Gait Genesis Through Emergent Ordering of RBF Neurons on Central Pattern Generator for Hexapod Walking Robot

Jan Feber, Rudolf Szadkowski, and Jan Faigl

Czech Technical University in Prague, Faculty of Electrical Engineering, Technicka 2, 166 27 Prague, Czech Republic,
{feberja1|szadkrud|faiglj}@fel.cvut.cz,
WWW home page: https://comrob.fel.cvut.cz

Abstract: The neurally based gait controllers for multi-legged robots are designed to reproduce the plasticity observed in animal locomotion. In animals, gaits are regulated by Central Pattern Generator (CPG), a recurrent neural network producing rhythmical signals prescribing each leg's action timing, leading to coordinated motion of multiple legs. The biomimetic CPG-RBF architecture, where leg motion timing is encoded by Radial Basis Function (RBF) neurons coupled with CPG, is used in recent gait controllers. However, the RBF neurons coupling is usually parameterized by the supervisor. Therefore, the RBF parameters get outdated when the CPG signal's wave-form changes. We propose self-supervised dynamics for RBF parameters adapting to a given CPG and producing the required gait rhythm. The method orders the leg activity with respect to inter-leg coordination rules and maps the activity onto CPG states. The proposed dynamics produce rhythmic control for three different hexapod gaits and adapts to the CPG parametric changes.

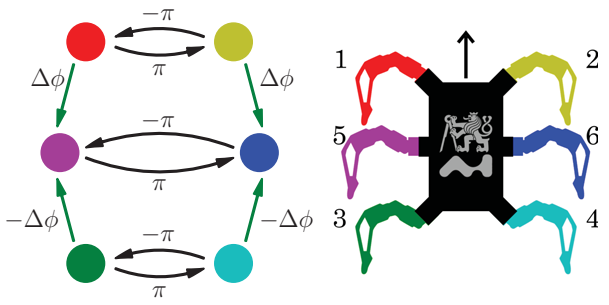

Figure 1: On the left: schema of leg motion phase ϕ_i relations. Each vertex represents the motion phase of the leg of the corresponding color on the corresponding anatomic position. The black oriented edges indicate repulsive forces maintaining antiphase; the green edges indicate keeping a specified distance from the other phase. The colors and positioning correspond to the robot schema on the right. On the right: the robot schema with colored and labeled legs, corresponding to labeling in Figs. 2 and 3. The arrow indicates the direction of the robot's movement.

1 Introduction

The biomimetic approach of the gait control is adopted in multi-legged robotics to imitate robustness and adaptability observed in animal locomotion [1]. The locomotion is driven by a neural network that continually controls and adapts to the environment during movement. In the context of the gait control, the essential part of the neural network is a Central Pattern Generator (CPG) [2], recurrently connected neurons generating rhythmical signals that drive the motion. The CPG is thus employed in many biomimetic multi-legged robot gait controllers [3].

In CPG-based controllers, the CPG drives the repetitive gait motion. During a regular motion, the gait can be described as a repeating sequence of leg movements, where each movement is performed at a certain motion phase. The motion phase is a hidden state that can be inferred from sensory feedback [4], or, as in our case, tracked by the CPG.

The CPG signal is periodic, and thus the state of the recurrent neural network representing the CPG creates a limit cycle, a closed trajectory in the state space. The CPG state then tracks the hidden motion phase of the gait [5, 6], where each leg movement corresponds to some CPG state. The method of mapping the leg movements onto the CPG state is determined by the selected architecture of the CPG-based controller.

In this paper, we focus on the architecture where the CPG is coupled with Radial Basis Function (RBF) neurons, where each RBF neuron fires at a particular CPG state [7, 8]. The RBF neuron activity dependes on the distance of the input point to the neuron's parameter point, i.e., the activity dependes on the radius around the fixed parameter determining vicinity in which the input point is. Hence, radial basis function neurons.

In our method, the RBF neurons are parameterized by the centers placed into the CPG's state space such that when the CPG state, representing the input, is near one of the centers, the corresponding RBF peaks in the activity. The RBF activations can then be used as motion phase encoding, motion primitive trigger, or couple multiple CPGs.

As far as we know, in all current CPG-RBF controllers, the RBF centers are set up by a supervisor. Such a prior parametrization assumes that the CPG properties remain unchanged during the locomotion. Due to the assumption of static properties, the CPG cannot be optimized (e.g., by Righetti's learning rule [9]) nor the entraining waveform can be changed, which poses a limitation to the adaptability of the system.

In this work, we propose a dynamic rule for RBF centers self-organization that generates gaits. The proposed method decomposes the RBF centers organization into two tasks. First, the organization of leg movements in phase space that is consistent with Inter-leg Coordination Rules (ICRs) [10] and given phase offset of consecutive legs' activity [11], providing phase relations within legs actions (see Fig. 1). The second task is the mapping of the organized leg movements onto CPG states. Both tasks are processed continually by the proposed dynamic rules and organize the RBF centers along the CPG's limit cycle, so the resulting rhythm produces a corresponding gait.

The method is implemented on a hexapod walking robot in the simulated environment, where we show that the proposed solution generates multiple gaits consistent with the ICRs. We also demonstrate the adaptive capabilities during change of CPG properties, where the proposed solution adapts to the changes.

2 Related Work

CPG-based controllers can be found in wearable and legged robotics. The controllers can consist of two submodules: amplitude control, providing the magnitude of actuation, and phase control, providing the actuation timing [12, 13]. The CPG is involved in the phase control, where the CPG state represents the motion phase. One of the distinguishing features of the CPG-based controllers is how the CPG state is mapped into the movement phase. Three types of CPG-to-motion mapping can be distinguished: continuous, binary-phase switch, and its generalization multi-phase switch, characterized as follows.

Continuous mapping reshapes the CPG signal into the motion command with continuous function. In [14], where the CPG signal is empirically reshaped into joint angle command. The authors of [15] interpret the CPG output as a foot tip position that is transformed into joint angles by inverse kinematics. In [16], the CPG output is directly fed as an input of a PD controller that transforms the CPG output into angles of leg joints. Continuous maps depend on the wave-form of the CPG, which might limit the system adaptation, as the wave-form changes non-linearly with changing CPG parameters.

In contrast, using the CPG as a binary switch makes the system independent of the exact shape of the wave-form and rather uses the CPG as a timing generator. The CPG can be used for switching between the stance and swing leg motion modes, where each mode has its control rules [12, 17]. The switching approach can be combined with a continuous mapping approach as in [18], where two different CPG output shapers are defined for the stance and swing motion modes, respectively. Using the CPG as a switch between stance and swing leg motions is straightforward; however, it forces the architecture to contain at least one CPG per leg as each leg needs its swing/stance timing. In multi CPG networks, the different gaits are implemented by learning the correct connectivity between CPGs, which might be difficult as the interaction between CPGs is generally non-linear. The networks of multiple CPGs can be avoided by generalizing the binary-phase into a multi-phase approach provided by CPG-RBF architecture.

Table 1: Phase offset of consecutive legs for corresponding gait pattern

gait pattern:	tripod	transition	wave
phase offset $\Delta\phi$:	π	$2\pi/3$	$\pi/3$

The CPG-RBF architecture has been recently proposed in [19], and it provides a straightforward representation of the map between CPG states and motion phase using the RBF layer. The straightforward motion phase representation is utilized in [8], where the RBF output is used to learn the amplitude control with reinforcement learning mechanisms. The RBF neurons themselves can be trained to adapt general CPG with the periodic Grossberg rule. It is presented in [7], where the learning rule is used to couple two CPG layers specialized in sensory estimation and motor phase control.

In the context of gait generation, each of the three approaches has a different way of parameterizing the gait. While there are already proposed methods for gait learning for continuous and binary mapping approaches [14, 13, 12, 18], there is none for the CPG-RBF architecture. The aforementioned CPG-RBF controllers have RBF centers set by a supervisor, limiting the system to static CPG's properties. Since we aim to increase the adaptability capabilities of the CPG-RBF controllers, we propose a self-organizing method for the RBF centers that generate the desired gait patterns for the hexapod walking robot.

3 Problem Statement

The gait phase controller provides the timing for each i-th leg to coordinate the leg movement. We focus on the movement patterns that are consistent with three inter-leg coordination rules (ICRs) observed from hexapod insect gaits [10]: (i) while a leg is lifted-off, suppress the lift-off of the consecutive leg; (ii) if the leg touches the ground, initiate the lift-off of the consecutive leg; (iii) do not lift off the contralateral legs at the same time. The swing duration given by the phase offset $\Delta\phi$ [11] then determines the exact motion pattern (see Tab. 1) as it is depicted in Figs. 2 and 3, where the motion states of the feasible gaits are visualized with the color labeling as in Fig. 1.

For the CPG-RBF phase controller, we encode the coordinated timing by coupling RBF neurons to a single CPG, where each i-th RBF neuron drives the corresponding i-th leg. Generally, the CPG state evolution, $\boldsymbol{y}(t) \in \mathbb{R}^D$, can be modeled by the differential equation $\dot{\boldsymbol{y}} = f(\boldsymbol{y}(t), c(t))$, where the dot notation represents differential with respect

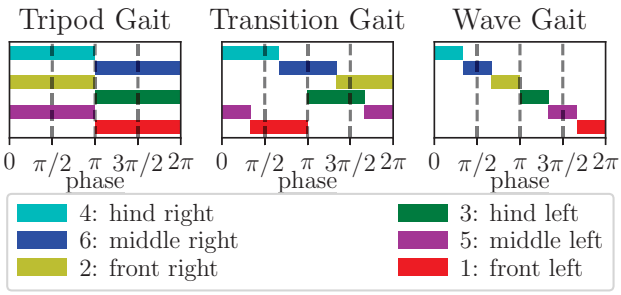

Figure 2: Visualization of legs' activity during the repeating phase for the desired gait patterns. The color bar represents the duration of the corresponding leg's swing action. Note that the ordering and relative distance (phase offset) of actions are important, not their particular position within the phase. The gaits have more valid ordering options, and the figure represents only one of the possible orderings.

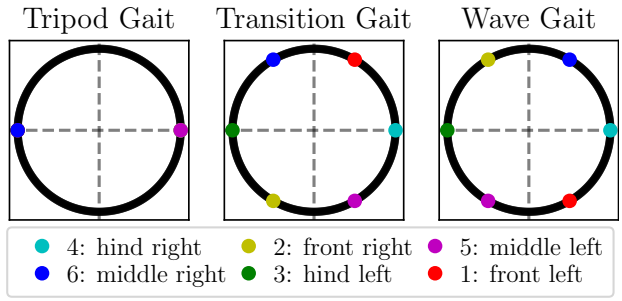

Figure 3: Example visualization of motion phases ϕ_i (colorful dots) correctly ordered within the phase (black circle) to produce the desired gait patterns. The interval $[0, 2\pi)$ representing phase is depicted as a unit circle using $\cos(\phi)$ for the horizontal axis and $\sin(\phi)$ for the vertical axis for $\phi \in [0, 2\pi)$. Note that the ordering and relative distance (phase offset) of actions are important, not their particular position within the phase. The gaits have more valid ordering options and the figure represents only one of the possible orderings. The motion phases for simultaneously activated legs in tripod gait are overlapping.

to time, and the system f contains a limit cycle attractor, $\boldsymbol{y}_0 \subset \mathbb{R}^D$: a looped trajectory to which all neighboring states converge. Thus, after the convergence, the CPG is T-periodic, $\boldsymbol{y}(t) = \boldsymbol{y}(t+T)$, and the CPG generates a periodic signal. The CPG state is an input for the RBF neuron activation $\varphi(\boldsymbol{y}; \boldsymbol{w}) = \exp\left(-\psi \|\boldsymbol{y} - \boldsymbol{w}\|_2^2\right)$, where the center $\boldsymbol{w} \in \mathbb{R}^D$ and hyperparameter ψ determines the timing and duration of activation, respectively. The RBF neuron peaks when the CPG state is close to the RBF center, $\boldsymbol{y}(t) \approx \boldsymbol{w}(t)$, generating periodic peaks. For each i-th leg, there is a corresponding RBF neuron with center $\boldsymbol{w}_i$, which peaks trigger the lift-off.

For a general CPG, the centers $\boldsymbol{w}_i$ cannot be placed a priory, as the shape of the limit cycle $\boldsymbol{y}_0$ is not known. Moreover, the limit cycle can change its shape dynamically with changing parametrization of CPG dynamics $f(\cdot)$ or different CPG input $c(t)$. Thus the centers $\boldsymbol{w}_i$ of RBF neurons need to be dynamically adjusted to drive the locomotion according to the coordination rules and given phase offset $\Delta\phi$.

4 Method

We propose dynamic rules that form feasible gait patterns by organizing the RBF centers $\boldsymbol{w}_i$ on the CPG's limit cycle $\boldsymbol{y}_0$ while respecting the ICRs and maintaining the given phase offset $\Delta\phi$. The task is decoupled into two subtasks: (i) order the lift-offs of each i-th leg into a sequence and (ii) map the sequence onto the CPG limit cycle.

The i-th RBF center dynamics are given by the periodic Grossberg rule [7]

$$\dot{\boldsymbol{w}}_i = (\boldsymbol{y}(t) - \boldsymbol{w}_i(t))\, p_i(t), \tag{1}$$

that pushes the center $\boldsymbol{w}_i$ towards the states $\boldsymbol{y}(t)$ at which the *target signal* $p_i(t) \in [0, 1]$ is nonzero. In the previous work [7], the target signal is given by a supervisor. We introduce a new method for forming the target signal

$$p_i(t) = \varphi\left(\hat{\phi}(t); \phi_i(t)\right) \tag{2}$$

where $\phi_i \in [0, 2\pi)$ is the phase of the i-th leg lift-off that determines the sequence of lift-offs, and $\hat{\phi} \in [0, 2\pi)$ that maps the $[0, 2\pi)$ phases onto the limit cycle $\boldsymbol{y}_0$. As the phase is within the circular space $\mathbb{S}^1$, we define the circle metric as $||\phi - \phi'|| = \min(|\phi - \phi'|, 2\pi - |\phi - \phi'|)$, which gives the closest distance between two phases. For both variables, ϕ_i and $\hat{\phi}$, we present their dynamics in the following sections.

4.1 Organizing Phase of Legs Activity

The motion start phases ϕ_i of each i-th leg must be ordered within the interval $[0, 2\pi)$, where the ordering has to be consistent with the ICRs, and phase offset $\Delta\phi$. Both constraints can be defined as distances between phases ϕ_i: (i) front (hind) leg i is shifted by $\Delta\phi$ with respect to ipsilateral middle leg j, $||\phi_i - \phi_j|| = \Delta\phi$; (ii) contralateral legs i and j are in anti-phase $||\phi_i - \phi_j|| = \pi$, see Fig. 1.

We present the following dynamics to order randomly initialized phases ϕ_i

$$\dot{\phi}_i(t) = \sum_j^6 \alpha_{ij} \left(\Delta_{ij} + \text{sign}\left(\phi_i - \phi_j\right) ||\phi_i - \phi_j||\right), \tag{3}$$

where $\Delta_{ij} \in [-\pi, \pi]$ parameterize the target offset between ϕ_i and ϕ_j, and $\alpha_{ij} \in \mathbb{R}^+$ weighs the relation influence. The relationships between contralateral legs are parameterized as $\Delta_{2,1} = \Delta_{4,3} = \Delta_{6,5} = -\pi$ and $\Delta_{1,2} = \Delta_{3,4} = \Delta_{5,6} = \pi$. The relation between ipsilateral legs are $\Delta_{2,6} = \Delta_{1,5} = \Delta\phi$ and $\Delta_{4,6} = \Delta_{3,5} = -\Delta\phi$. The above-defined relations have set $\alpha_{ij} = 1$ while the rest is turned off by $\alpha_{j'i'} = 0$. The variable ϕ_i serves as a parameter of the target signal (2), the next input is the phase estimation.

4.2 Mapping the Legs Activity onto the Limit Cycle Using Phase Estimation

The target signal (2) should have the same periodicity as the CPG; however, the periodicity of a general CPG nor its frequency cannot be obtained analytically. However, the frequency is needed to modulate the phase of the target signal $\hat{\phi}(t)$, and thus the frequency must be learned. We propose to dynamically learn the CPG frequency by coupling the CPG with the pivot RBF neuron and measuring the RBF activity period to determine the CPG's frequency.

First, the randomly initialized pivot RBF center $\boldsymbol{w}_{\text{sig}} \in \mathbb{R}^D$ must get close to the limit cycle. The pivot center $\boldsymbol{w}_{\text{sig}}(t)$ is attracted to the CPG state $\boldsymbol{y}(t)$ if the CPG state is within ε-neighborhood of the center

$$\dot{\boldsymbol{w}}_{\text{sig}} = \left(1 - \left\|\boldsymbol{y}(t) - \boldsymbol{w}_{\text{sig}}\right\|^2 \varepsilon^{-2}\right) \left(\boldsymbol{y}(t) - \boldsymbol{w}_{\text{sig}}\right) \mu(t) \quad (4)$$

$$\mu(t) = \begin{cases} 1 & \left\|\boldsymbol{y}(t) - \boldsymbol{w}_{\text{sig}}\right\| < \varepsilon \\ 0 & \text{otherwise} \end{cases}. \quad (5)$$

The neighborhood radius ε itself is dynamic, making the ε-neighborhood expand when the CPG state is outside and contract when the CPG state is closer than half the radius:

$$\dot{\varepsilon} = \left(\text{clip}\left(\left\|\boldsymbol{y}(t) - \boldsymbol{w}_{\text{sig}}\right\| \varepsilon^{-1}\right) - 0.5\right)^3 \gamma(t) \quad (6)$$

$$\gamma(t) = \begin{cases} \sigma_1 & \text{clip}\left(\left\|\boldsymbol{y}(t) - \boldsymbol{w}_{\text{sig}}\right\| \varepsilon^{-1}\right) < 0.5 \\ \sigma_2 \varepsilon^2 & \text{otherwise} \end{cases}, \quad (7)$$

where $\text{clip}(x) = \min(1, \max(0, x))$ and hyperparameters $\sigma_1 = 80, \sigma_2 = 2$ are set empirically. As the pivot center $\boldsymbol{w}_{\text{sig}}(t)$ converges to the limit cycle, the pivot RBF activation $p_{\text{sig}} = \varphi(\boldsymbol{y}(t); \boldsymbol{w}_{\text{sig}})$ produces T-periodic pulses.

From the pivot RBF activation p_{sig}, we extract the frequency of the CPG by adjusting descend of the variable

$$\dot{s} = \begin{cases} (1-s)\xi & p_{\text{sig}}(t) \approx 1 \\ -a(t) & \text{otherwise} \end{cases}, \quad (8)$$

where ξ is large enough to reset s to 1 when the pivot activation peaks ($p_{\text{sig}}(t) \approx 1$), and $a \in \mathbb{R}^+$ determines the slope of the descend. If the slope of a has such a value that descends from $s(t_1) = 1$ to $s(t_1 + T) = 0$, then a is the CPG frequency. Thus slope a is adjusted as follows

$$a(t_+) := \begin{cases} a(t_-) + \kappa s(t_-) & p_{\text{sig}}(t) \approx 1 \\ a(t_-) & \text{otherwise}, \end{cases} \quad (9)$$

which increases the frequency if $s(t_1 + T) > 0$ and decreases the frequency if $s(t_1 + T) < 0$. The hyperparameter κ is empirically set to 0.02. The obtained CPG frequency is used to estimate the CPG phase

$$\hat{\phi}(t) = 2\pi\left(1 - s(t)\right). \quad (10)$$

After the convergence of the CPG estimation $\hat{\phi}(t)$ and lift-off ordering ϕ_i, the variables $p_i(t)$ of (2) produce the target signal that orders the RBF centers of each i-th leg to produce the gait pattern; which is demonstrated in the following section.

5 Results

The feasibility of the proposed method has been validated by experimental deployment in several scenarios to demonstrate the adaptability of the developed solution. We used Euler's method to run the dynamic system consisting of the proposed equations, running for 20000 iterations with the step size 0.01. The correctness of the generated rhythm for different gaits is demonstrated using modified signals, obtained from the RBF neuron activations $\varphi(\boldsymbol{y}, \boldsymbol{w}_i)$, to trigger the predefined swing movement, followed by a predefined stance movement for the robot's legs in *CoppeliaSim*[1] simulator. The methods have been implemented in *Python 3* and a hexapod model *PhantomX MK-III* has been used to run the simulations. The system's adaptability has been evaluated for two different CPG models.

The first CPG model is Matsuoka oscillator given by

$$\tau_1 \dot{v}_1 = \text{h}(u_1) - v_1, \quad (11)$$

$$\tau_1 \dot{v}_2 = \text{h}(u_2) - v_2, \quad (12)$$

$$\tau_2 \dot{u}_1 = -u_1 - \text{h}(u_2)\beta_1 - v_1\beta_2 + 1, \quad (13)$$

$$\tau_2 \dot{u}_2 = -u_2 - \text{h}(u_1)\beta_1 - v_2\beta_2 + 1, \quad (14)$$

$$\text{h}(x) := \max(x, 0), \quad (15)$$

where the hyperparameters are set to $\tau_1 = 0.5; \tau_2 = 0.25; \beta_1 = \beta_2 = 2.5$, and the function $h(x)$ represents the rectifier (i.e., ReLU function). For the Matsuoka oscillator the CPG's state is $\boldsymbol{y} = (u_1, u_2, v_1, v_2) \in \mathbb{R}^4$.

The second CPG is Van der Pol's oscillator (VdP), given by

$$\dot{u} = v, \quad (16)$$

$$\dot{v} = \zeta\left(1 - u^2\right) v - u, \quad (17)$$

where, ζ is a parameter indicating the strength of damping and the CPG's state is $\boldsymbol{y} = (u, v) \in \mathbb{R}^2$.

5.1 Adaptability Experiments

The system's adaptability to different limit cycle shapes is demonstrated by learning the transition gait in four different scenarios.

1. Using unperturbed Matsuoka oscillator as the CPG model.
2. Using Matsuoka oscillator synchronized with four other coupled Matsuoka oscillators to demonstrate the method's adaptability to small perturbations.
3. Using the VdP with $\zeta = 3$ to demonstrate the usage on a different oscillator.
4. Using the VdP with the changed parameter $\zeta = 1$ to show the adaptability to change of the oscillator parameter.

[1] https://www.coppeliarobotics.com

In scenario 2. the motion phases ϕ_i, their respective RBF centers $\boldsymbol{w}_i$, and the parameter a are initialized to the transition gait values, which were previously successfully learned with the unperturbed CPG. The RBF neurons provide the rhythmical input, producing the signal $p_{\text{sig}}(t)$ based on the position of the center $\boldsymbol{w}_{\text{sig}}$ with the dynamic vicinity.

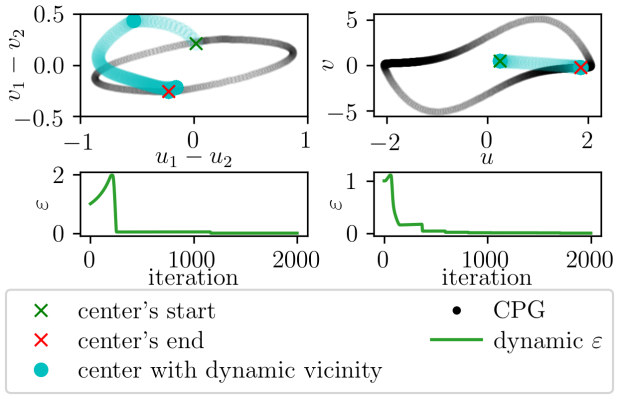

Figure 4: Plot of $\boldsymbol{w}_{\text{sig}}$ RBF center's dynamics based on its dynamic vicinity given by the dynamic radius ε. The upper plots show the center's movement (cyan path from green cross to red cross) in the CPG's state space towards the CPG's limit cycle (black closed shape). The lower plots show the progression of the center's respective ε. The unperturbed Matsuoka oscillator is shown on the left, and the VdP is on the right. A darker shade of the cyan and black colors signalizes that more time is spent at the corresponding place.

The CPG's phase is estimated as $\hat{\phi}(t)$ to map the centers $\boldsymbol{w}_i \in \mathbb{R}^D$ on the CPG's limit cycle, by learning the frequency a, (9), and pivot center $\boldsymbol{w}_{\text{sig}}$, (4). A showcase of $\boldsymbol{w}_{\text{sig}}$ center's attraction to the CPG's limit cycle together with the progress of its dynamic vicinity radius ε, (6), is shown in Fig. 4 for both types of the oscillators with differing limit cycle shapes of a prior unknown shape.

The frequency a is learned based on the signal generated by the RBF neuron corresponding to the center $\boldsymbol{w}_{\text{sig}}$. The learning process of a is shown in Fig. 5. In Fig. 6, we provide a learning process of a for Matsuoka oscillator with differently initialized a. In all the cases, the frequency a successfully converges. The progression and convergence of $s(t)$, estimating the CPG's phase growth, and the modulated learning signal $p_{\text{sig}}(t)$ are presented in Fig. 7, where the fact that a converges can be seen in convergence to zero of local minimum values, marked by the orange line. As mentioned in Section 4, if learned correctly, the value of $s(t)$ declines from one to zero during each period, i.e., at the $p_{\text{sig}}(t)$ signal pulse occurs, $s(t)$ equals zero and it is reset to one.

Based on the estimated phase, the centers are organized around the CPGs' limit cycles, as demonstrated by the transition gait for Matsuoka and VdP oscillators shown in Fig. 8, where centers are organized around the CPGs limit cycles of various a prior unknown shapes.

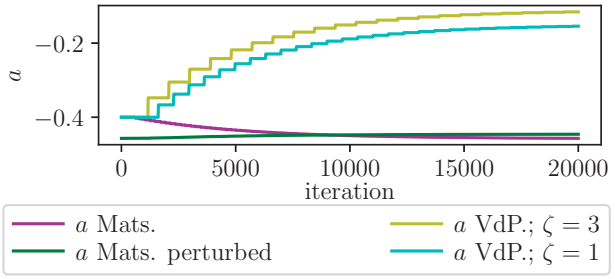

Figure 5: The progression of the learned frequency a for perturbed (green) and unperturbed (magenta) Matsuoka oscillator, and the VdP oscillator with the parameter $\zeta = 3$ (yellow) and $\zeta = 1$ (cyan). The initialization value of a is -0.4 for the VdP and unperturbed Matsuoka oscillators. For the perturbed Matsuoka oscillator, a is initialized as the final value of a of the unperturbed one (i.e., the value of the magenta line at the 20000-th iteration is the initial value of the green line). All the cases converge to a stable value. The values $s(t)$ dependent on a are visualized in Fig. 7.

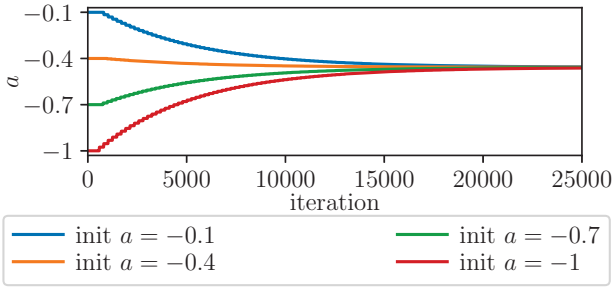

Figure 6: The progression of the learned frequency a for unperturbed Matsuoka oscillator for four different initialization values of a. The plot shows that the frequency a correctly converges to the same value in all the cases.

5.2 Different Gait Patterns Experiment

The ability to generate different gait patterns is demonstrated using Matsuoka neural oscillator. The motion phases $\phi_i \in [0, 2\pi)$ interact with each other according to the given phase offset $\Delta\phi$ of two consecutive leg's actions, and to ICRs, as shown in Fig. 9 for all three gait patterns. The correctness of the process can be observed by comparing the results with schema,[2] describing the gait patterns in Fig. 3. The process of ordering the motion phases within the phase is independent of the used CPG model.

The successful mapping of the RBF centers' for three different gait patterns with the use of Matsuoka oscillator is shown in Fig. 10. The signals produced via the centers'

[2]Note that important are the relative positions, i.e., ordering of the motion phases with correct distance (phase offset) between them.

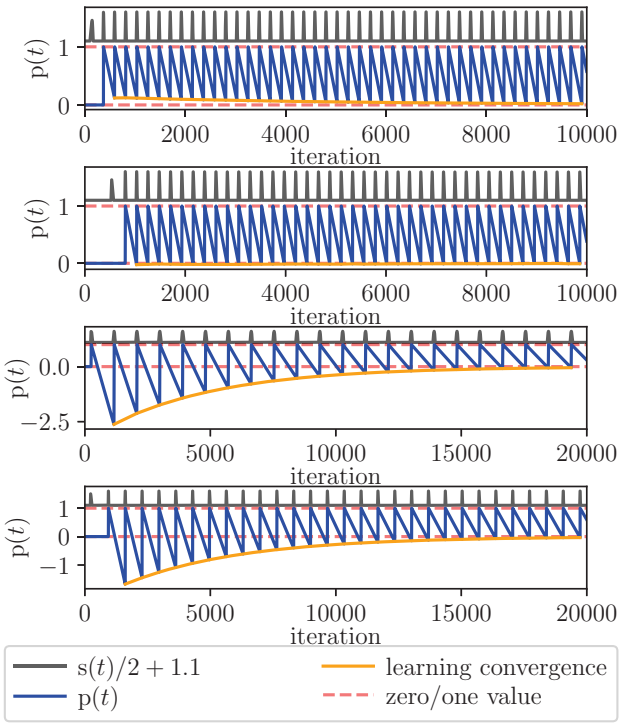

Figure 7: Results of learning $s(t)$ (blue), estimating the phase growth by the proposed relation (10). If learned correctly, $s(t)$ equals to zero at the moment of $p_{\text{sig}}(t)$ signal's (gray line) pulse occurrence, which resets the value of $s(t)$ to one. Hence, the minimum of $s(t)$ in each period should equal to zero, if learned correctly. The plot shows that the values of the periods' minimums (orange line) are correctly converging to zero. The plots correspond to Matsuoka unperturbed and perturbed oscillator and to the VdP with $\zeta = 3$ and $\zeta = 1$ from top to down, respectively. The decline of $s(t)$ depends on the value of the frequency $a(t)$ shown in Fig. 5. The signal $p_{\text{sig}}(t)$ is modulated for improved visibility.

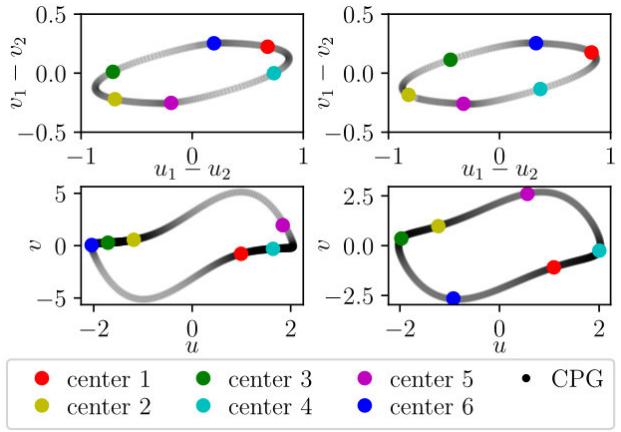

Figure 8: The final state of the RBF centers (colorful dots) placed onto the limit cycles (closed black curve) of unperturbed (upper left) and perturbed (upper right) Matsuoka oscillator and the VdP with parameter $\zeta = 3$ (lower left) and $\zeta = 1$ (lower right).

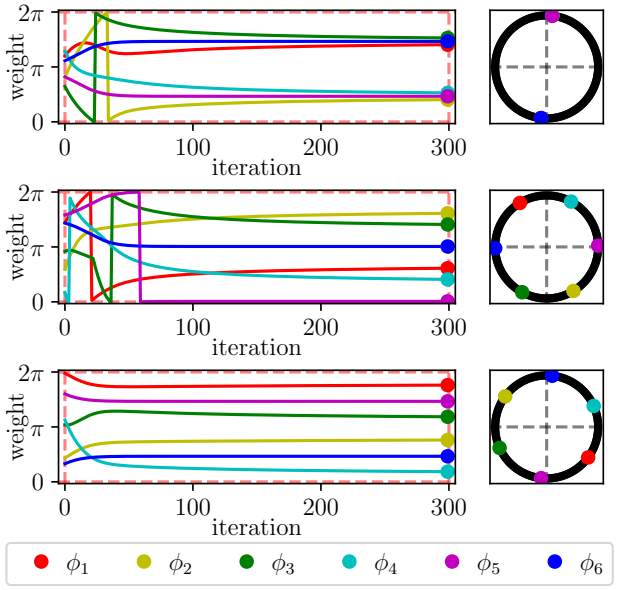

Figure 9: On the left: the first 300 iterations of ϕ_i self-organizing motion phases, where the vertical axis represents the value of the motion phase at each iteration given by the horizontal axis. On the right: the final positions at the 20000-th iteration of the organized motion phases, depicted as a unit circle using $\cos(\phi)$ for horizontal axis and $\sin(\phi)$ for vertical axis for $\phi \in [0, 2\pi)$ (compare with desired patterns in Fig. 2). The plots correspond to the tripod, transition, and wave gaits from top to down, respectively. Note that motion phases for simultaneously activated legs in the tripod gait are overlapping.

RBF neurons producing gait pattern rhythm are visualized in Fig. 11.

The RBF signals are used in the *CoppeliaSim* simulator to make the hexapod robot walking, as shown by snapshots of one gait cycle for each of the given gait patterns in Fig. 12.

The experiments demonstrated that the proposed mechanism successfully produced rhythm for the desired gait patterns on both the CPG models with different dimensionality and different shape of their limit cycles.

5.3 Discussion

The current CPG-RBF controllers require setting the gait-pattern-determining RBF neurons parameters by a supervisor, assuming that the CPG produces a signal of unchanging wave-form and frequency. However, this assumption limits the architecture's ability to adapt to changing CPG parameters, enabling higher frequency (thus faster movement) or adaptation after a change of the synchronizing signal. Our method enables the change of the CPG parameters, and therefore improves the adaptability to evolving conditions for the CPG-RBF architectures.

The gait pattern rhythm is generated by correctly ordered RBF centers $\boldsymbol{w}_i \in \mathbb{R}^D$ (see Figs. 8 and 10), corre-

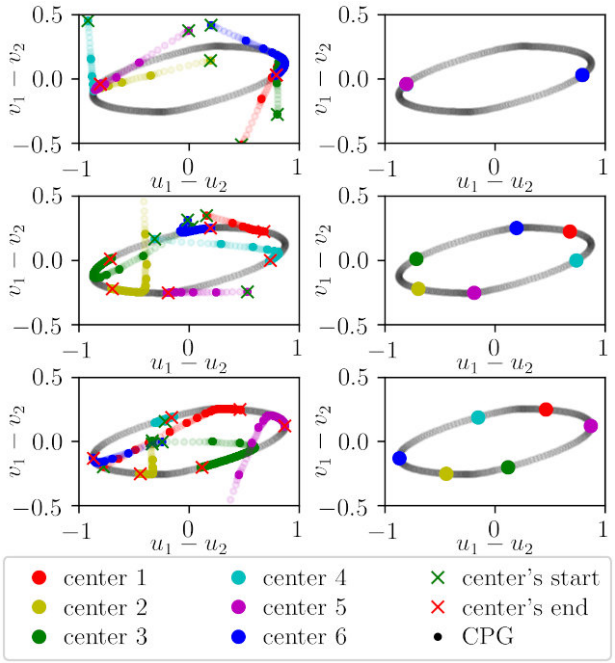

Figure 10: The plot shows the progress and final state of the RBF centers organizing along the limit cycle (black closed shape) of Matsuoka oscillator to produce the desired gait patterns. The centers' movement (colorful paths from green crosses to red crosses) in the CPG's state space are shown on the left-hand side. A darker shade of the colors signalizes more time spend in that place. The centers' final states (the colorful dots) are shown on the CPG's limit cycle on the right-hand side. The gait patterns are tripod, transition, and wave gaits from top to down, respectively. The rhythm invoked by the centers' respective RBF neurons producing the gaits is presented in Fig. 11. Note that the centers for simultaneously activated legs in the tripod gait are overlapping.

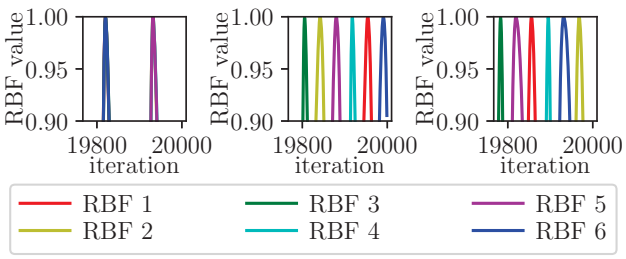

Figure 11: The rhythm invoked by the RBF neurons corresponding to the RBF centers from Fig. 10 at the end of the learning process. The plots show one period of the produced tripod, transition, and wave gaits, from left to right, respectively, using unperturbed Matsuoka oscillator. Note that signals for simultaneously activated legs in tripod gait are overlapping. The snapshot of the signals' use in the simulation is shown in Fig. 12.

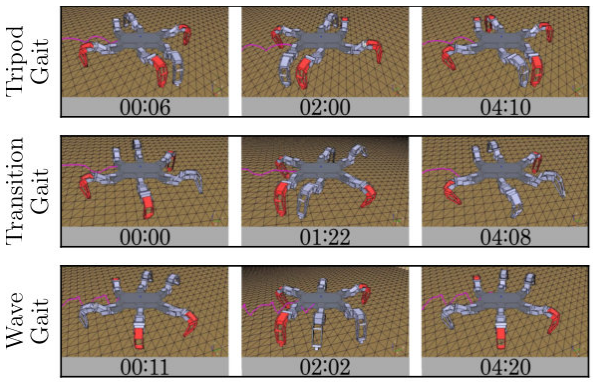

Figure 12: The showcase of one gait cycle of each of the three gait patterns (tripod, transition, and wave) simulated in *CoppeliaSim* simulator. The robot moves from left to right. The timestamps on the gray background are in format *seconds:frames*, where each second represents 25 frames. The time 00:00 marks the start of the gait cycle, beginning with the hind right leg's swing.

sponding to legs actions, around the CPG's limit cycle. The centers $\boldsymbol{w}_i$ produce signals via their respective RBF neurons if the CPG's state is close enough to the corresponding center. The produced signal provides a rhythm for the required gait pattern if the centers' ordering along the limit cycle respects ICRs and the given phase offset $\Delta\phi$ of consecutive legs, determining the required gait pattern. As shown in Fig. 12, we successfully produced three desired gait patterns for the hexapod walking robot, tripod, transition, and wave gaits, described in Figs. 2 and 3.

The model learns parameters which have real-wolrd meaning. The phases ϕ_i represent the respective leg's swing phase start within the walking cycle. The placing of the centers $\boldsymbol{w}_i$ around the limit cycle represents the respective leg's swing phase start within the repeating walking cycle. Slope a represents the frequency of the used CPG. The pivot center $\boldsymbol{w}_{\text{sig}}$ marks the start of the walking cycle on the CPG limit cycle. Hence, the learning process is explainable, which is an advantage in comparison with black-box approaches.

The current method does not enable the generation of gait rhythm for different numbers of legs than six without further modifications. In our future work, we would like to explore the ICRs possibilities in automatically generating gait patterns for any number of legs. Robots with differing numbers of legs exist and malfunctions of the robot are also possible, requiring to learn to walk with damaged or missing limbs while deployed on a mission.

6 Conclusion

In this work, we propose and test self-supervised dynamics for organizing the RBF centers producing rhythm for required gait patterns. The method improves CPG-RBF controllers' gait-generating adaptability towards a change

of CPG properties. The method decouples the gait rhythm generating problem into two tasks, the legs activity ordering, and the CPG phase estimation, leading to mapping the legs' activity ordering onto CPG's states. The ordering of legs' activity within the phase is driven by biomimetic inter-leg coordination rules and given phase offset of consecutive legs' activity, determining the required gait pattern. The phase estimation is based on estimating the phase growth (phase angular velocity) from the signal with the period equal to the CPG's period. Combining the proposed mechanisms enables mapping the RBF centers, corresponding to ordered actions, onto CPG's limit cycle. The phase controller produces the rhythm for three desired gait patterns, tripod, transition, and wave gaits.

We demonstrate the correct functionality of the proposed method, including showcase from *CoppeliaSim* simulator, where the generated gait pattern rhythms are used to invoke the movement of the simulated hexapod walking robot. The results are demonstrated for two different CPG models, Matsuoka neural oscillator with/without a rhythmical input from other coupled CPGs, and Van der Pol's oscillator with two different parameter settings.

In our future work, we aim to extend the model to generate gait patterns for robots with differing numbers of legs.

Acknowledgments – This work has been supported by the Czech Science Foundation (GAČR) under research project No. 21-33041J.

References

[1] A. Ayali, A. Borgmann, A. Büschges, E. Couzin-Fuchs, S. Daun-Gruhn, and P. Holmes, "The comparative investigation of the stick insect and cockroach models in the study of insect locomotion," *Current Opinion in Insect Science*, vol. 12, pp. 1–10, 2015.

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

[3] J. Yu, M. Tan, J. Chen, and J. Zhang, "A survey on cpg-inspired control models and system implementation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 3, pp. 441–456, 2014.

[4] K. W. Wait and M. Goldfarb, "A biologically inspired approach to the coordination of hexapedal gait," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2007, pp. 275–280.

[5] T. Yan, A. Parri, V. R. Garate, M. Cempini, R. Ronsse, and N. Vitiello, "An oscillator-based smooth real-time estimate of gait phase for wearable robotics," *Autonomous Robots*, vol. 41, no. 3, pp. 759–774, 2017.

[6] C. Maufroy, H. Kimura, and K. Takase, "Towards a general neural controller for quadrupedal locomotion," *Neural Networks*, vol. 21, no. 4, pp. 667–681, 2008.

[7] R. Szadkowski and J. Faigl, "Neurodynamic sensory-motor phase binding for multi-legged walking robots," in *International Joint Conference on Neural Networks (IJCNN)*, 2020, pp. 1–8.

[8] M. Thor, T. Kulvicius, and P. Manoonpong, "Generic neural locomotion control framework for legged robots," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–13, 2020.

[9] L. Righetti, J. Buchli, and A. J. Ijspeert, "Dynamic hebbian learning in adaptive frequency oscillators," *Physica D: Nonlinear Phenomena*, vol. 216, no. 2, pp. 269–281, 2006.

[10] V. Dürr, J. Schmitz, and H. Cruse, "Behaviour-based modelling of hexapod locomotion: linking biology and technical application," *Arthropod Structure & Development*, vol. 33, no. 3, pp. 237–250, 2004, arthropod Locomotion Systems: from Biological Materials and Systems to Robotics.

[11] W. Chen, G. Ren, J. Zhang, and J. Wang, "Smooth transition between different gaits of a hexapod robot via a central pattern generators algorithm," *Journal of Intelligent & Robotic Systems*, vol. 67, no. 3, pp. 255–270, 2012.

[12] H. Chung, C. Hou, and S. Hsu, "A cpg-inspired controller for a hexapod robot with adaptive walking," in *CACS International Automatic Control Conference (CACS)*, 2014, pp. 117–121.

[13] H. Yu, W. Guo, J. Deng, M. Li, and H. Cai, "A cpg-based locomotion control architecture for hexapod robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2013, pp. 5615–5621.

[14] L. Xu, W. Liu, Z. Wang, and W. Xu, "Gait planning method of a hexapod robot based on the central pattern generators: Simulation and experiment," in *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2013, pp. 698–703.

[15] G. Zhong, L. Chen, Z. Jiao, J. Li, and H. Deng, "Locomotion control and gait planning of a novel hexapod robot using biomimetic neurons," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 2, pp. 624–636, 2018.

[16] W. Ouyang, H. Chi, J. Pang, W. Liang, and Q. Ren, "Adaptive locomotion control of a hexapod robot via bio-inspired learning," *Frontiers in Neurorobotics*, vol. 15, p. 1, 2021.

[17] Y. Fukuoka, H. Kimura, Y. Hada, and K. Takase, "Adaptive dynamic walking of a quadruped robot 'tekken' on irregular terrain using a neural system model," in *IEEE International Conference on Robotics and Automation (ICRA)*, vol. 2, 2003, pp. 2037–2042.

[18] L. Bai, H. Hu, X. Chen, Y. Sun, C. Ma, and Y. Zhong, "Cpg-based gait generation of the curved-leg hexapod robot with smooth gait transition," *Sensors*, vol. 19, no. 17, 2019.

[19] M. Pitchai, X. Xiong, M. Thor, P. Billeschou, P. L. Mailänder, B. Leung, T. Kulvicius, and P. Manoonpong, "Cpg driven rbf network control with reinforcement learning for gait optimization of a dung beetle-like robot," in *Artificial Neural Networks and Machine Learning – ICANN 2019: Theoretical Neural Computation*, I. V. Tetko, V. Kůrková, P. Karpov, and F. Theis, Eds. Cham: Springer International Publishing, 2019, pp. 698–710.