Neuromorphic Building Blocks for Locomotion Pattern Generation*

Vaibhav Gandhi
Faculty of Science and Technology
Middlesex University London
London, United Kingdom
0000-0003-1121-7419

Zhijun Yang
Central Institute for Technology Innovation
Jack Technology Co.Ltd
Taizhou City, China
0000-0003-2615-4297

***Abstract*—The central pattern generators network (CPGs) plays an important role in motion control which enables creatures to interact with the world. A novel neuromorphic circuit model presented in this work can be used as the simple building blocks for prescribing more complex, coordinated motor patterns. The circuit demonstrates its capability in generating the activity frequency and duty cycle, independently adjustable by a small set of model parameters. The simulation outcomes also show that the circuit can implement the parallel and distributed algorithms for building the artificial CPGs to drive motors.**

***Index Terms*—central pattern generation, neuromorphic circuit, parallel algorithms, locomotion, simulation**

I. Introduction

There are evidences showing that walking with adaptive gait patterns is the outcome of interaction between the innate neural mechanisms and postnatal maturation and development [1] [2]. A neonatal vertebrate, including even the precocial animals, experiences an inept process of learning walking before becoming fully adaptive to a complex terrain [3]. This learning process starts with reflexes reflecting involuntary responses to stimuli. It may involve a complex sequence of activities for sensorimotor integration, synchronization and coordination of cortical neuron populations and muscles. After the relevant cortical regions are well acquainted with the external world, the animals are considered as trained and represent the most capable walking machine in nature.

Many theoretical and experimental approaches have been proposed intending to decipher the mechanisms underlying the control of locomotion while presenting its artificial intelligence counterpart [4] [5]. The central pattern generators network (CPGs), a neural circuit located at the lower thoracic and lumbar areas of the spinal cord [6], has been identified responsible for generating biological rhythmic patterns without involvement of higher nervous systems as well as sensory inputs [7]. CPGs have been studied extensively in its neurophysiological and anatomical rationale (referring to [8] for a review), its neuromorphic realization [9]–[11], and applications in robotics [12]. Two research frameworks, i.e., the closed- and open-loop schemes, are commonly used. In the closed-loop scheme, as a part of the sensorimotor system, CPGs are the actuators driven either directly by sensors or indirectly via the nervous systems like the brainstem or basal ganglia. However, in the open-loop scheme, the CPGs behaviors are explored by using neuronal receptor agonists and antagonists or mechanical operations to cut the links of the CPGs from other neural mechanisms and sensory inputs [13]. Thereby, the endogenous neural activities can be investigated without interference of outliers external to CPGs property.

Based on the parallel and distributed algorithms, namely, the scheduling by edge reversal (SER) and scheduling by multiple edges reversal (SMER) [14], a generalized artificial CPGs architecture which is able to generate arbitrary rhythmic gait patterns for legged locomotion is proposed [15] [16]. The architecture is based on a set of oscillatory building blocks (OBBs) which can be used to build an instance of CPGs of arbitrary topology for a specific, tailor-designed rhythmic pattern. A set of these CPGs instances can then be connected to achieve a set of different patterns and transitions between these patterns [13]. The OBBs-based CPGs instance has been implemented in mixed-signal very large-scale integrated circuit (VLSI) in 0.35um CMOS process [11]. Despite the advantages of using VLSI chip technology, e.g., small power consumption and light footprint, the costs of time and resources are not easily affordable, and the required expertise in design is remarkable. Considering that an invertebrate animal may only have few legs with a limited number of degrees of freedom (DoFs) for each leg, a more worthwhile and flexible approach might be to use the discrete electronic components for building a CPGs architecture for arbitrary rhythmic generation and transition. In recognition of the viability of the new implementation of the OBBs-based CPGs, in this work we propose two neuromorphic circuit plans for realizing SER- and SMER-based OBBs, respectively. The circuits are then simulated, and their features briefly discussed.

The description of the work is organized as follows. Section II is a brief introduction to the SER and SMER algorithms, and the creation of instances of OBBs modules corresponding to the algorithm. Section III illustrates the design of the neuromorphic OBBs modules by using out-of-the-shelf tools. Section IV displays the simulation outcomes of the circuits. Finally, Section V concludes the work and suggests the future

* This work was partly funded by a UK EPSRC grant EP/P00542X/1, and a China Jack Technology Co.Ltd. research grant. Correponding authors: V.Gandhi@mdx.ac.uk, zhijuny@hotmail.com.

directions.

II. SER AND OBBS MODULES

A. SER Algorithm

Suppose a node is coupled with other nodes in a mutually exclusive way, i.e., at any time instant only one node in this group of nodes is active because it occupies the resources shared between the other nodes and itself, while the others are idle. When the active node stops firing it releases the resources so that the coupled nodes occupy the resources. Some of these nodes become active if they occupy the shared resources from all of their coupled nodes. This process will repeat indefinitely to form a fixed excitatory-inhibitory periodic cycle, which has potential to be used to mimic the rhythmic patterns of muscle-driven system like the coupled extensor and flexor. A simple, directed graph of the SER dynamics is shown in Fig. 1. The formal theoretic description of the algorithm can be found in [14] [15].

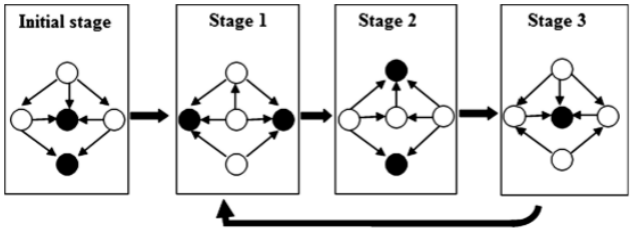

Fig. 1. The graph dynamics of the SER algorithm. The black node represents an active node. The edges with arrows between nodes represent the attachment of the shared resources. After an initial stage a 3-stage periodic cycle is formed in which a node will fire once (reproduced from [16]).

In a more generalized SMER algorithm, unlike the SER algorithm in which any two coupled nodes share only one unit of resources represented by an arrow, any two coupled nodes may share a different number of resources. This sharing of resources is now represented by multiple arrows between two coupled nodes, as shown in Fig. 2.

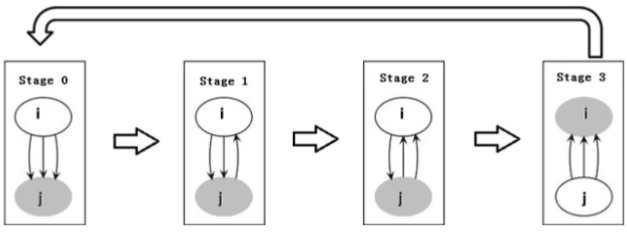

Fig. 2. The graph dynamics of the SMER algorithm. The gray color represents an active node. The nodes i and j have their reversibility ri=3, rj=1, respectively (reproduced from [16].

In SMER, a node has an attribute, namely reversibility. The reversibility indicates the number of resources a node must have in order to be active, and after its activity, the same number of resources that the node will release to its coupled counterpart. In the example shown in Fig. 2, the reversibility of node i and j, i.e., r_i and r_j, is 3 and 1, respectively. The graph dynamics shows that node i fires once which node j fires three times, in a period of 4 stages, due to their different reversibility.

B. Analog OBB Modules

In order to implement the SER/SMER algorithms in analog circuit, the algorithms parallel and distributed characteristics have been converted to the neural network building blocks represented by the discrete mathematics formulae as shown below. Here the node is converted to a neuron, and the node's reversibility is converted to a set of neuronal parameters including the neuron membrane potential, neuron output voltage, threshold and coupling weights [15].

$$V_M^i(n+1) = V_M^i(n) + w_{ji}v_j(n) + w_{ii}v_i(n) \quad (1)$$

$$V_M^j(n+1) = V_M^j(n) + w_{ij}v_i(n) + w_{jj}v_j(n) \quad (2)$$

$$V_i(n) = max(0, sgn(V_M^i(n) - \theta_i)) \quad (3)$$

$$V_j(n) = max(0, sgn(V_M^j(n) - \theta_j)) \quad (4)$$

where in Eqn.1, $V_M^i(n+1)$ is neuron i membrane potential at time instant $n+1$. w_{ji} is the synaptic weight of neuron j connecting neuron i. w_{ii} is the internal negative feedback coupling strength of neuron i for maintaining its stability. θ_i and θ_j are the thresholds of neuron i and j, respectively. $v_i(n)$ and $v_j(n)$ are the neuron i and j outputs, respectively. $sgn(*)$ takes the sign of the operation, $max(*)$ takes the maximum value of the operation.

The relation between the neuronal threshold, weights and the node reversibility is shown below.

$$\begin{cases} \theta_i = max(r_i, r_j)/(r_i + r_j - gcd(r_i, r_j)) \\ w_{ij} = max(r_i, r_j)/r^{'} \\ \theta_j = (min(r_i, r_j) - 1)/(r_i + r_j - gcd(r_i, r_j)) \\ w_{ji} = min(r_i, r_j)/r^{'} \end{cases} \quad (5)$$

where $gcd(*)$ takes the greatest common divisor. $r^{'} = h(r)$ where $h(*)$ is a function of the highest integer level and multiplying it by 10, e.g., $h(256) = 1000$, $r^{'}$ is used for normalisation. In addition, we have $w_{ii} = -w_{ij}$, $w_{jj} = -w_{ji}$.

The neural model has been simulated for mimicking the behavior of CPGs mechanism [15] [16]. In this work, we show a simple neuromorphic circuit design to implement the model with similar operational performance compared to the expensive VLSI circuit realisation.

III. NEUROMORPHIC OBBS MODULES

As described in section II, the design of the SMER-based OBBs modules takes the SER-based OBBs module as its basis. In this section, we start with designing the SER-based OBBs module. The SMER-based OBBs modules can be realized by adjusting the bias voltages of the SER-based modules. Similar to the work in [17], an OBBs module only outputs a high or a low level voltage representing the envelope of the neuronal population bursting activity. The outputs can be used to drive

the robot joint motor for one DoF. The dynamical detail of the neuronal spikes are not considered for simplicity. The free circuit simulator LTSpice XVII running on a Windows 10 laptop with Intel core i5 is used as the circuit simulator.

A. SER-based OBBs Module

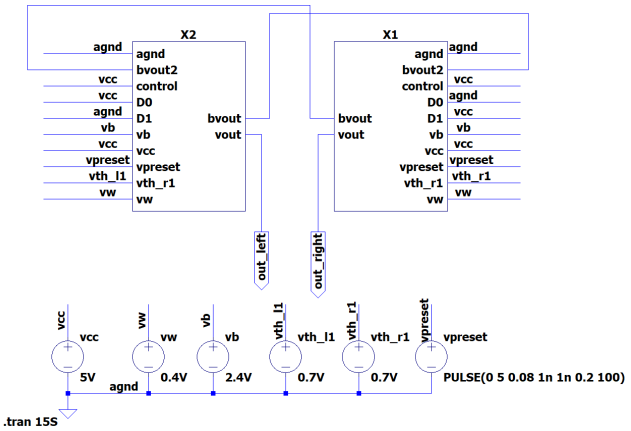

Fig. 3. The block diagram of a SER/SMER-based OBBs module.

Here a simple SER-based OBBs module consisting of two coupled neurons is considered. Fig. 3 shows the block diagram of two neurons connected in such a mutual exclusive way that only one neuron can be active at any time while the other neuron must be idle. The neuron circuit is composed of analog and digital parts, hence a mixed-signal circuit. Four bias voltage sources are used for the OBBs module, apart from the power supply and a system preset signal. A neuron is connected with its coupled counterpart via the signal bvout, which, when the neuron stops firing, kicks off the activity of its coupled neuron.

B. Analog Part

As shown in Fig. 4, a capacitor $C1$ of $330nF$ works as the membrane capacitor of the neuron. There are 6 P-channel and 5 N-channel Metal Oxide Semiconductor Field Effect Transistors (PMOS and NMOS). $M1$ to $M4$ constitute a current mirror with $M4$ controlling the current strength via its gate voltage vw to charge the capacitor $C1$. Three analog stages follow the current mirror taking the capacitor $C1$ voltage as the input. The capacitor voltage is obtained using Eqn. 6 (for a detailed derivation see [17]). The process of charging $C1$ represents the active period of the neuron. The PMOS $M6$ serves as a potentiometer with its source-drain resistance adjustable by the bias voltage vb.

$$V_{C1} = I_{DO} \cdot \frac{W}{L} e^{\frac{v_w}{nU_T}} \cdot \frac{t}{C_1} \tag{6}$$

Where I_{DO} is a process dependant parameter, n the sub-threshold slope factor, U_T the thermal voltage, W and L are the width and length of $M1$ transistor, respectively. When the voltage on $C1$ is less than the voltage at the source of $M7$ then $M7$ is switched off, and $M6$ is resistive so that the drain of $M7$ has the high level (here $5V$). This is used as the input to the inverter composed of $M8$ and $M9$, whose switching threshold is $V_{th} = \frac{rv_{dd}}{1+r}$, where v_{dd} is the supply voltage of $5V$, r is a factor comparing the driving strength of the PMOS and NMOS transistors, related to their channel width and length ratio, in our case $r = 2.1$. The switching voltage in our design is $V_{th} = 1.75V$. When $C1$ voltage becomes greater than $M7$ source voltage, which is set lower than the next stage inverter switching threshold V_{th}, then $M7$ is switched on. The inverter output becomes high which represents the end of activity of this neuron by driving the digital part to output a high level. This high level signal connects to the gates of $M3$ and $M5$, switching off $M3$ which stops charging the capacitor $C1$, and switching on $M5$ which discharges the capacitor $C1$. Meanwhile, the inverter output drives the couple neuron to be active.

C. Digital Part

The digital part of the OBBs module is composed of two functional components. One, the $X1$ component in Fig. 5, is a RS flip-flop to form the neuron output based on the ramp-shaped capacitor voltage. The other, $X2$ and $X3$, is also a RS flip-flop and peripheral combinational circuit as the selector of the active neuron in the pair which starts the mutually inhibited oscillation. In addition, this circuit is also responsible for stopping the neuron activity and kicking off the activity of the other coupled neuron, depending on the system parameters. These two tasks are fulfilled by using the output signal Y to control the gate voltage of PMOS $M3$ as a switch for charging the capacitor $C1$, and the gate voltage of NMOS $M5$ for discharging the capacitor.

IV. Simulation Outcomes

The circuit has been simulated using LTSpice XVII. Different from the previous mixed-signal VLSI circuit design for neuromorphic stepping pattern generation [17], where a coupled OBBs module has a set of 7 bias voltages, this design has only a set of 4 bias voltages, i.e., $\{v_w, v_b, vth\_right, vth\_left\}$. vw is the synaptic weight for charging the capacitor, which is able to control the periodic activity duty cycle and frequency. vb is a fixed voltage for introducing a pull-up resistance by using PMOS $M6$. vth_right and vth_left are the threshold voltage for the right and the left neuron, respectively.

We consider the CPGs is controlled by the descending signals from the cortex system without the ascending signals from the sensors. However, it is possible to add extra control signals to include the sensor feedback. We show here the OBBs module is able to display outputs of different frequencies and duty cycles by adjusting the model parameters.

A. Control of Frequency

When animals walk with different gait patterns and speeds, their leg joints are driven by CPGs with variable activity frequencies. By adjusting one of the model parameters, vw alone, the speed of charging the capacitor $C1$ is changed (see

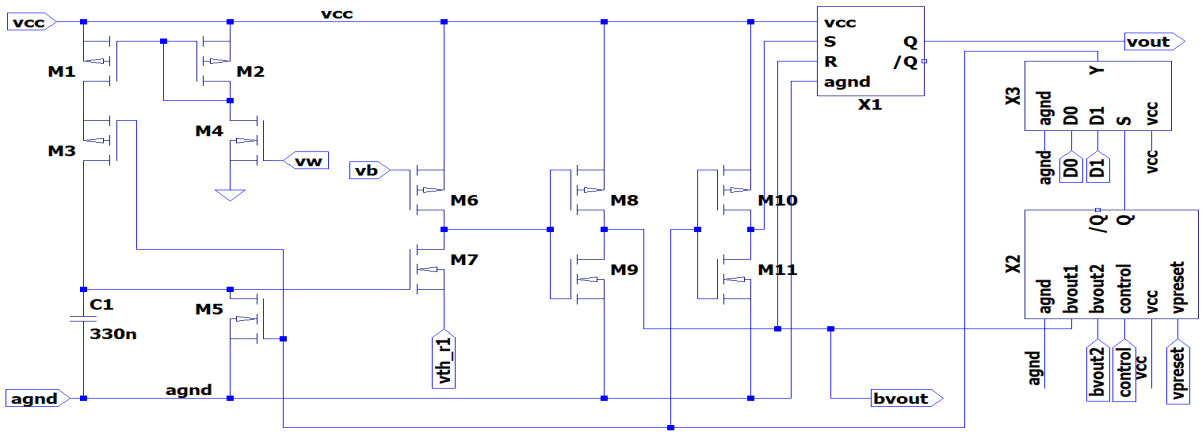

Fig. 4. The exemplary schematic diagram of the right neuromorphic neuron.

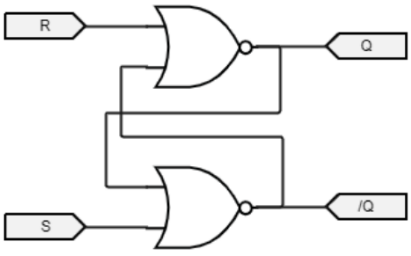

Fig. 5. The digital component $X1$ is a typical RS flip-flop.

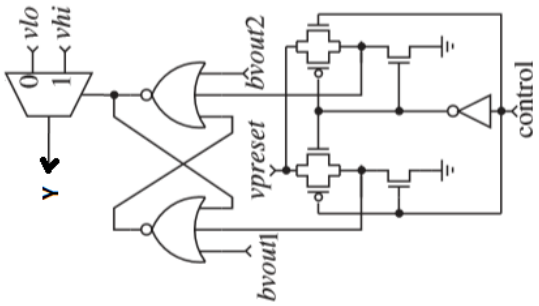

Fig. 6. The digital component $X1$ is a typical RS flip-flop.

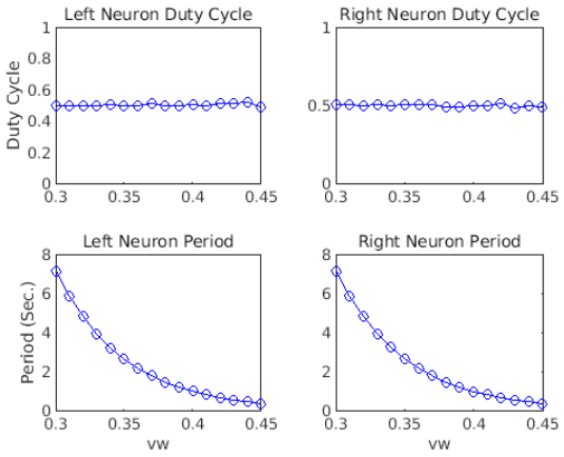

Fig. 7. Relation between the neuron firing frequency and the capacitor charging strength controlled by vw.

Fig. 7). As the capacitor voltage is compared with $M7$ source voltage which is fixed, the bigger the vw value is, the faster the capacitor voltage becomes larger than the $M7$ source voltage. When this happens, $M7$ is switched on. Then the input to the inverter composed of $M8$ and $M9$ jumps to the low level from the high level as the $M7$ source voltage is set to a value smaller than the inverter's threshold voltage. Hence, the inverter output becomes high to reset the RS flip-flop $X1$, indicating the end of its activity. Meanwhile, it shuts down the current mirror, discharge the capacitor via the digital circuits $X2$ and $X3$, as shown in Fig. 6.

The simulation in Fig. 7 shows that the coupled neurons activity frequency depends on the synaptic weight like bias voltage vw. When vw is changed within the range of $[0.3, 0.45]$, different frequencies of CPGs driving signal can be achieved while the duty cycles of the signal keeps unchanged. The range of the signal frequency, in the range of $[0.125Hz, 3Hz]$, is biologically plausible [10].

B. Control of Duty Cycle

The duty cycle is another target for a coupled neuron to be controlled. Terrestrial animals alter locomotion speed by preferentially decreasing the stance phase of leg movements. Consequently, the duty cycles of swing and stance change with locomotion period [2]. The synaptic weight vw is the only variable which needs to be adjusted if adapting the activity frequency while keeping the duty cycle unchanged. However, two variables, i.e., the threshold of one neuron and the synaptic weight that two neurons couple with each other, need to be tuned to change the duty cycles while keeping the frequency unchanged.

Table I presents the outcomes of 10 simulations. It is apparent that the right neuron pulse width decreases while the left neuron pulse width increases with the monotonic increase

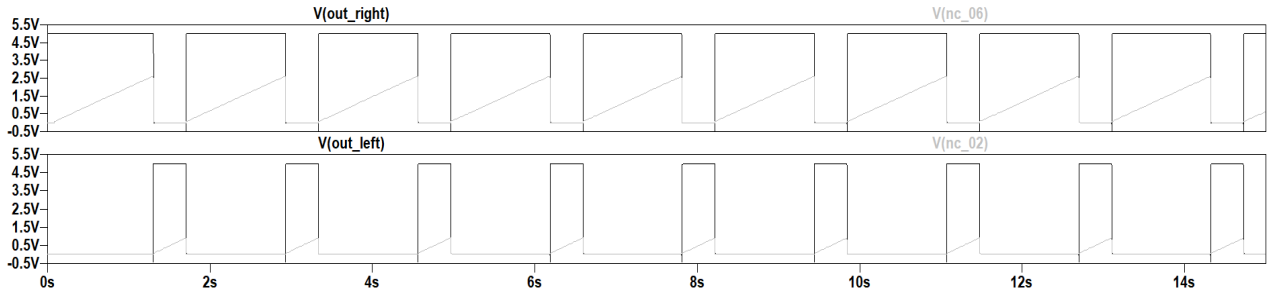

Fig. 8. Two coupled neurons activity. The dark-colored square waves are the outputs of the neurons; The grey-colored ramp waves are the capacitors voltage of the neurons. Model parameters: vth_l1 (left neuron threshold) = $0V$, vth_r1 (right neuron threshold) = $1.7V$, vb (bias voltage) = $2.4V$, vw (synaptic weight/gate voltage) = $0.379V$. Right cycle equals to left cycle $1.62S$, right pulse $1.22S$, left pulse $0.4S$.

Table I: CHANGE OF DUTY CYCLE WITH FIXED PERIOD

Model Parameters (V)*		**Output Signal Parameters (Sec.)**			
vth_l1	vw	right pulse	right cycle	left pulse	left cycle
-0.75	0.368	1.52	1.58	0.06	1.59
-0.5	0.372	1.4	1.59	0.19	1.6
-0.25	0.376	1.3	1.6	0.3	1.6
0.0	0.379	1.22	1.62	0.4	1.62
0.25	0.384	1.11	1.58	0.47	1.59
0.5	0.387	1.05	1.59	0.63	1.59
0.75	0.39	0.99	1.6	0.61	1.6
1.0	0.393	0.93	1.59	0.67	1.59
1.25	0.395	0.89	1.62	0.73	1.62
1.5	0.398	0.84	1.6	0.76	1.6

*vth_l1 is the left neuron threshold and vw is the synaptic weight to charge the capacitor. The other two model parameters remain unchanged, i.e., right neuron threshold vth_r1=1.7V, and bias voltage vb=2.4V.

of two model parameters, here the left neuron threshold and the synaptic weight. Two other model parameters, i.e., the right neuron threshold and the bias voltage vb are not changed. The activity frequency of two neurons remains the same within a tolerable range of measurement errors and circuit mismatch. Fig. 8 shows one case (the fourth instance in Table I) in which two neurons are firing with different duty cycles. It is interesting to see that this instance approximates one SMER instance which has two nodes of different reversibility values 3 and 1, respectively. Thereby in one cycle the node with its reversibility value of 3 will fire for only the one-third time as its coupled node.

V. Concluding Remarks

In this work, a novel neuromorphic circuit design is proposed by using the off-the-shelf discrete electronic components. The circuit is shown capable of implementing the parallel and distributed algorithms for generating rhythmic patterns for legged locomotion. A set of model parameters are used to achieve the independent adaptation of the neuron activity frequency and duty cycle, which are important considerations for CPGs measurement. The future works include the construction of the complete locomotion CPGs network including a range of coordinated gait patterns and transitions between different patterns, based on the proposed OBBs modules.

References

[1] N. Dominici, Y.P. Ivanenko, G. Cappellini, et al., "Locomotor primitives in newborn babies and their development," Science, vol. 334, pp. 997–999, 2011.

[2] S. Grillner, "Control of locomotion in bipeds, trtrapods, and fish," Comprehensive Physiology, doi:10.1002/cphy.cp010226, 2011.

[3] C.V. Hole, J. Goyens, S. Prims, et al., "How innate is locomotion in precocial animals? A study on the early development of spatio-temporal gait variables and gait symmetry in piglets," J. Exp. Biology, vol. 220, pp. 2706-16, 2017.

[4] L. Ruder, A. Takeoka, and S. Arber, "Long-distance descending spinal neurons ensure quadrupedal locomotor stability," Neuron, vol. 92(5), pp. 1063-78, 2016.

[5] A. Prochazka, "Comparison of natural and artificial control of movement," IEEE Trans. Rehab. Eng., vol. 1, pp. 7-17, 1993.

[6] O. Kiehn, and S.J. Butt, "Phsiological, anatomical and genetic identification of CPG neurons in the developing mammalian spinal cord," Prog. Neurobiol., vol. 70(4), pp. 347-61, 2003.

[7] I. Steuer and P.A. Guertin, "Central pattern generators in the brainstem and spinal cord: an overview of basic principles, similarities and differences," Reviews in Neurosci., vol. 30(2), pp. 107-64, 2019.

[8] S. Grillner and A.E. Manira, "Current principles of motor control, with special reference to vertebrate locomotion," Physiol. Rev., vol. 100, pp. 271-320, 2020.

[9] K. Nakada, T. Asai, and Y. Amemiya, "An analog CMOS central pattern generator for interlimb coordination in quadruped locomotion," IEEE Trans. Neural Netw., vol. 14(5), pp. 1356–1365, 2003.

[10] S. Still, K. Hepp, and R. J. Douglas, "Neuromorphic walking gait control," IEEE Trans. Neural Netw., vol. 17(2), pp. 496–508, 2006.

[11] Z.Yang, K. Cameron, L. Lewinger, A.F. Murray, and B. Webb, "Neuromorphic control of stepping pattern generation: a dynamic model with analog circuit implementation," IEEE Trans. Neural Netw Learning Sys., vol. 23(3), pp. 373-84, 2012.

[12] J. Hwangbo, J. Lee, A. Dosovitskiy, et al., "Learning agile and dynamic motor skills for legged robots," Science Robotics, vol. 4, pp. 1-13, 2019.

[13] S. Daun, C. Mantziaris, T. Toth, A. Büschges, N. Rosjat, "Unravelling intra- and intersegmental neuronal connectivity between central pattern generating networks in a multi-legged locomotor system," PLoS ONE, vol. 14(8), doi: 10.1371/journal.pone.0220767, 2019.

[14] V.C. Barbosa, E. Gafni, "Concurrency in heavily loaded neighbourhood-constrained system," ACM Trans. on prog. Lang. and sys., vol. 11(4), pp. 562–584, 1989.

[15] Z. Yang and F.M.G. Franca, "The discrete artificial locomotion CPG architecture and its analog asymmetric Hopfield-like conversion," Biol. Cybern., vol. 89(1), pp.34-42, Springer, 2003.

[16] Z. Yang, D. Zhang, M. Rocha, P.Lima, M. Karamanoglu, F.M.G. Franca, "Prescription of rhythmic patterns for legged locomotion", Neural Computing and Applications, vol.28(11), pp. 3587-3601, Springer, 2017.