scenarios [10]. Our work paves the way for developing novel active sensing neuromorphic solutions by drawing biological principles from neuroscience and translating them to computational primitives.

When it comes to innervating robots, there is no scarcity of communication frameworks between ROS and most of the popular neuron simulators [51]. Currently, however, there is no coupling between the most recently introduced neuromorphic chip, Loihi, and ROS. To propose an end-to-end solution, we had to bridge the gap between our on-chip CPG network and the robot’s joints, by proposing a real-time intercommunication framework. For a reasonable communication to take place, the control commands need to be generated before they are required by the robot. Our framework established a faster-than-real-time intercommunication between ROS and Loihi that can further incorporate more computationally intensive Loihi networks that would communicate with ROS and add up to the functionality of the robot.

Overall, an autonomous robot should, among others, 1) be robust to a noisy neural representation, 2) adapt to a fast changing environment, and 3) learn with no or limited supervision or reinforcement. The emergence of neuromorphic computing calls for a bottom-up rethinking of the real-time control algorithms for robots, that can seamlessly integrate into non-Von Neumann hardware, promising unparalleled energy-efficiency and a robust yet versatile alternative to the brittle inference-based AI solutions. This paper brings us closer to realize this promise, focusing on the CPG, a crucial mechanism for legged robots.

5 CONCLUSION

Effective as they may have become, robots still cannot duplicate a range of human behaviors, such as responding to changing environments using error-prone sensors. This work draws from the structure of the brain areas associated with the targeted behavior as well as the recently identified principles for information processing in the brain and introduces a neuromorphic approach in designing robotic controllers. The reported robotic behavior was achieved by emulating the connectome and the underlying types of brain cells, not through learning. This suggests that the co-development of brain-morphic algorithms and neuromorphic hardware for solving scalable robotic problems is a direction worth pursuing.

REFERENCES

- [1] David Acton and Gareth B Miles. 2015. Stimulation of glia reveals modulation of mammalian spinal motor networks by adenosine. *PLoS one* 10, 8 (2015).

- [2] Eric A Antonelo, Benjamin Schrauwen, Xavier Dutoit, Dirk Stroobandt, and Marnix Nuttin. [n.d.]. Event detection and localization in mobile robot navigation using reservoir computing. In *International Conference on Artificial Neural Networks*. Springer, 660–669.
- [3] Jose Hugo Barron-Zambrano, Cesar Torres-Huitzil, and Bernard Girau. 2010. Hardware implementation of a CPG-based locomotion control for quadruped robots. In *International Conference on Artificial Neural Networks*. Springer, 276–285.
- [4] Robert Bogue. 2012. Robots for space exploration. *Industrial Robot: An International Journal* 39, 4 (2012), 323–328.
- [5] Pietro-Luciano Buono and Martin Golubitsky. 2001. Models of central pattern generators for quadruped locomotion I. Primary gaits. *Journal of mathematical biology* 42, 4 (2001), 291–326.
- [6] B Bussel, A Roby-Brami, O Rémy Nériss, and A Yakovlev. 1996. Evidence for a spinal stepping generator in man. *Spinal Cord* 34, 2 (1996), 91–92.
- [7] Steven Condamine, Raphael Lavoie, Dorly Verdier, and Arlette Kolta. 2018. Functional rhythmic domains defined by astrocytic networks in the trigeminal main sensory nucleus. *Glia* 66, 2 (2018), 311–326.
- [8] Alessandro Crespi and Auke Jan Ijspeert. 2006. AmphiBot II: An amphibious snake robot that crawls and swims using a central pattern generator. In *Proceedings of the 9th international conference on climbing and walking robots (CLAWAR 2006)*. 19–27.
- [9] Alessandro Crespi, Konstantinos Karakasiotis, Andre Guignard, and Auke Jan Ijspeert. 2013. Salamandra robotica II: an amphibious robot to study salamander-like swimming and walking gaits. *IEEE Transactions on Robotics* 29, 2 (2013), 308–320.
- [10] Antoine Cully, Jeff Clune, Danesh Tarapore, and Jean-Baptiste Mouret. 2015. Robots that can adapt like animals. *Nature* 521, 7553 (2015), 503–507.
- [11] Angela Davids. 2002. Urban search and rescue robots: from tragedy to technology. *IEEE Intelligent Systems* 17, 2 (2002), 81–83.
- [12] Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, et al. 2018. Loihi: A neuromorphic manycore processor with on-chip learning. *IEEE Micro* 38, 1 (2018), 82–99.
- [13] Maurizio De Pittà, Nicolas Brunel, and Andrea Volterra. 2016. Astrocytes: Orchestrating synaptic plasticity? *Neuroscience* 323 (2016), 43–61.
- [14] Fred Delcomyn. 1999. Walking robots and the central and peripheral control of locomotion in insects. *Autonomous Robots* 7, 3 (1999), 259–270.
- [15] Elisa Donati, Federico Corradi, Cesare Stefanini, and Giacomo Indiveri. 2014. A spiking implementation of the lamprey’s Central Pattern Generator in neuromorphic VLSI. In *2014 IEEE Biomedical Circuits and Systems Conference (BioCAS) Proceedings*. IEEE, 512–515.
- [16] Elisa Donati, Fernando Perez-Peña, Chiara Bartolozzi, Giacomo Indiveri, and Elisabetta Chicca. 2018. Open-loop neuromorphic controller implemented on vlsi devices. In *2018 7th IEEE International Conference on Biomedical Robotics and Biomechanics (Biorob)*. IEEE, 827–832.
- [17] Tommaso Fellin, Michael M Halassa, Miho Terunuma, Francesca Succol, Hajime Takano, Marcos Frank, Stephen J Moss, and Philip G Haydon. 2009. Endogenous nonneuronal modulators of synaptic transmission control cortical slow oscillations in vivo. *Proceedings of the National Academy of Sciences* 106, 35 (2009), 15037–15042.
- [18] Matthias O Franz and Hanspeter A Mallot. 2000. Biomimetic robot navigation. *Robotics and autonomous Systems* 30, 1 (2000), 133–153.
- [19] Steve B Furber, Francesco Galluppi, Steve Temple, and Luis A Plana. 2014. The spinnaker project. *Proc. IEEE* 102, 5 (2014), 652–665.
- [20] D Gamez, Andreas K Fjeldland, and E Lazdins. 2012. iSpoke: a spiking neural interface for the iCub robot. *Bioinspiration & biomimetics* 7, 2 (2012), 025008.
- [21] Sten Grillner. 1975. Locomotion in vertebrates: central mechanisms and reflex interaction. *Physiological reviews* 55, 2 (1975), 247–304.
- [22] Pierre A Guertin. 2009. The mammalian central pattern generator for locomotion. *Brain research reviews* 62, 1 (2009), 45–56.
- [23] Daniel Gutierrez-Galan, Juan P Dominguez-Morales, Fernando Perez-Peña, Angel Jimenez-Fernandez, and Alejandro Linares-Barranco. 2019. NeuroPod: a real-time neuromorphic spiking CPG applied to robotics. *Neurocomputing* (2019).
- [24] Auke Jan Ijspeert. 2008. Central pattern generators for locomotion control in animals and robots: a review. *Neural networks* 21, 4 (2008), 642–653.
- [25] LS Illis. 1995. Is there a central pattern generator in man? *Spinal Cord* 33, 5 (1995), 239–240.
- [26] James P Lund and Arlette Kolta. 2006. Generation of the central masticatory pattern and its modification by sensory feedback. *Dysphagia* 21, 3 (2006), 167–174.
- [27] David A McCrea and Ilya A Rybak. 2007. Modeling the mammalian locomotor CPG: insights from mistakes and perturbations. *Progress in brain research* 165 (2007), 235–253.
- [28] Paul A Merolla, John V Arthur, Rodrigo Alvarez-Icaza, Andrew S Cassidy, Jun Sawada, Filipp Akopyan, Bryan L Jackson, Nabil Imam, Chen Guo, Yutaka Nakamura, et al. 2014. A million spiking-neuron integrated circuit with a scalable communication network and interface. *Science* 345, 6197 (2014), 668–673.
- [29] Ben Mitchinson, M Pearson, Anthony G Pipe, and Tony J Prescott. 2011. Biomimetic robots as scientific models: a view from the whisker tip. *Neuromorphic and Brain-Based Robots* (2011), 23–57.
- [30] Philippe Morquette, Dorly Verdier, Aklessio Kadala, James Féthière, Antony G Philippe, Richard Robitaille, and Arlette Kolta. 2015. An astrocyte-dependent mechanism for neuronal rhythmicogenesis. *Nature neuroscience* 18, 6 (2015), 844.
- [31] Kenzo Nonami, Farid Kendoul, Satoshi Suzuki, Wei Wang, and Daisuke Nakazawa. 2010. *Autonomous flying robots: unmanned aerial vehicles and micro aerial vehicles*. Springer Science & Business Media.
- [32] H Rheinallt Parri, Timothy M Gould, and Vincenzo Crunelli. 2001. Spontaneous astrocytic Ca²⁺ oscillations in situ drive NMDAR-mediated neuronal excitation. *Nature neuroscience* 4, 8 (2001), 803–812.
- [33] Fernando Perez-Peña, Arturo Morgado-Estevéz, Alejandro Linares-Barranco, Angel Jimenez-Fernandez, Francisco Gomez-Rodriguez, Gabriel Jimenez-Moreno, and Juan Lopez-Coronado. 2013. Neuro-inspired spike-based motion: from dynamic vision sensor to robot motor open-loop control through spike-VITE. *Sensors* 13, 11 (2013), 15805–15832.
- [34] Ioannis Polykretis, Vladimir Ivanov, and Konstantinos P Michmizos. 2018. A Neural-Astrocytic Network Architecture: Astrocytic calcium waves modulate synchronous neuronal activity. In *Proceedings of the International Conference on Neuromorphic Systems*. 1–8.
- [35] Ioannis E Polykretis, Vladimir A Ivanov, and Konstantinos P Michmizos. 2019. Computational Astrocyte: Astrocytes encode inhibitory activity into the frequency and spatial extent of their calcium elevations. In *2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*. IEEE, 1–4.
- [36] Ioannis E Polykretis, Vladimir A Ivanov, and Konstantinos P Michmizos. 2019. Computational Astrocyte: Astrocytes encode inhibitory activity into the frequency and spatial extent of their calcium elevations. In *2019 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*. IEEE, 1–4.
- [37] Kira E Poskanzer and Rafael Yuste. 2016. Astrocytes regulate cortical state switching in vivo. *Proceedings of the National Academy of Sciences* 113, 19 (2016), E2675–E2684.
- [38] Ludovic Righetti and Auke Jan Ijspeert. 2006. Programmable central pattern generators: an application to biped locomotion control. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. IEEE, 1585–1590.
- [39] Horacio Rostro-Gonzalez, Pedro Alberto Cerna-Garcia, Gerardo Trejo-Caballero, Carlos H Garcia-Capulin, Mario Alberto Ibarra-Manzano, Juan Gabriel Avina-Cervantes, and César Torres-Huitzil. 2015. A CPG system based on spiking neurons for hexapod robot locomotion. *Neurocomputing* 170 (2015), 47–54.
- [40] Guillaume Sartoretti, Samuel Shaw, Katie Lam, Naixin Fan, Matthew Travers, and Howie Choset. 2018. Central pattern generator with inertial feedback for stable locomotion and climbing in unstructured terrain. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1–5.
- [41] Johannes Schemmel, Daniel Brüderle, Andreas Gribbl, Matthias Hock, Karlheinz Meier, and Sebastian Millner. 2010. A wafer-scale neuromorphic hardware system for large-scale neural modeling. In *Circuits and systems (ISCAS), proceedings of 2010 IEEE international symposium on*. IEEE, 1947–1950.
- [42] Sumit Bam Shrestha and Garrick Orchard. [n.d.]. SLAYER: Spike layer error reassignment in time. In *Advances in Neural Information Processing Systems*. 1412–1421.
- [43] SR Soffe and Alan Roberts. 1982. Tonic and phasic synaptic input to spinal cord motoneurons during fictive locomotion in frog embryos. *Journal of Neurophysiology* 48, 6 (1982), 1279–1288.
- [44] Silke Steingrube, Marc Timme, Florentin Wörgötter, and Poramate Manoonpong. 2010. Self-organized adaptation of a simple neural circuit enables complex robot behaviour. *Nature physics* 6, 3 (2010), 224–230.
- [45] Susanne Still, Klaus Hepp, and Rodney J Douglas. 2006. Neuromorphic walking gait control. *IEEE transactions on neural networks* 17, 2 (2006), 496–508.
- [46] Guangzhi Tang and Konstantinos P Michmizos. 2018. Gridbot: An autonomous robot controlled by a Spiking Neural Network mimicking the brain’s navigational system. In *Proceedings of the International Conference on Neuromorphic Systems*. 1–8.
- [47] Guangzhi Tang, Ioannis E Polykretis, Vladimir A Ivanov, Arpit Shah, and Konstantinos P Michmizos. 2019. Introducing astrocytes to a neuromorphic processor: Synchronization, local plasticity and edge of chaos. In *Proceedings of the 7th Annual Neuro-inspired Computational Elements Workshop*. 1–9.
- [48] Guangzhi Tang, Arpit Shah, and Konstantinos P Michmizos. 2019. Spiking neural network on neuromorphic hardware for energy-efficient unidimensional SLAM. *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2019).
- [49] Nicolas Van der Noot, Auke J Ijspeert, and Renaud Ronsse. 2015. Biped gait controller for large speed variations, combining reflexes and a central pattern generator in a neuromuscular model. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 6267–6274.
- [50] C Von Euler. 1983. On the central pattern generator for the basic breathing rhythmicity. *Journal of Applied Physiology* 55, 6 (1983), 1647–1659.

- [51] Philipp Weidel, Mikael Djurfeldt, Renato C Duarte, and Abigail Morrison. 2016. Closed loop interactions between spiking neural network and robotic simulators based on MUSIC and ROS. *Frontiers in neuroinformatics* 10 (2016), 31.
- [52] Louis Whitcomb, Dana R Yoerger, Hanumant Singh, and Jonathan Howland. 2000. Advances in underwater robot vehicles for deep ocean exploration: Navigation, control, and survey operations. In *Robotics Research*. Springer, 439–448.