Design of Synthetic Central Pattern Generators Producing Desired Quadruped Gaits
Matteo Lodi, Andrey Shilnikov, and Marco Storace, Senior Member, IEEE

Abstract—This paper is concerned with a method for design and analysis of specific neuronal networks, called central pattern generators (CPGs), which produce primary rhythmic patterns in animals. In particular, the paper is focused on synthetic CPGs made up of few basic elements and governing quadrupeds' gaits and gait transitions, under the control of an external drive. The method combines the principles of bifurcation theory, geometric properties of symmetry, and numerical analysis based on the recently proposed toolbox CEPAGE. The method is applied to two CPGs, one bio-inspired and one purely synthetic. In both the cases, the method provides a way to obtain a desired sequence of gaits by continuously changing a bifurcation parameter related to the external drive.

Index Terms—Central pattern generators, dynamical systems, bifurcation analysis.

I. INTRODUCTION

THE motor circuits in the spinal cord that control locomotion are commonly referred to as central pattern generators (CPGs). A CPG is a neuronal network that is capable of generating an organized pattern of motor activity independently of sensory inputs, thus producing primary rhythmic behaviors such as respiration, mastication, sucking, crawling, flying, swimming and walking [1]. In vertebrates, the planning of muscle activity involves many supra-spinal networks, which activate the CPGs that determine the gaits [2]–[7]. The CPG functions include selecting which muscles are to be activated, how intensely and for how long, thus allowing patterns of movements of widely varying strengths and speeds, whereas the supra-spinal networks drive the outputs from the CPG (allowing gait changes and adaptation to obstacles and uncertainties during ambulatory excursions [8]) on the basis of both sensory feedback pathways and vestibular pathways [9]–[11]. This combination of closed-loop and open-loop control systems allows obtaining a robust control of locomotion, characterized by rhythmicity (the specific periodic pattern provided by the CPG), stability against perturbations and noise (the pattern corresponds to self-sustained oscillation due to a stable limit cycle), adaptability (owing to the feedback pathways), and variety (by changing the gaits) [12]–[16].

One of the fundamental challenges in motor systems neuroscience is discovering the intrinsic functional mechanisms of CPG networks and the way in which they integrate descending inputs from the brain-stem, which are in turn under the control of basal ganglia and cortex [3], [17]. This challenge is faced both by biology and related disciplines – whose main aim is fully understanding the CPG physiological structure and functionality – and by nonlinear dynamics, whose main aims are understanding the functionality of the underlying mechanisms and modeling with the simplest dynamical networks either a real structure (bio-inspired CPG) or just specific functionalities (synthetic CPG), either by resorting to group theory [18] or to multi-parameter bifurcation theory [19]–[21]. Moreover, the main aim of engineering is designing and implementing CPGs on embedded circuits for specific applications [12], [22], mainly in the fields of bio-robotics [23], [24] and rehabilitation [25], [26].

Then, the acronym CPG is used to denote both the real neuronal network (which in vertebrates can be composed of hundreds/thousands of neurons) and its model. Since in the real network there are groups of neurons that behave coherently and whose concerted activity can be modeled as a unique functional module (called in many ways, e.g., cell, unit, oscillator, neuron), the CPG intended as model is always composed of few cells. The complete CPG function is the result of neural circuits containing these modules as elementary blocks. Henceforth, unless otherwise stated, CPG will denote a model of a real neuronal network.

In this paper we propose a method for designing and analyzing CPGs, based on multi-parameter bifurcation theory. Of course, the method is independent of the tools used to implement it, but here we will use a recently proposed software tool (called CEPAGE) [27].

The proposed strategy is illustrated through two case studies, related to locomotion and gait transitions in quadrupeds, which are in turn novelty elements of this paper. The first case study is an 8-cell bio-inspired CPG controlling gaits in quadrupeds [28]. Despite the complexity of both the real CPG and its 40-cell model described in [28], the proposed 8-cell CPG model is able to capture the main functional behaviors of the real CPG. This has a twofold advantage: firstly, the simplification points out the role played by the main
components of the network; secondly, the reduced network has a lower computational complexity and then it can be exploited to improve simulation speed or to implement an embedded system able to mimic the network behavior in a real-time environment.

The second case study is a 4-cell purely synthetic CPG, which is designed to obtain the same gait transitions as before. In both cases, by following some prescribed steps, we obtain the desired gait transitions by acting on a bifurcation parameter modeling the supra-spinal networks driving.

This paper is structured as follows. Section II briefly describes the working framework, i.e., how a CPG can be modeled and the main features of CEPAGE. The proposed design and analysis strategy is described in Sec. III, whereas the two case studies are analyzed in Secs. IV and V. Finally, some conclusions are drawn in Sec. VI.

II. CPGs AND CEPAGE

In this section we briefly describe the building elements of a CPG model, the phase-difference representation used in CEPAGE to analyze them, and the main toolbox features.

A. CPG Models

A CPG model is basically defined by two elements:

- the cell, which can be a neuron model (e.g., Hodgkin-Huxley, FitzHugh-Nagumo, Morris-Lecar, Hindmarsh-Rose, integrate-and-fire models) or another oscillator (e.g., Kuramoto, Hopf, Van der Pol);
- the connections (synapses) between cells, which define the CPG topology and can be electrical, chemical inhibitory or chemical excitatory.

A third possible element (CPG input, when it does not work autonomously) is the brain-stem drive, which brings an input to the CPG from supra-spinal networks, allowing gait changes. In this paper, the synaptic actions have no dynamics and are modeled according to the fast threshold modulation paradigm [29], as follows:

\[
\begin{align*}
    h_{\text{in}}(V_i, V_j) &= \frac{E_{\text{in}} - V_i}{1 + e^{\nu(V_j - \theta)}} \\
    h_{\text{ex}}(V_i, V_j) &= \frac{E_{\text{ex}} - V_i}{1 + e^{\nu(V_j - \theta)}},
\end{align*}
\]

where \(E_{\text{in}}\) and \(E_{\text{ex}}\) are the inhibitory and excitatory synapses reverse potentials, respectively, whereas \(\nu\) and \(\theta\) act on the chemical synapses activation function shape.

B. Analysis Strategy

In this paper the CPG are analyzed following the so-called phase-lag or phase-difference representation of oscillatory or bursting cells coupled in a network [13], [14], [16], [30], [31], which allows checking the existence and stability of rhythmic patterns generated by the network by using standard tools of nonlinear dynamics. A first assumption underlying this method is that all cells remain oscillatory with relatively close temporal characteristics. This means that each i-th cell stays on a structurally stable periodic orbit \(\tilde{x}_i(t)\) of period \(T_i\) and that this orbit can be mapped (through the modulo function) to a phase variable \(\phi_i \in [0, 1)\) so that \(\phi_i\) is reset to 0 when \(V_i\) grows over a threshold \(V_{th}\).

The phase-difference representation of the network employs \(N - 1\) state variables describing phase differences between the reference cell 1 and the other network cells: \(\Delta \phi_{1i}(t) = (\phi_i(t) - \phi_1(t)) \mod 1\) (i = 2, ..., N). The time evolution of these state variables is unknown a priori and is usually determined numerically by integrating multiple initial conditions of (1) to reveal possible multi-stability.

From a numerical standpoint, the phase differences can be computed as follows. Let \(t_i(k)\) be the k-th time at which the membrane voltage \(V_i\) of the i-th cell overcomes the threshold \(V_{th}\). The phase lag \(\Delta \phi_{1i}(k)\) between the i-th cell and the reference cell 1 can be numerically computed as follows:

\[
\Delta \phi_{1i}(k) = \frac{t_i(k) - t_1(k)}{T_1} \mod 1,
\]

where \(T_1\) is the period of the first cell. As the time progresses these phase lags can converge and stabilize at some stable phase-locked states, possibly more than one (multi-stability of the network).

This representation is adopted also in Motiftoolbox [32].
CEPAGE is an object-oriented toolbox for simulation and analysis of CPGs [27]. It has a two-layer organisation: the outer layer is a MATLAB interface that makes it easy to specify the CPG configuration and offers tools for data analysis and visualization; the inner layer is used for numerical integrations and is based on Boost C++ libraries and on MEX files. The MATLAB layer provides flexibility to CEPAGE, since it makes it easy to add new neuron and synapse models to be simulated and new functionalities to the package by extending the base classes. Moreover, MATLAB allows the user to write very concise and clear scripts, which nonetheless retain the full power and speed of the underlying C/C++ code.

Figure 1 shows the functional relationships between classes (gray boxes), main methods (solid ellipses) and corresponding output data (white boxes). The dashed ellipses denote external analysis tools that can be applied to the obtained data. Parallel computation, MEX files and the Boost C++ libraries are used to reduce the simulation times. The classes neuron, synapse and CPG describe a single cell, a synapse and a CPG, respectively.

The main toolbox functionalities are:

- **Simulation of CPGs**: by using method `sim` of class `CPG`, the user can easily obtain the time evolution of the state variables describing the network; it is also possible to start parallel simulations from different initial conditions. If only one initial condition is considered, it is possible to use the `simpplot` method, which also plots the state evolution;

- **Limit cycle continuation**: this functionality is useful when one wants to detect limit cycle bifurcations; through the method `writeContinuationInterface`, it is possible to generate AUTO [33] or MATCONT [34] files for the limit cycle continuation;

- **CPG phase difference simulation**: the method `getPhaseRepresentation` of class `CPG` allows obtaining the evolution of the phase differences for the CPG cells; also in this case, parallel computations can be exploited to integrate the system starting from different initial conditions. The simulation results can then be plotted through the `plotPhaseSpace` method. This functionality can be used to obtain a brute-force bifurcation diagram of the phase differences, but turns out to be very time consuming for relatively large networks;

- **CPG approximate phase difference simulation**: the method `computeApproxVectorField` of class `CPG` is useful to carry out brute-force (i.e., based on numerical integrations and Poincaré sections [35]) analysis of the phase differences between cells reducing the simulation times. The approximate solution works accurately only for weakly-coupled networks and is computed starting from the so-called Phase Resetting Curve (PRC) [36], which can be computed through the method `computePRC` of class `neuron model`;

- **Phase difference continuation**: the approximate formulation allows also knowing the vector field that describes the phase difference evolution, making it possible a continuation analysis of the patterns generated by the network. CEPAGE can automatically generate files through the method `writeApproxVectorField`, which can be used to carry out continuation analysis with AUTO or MATCONT.

### III. Synthetic CPG Design Method

CEPAGE can be used as a tool to design a synthetic CPG able to generate some specific gaits typical of quadrupeds (i.e., trot, walk, bound, rotary gallop, transverse gallop) either by varying a bifurcation parameter in an assigned (e.g., bio-inspired) CPG with fixed structure or by designing (including the structure) a purely synthetic CPG. In both cases, our goal is finding – for the cells or synapses directly depending on the brain-stem drive through the parameter $\alpha$ – proper functions of $\alpha$ that allow obtaining the desired gaits and gait transitions. To this end, according to the framework described in Sec. II-A, we introduce an explicit dependence on $\alpha$ of some parameters and we choose piecewise-linear (PWL) functions, connecting points detected through bifurcation analysis.

Table I shows the main characteristics of each gait we want to achieve. The duty cycle properties of each gait are common for many quadrupeds, whereas gait amplitude and frequency depend on each specific animal. In this work (case study 1), we focus on the amplitude and frequency values typical for a mouse [28]. A representation of the different gaits is provided as supplemental material.

We assume that each limb is driven by a cell, then we will consider CPGs containing at least four cells. The proposed strategy can be used for any gait with left-right symmetry. It can be applied also to asymmetric gaits (possibly with few changes, as shown in Sec. V-E).

The proposed design steps to obtain a specific symmetric gait are as follows:

- **Step 1**: we analyze a simple structure (which appears more than once in an assigned CPG or is used as building
GAIT CHARACTERISTICS IN TERMS OF DUTY-CYCLE ($dc$) AND PHASE DIFFERENCES BETWEEN LEGS ($L$ = LEFT, $R$ = RIGHT, $F$ = FORE, $H$ = HIND)

<table>
<thead>
<tr>
<th>Gait</th>
<th>$dc$</th>
<th>$\Delta \phi_{RF-LF}$</th>
<th>$\Delta \phi_{RF-LH}$</th>
<th>$\Delta \phi_{RF-RH}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Walk</td>
<td>0.25</td>
<td>0.5</td>
<td>0.75</td>
<td>0.25</td>
</tr>
<tr>
<td>Trot</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Bound</td>
<td>0.65</td>
<td>0</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td>Transverse gallop</td>
<td>0.6</td>
<td>0.4(0.9)</td>
<td>0.7(0.6)</td>
<td>0.6</td>
</tr>
<tr>
<td>Rotary gallop</td>
<td>0.6</td>
<td>0.1(0.9)</td>
<td>0.4(0.7)</td>
<td>0.6</td>
</tr>
</tbody>
</table>

IV. CASE STUDY 1: BIO-INSPIRED CPG

In this section we will show how CEPAGE can be used to analyze a 8-cell CPG and to set its parameters in order to generate all the gaits listed in Table I.

In particular, in [28] a quadrupedal 40-cell CPG is described and analyzed, which is able to generate trot, walk and bound. The brain-stem drive acts directly on some CPG cells through the parameter $\alpha$, ranging in the interval $[0, 1]$.
A. Analysis

We analyzed the CPG behavior by varying the bifurcation parameter \( \alpha \), as in [28].

Each flexor cell eventually produces the same periodic spiking pattern, but with different phase. Figure 3 shows the spiking frequency \( f \) (upper panel) and duty cycle \( dc \) vs. \( \alpha \) for each flexor cell. It is evident that both \( f \) and \( dc \) increase with \( \alpha \) and this is perfectly coherent with the results reported in [28].

The stable phase differences \( \Delta \phi_{ij} \) \( (j = 2, 3, 4) \) vs. \( \alpha \) are shown in the three upper panels of Fig. 4. By varying \( \alpha \), the CPG is able to produce walk (region W), trot (region T) and bound (region B). These brute-force bifurcation diagrams have been obtained by using CEPAGE to simulate the CPG by increasing (black lines) and decreasing (gray lines) \( \alpha \) values. The comparison points out the presence of a bistability interval between regions T and B. The corresponding membrane voltages \( V_i(t) \) for the flexor cells over a window of 600ms are shown in the bottom panels, where the color code is the same as for the cells in Fig. 2.

Figure 5 shows how the bifurcation diagram for \( \Delta \phi_{12} \) changes by removing some synaptic connections. These results are coherent with biophysical experiments where some CPG cells are genetically ablated [43] and are completely similar to those obtained in [28] for the 40-cell CPG.

Because in our reduced model the cells removed in the original 40-cell CPG are no longer included, we modified the synaptic efficacies as follows (the reader not familiar with physiological details is referred to [28] for deeper insights about the removed cells/connections):

- \( V_0V \): we decreased the synaptic efficacies of the inhibitory connections between the flexor cells 1-2 and 3-4, since the cell \( V_0V \) in the 40-cell CPG is involved in one of the two possible inhibitory connections (the other connection involves the cell \( V_0D \)) between the considered flexor cells.
- \( V_0D \) and \( V_0V \): we removed the inhibitory connections between the flexor cells 1-2 and 3-4, for the reasons explained above.
- \( V3 \): we removed the excitatory connections between the flexor cells 1-2 and 3-4, for similar reasons.

In the upper panels, due to the lower strength of the inhibitory connections between left and right flexor cells, region T disappears, whereas region B is larger than in Fig. 4. Moreover, region TG appears, meaning that the quadruped can generate a transverse gallop gait. Black and gray lines have the same meaning as in Fig. 4 and reveal the presence of bistability in two transition regions.

In the last two cases the interpretation is quite direct: due to the absence of inhibitory (excitatory) connections between left side and right side, the CPG is able to generate only in-phase (anti-phase) patterns. This prevents the quadruped from producing bound (bottom-left panel) or walk and trot (bottom-right panel).
Summarizing, our analysis shows that the 8-cell CPG has the same behaviors as the 40-cell CPG and can produce up to four gaits (only three if we keep unchanged the CPG structure), among those listed in Tab. I. Now, we want to see if it is possible obtaining all the five gaits listed in the table, by taking the CPG structure fixed and acting only on the way $g_{ij}^{ex}$ and $D_1$ depend on the brain-stem parameter $\alpha$. So, after an analysis problem, now we face a design problem, following the steps described in Sec. III.

B. Step 1

We analyze a fore flexor-extensor pair (see the gray dashed rectangle in Fig. 2). Figure 6 shows the asymptotic flexor cell duty cycle $dc$ vs. parameter $D_1$ (see Appendix A, last equation of system (5)) of the same cell. The minimum and maximum $dc$ values we want to generate are 0.25 (walk) and 0.65 (bound). Then $D_1$ can range between 0.0043 and 0.09 and we define it as a non-decreasing PWL function $D_1(\alpha)$. This choice allows obtaining the same duty cycle for different values of $\alpha$ and, consequently, we can obtain different gaits sharing the same duty cycle. Figure 7 shows the chosen function $D_1(\alpha)$ in the considered example.

C. Step 2

We analyze the sub-structure within the gray solid box in Fig. 2, where the fore flexor-extensor pairs are identical. Through CEPAGE, we carry out a two-dimensional bifurcation analysis of the stable phase difference $\Delta \phi_{12}$ with respect to $\alpha$ and $g^{ex} (= g_{12}^{ex} = g_{21}^{ex})$. Figure 8 shows the obtained brute-force bifurcation diagram.

In the blue region, the (unique) stable equilibrium point has a phase coordinate $\Delta \phi_{12} = 0$ (in-phase). In the yellow region, the (unique) stable equilibrium point has a locked phase $\Delta \phi_{12} = 0.5$ (anti-phase). In the third intermediate region, instead, two stable equilibria coexist; the diagram shows the one with phase $0 < \Delta \phi_{12} < 0.5$. The second equilibrium (not shown) has phase $1 - \Delta \phi_{12}$. This is the reason because of the diagram colorbar ranging from 0 through 0.5. On the whole, we can obtain any phase difference between 0 and 1.

The red curves mark supercritical pitchfork bifurcations, obtained again through CEPAGE (brute-force approach).

We remark that, despite the fact that the presence of bistability makes the produced patterns less robust, for the asymmetric gaits we can obtain mono-stability by breaking the symmetry, as we will see below. For the symmetric gaits, the bistability just means that the limbs move in the reverse order, but the gait remains the same.

At this point, we can define a function $g^{ex}(\alpha)$ so as to have a continuous sequence of gaits. The chosen function is shown in Fig. 8 (black PWL curve).

D. Step 3

Step 3 is related to the analysis of the CPG sub-network within the black dashed box in Fig. 2. Cell 1 depends on...
α through $D_1(α)$, whereas cell 4 initially depends on two parameters through $D_4(α, ΔD) = D_1(α) + ΔD$.

CEPAGE provided the 2D bifurcation diagram shown in Fig. 9 for the equilibrium values of $Δφ_{14}$ with respect to $α$ and $ΔD$. By properly choosing $ΔD$ as a PWL function of $α$ connecting the values selected at the end of Step 1 (marked by black dots), we can obtain a function $ΔD_4(α)$ ensuring the desired phase lags $Δφ_{14}$ between fore and hind legs.

If we want to obtain symmetric gaits only, we can design the right part of the CPG as identical to the analyzed subnetwork. On the contrary, if we want to obtain also asymmetric gaits, we have to design differently the two sides. In particular, in this case study, we can define two functions $ΔD_3(α)$ and $ΔD_4(α)$ (one for each side, right/left) so as to have a continuous sequence of gaits. The chosen functions are shown in Fig. 9: the dashed red line is related to the left legs and the black solid line to the right legs.

E. Step 4

Finally, we check the designed CPG by carrying out the same bifurcation analysis as in Fig. 4, by setting $g_{12}^{cs} = g_{21}^{cs} = g_{34}^{cs} = g_{43}^{cs} = g_{43}^{ex}(α)$, $D_2(α) = D_1(α)$, $D_3(α) = D_1(α) + ΔD_3(α)$ and $D_4(α) = D_1(α) + ΔD_4(α)$, by using the PWL functions of $α$ obtained through the previous steps. The result is shown in Fig. 10. The upper panels show the bifurcation diagrams obtained by applying the chosen functions and point out the correct sequence of gaits. The bottom panels show the corresponding evolution of the steady-state membrane voltages $V_i(t)$ for the flexor cells in the five regions, over a window of 600ms and with voltages ranging in the interval $[-60, -10]$mV (the color code is the same as for the cells in Fig. 2). As pointed out in Sec III, each voltage has its own duty cycle, amplitude, frequency, and phase, which determine on the whole the corresponding gait.

V. Case Study 2: A Synthetic CPG

In this section we show how to design a 4-cell purely synthetic CPG in order to generate the same gaits as before. The chosen neuron model is the modified FitzHugh-Nagumo model described in [21] and reported in Appendix B for ease of reference. In this model, all variables are normalized and dimensionless. We use the synapse model (2), with $v = 0.3$, $θ = 0$, $E_{in} = -1.5$ and $E_{ex} = 1$.

In this case, we consider only the phase relationships between limbs for each gait, i.e., we focus on the times of maximum contact between limb and ground. For the sake of simplicity, in this example we neglect the duty cycle, which accounts for the duration of the contact. In other words, this network is only a rhythm generator, that would require either a more complex cell model or further cells (e.g., a pattern formation network and motor neurons, as proposed in [44]) to become a realistic CPG, able to modulate also duty cycles, amplitudes and frequencies of the cells driving flexor and extensor muscles. With this caveat in mind, henceforth the network will be called anyway CPG.

Our goal in this second case study is to design a synthetic CPG that, for a given parameter setting, produces only one stable motif, in order to ensure robustness for the generated pattern.

Some synapses are fixed whereas others depend on the bifurcation parameter $α ∈ [0, 1]$, in order to make the CPG able to switch between the desired gaits.

The complete CPG reference structure is shown in Fig. 11. Actually, the design strategy starts from a simpler block, i.e., the HCO within the gray dashed box. With respect to a standard HCO (containing only inhibitory synapses, light-gray connections ending with filled circles), here we add also excitatory synapses (ending with filled squares), whose strengths $g_{ij}^{ex}$ depend on $α$.

The second step in the design involves two HCOs (made up of cells 1-2 and 3-4), that are connected through the vertical gray inhibitory synapses.
Fig. 12. Maximum convergence time of the phase difference $\Delta\phi_{12}$ to the equilibrium point in the HCO for $g_{12}^{ex} = g_{21}^{ex} = 0$.

Fig. 13. (Color online) Step 1: two-dimensional bifurcation diagram for the excitatory synaptic efficacies of the HCO (gray dashed box in Fig. 11).

Step 3 involves also the dark-gray inhibitory (or excitatory) synapses, whose strengths $g_{ij}^{in}$ depend on $\alpha$. Each step requires some analysis (carried out with CEPAGE), which is described in detail in the following.

A. Step 1

First of all, we set the strength of the inhibitory synapses, which will be taken as a reference for the whole design process. Since the HCO has always a stable equilibrium point for the phase difference, we can set the synaptic efficacy $g_{ij}^{in}$ (the same for both connections $1 \rightarrow 2$ and $2 \rightarrow 1$) according to the desired convergence time scale. Figure 12 shows the maximum convergence time of the phase difference $\Delta\phi_{12}$ to the equilibrium point for $g_{12}^{ex} = g_{21}^{ex} = 0$. We choose $g_{ij}^{in} = 4$ in order to have convergence times in the scale of some normalized units of time.

Now, we have to set the strengths of the excitatory synapses. To this end, we obtain a two-dimensional bifurcation diagram showing the equilibrium phase difference $\Delta\phi_{12}$ with respect to $g_{12}^{ex}$ and $g_{21}^{ex}$ (see Fig. 13). White pixels mark the presence of multiple stable equilibria. The white region is due to the presence of a subcritical pitchfork bifurcation along the main diagonal, which degenerates in a fold bifurcation outside the diagonal (due to symmetry breaking).

Points W/T, TG, B in the figure mark the pairs chosen to reproduce different gaits with the complete CPG, on the basis of the corresponding left-right phase difference (see Table I): walk and trot (W/T, anti-phase LR alternation), bound (B, in-phase LR alternation), transverse gallop (TG, almost in-phase LR alternation). Figure 14 shows the chosen functions $g_{12}^{ex}(\alpha)$ (black solid line) and $g_{21}^{ex}(\alpha)$ (gray dashed line).

B. Step 2

The bottom HCO is identical to the top one, with $g_{12}^{ex}$ and $g_{21}^{ex}$ set to point A, in order to have left-right alternation with $\Delta\phi_{12} = 0.5$. Here, we analyze the CPG behavior changes with respect to the strength $g$ of the 4 mid-gray inhibitory synapses shown in Fig. 11.

The 1D bifurcation diagram in Fig. 15 shows the equilibrium value of the phase difference $\Delta\phi_{14}$ with respect to $g$. The bifurcation diagram contains three regions, whose edges are marked by dashed vertical lines. In the left region there is no phase locking (i.e., the CPG works out of an Arnold tongue), due to the too low value of $g$. In the right region, the $g$ strength approaches $g_{ij}^{in} = 4$ and further stable equilibria appear, thus producing undesired multi-stability.

Then we set $g$ to a value within the central region. In order to ensure structural stability, we choose $g = 2$.

C. Step 3

Now we want to set the strength $g_{c}$ of the two dark-gray inhibitory synapses shown in Fig. 11 in order to generate all the desired front-hind alternations, corresponding to different rhythms.

To this end, Fig. 16 provides one-dimensional bifurcation diagrams showing the stable equilibrium phase differences $\Delta\phi_{12}$ and $\Delta\phi_{14}$ with respect to $g_{c}$, for the HCO configured in the points W, T (black lines), TG (gray lines), B (light-gray lines) in Fig. 13.

We want to ensure that $g_{c}$ is set to a value that (i) does not alter the existing LR phase difference and (ii) provides the desired FH phase difference. The upper bifurcation diagram in Fig. 16 shows the actual stable equilibrium phase differences $\Delta\phi_{12}$ versus $g_{c}$ (solid lines) and those set during step 1...
Fig. 16. Step 3: one-dimensional bifurcation diagram showing the stable equilibria with respect to $g_c$ in the case of inhibitory synapses.

Fig. 17. Step 3: one-dimensional bifurcation diagram showing the stable equilibria with respect to $g_c$ in the case of excitatory synapses.

(cross markers). It is evident that for the parameter settings $W$, $T$ and $TG$ the desired equilibrium value of $\Delta \phi_{12}$ is kept for any $g_c$, whereas for the parameter setting $B$ only $g_c$ values lower than about 0.2 allow keeping the desired equilibrium value of $\Delta \phi_{12}$.

About condition (ii), from the lower bifurcation diagram we deduce that for the parameter setting $W$, $T$ (black line) we can only have a delay between fore and hind limbs ($\Delta \phi_{14}>0.5$) and acting on $g_c$ we can control this delay over a reasonable interval (with $\Delta \phi_{14}$ ranging from 0.5 to about 0.8). On the contrary, for the other two settings we can only have an advance of the fore limb with respect to the hind limb ($\Delta \phi_{14}<0.5$) and acting on $g_c$ we can control this delay over a small interval (with $\Delta \phi_{14}$ ranging from about 0.4 to 0.5).

If the nature of the synaptic connections is changed to excitatory, we obtain the bifurcation diagrams shown in Fig. 17. A direct comparison of Figs. 16 and 17 makes it evident that the two kinds of connections have a complementary effect. This suggests that in the case $W$, $T$ inhibitory connections can be favorably used to obtain a prescribed delay between fore and hind limbs, whereas excitatory connections are better to obtain a prescribed advance. Similarly, in the cases $TG$ and $B$ inhibitory (excitatory) connections can be used to obtain a prescribed advance (delay).

Among the allowed $g_c$ values, we choose the one corresponding to the equilibrium value of $\Delta \phi_{14}$ closest to the desired rhythm (see Table 1), thus obtaining the functions $g_{13}^{in}(\alpha) = g_{24}^{in}(\alpha)$ (black solid curve) and $g_{13}^{ex}(\alpha) = g_{24}^{ex}(\alpha)$ (gray dashed curve) shown in Fig. 18.

D. Step 4

Figure 19 shows the stable equilibrium values of the phase differences (upper panel) and the time evolution of the normalized membrane voltages (lower panels) by changing $\alpha$ to obtain the desired rhythms: walk (region $W$), trot (T), transverse gallop (TG), bound (B).

The width of the time axes in the lower panels is 50 units of time (notice that the model used in this case study, described in Appendix B, is normalized and uses dimensionless variables).

E. Asymmetric Rhythms

If we want to add to the rhythm sequence also asymmetric rhythms, the procedure described for step 3 in the case of symmetric rhythms must change. In the complete CPG, cells 1 and 2 are initially assumed to be not connected, in order to avoid LR synchronization, whereas cells 3 and 4 remain connected through the synapses with PWL functions $g_{34}^{ex}(\alpha) = g_{12}^{ex}(\alpha)$ and $g_{43}^{ex}(\alpha) = g_{21}^{ex}(\alpha)$ (see Fig. 14).

Now we obtain again a bifurcation diagram with respect to $g_c$ (as in Figs. 17 and 18), to choose proper values of $g_c$ and a proper PWL function $g_c(\alpha)$.

Finally, we choose proper values of $g_{12}^{ex}$ and $g_{21}^{ex}$ (as in Fig. 13) and related PWL functions $g_{12}^{ex}(\alpha)$ and $g_{21}^{ex}(\alpha)$ (as in Fig. 14), by keeping unchanged $g_{34}^{ex}(\alpha)$ and $g_{43}^{ex}(\alpha)$.

Figure 20 shows the stable equilibrium values of the phase differences (upper panel) and the time evolution of the neuron voltages (lower panels) by changing $\alpha$ to obtain the desired rhythms: walk (region $W$), trot (T), transverse gallop (TG), rotary gallop (RG), and bound (B).
The main features of the proposed design strategy can be summarized as follows:

- parallel development of analysis and design, based on multi-parameter bifurcation theory;
- combination of local analysis (and related design of some local properties/parameters of the CPG) and global analysis, to ensure structural stability of the overall system;
- use of a bifurcation parameter modeling the brain-stem drive coming from the supra-spinal networks to properly govern gait transitions through the nonlinear functions \( g_{ij}^a(\alpha) \).

The method has been applied to model with relatively simple dynamical networks either a real structure (first case study, reduced-complexity version of a bio-inspired CPG) or just specific quadrupeds’ functionalities (second case study, synthetic CPG), by resorting to the toolbox CEPAGE for efficient numerical analysis. After proper robustness analysis with respect to cell and synapse models and after properly relating the parameter \( \alpha \) to sensory inputs (in order to introduce also an effective closed-loop control provided by the CPG), the obtained results can find applications in the fields of bio-robotics [23], [24] and rehabilitation [25], [26]. Moreover, we will have to introduce a direct sensory feedback to properly adjust the gait in the presence of mechanical perturbations, for instance, if one leg cannot find a foothold [45], [46].

To conclude, we briefly address the physical implementation problem related to applications. As pointed out in [12], a CPG-based locomotion control is usually programmed in software and running on hardware (microcontroller, DSP, FPGA or dedicated hardware). Providing an overview on possible hardware implementations, which (except purely digital solutions) depend on the specific choice of cell and synapse models, is out of the scope of this paper. About this issue, the reader is kindly referred to surveys such as [12] and [47] or to specific studies related to the cited applications [48], [49].

### APPENDIX A

#### CASE STUDY 1

The model employed in the first case study is [28]

\[
\begin{align*}
C \frac{dV_i}{dt} &= -I_{Na} - I_L - I_D^{(\alpha)}(\alpha) + I_{syn}^{(i)} \\
\frac{dh}{dt} &= h_\infty - h \\
I_L &= g_L \cdot (V_i - E_L) \\
I_{Na} &= g_{Na} \cdot m \cdot h \cdot (V_i - E_{Na}) \\
m &= \left(1 + e^{\frac{V_i - V_{Na}}{k_m}}\right)^{-1} \\
h_\infty &= \left(1 + e^{\frac{V_i - V_{Na}}{k_h}}\right)^{-1} \\
\tau &= \tau_0 + \frac{\tau_M - \tau_0}{\cosh(\frac{V_i - V_{Na}}{k_t})}
\end{align*}
\]

The synapses parameters are \( \nu = 0.3 \, mV^{-1}, \theta = -30 \, mV \), \( E_{ex} = -10 \, mV \) and \( E_{inh} = -75 \, mV \), whereas the constant synaptic strengths are listed in Tab. II.

### APPENDIX B

#### TABLE II

<table>
<thead>
<tr>
<th>Connections</th>
<th>Value [mS]</th>
</tr>
</thead>
<tbody>
<tr>
<td>( g_{11}^{(D)} ), ( g_{12}^{(D)} ), ( g_{21}^{(D)} ), ( g_{22}^{(D)} )</td>
<td>0.228</td>
</tr>
<tr>
<td>( g_{11}^{(D)} )</td>
<td>2.853</td>
</tr>
<tr>
<td>( g_{12}^{(D)} )</td>
<td>0.0221</td>
</tr>
<tr>
<td>( g_{21}^{(D)} )</td>
<td>0.298</td>
</tr>
<tr>
<td>( g_{22}^{(D)} )</td>
<td>0.448</td>
</tr>
<tr>
<td>( g_{11}^{(D)} )</td>
<td>0.1272</td>
</tr>
<tr>
<td>( g_{12}^{(D)} )</td>
<td>0.0545</td>
</tr>
</tbody>
</table>

\[ V_i(t) \text{ for the CPG cells in the four regions, ranging in the interval } [-1, 1] \text{ over a window of 50 units of time (color code as for the cells in Fig. 11).} \]

We remark once more that in this case (contrary to the first case study), we focused on the phase differences only, since amplitudes, frequencies and duty cycles of the cell voltages can be properly modulated only by using a more complete CPG model. Inasmuch as this paper is focused on the design method, the cell model was used as is and the method was applied in order to make the CPG generate the correct phase differences. This is the reason why in this case the voltages \( V_i(t) \) differ in the phase only.
The synthetic strengths of the excitatory synapses depend on $\alpha$ as follows:

\[
\begin{align*}
\sigma_{12}^{\alpha} &= 8 \sigma_{21}^{\alpha} = 115.98 a^{10} - 231.71a^9 + 25.54a^8 \\
+ 329.37a^7 - 407.13a^6 + 235.88a^5 - 76.053a^4 \\
+ 13.751a^3 - 1.1155a^2 + 0.16808 \quad (6)
\end{align*}
\]

## APPENDIX B

### CASE STUDY 2

The model used in the second case study is [21]

\[
\begin{align*}
\frac{dV_i}{dt} &= V_i - V_i^3 - x_i + I + \beta I_{syn}^{(i)} \\
\frac{dx_i}{dt} &= \left( \frac{1}{1 - e^{-10V_i}} - x_i \right) \\
\end{align*}
\]

where $I = 0.5$, $\beta = 10^{-3}$ and $\epsilon = 0.3$.

## ACKNOWLEDGMENTS

Marco Storace would like to thank his colleague and friend Federico Bizzarri for stimulating discussions. The authors also thank Simon Danner for providing benchmark results for case study 1.

## REFERENCES


Matteo Lodi was born in Genoa, Italy, in 1991. He received the Laurea (M.Sc.) five-year degree (*summa cum laude*) in electronic engineering from the University of Genoa, Italy, in 2015, where he is currently pursuing the Ph.D. degree in electrical engineering. He was a Visitor to Georgia State University, Atlanta, USA, in 2016. His main research interests are in the area of modeling of nonlinear systems (hysteresis and networks of biological neurons), bifurcation analysis, and nonlinear dynamics.

Andrey Shilnikov was born in Nizhny Novgorod, Russia, in 1962. He received the M.S. degree in mathematics and physics and the Ph.D. degree in differential equations incl. mathematical physics from the University of Nizhny Novgorod, Russia, in 1984 and 1990, respectively. He was a Post-Doctoral Fellow with UC Berkeley from 1993 to 1996 and a Royal Society Post-Doctoral Fellow with Cambridge University, U.K., from 1994 to 1995. He held visiting positions at UC Berkeley, Georgia Institute of Technology, and Cornell University. In 2000, he joined Georgia State University (GSU), where he is currently a Professor of applied mathematics and mathematical neuroscience with a joint appointment at the Neuroscience Institute and the Department of Mathematics and Statistics. He is also a Faculty Member with the Center for Nonlinear Science, Georgia Institute of Technology, and a member of the Center for Behavioral Neuroscience, GSU. His original area of expertise is the theory of applied dynamical systems and global bifurcations. He studies dynamics and their origin in diversely phenomenological systems and in exact models from life sciences. Of his special interest is a new emergent cross disciplinary field known as mathematical neuroscience. Its scopes include nonlinear models of individual neurons and networks. His laboratory develops advanced mathematical tools paired with sophisticated computations. He is an author of about 100 scholarly publications, including several advanced textbooks on dynamical systems. He presented many plenary and invited talks at various meetings and colloquium talks at national universities and around the globe, and co-organized more than 30 conferences, workshops, and special sessions nation- and worldwide. He currently serves on the Editorial board of the *Journal of Mathematical Neuroscience*, the *Journal of Frontiers of Applied Mathematics*, and the *Journal of Discontinuity, Nonlinearity and Complexity*.

Marco Storace (M’01–SM’14) was born in Genoa, Italy, in 1969. He received the Laurea (M.Sc.) five-year degree (*summa cum laude*) in electronic engineering and the Ph.D. degree in electrical engineering from the University of Genoa in 1994 and 1998, respectively. He was a Visitor to EPFL, Lausanne, Switzerland, in 1998 and 2002, respectively. Since 2011, he has been a Full Professor with the Department of Electrical, Electronic, Telecommunications Engineering and Naval Architecture, University of Genoa. He is the author or co-author of about 130 scientific papers, more than an half of which have been published in international journals. His main research interests are in the area of nonlinear circuit theory and applications, with emphasis on (circuit) models of nonlinear systems (e.g., hysteresis and biological neurons), methods for the piecewise-linear approximation of nonlinear systems and for the consequent circuit synthesis, bifurcation analysis, and nonlinear dynamics. He is a member of the IEEE Technical Committee on Nonlinear Circuits and Systems. He served as an Associate Editor of the *IEEE Transactions on Circuits and Systems II* from 2008 to 2009.