Bioinspiration & Biomimetics

PAPER

Towards autonomous locomotion: CPG-based control of smooth 3D slithering gait transition of a snake-like robot

To cite this article: Zhenshan Bing *et al* 2017 *Bioinspir. Biomim.* **12** 035001

Bioinspiration & Biomimetics

RECEIVED
29 December 2016

REVISED
28 February 2017

ACCEPTED FOR PUBLICATION
3 March 2017

PUBLISHED
4 April 2017

PAPER

Towards autonomous locomotion: CPG-based control of smooth 3D slithering gait transition of a snake-like robot

Zhenshan Bing[1], Long Cheng[1], Guang Chen[1], Florian Röhrbein[1], Kai Huang[2,3] and Alois Knoll[1]

[1] Fakultät für Informatik, Technische Universität München, Germany
[2] Sun Yat-Sen University, People's Republic of China
[3] Author to whom any correspondence should be addressed

E-mail: bing@in.tum.de, chengl@in.tum.de, guang@in.tum.de, florian.roehrbein@in.tum.de, huangk36@mail.sysu.edu.cn and knoll@in.tum.de

Keywords: snake robot, slithering, gait transition, CPG

Abstract

Snake-like robots with 3D locomotion ability have significant advantages of adaptive travelling in diverse complex terrain over traditional legged or wheeled mobile robots. Despite numerous developed gaits, these snake-like robots suffer from unsmooth gait transitions by changing the locomotion speed, direction, and body shape, which would potentially cause undesired movement and abnormal torque. Hence, there exists a knowledge gap for snake-like robots to achieve autonomous locomotion. To address this problem, this paper presents the smooth slithering gait transition control based on a lightweight central pattern generator (CPG) model for snake-like robots. First, based on the convergence behavior of the gradient system, a lightweight CPG model with fast computing time was designed and compared with other widely adopted CPG models. Then, by reshaping the body into a more stable geometry, the slithering gait was modified, and studied based on the proposed CPG model, including the gait transition of locomotion speed, moving direction, and body shape. In contrast to sinusoid-based method, extensive simulations and prototype experiments finally demonstrated that smooth slithering gait transition can be effectively achieved using the proposed CPG-based control method without generating undesired locomotion and abnormal torque.

1. Introduction

To meet the growing need for robotic mobility implementations such as disaster rescue, factory pipe maintenance, and terrorism surveillance, a considerable number of snake-like robots for ground locomotion have been developed in the past decades [1–3]. Early versions of snake-like robots were equipped with passive wheels, which could achieve stable and fast planar locomotion by swinging their bodies. Such robots, however, lacked the ability of moving in varied topography [2, 4]. Due to this drawback, more attention has been focused on snake-like robots with 3D locomotion ability. By changing the internal shape of their bodies, snake-like robots with lateral and dorsal connected modules can achieve 3D locomotion, which make them more adaptive to different kinds of terrain [3, 5].

Although the locomotion control of snake-like robots has been widely investigated, many crucial problems still exist. An indispensable aspect of autonomous locomotion, smooth gait transition, is one of the unsolved problems [6, 7]. Unsmooth gait transition could cause undesired locomotion, like deflecting direction or even movement failure. The reason is that the transition process requires another sinusoid wave which may have different phases, amplitude, or frequency. Such changes on the sinusoid wave could lead to discontinuous commands for the joint position control. Those discontinuous joint positions would have an impact on maintaining the desired locomotion mode. A more serious problem, the accompanying high torque produced during the gait transition may damage the gearbox and result in high energy consumption. Therefore, it is desirable to control smooth gait transition without generating undesired movements and abnormal torque [7]. Unlike other developed gaits for 3D locomotion snake-like robots, the slithering gait (figure 1) is regarded as a promising gait for autonomous locomotion since the moving direction is aligned with the body length direction. Vision sensors in the head module can directly reflect the vis-

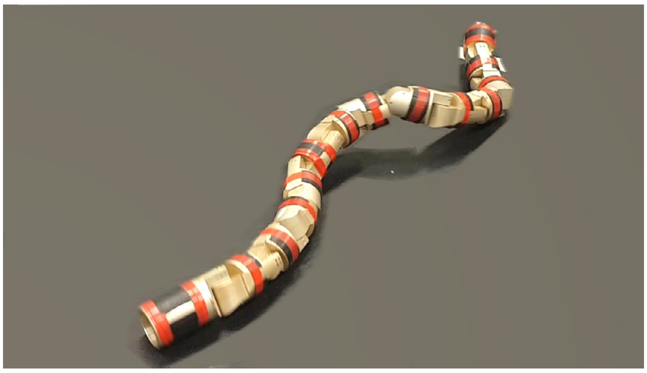

Figure 1. The snake-like robot is slithering forward. This snake-like robot is modular designed. Each joint axis is orthogonal to its neighbors with a rotation range of ±90°.

ual information of the upcoming environment during locomotion. Normally, basic autonomous locomotion scenarios require the changing of speed, moving direction, and body shape to avoid or pass through obstacles. Therefore, gait transition processes are more frequently demanded in slithering locomotion implementations.

As a promising solution for smooth gait transition, the CPG-based control method can generate smoothly self-adjusted rhythmic signals for cyclic motions, particularly suitable for snake-like robots with redundant degrees of freedom [8]. Although several CPG-based methods have been adopted to control the planar movement of snake-like robots under several gaits [7, 9, 10], the effectiveness of such methods to smooth the 3D slithering gait transition of snake-like robots has not yet been fully studied. The reasons are multi-fold.

First of all, 3D slithering is more complex than slithering in the 2D plane because it requires two waves in the lateral and dorsal planes to drive the robot [11]. This complexity leads to unstable gait transition of the robot. Additionally, when the robot is slithering, the body forms itself into a cylinder shape along the longitudinal direction. As a consequence, the robot suffers from the unbalanced geometry and swings the head module in a wide range, which decreases the stability of the vision sensor inside [12]. Stability of the vision sensor is vital for obtaining environment information, a central issue for autonomous locomotion [13]. Secondly, the CPG needs time to compute the operating patterns, usually on MCU (micro control unit) with limited computing power [14, 15]. The length of this computing time would directly affect the control performance. For snake-like robots with redundant degrees of freedom, the computing load is more demanding as greater degrees of freedom means more CPGs to be calculated on the microprocessor with limited computing power [16]. Last, the abnormally high torque generated by the unsmooth gait transition process is difficult to measure and unfounded in literature. However, this high torque is directly related to the locomotion stability and energy efficiency [7]. Hence, these issues make it challenging to study smooth slithering gait transition control and demonstration.

To investigate a smooth slithering gait transition of 3D snake-like robots, this paper studies a CPG-based control strategy. In particular, a random moving target tracking scenario is simulated to examine the smooth gait transition process. In terms of the smoothness, the robot trajectory and the output torque are targeted, which can reflect locomotion accuracy and energy consumption. The contributions of this work are summarized as follows.

- Based on the gradient theory, a lightweight CPG model with fast computing time is designed to control the 3D locomotion of a snake-like robot. Our CPG model is at least three times faster than the other widely adopted CPG models in the literature. An amplitude bias term is integrated in the proposed CPG model to change the signal's amplitude center.
- By shaping the body into a more stable geometry, the slithering gait is modified and achieved by the proposed CPG model. As autonomous locomotion frequently requires gait transition processes, we simulate a scenario in which the snake robot locks on and tracks a random moving object, combining the camera sensor in the head module.
- We first conducted prototype measurements on the output torque for the slithering gait transition process. Combined with the trajectory simulation results, the control method based on the proposed CPG model can effectively avoid undesirable movement and abnormal torque.

The remainder of this paper is organized as follows: section 2 briefly presents related work. The CPG mathematical model is deducted and analyzed in section 3. Smooth gait transition processes are analysed compared CPG-based method to sinusoid-based method and then slithering gait is modeled in section 4. Torque and trajectories of the slithering gait are simulated in a tracking scenario in section 5. In section 6, the mechanical and electronic hardware of our snake-like robot are introduced and extensive experiments results are presented. Section 7 concludes this paper with the discussion and the presentation of future work. Similar gait results are presented in the appendix.

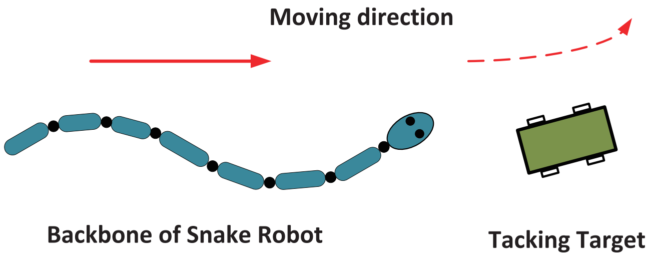

Figure 2. Scenario of snake-like robot tracking a moving target.

2. Related work

As a potential gait for autonomous serpentine locomotion [17, 18], slithering is highlighted due to fact that the moving direction is aligned with the body length direction [19], which can help the robot obtain the environment information during locomotion as shown in figure 2. Through adopting this gait, robots are able to lift parts of their body off the ground [19], which be achieved by 3D locomotion, as opposed to planar slithering robots that swing forward by using passive wheels [4, 20, 21].

Inspired by the morphology of real snakes, the gait of snake-like robots is usually controlled with serpenoid curves, which was firstly presented by Hirose [22]. To generate these serpenoid curve signals, the locomotion control architectures can be classified into three types: sinusoid-based, model-based, and CPG-based. The most widely adopted control strategy is the sinusoid-based method [23].

Through abundant experiments, the sinusoid-based method has been proven simplicity for imitating real snake locomotion and generating gaits of snake-like robots [12, 24, 25]. The ACM-R3 [2], controlled by active cord mechanism, consisted of 20 links and was capable of only 2D motion. The diverse locomotion was accomplished by propagating a wave in the form of the serpenoid curve throughout the robot.

For those snake-like robots with 3D locomotion ability, Choset [26] proposed a parameterized gait equation by simplifying the serpenoid curve into sine functions, which can achieve a variety of gaits for different application scenarios. Based on gait equation, Melo [27] conducted numerous experiments to select appropriate parameters for repeatable gaits under mechanical constraints and theoretical rules for modular snake robots. According to the experiments results, he proposed several indoor and outdoor gaits for a modular snake robot [27].

One major strength of the sinusoid-based approach is its simplicity because the important parameters that influence the gaits are predefined, such as phase, frequency, and amplitude. However, the sinusoid-based method inherently depends on time, which may cause undesirable movements during gait transition [6]. The reason is that the transition process requires a change in the aforementioned parameters of control signals. Efforts have been made to use the curve-based approach for the gait transition problem. Tanaka [28] achieved smooth gait transition of the body shape by sequential control of the shape controllable points. However, the approach was only validate for snake robots move in a two-dimensional plane.

Model-based control method requires an accurate mathematica model of the kinematic [29] and dynamic [30] of robot as well as the friction model [31] between the robot and the environment. The advantage of this method is that the robot can make an accurate move based on the torque calculation. While strongly depending on the accurate model, this method lacks the ability of adaptation. The control will fail if the model is not accurate enough, or the snake is in the environment having uncontrolled factors. Therefore, this method does not work well for locomotion in uncertain environment.

As CPGs can autonomously produce rhythm signals without external input [16], researchers have successfully implemented CPG knowledge to robotic locomotion [9, 10, 32, 33] focusing on different aspects. For instance, using the Kuramoto oscillator [34] as a CPG neuron, Crespi [35] built anguilliform swimming salamander robots to achieve the switch of swimming and crawling gaits. Ma [4] used a cyclic inhibitory CPG network with feedback to control the serpentine locomotion of a snake-like robot. Then, Lu [36] extended Ma's work to achieve multi-gaits transition CPG-based controller. But the transition process was not mentioned. Wu [37] added sensory inputs to the CPG-based locomotion control system to investigate steering for collision avoidance behaviour of a snake robot. By adopting the Hopf oscillator as the CPG model, Seo and Slotine [38] developed an open-loop CPG model for a turtle-like robot while successfully generating reference trajectories for fin motions. Except for those widely adopted CPG models, the van der Pol oscillator [39], the Matsuoka oscillator [40], and the Rayleigh oscillator [41] have also been implemented in many field robots [42]. However, the online execution time of those CPG models have been rarely investigated.

The greater flexibility and robustness of the CPG network could result in more general design strategies for autonomous locomotion [8]. Ma [7] developed an activation function in the CPG model to control the body shape and validated the effectiveness by simulating the joints' torque. The drawback of this work is a lack of supporting prototype experiments and their

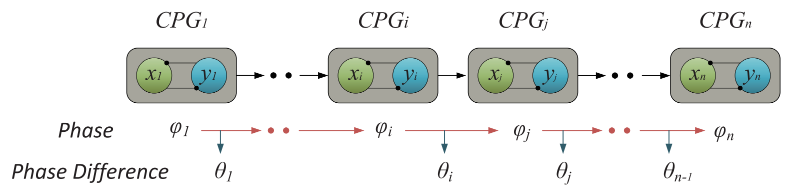

Figure 3. Parameters setting of CPG net with chain-type.

snake-like robot only moves in a 2D plane. Dorge [6] stated that smooth transition may not be insufficient to ensure desired movement. The decentralized controller used by Dorge propagated the second gait down the snake-like robot, during the gait transition process. However, these authors do not present the assessment criteria for effective and repeatable transition gaits or for supported prototype experiments in their paper.

As described above, this paper focuses on a CPG-based control method for smoothing 3D slithering gait transition. To achieve this, a lightweight CPG model is designed and compared to those in literature. We also model and investigate slithering gait in a moving object tracking scenario toward autonomous locomotion.

3. CPG model based on gradient method

The chain-type CPG network is one of the most common topological structures. This structure forms a chain of oscillators coupling the neighbor oscillators [43].

Inspired by Kopell [44], we design an oscillator model for the chain-coupled CPG network based on our previous work [45]. The convergence behavior of the CPG network is based on the gradient system, which can adjust the output signal's frequency, phase difference, and amplitude as required. More importantly, the model is lightweight with less online execution time.

3.1. Mathematical model of CPG network

The concept of *Gradient System* comes from the gradient field [46]. For any point M in space region G, there is a certain scalar function $V(M)$ corresponding to point M. Then, $V(M)$ is one certain scalar field in space region G. If there is a gradient function grad$V(M)$ corresponding to the point M, then a gradient field can be determined. The gradient field is generated by the scalar field $V(M)$, called the potential of the gradient field. In space region G, the potential of any point decreases against the gradient direction , as shown in (1).

$$\frac{\mathrm{d}x}{\mathrm{d}t} = -\frac{\partial V}{\partial x}, \quad \vee x \in G \tag{1}$$

If the gradient system has a minimal value x^* in G, then all the vector x will converge to x^*, which satisfies:

$$\left.\frac{\partial V}{\partial x}\right|_{x=x^*} = 0, \quad \left.\frac{\partial^2 V}{\partial x^2}\right|_{x=x^*} > 0 \tag{2}$$

The gradient system has an important property; any initial state x_0, will converge to the minimal value in region G during finite time and remain stable since then.

According to (2), we can design a chain-type CPG network to realize a fixed phase difference among all the CPG models. The phase differences among all CPGs can be seen as a group of state vectors. A global convergence gradient system is assumed in G, which is composed of those state vectors. The vectors in G will converge to the extreme point of the gradient system in finite time. If the extreme point of the gradient system is different from the target phase vector we set, the CPG network will converge to the desired phase difference from any position in finite time. As shown in figure 3, the chain-type CPG network is composed of n neurons with the same parameters. Suppose that the phase of the ith CPG is $\varphi_i(t)$, and the phase difference between two neighbouring CPGs ith, jth is $\theta(t)$, then the desired phase difference decided by the gait generator is $\tilde{\theta}_i$.

Consider the chain-type CPG network as a directed graph. The topological structure of the chain-type CPG network can be described by the incidence matrix. Set the incidence matrix $T = \{a_{i,j}\}_{(n-1)\times n}$ and satisfies:

$$\{a_{i,j}\} = \begin{cases} -1, & \text{from } j \text{ to } i \\ 1, & \text{from } i \text{ to } j \\ 0, & i \text{ and } j \text{ are not adjacent} \end{cases} \tag{3}$$

Therefore, the incidence matrix T for chain-type CPG network is

$$T = \begin{bmatrix} 1 & -1 & & & 0 \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ 0 & & & 1 & -1 \end{bmatrix}_{(n-1)\times n} . \tag{4}$$

Thus, the relationship between the phase difference and the phase vector is as follows:

$$\Theta = T\Phi \tag{5}$$

where phase difference vector is $\Theta = [\theta_1, \theta_2, ..., \theta_{n-1}]^T$, the phase vector is $\Phi = [\varphi_1, \varphi_2, ..., \varphi_{n-1}, \varphi_n]^T$.

In order to design the potential function, Ψ_i is used as the generalized coordinates of the gradient system, which satisfies:

$$\Psi_i = \begin{cases} \varphi_1 - \varphi_2 = \theta_1, & i = 1 \\ \varphi_{n-1} - \varphi_n = \theta_n, & i = n-1 \\ \varphi_{i+1} + \varphi_{i-1} - 2\varphi_i = \theta_{i-1} - \theta_i, & \text{otherwise} \end{cases} \tag{6}$$

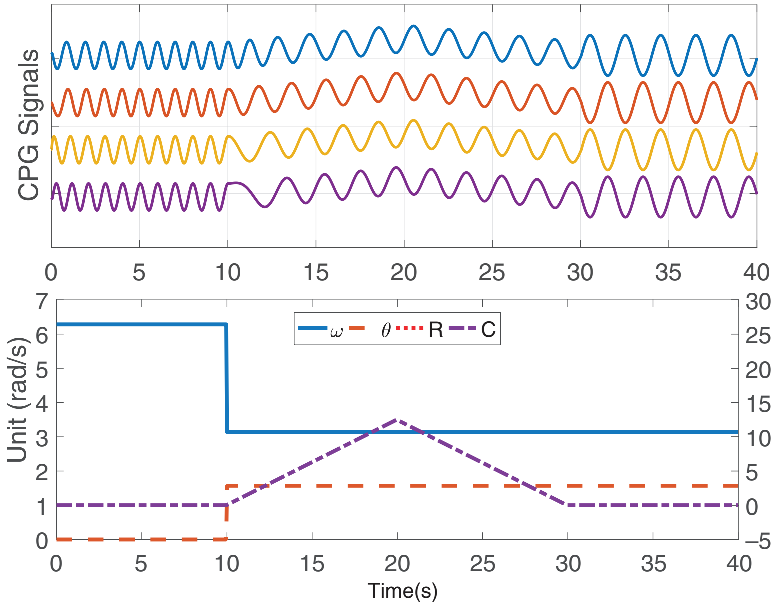

Figure 4. CPG output waves with changing parameters. The upper figure shows four CPG output signals. The bottom figure shows different sets of CPG parameters. Phase difference changes from 0 to π, with θ changing from 0 to π, at $t = 10$ s. Frequency changes from 1 Hz to 0.5 Hz, with ω_i changes from 2π to π, at $t = 10$ s. Amplitude bias rises up from 0 to 12 at $t = 20$ s, and then falls off to 0 at $t = 30$ s. Amplitude goes up from 20 to 30 at $t = 30$ s.

Then the potential function of a parabolic system is as follows:

$$V(\Psi) = \sum_{i=1}^{n} \mu_i (\psi_i - \psi_i)^2 \tag{7}$$

where μ_i is the coefficient of the convergence velocity and ψ_i is the generalized coordinates of the desired phase differences. According to (1), $V(\Psi)$ can be seen as the gradient system. ψ_i is the desired phase difference, represented in the new coordinate constructed by vector Ψ. Thus, the gradient system described by the new coordinate Ψ is:

$$\frac{\mathrm{d}V(\Psi)}{\mathrm{d}t} = -\frac{\partial V(\psi_1, \psi_2, \ldots, \psi_n)}{\partial(\psi_1, \psi_2, \ldots, \psi_n)} \tag{8}$$

Transfer the equation into the original coordinate Θ, we get:

$$\frac{\mathrm{d}V(\theta_i)}{\mathrm{d}t} = -\frac{\partial V(\Psi)}{\partial \theta_i} = -\sum_{i=1}^{n} \frac{\partial V(\Psi)}{\partial(\psi_i)} \frac{\partial \psi_i}{\partial \theta_i} \tag{9}$$

Then we can expend (9) into:

$$\frac{\mathrm{d}\theta_i}{\mathrm{d}t} = \begin{cases} -2\mu_1(\theta_1 - \tilde{\theta}_1) - 2\mu_2(\theta_1 - \theta_2 - \tilde{\theta}_1 + \tilde{\theta}_2), & i = 1 \\ 2\mu_{n-1}(\theta_{n-1} - \theta_n - \tilde{\theta}_{n-1} + \tilde{\theta}_n) & \\ -2\mu_n(\theta_{n-1} - \tilde{\theta}_{n-1}), & i = n-1 \\ 2\mu_{i-1}(\theta_{i-1} - \theta_i - \tilde{\theta}_{i-1} + \tilde{\theta}_i) & \\ -2\mu_i(\theta_i - \theta_{i+1} - \tilde{\theta}_{i-1} + \tilde{\theta}_{i+1}), & \text{otherwise} \end{cases} \tag{10}$$

Finally, the gait generator model for the chain-type CPG-network is obtained as follows:

$$\begin{bmatrix} \dot{\varphi}_1 \\ \dot{\varphi}_2 \\ \vdots \\ \dot{\varphi}_n \end{bmatrix} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \vdots \\ \omega_n \end{bmatrix} + A \begin{bmatrix} \varphi_1 \\ \varphi_2 \\ \vdots \\ \varphi_n \end{bmatrix} + B \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_{n-1} \end{bmatrix} \tag{11}$$

where A is,

$$A = \begin{bmatrix} -\mu_1 & \mu_2 & & & 0 \\ \mu_2 & -2\mu_2 & \mu_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \mu_{n-1} & -2\mu_{n-1} & \mu_{n-1} \\ 0 & & & \mu_n & -\mu_n \end{bmatrix}_{n \times n} \tag{12}$$

and B is,

$$B = \begin{bmatrix} 1 & & & 0 \\ -1 & 1 & & \\ & -1 & \ddots & \\ & & \ddots & 1 \\ 0 & & & -1 \end{bmatrix}_{n \times (n-1)} \tag{13}$$

ω_i in (11) is the integration constant, which is also the frequency of the CPG signal. To output the same frequency signals, we set $\omega_1 = \omega_2 = \ldots = \omega_n$. The convergence rate of the system is decided by the matrix A, which increases with the value of μ_i.

Next, two PD controllers are adopted to ensure the convergence of the amplitude [35] and the amplitude bias,

$$\ddot{r}_i = a_i \left[\frac{a_i}{4}(R_i - r_i) - \dot{r}_i \right] \tag{14}$$

$$\ddot{c}_i = a_i \left[\frac{a_i}{4}(C_i - c_i) - \dot{c}_i \right] \tag{15}$$

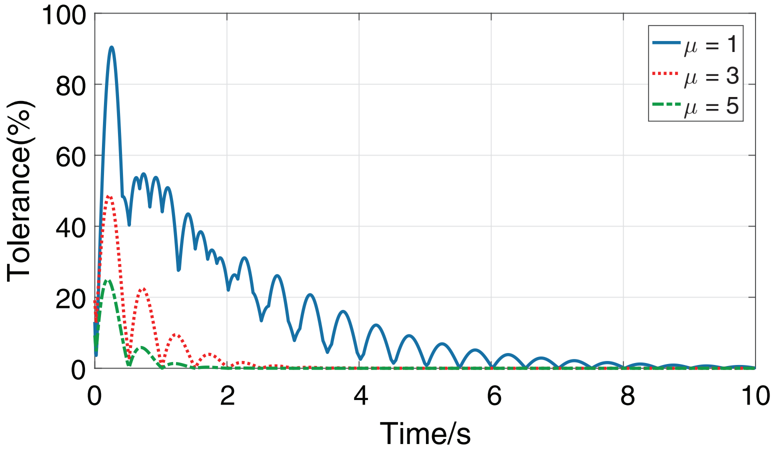

Figure 5. The synchronization speed of a three-CPG network changes with the parameter μ. Y axis is the tolerance between adjacent CPG.

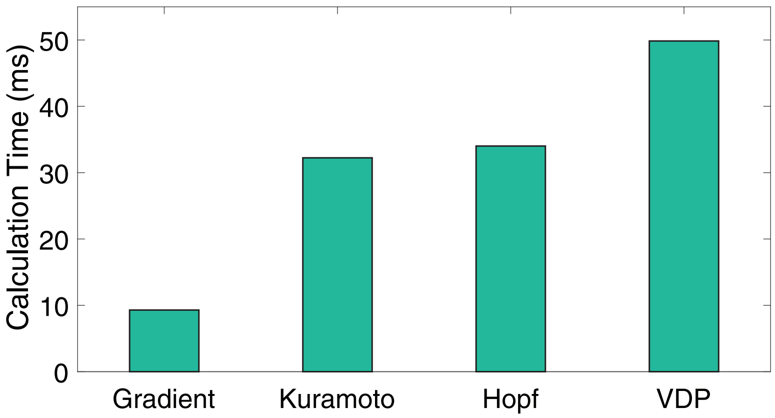

Figure 6. The execution time on Atmega 328 for 10 iteration of four kinds of CPG models. Each iteration step is 5 ms. Gradient, Kuramoto, Hopf, and VDP are short for the CPG models based on the Kuramoto oscillator, the Hopf oscillator, and the Van der Pol oscillator, respectively.

where, the parameter R_i determines the stable amplitude. The parameter C_i dominates the biased value of the stable amplitude. r_i, c_i are the amplitude and the amplitude bias of the ith oscillator, respectively. In summary, the single neuron CPG model can be written as,

$$\begin{cases} \dot{\varphi}_i &= \omega_i + A\{i,:\}\cdot\Phi + B\{i,:\}\cdot\tilde{\Theta} \\ \ddot{r}_i &= a_i\left[\dfrac{a_i}{4}(R_i - r_i) - \dot{r}_i\right] \\ \ddot{c}_i &= a_i\left[\dfrac{a_i}{4}(C_i - c_i) - \dot{c}_i\right] \\ x_i &= c_i + r_i\sin(\varphi_i); \end{cases} \tag{16}$$

where $A\{i,:\}$ is the ith row vector in (12), $B\{i,:\}$ is the ith row vector in (13). Φ is the vector of the CPG neurons' phase, and $\tilde{\Theta}$ is the vector of the phase difference among the CPG neurons. The state variable φ_i is the phase of the ith oscillator, respectively. The positive constants a_i and b_i are used to adjust the convergence speed of the amplitude. The variable x_i is the rhythmic output signal integrated by the phase ϕ_i, the amplitude r_i, and the amplitude bias c_i. To evaluate the output signals of the proposed CPG model, we plot four CPG outputs by adjusting the parameters with time, as seen in figure 4. The CPG network changes the phase difference θ from 0 to π, the frequency ω from 0.5 Hz to 1 Hz, respectively, at $t = 10$, then linearly changes the amplitude bias C from 0° to 15°, between $t = 10$ and $t = 20$. The amplitude R is changed from 20° to 30°, at $t = 30$.

To evaluate the convergence speed of the proposed CPG model, we define the convergence tolerance as (17), where n is the number of CPG neurons and the target phase difference among the CPG network is set as 0. Therefore, the signals will reach the peak value at the same pace in finite time. When the tolerance is below 1%, the convergence is treated as finished.

$$\text{tolerance} = \sum_{i=1}^{n-1}\left(\frac{1}{n-1}|(x_i - x_j)|\right), \quad j = i + 1 \tag{17}$$

The parameter μ in (12), is related to the speed of the synchronization. Based on the tolerance definition in (17), the relationship between the convergence rate and the parameter μ is shown in figure 5. Therefore, we can conclude that the convergence rate increases with μ.

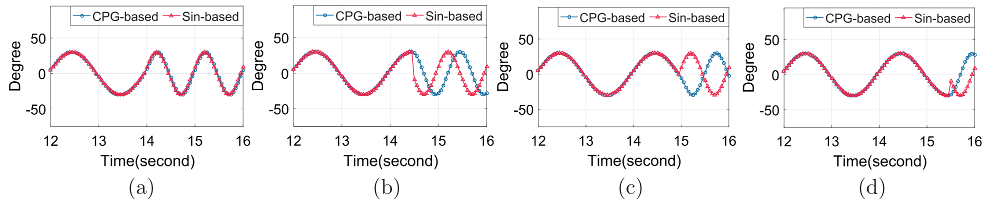

Figure 7. Changing in frequency from 1 Hz to 2 Hz, at four different times in a signle period. (a) $t = 14$ s. (b) $t = 14.5$ s. (c) $t = 15$ s. (d) $t = 15.5$ s.

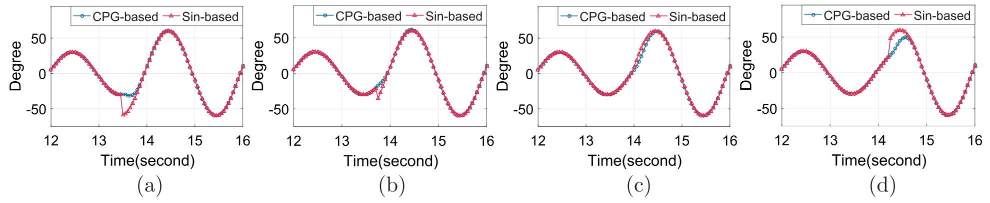

Figure 8. Changing in amplitude from 30° to 60°, at four different times in a single period. (a) $t = 13.5$ s. (b) $t = 13.75$ s. (c) $t = 14$ s. (d) $t = 14.25$ s.

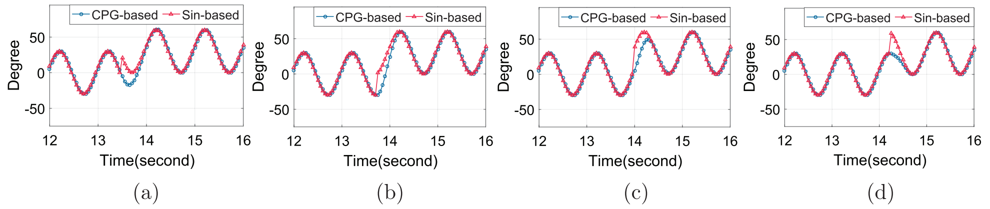

Figure 9. Changing in amplitude bias from 0° to 30°, at four different times in a single period. (a) $t = 13.5$ s. (b) $t = 13.75$ s. (c) $t = 14$ s. (d) $t = 14.25$ s.

3.2. Comparison with other CPG models

Those widely adopted CPG models are based on different nonlinear oscillators; for example, the Kuramoto oscillator model was adopted by Ijspeert and Shugen Ma [7, 35], the Hopf oscillator model was adopted by Seo and Slotine [38], and the Van Der Pol oscillator was adopted by Yu [39]. Many of these models can generate waves with desired properties, like frequency, amplitude and phase difference. However, the execution time on MCU(Micro Control Unit) is rarely investigated, a critical factor that influences the performance of the CPG model in real-life implementations. To examine the advantage of the prososed CPG model, experiments are conducted on an Arduino Nano board(Atmel Atmega 328) to inspect the online execution time of those CPG models.

The fourth-order Runge-Kutta is adopted to solve the differential equations of a three-CPG network. To ensure the solution accuracy of coupled differential equations, we set the CPG output period as 50 ms and the step-size as 5 ms. The online executing experiment results are shown in figure 6. We can observe that the online execution time for ten steps of the proposed CPG model is about 10 ms, which can easily satisfy the required signal output period. Since the total online execution involves on-bus data transmit, other CPG models may consume more that 50 ms, which means a potential failure of output period. Therefore, the conclusion can be made that the proposed CPG model is lighter in weight compared to other CPG models.

Table 1. Description of the parameters in extended gait equation.

Items	Descriptions	Value
n	Module subscript	$1 \sim 16$
N	Module numbers	16
$C_{\text{odd}}, C_{\text{even}}$	Offset in lateral and dorsal plane	0°, 0°
$A_{\text{odd}}, A_{\text{even}}$	Amplitude in lateral and dorsal plane	60°, 40°
Ω_{odd}	Spatial frequency in lateral plane	$n \cdot 3.5/N \cdot \pi$
Ω_{even}	Spatial frequency in dorsal plane	$n \cdot 7/N \cdot \pi$
$\omega_{\text{odd}}, \omega_{\text{even}}$	Time frequency in lateral and dorsal plane	1, 2
$x_{\text{odd}}, x_{\text{even}}$	Cycle numbers	1.75, 3.5
y, z	Linear coefficient	0.3, 0.7

4. Smooth transition of slithering gait

This section will discuss the effectiveness of the control method based on our proposed CPG model for smoothing the control signals during gait transition contrasted with sinusoid-based method. The slithering gait is then modelled and achieved, based on our CPG method for further evaluation in simulations and experiments.

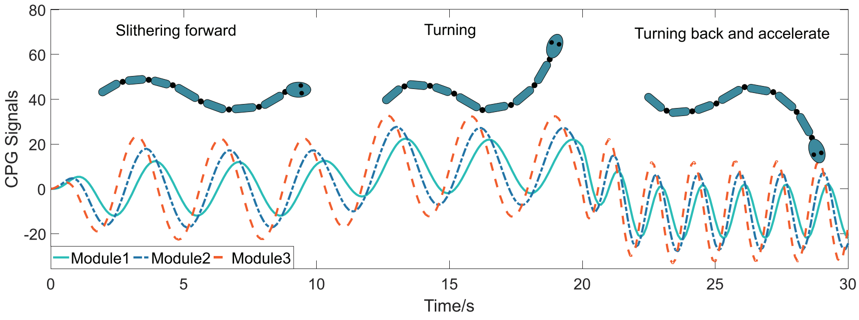

Figure 10. Illustration of the first three CPG outputs when the snake-like robot performs slithering gait. The robot slithers forward until $t = 10$ s, then turns to left until $t = 20$, finally turns back and accelerates.

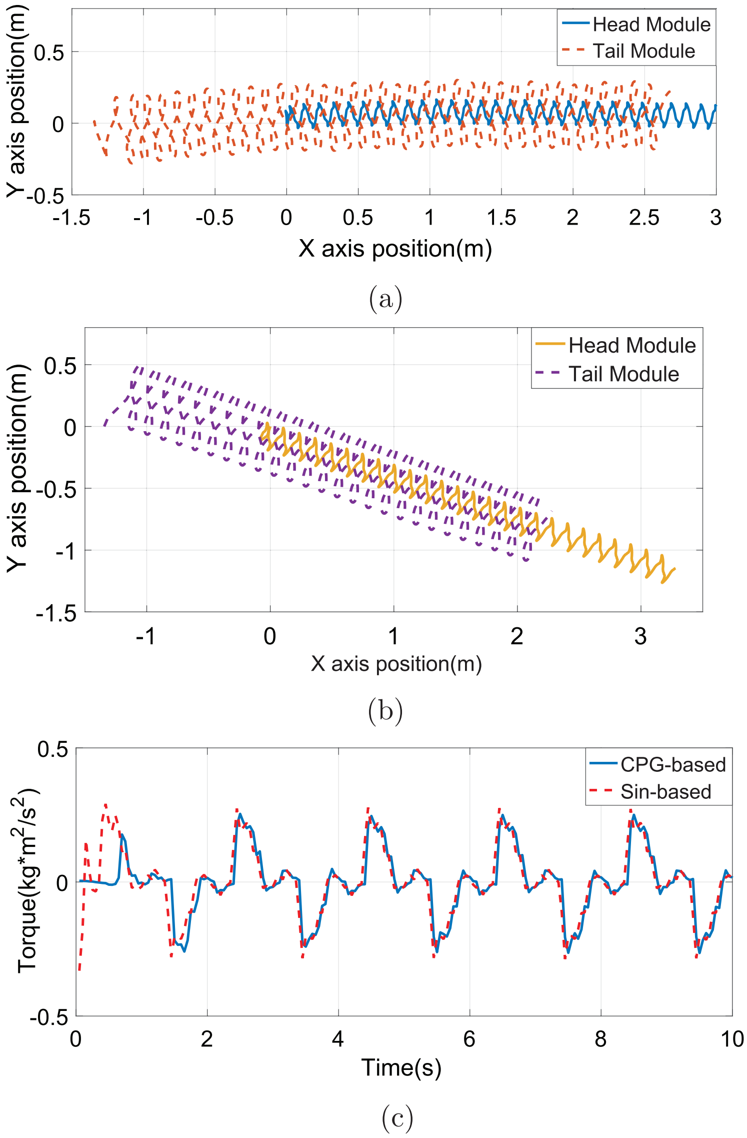

Figure 11. Trajectory and torque comparison for slithering forward. (a) Trajectory of slithering forward based on CPG method. (b) Trajectory of slithering forward based on sinusoid method. (c) Torque contrast of CPG and sinusoid based methods.

4.1. Gait transition analysis

As presented in the literature, a smooth transition is dependent on maintaining the gait properties during the transition, not maintaining gait properties will cause undesired movement. Normally, the transition process requires a change of parameters of control signals, including frequency, amplitude, amplitude bias, and phase difference. However, sinusoid-based control method inherently depends on time. During the gait transition process, each joint position will change to another sinusoid wave, which probably has different values and direction.

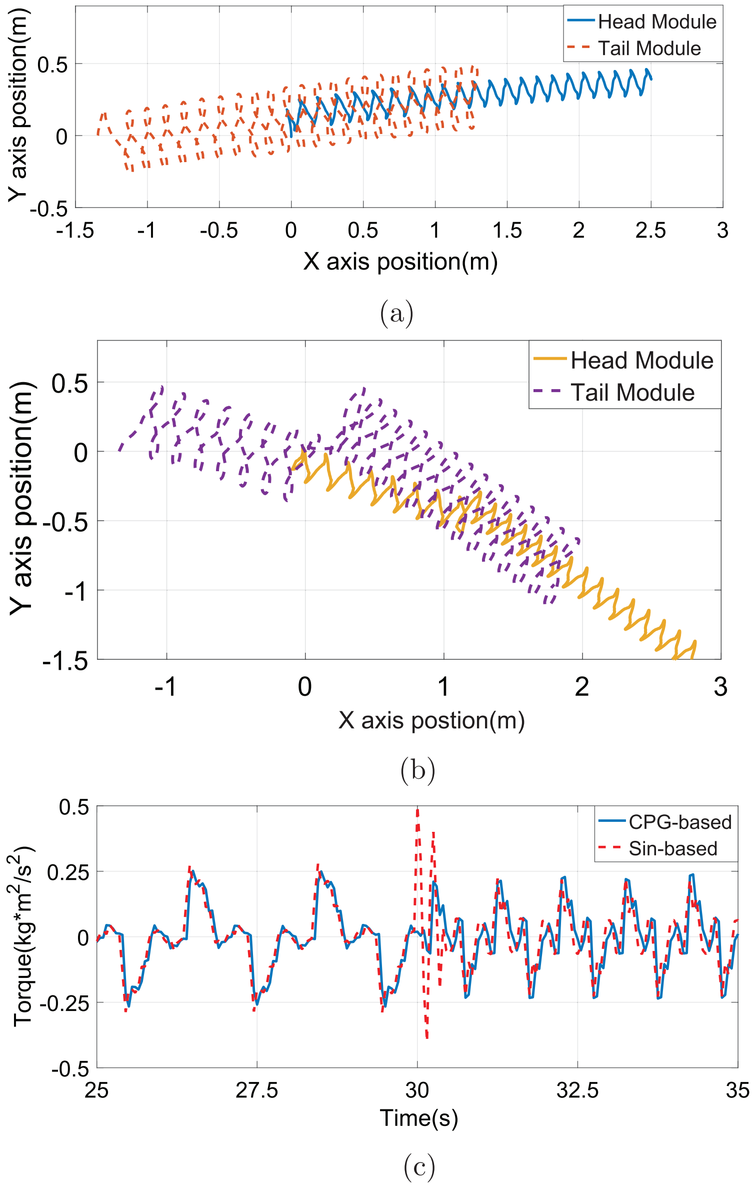

Figure 12. Trajectory and torque comparison for slithering acceleration. (a) Trajectory of slithering acceleration based on CPG method. (b) Trajectory of slithering acceleration based on sinusoid method. (c) The accelerating slithering torque contrast of CPG and sinusoid based methods.

In contrast, the CPG-based control method can generate continuous output signals due to self-adjusting ability. As a result, when the transition occurs, the CPG can smoothly switch to the new state of parameters. The numerical analysis results are shown in figures 7–9.

In figures 7(a)–(d), the frequency of the signal is changed from π to 2π, at $t = 14$ s, $t = 14.5$ s, $t = 15$ s, and $t = 15.5$ s, respectively. Since the sinusoid-based method inherently relies on time, it generates different discontinuous waves at different times. An obvious set-point jump over 30° and a change of wave traveling direction can be observed in figures 7(b) and (c). In contrast, our CPG-based method exhibits a continuous transition process, no matter when the transition occurs. In figures 8(a)–(d), the amplitude of the signal is changed from 30° to 60°, at $t = 13.5$ s, $t = 13.75$ s, $t = 14$ s, and $t = 14.25$ s, respectively. As shown in figures 9(a)–(d), the discontinuity can also be observed when changing the amplitude bias from 0° to 30°, at $t = 13.5$ s, $t = 13.75$ s, $t = 14$ s, and $t = 14.25$ s, respectively. In general, although the discontinuity results are not as obvious as figure 7, the CPG-based method still exhibits more smooth waves compared to the sinusoid-based method. According to the results, the time-varying discontinuous waves will cause undesired movement when all the errors are integrated to the joints of the robot. Therefore, it is meaningful to adopt the CPG-based method for scenarios with frequent gait parameter transitions.

4.2. Slithering gait transition

As a promising type of gait of a snake-like robot for autonomous locomotion, slithering is adopted by snakes to move forward with a S-like shape. The challenge to utilize this gait, is stability due to the relatively small base, compared to other gaits such as sidewinding or rolling. Especially in an object tracking scenario, the snake-like robot is required to change the

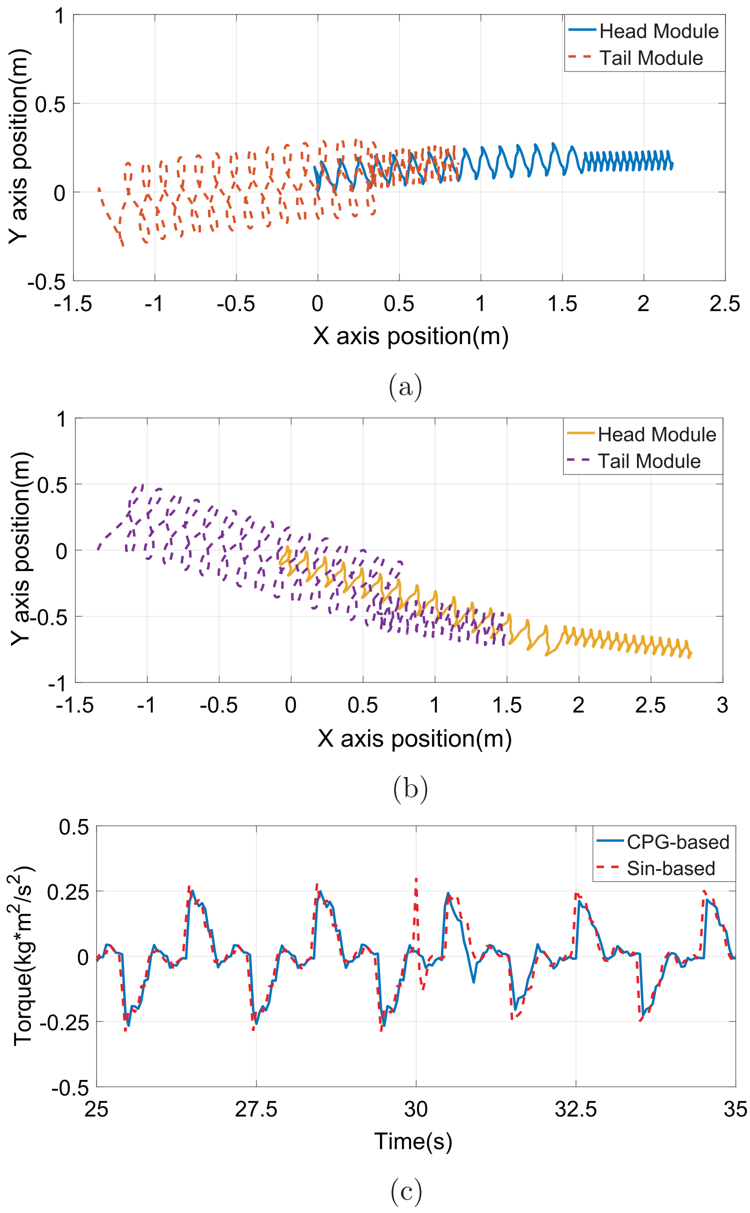

Figure 13. Slithering body shape change. (a) Trajectory for changing body shape based on CPG method. (b) Trajectory for changing body shape based on sinusoid method. (c) Torque contrast of CPG and sinusoid based methods.

locomotion speed and direction according to the target. The slithering gait is modeled as (18). The parameters and values are listed in table 1.

$$\begin{aligned}\alpha(n,t)_{\text{even}} &= C_{\text{even}} + P\cdot A_{\text{even}}\cdot\sin(\Omega_{\text{even}}\cdot n\\&\quad+\omega_{\text{even}}\cdot t)\\ \alpha(n,t)_{\text{odd}} &= C_{\text{odd}} + P\cdot A_{\text{odd}}\cdot\sin(\Omega_{\text{odd}}\cdot n\\&\quad+\omega_{\text{odd}}\cdot t+\delta)\\ \Omega &= (w+x\cdot\frac{2}{N})\cdot\pi\\ P &= (\frac{n}{N}\cdot z+y)\quad\in[0,1]\quad\forall n\in[0,N]\end{aligned}\tag{18}$$

In order to track and follow moving objects, the snake-like robot is required to change locomotion speed and direction. These are directly related to the frequency and amplitude bias in (18). The slithering gait transition process is depicted in figure 10 and the corresponding CPG signals are listed below. The CPG output waves for the first three modules are represented by the solid line, dotted line, and dash line, respectively. These CPG commands can make the snake-like robot slither forward before $t = 10$ s. Then, the snake changes direction to the left by increasing amplitude bias C_{odd} in (18). At $t = 20$ s, the robot accelerates and turns back by doubling the frequency ω_{even}, ω_{odd}, and decreasing amplitude bias C_{odd}. No obvious sudden change of output waves can be observed during the process.

5. Simulations

Two aspects will be demonstrated in simulations via a scenario of tracking a moving target comparing the CPG-based method with the sinusoid-based method. The

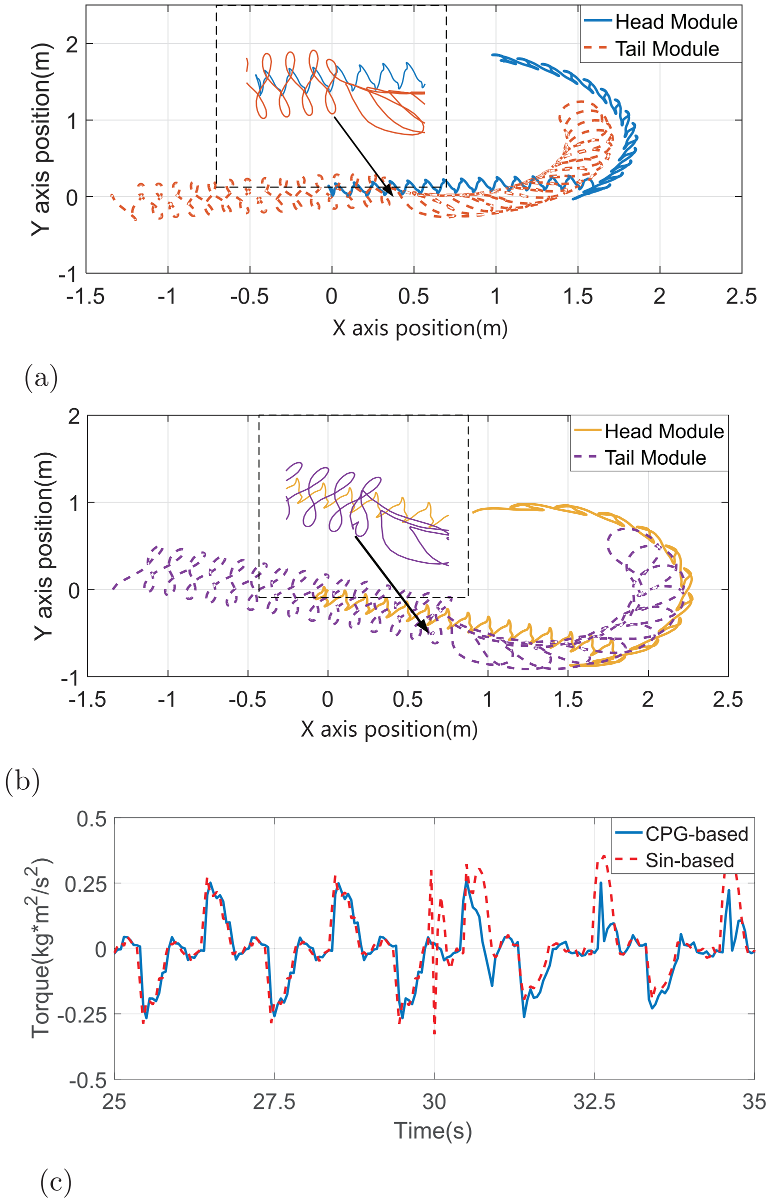

Figure 14. Slithering turning direction. (a) Trajectory for turning direction based on CPG method. (b) Trajectory for turning direction based on sinusoid method. (c) Torque contrast of CPG and sinusoid based methods.

CPG-based control method has two advantages. First, the CPG-based control method can ensure smooth gait trajectory, when the body shape and locomotion speed are changed. Second, the CPG-based control method can effectively decrease the abnormal torque during transition. All simulations are conducted in Virtual Robot Experimentation Platform (V-Rep EDU Version).

The trajectories and torque comparison results are shown in figures 11–14. The slithering process are shown in figures 11(a) and (b). The trajectories of head and tail modules of the robot are represented by a solid and dash line, respectively. At the initial position, the snake robot lies along the *x* axis and the head coordinates are (0,0). By comparing the two figures, one can see that the sinusoid-based method generates an inclination angle deviated from the body direction. However, the CPG-based method trajectory is almost aligned with the body direction indicating an accurate locomotion direction control effect. The torque curves are shown in figure 11(c). The dash line exhibits jitter phenomenon when the snake starts to move, the torque curve of the proposed CPG-based method fluctuates slightly at initiation of movement.

During the tracking process, the snake-like robot is required to accelerate or decelerate. Figure 12 shows the acceleration process. The CPG-based method maintains the direction of movement when doubling the forward speed (figure 12(a)). On the contrary, the sinusoid-based method deflects to another direction when accelerating (figure 12(b)). Meanwhile, a sudden change of the torque curve can be observed in figure 12(c).

To increase the throughput capacity for narrow terrain, a change in body shape is necessary. By adjusting the amplitude, the snake robot can contract the body, as shown in figure 13. After the transition, the trajectories of the robot shrink together and

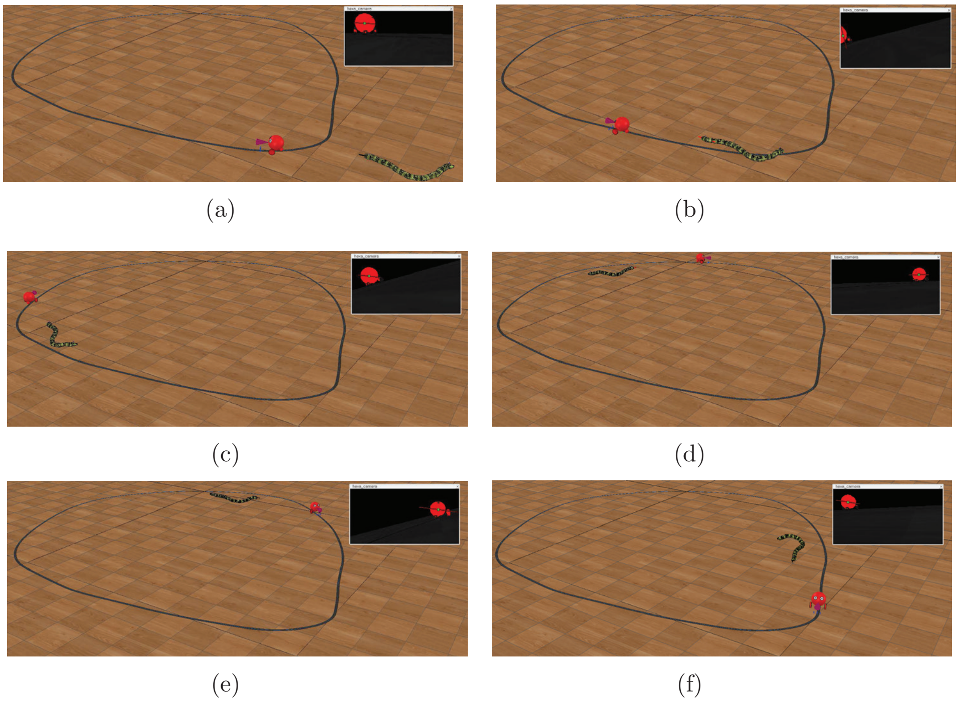

Figure 15. Montage of the simulation scenario: the snake-like robot locks on and tracks the red car moving along the black path. The vision image is presented in the top right of each figure. The red spot means the relative position of the car in the view field of the robot. (a) Start to track the red moving car, at $t = 0$. (b) Track the car along an approximate straight-line, at $t = 45$. (c) Turn right to follow the target, at $t = 120$. (d) Turn right to follow the target, at $t = 200$. (e) Turn right to follow the target, at $t = 270$. (f) Reach the finish point, at $t = 320$.

continue moving forward. In contrast to the CPG-based method (figure 13(a)), the sinusoid-based trajectories (figure 13(b)) deviate slightly from the previous direction. The torque generates a jerky jump of set points, as shown in figure 13(c).

Another requirement for tracking locomotion ability is turning direction (figure 14). The snake-like robot turns to the left when $t = 30$. Although it is not so obvious, the discontinuity of the body trajectories are still apparent, as compared to figures 14(a) and (b). The sharp edge of the trajectories in the enlarge figure is not acceptable, especially for unstable direction turning locomotion. As expected, the torque curves (figure 14(c)) demonstrate an abnormally large value. The montage of the tracking process is shown in figure 15. Here, we can find that the red car moves along the black random shape path on the ground. The snake-like robots tracks the target red car by identifying the color. The color filtered image indicating the position and orientation of the red car is shown at the top right of each figure.

6. Prototype experiments

Our snake-like robot has a modular design consisting 13 actuated modules and a head module (figure 1). The modules communicate with each other via I^{2}C bus. All the output shafts are alternately aligned with the robot's lateral and dorsal planes to generate 3D locomotion. Each module is connected to the adjacent modules and allows a full 180° rotation.

Table 2. Overview of Snake-like robot specifications.

Items	Discriptions
Dimensions	Diameter 60 mm
	Length 70 cm
Mass	Module 0.3 kg
	Full 2 kg
Actuation	Max torque 12.8 Kg · cm
	Max speed 0.07 s/60°
Power	7.4 V DC
Communication	I^{2}C Bus
Sensing	Angular sensor MLX90316KDC

6.1. Overview mechanics

Each module contains a servo and a set of gears to actuate the joint. The DC servo (DS1509MG) has a maximum torque of12.8 Kg · cm and drives a gearbox with a reduction factor of 3.71. For the electronic hardware, each module has an Arduino Nano board, a printed circuit board and an angle sensor. The Arduino Nano board runs three tasks: controlling the servo, reading the joint angle, and communicating with the other modules. Table 2 summarizes the technical specifications of our snake robot.

6.2. Experiments

In section 3, the characters of the proposed CPG model were proved via numerical simulation and online execution experiments. In section 5, simulations are performed to prove that the

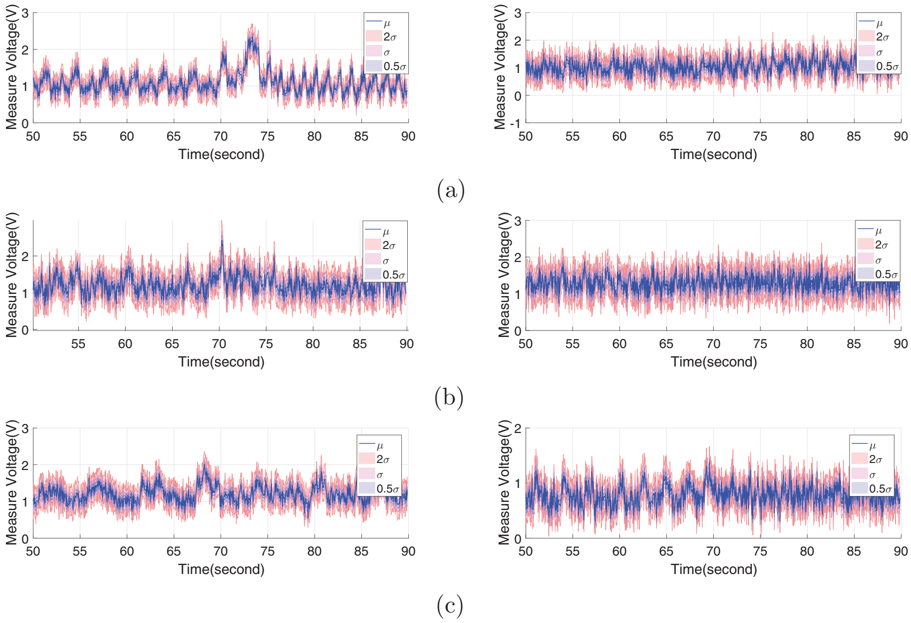

Figure 16. Voltage measurements of the high-load resistance. The left column figures depict the sinusoid-based method measurements. The right column figures depict the CPG-based method measurements. (a) Frequency changes from 0.75 Hz to 1.5 Hz of slithering gait. (b) Amplitude changes from 20° to 40° of slithering gait. (c) Amplitude bias changes from 0° to 10° of slithering gait.

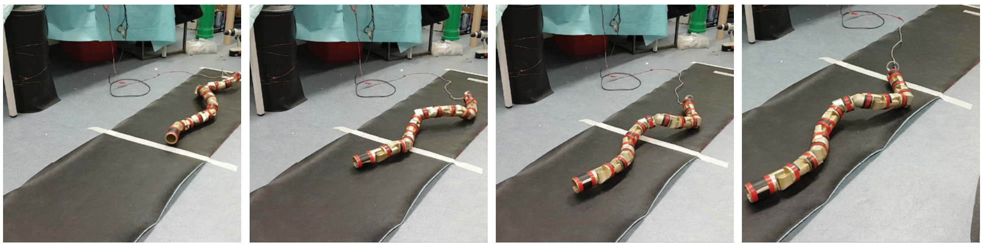

Figure 17. Montage of snake-like robot slithering forward. The forward speed is approximately 0.15 m s^{-1}.

proposed CPG-based method can achieve smooth transition when the body shape, locomotion direction, or the locomotion speed is changed. Now, we report the results of a set of experiments conducted on our snake-like robot to demonstrate that our CPG-based method can ensure smooth slithering gait transition as shown in figure 17. The CPG network is implemented on a Master Arduino Nano board, located in the tail module, which sends commands via I^2C bus. Meanwhile, the tail receives the angle position of each module. Every slave module runs a PD controller to control the servo so that the joint can reach the desired position.

The output torque of the snake-like robot is directly related to the current. Therefore, by measuring the current of the robot, we can obtain the changing trend of the output torque. A 0.1Ω/50W high-load resistance is stringed into the circuit and the voltage of the resistance is measured by oscilloscope. Thus, the changing trend of the torque can be presented by the voltage measurements.

In figure 16, the results of the sinusoid-based method and CPG-based method for slithering gait transition are in the left column and the right column, respectively. The average value is presented with solid lines and the standard deviation of each figure is represented with σ. Figure 16(a) describes the frequency transition of slithering gait, from 0.375 Hz to 0.75 Hz. The transition of the sinusoid-based method happens at $t \approx 73$ s, when an abnormal torque is generated. The transition of the CPG-based method does not show obvious abnormal torque. The amplitude transition of sidewinding gait from 20° to 40° is shown in figure 16(b), the similar abnormal torque occurs at $t \approx 70$ s in the left side column. On the contrary, by adopting the CPG-based method, the voltage measurements exhibit relative smooth processes. In figure 16(c), amplitude bias is adjusted from 20° to 40° at $t \approx 68$ s. Both figures on the left show the abnormal torque when starting the transition process. On the opposite side, the CPG-based method, the voltage measurements do not exhibit abnormal torque during the transition in the right side column.

Another viewpoint of the experimental torque results is the collision and friction between the snake robot and the ground. During the locomotion process, the robot inevitably drags itself against the ground and

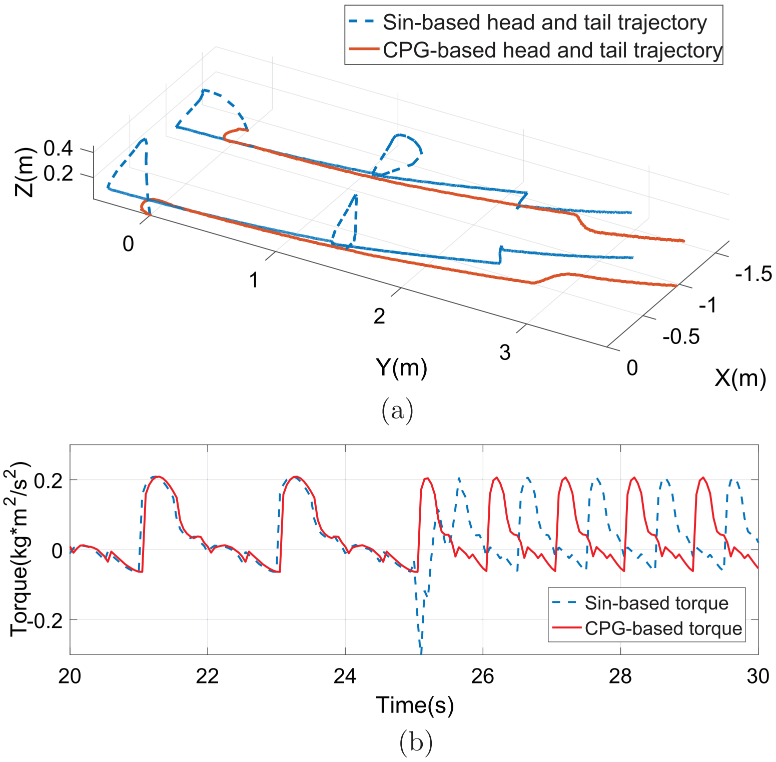

Figure A1. Snake-like trajectories and torque curves under rolling gait. (a) Sin-based and CPG-based trajectories under rolling gait. (b) Torque comparison during gait transition.

collide with the ground. As we can see in figure 16, the torque curve is consisted of rhythmic waves. Since the robot moves periodically, the torque wave also present periodical change. Except for the sudden change discussed above, the peak of the rhythmic wave is generated by the collision and the friction between the snake robot and the ground. The results in figure 16(a) also prove the finding, where the frequency of the torque wave is doubled when the frequency of the gait is changed from 0.375 Hz to 0.75 Hz.

7. Conclusions and future work

In this paper, we have proposed a lightweight CPG model with fast computing time. Then, a CPG-based locomotion control architecture for smooth slithering gait transition of snake-like robots has been presented and compared with the sinusoid-based method. Simulation and experiment results show that the snake-like robot can achieve smooth locomotion speed, moving direction, and body shape transition by tuning the CPG parameters.

For future work, we plan to integrate our snake-like robot in the Neurorobotics Platform (NRP) of the Human Brain Project [47]. We will adopt the spiking neural network to achieve neuromorphic mapping from sensors to the locomotion of the snake-like robot, with Spinnaker [48], a neuromorphic computing plantform. Since the proposed CPG-based control method can be used as the locomotion control layer and multi sensors information can be used as input layer, the future work mainly focuses on the layers transmitting the information to locomotion, aiming at autonomous locomotion of the snake-like robot.

Acknowledgment

The research leading to these results has received funding from the European Union Research and Innovation Programme Horizon 2020 (H2020/2014–2020) under grant agreement no. 720270 (Human Brain Project) and Chinese Scholarship Council.

Appendix

Similar results reported in this paper can also be observed with other kinds of gaits. As one of the most stable and common gaits, *rolling* is examined by changing the locomotion speed, body shape and moving direction, both on simulations and prototype experiments. The results are shown in figure A1. In the simulation, the snake is initially placed at the origin point, in alignment with the x axis. The snake-like robot rolls along the y axis. As shown in figure A1(a), the solid line is the trajectory based on the proposed CPG method, the dash line trajectory is based on the sinusoid method. At $t = 25$, the frequency is increased from 1 Hz to 2 Hz and undesireable locomotion trajectory can be observed at the start time. In figure A1(b), an abnormal torque in the dash line can be observed. At $t = 35$, the amplitude is increased from 20° to 40° and the body of the robot forms into an arc with larger curvature. The results show that We can also smooth the body shape transition to contrast the CPG-based method with the sinusoid-based method.

References

[1] Hirose S 1993 Biologically inspired robots: snake-like locomotors and manipulators (Oxford: Oxford University Press)

[2] Mori M and Hirose S 2002 Three-dimensional serpentine motion and lateral rolling by active cord mechanism ACM-R3 *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems* vol 1 pp 829–34

[3] Rollinson D, Ford S, Brown B and Choset H 2013 Design and modeling of a series elastic element for snake robots *ASME 2013 Dynamic Systems and Control Conf. p* V001T08A002 (American Society of Mechanical Engineers)

[4] Wu X and Ma S 2010 CPG-based control of serpentine locomotion of a snake-like robot *Mechatronics* **20** 326–34

[5] González Gómez J, Zhang H, Boemo E I and Zhang J 2006 Locomotion capabilities of a modular robot with eight pitch-yaw-connecting modules *Int. Conf. on Climbing and Walking Robots*

[6] Droge G and Egerstedt M 2012 Optimal decentralized gait transitions for snake robots *IEEE Int. Conf. on Robotics and Automation* pp 317–22

[7] Nor N M and Ma S 2013 Smooth transition for CPG-based body shape control of a snake-like robot *Bioinspir. Biomim.* **9** 016003

[8] Herrero-Carrón F, Rodríguez F B and Varona P 2011 Bio-inspired design strategies for central pattern generator control in modular robotics *Bioinspir. Biomim.* **6** 016006

[9] Wang Z, Gao Q and Zhao H 2017 CPG-inspired locomotion control for a snake robot basing on nonlinear oscillators *J. Intell. Robot. Syst.* **85** 209–227

[10] Sfakiotakis M and Tsakiris D P 2007 Neuromuscular control of reactive behaviors for undulatory robots *Neurocomputing* **70** 1907–13

[11] Astley H C, Gong C, Dai J, Travers M, Serrano M M, Vela P A, Choset H, Mendelson J R, Hu D L and Goldman D I 2015 Modulation of orthogonal body waves enables high maneuverability in sidewinding locomotion *Proc. Natl Acad. Sci.* **112** 6200–5

[12] Hatton R L, Knepper R A, Choset H, Rollinson D, Gong C and Galceran E 2013 Snakes on a plan: toward combining planning and control *2013 IEEE Int. Conf. on Robotics and Automation* pp 5174–81 (IEEE)

[13] Prautsch P and Mita T 1999 Control and analysis of the gait of snake robots *Proc. of the 1999 IEEE Int. Conf. on Control Applications* vol 1 pp 502–7 (IEEE)

[14] Maass W, Natschläger T and Markram H 2002 Real-time computing without stable states: a new framework for neural computation based on perturbations *Neural Comput.* **14** 2531–60

[15] Stefanini C *et al* 2012 A novel autonomous, bioinspired swimming robot developed by neuroscientists and bioengineers *Bioinspir. Biomim.* **7** 025001

[16] Stefanini C 2012 A novel autonomous, bioinspired swimming robot developed by neuroscientists and bioengineers *Bioinspir. Biomim.* **7** 025001

[17] Luo M, Pan Y, Skorina E H, Tao W, Chen F, Ozel S and Onal C D 2015 Slithering towards autonomy: a self-contained soft robotic snake platform with integrated curvature sensing *Bioinspir. Biomim.* **10** 055001

[18] Onal C D and Rus D 2013 Autonomous undulatory serpentine locomotion utilizing body dynamics of a fluidic soft robot *Bioinspir. Biomim.* **8** 026003

[19] Hu D L and Shelley M 2012 Slithering locomotion *Natural Locomotion in Fluids and on Surfaces* eds S Childress, A Hosoi, W W Schultz and J Wang (Berlin: Springer) pp 117–35

[20] Crespi A and Ijspeert A J 2006 AmphiBot II: an amphibious snake robot that crawls and swims using a central pattern generator *Proc. 9th Int. Conf. Climbing and Walking Robots* vol BIOROB-CONF-2006-001 pp 19–27

[21] Manzoor S and Choi Y 2016 A unified neural oscillator model for various rhythmic locomotions of snake-like robot *Neurocomputing* **173** 1112–23

[22] Ohno H and Hirose S 2001 Design of slim slime robot and its gait of locomotion *IEEE/RSJ Int. Conf. Intelligent Robots and Systems* vol 2 pp 707–15

[23] Liljebäck P, Pettersen K Y, Stavdahl O and Gravdahl J T 2012 *Snake Robots: Modelling, Mechatronics, and Control* (Berlin: Springer)

[24] Gong C, Travers M J, Astley H C, Li L, Mendelson J R, Goldman D I and Choset H 2016 Kinematic gait synthesis for snake robots *Int. J. Robot. Res.* **35** 100–13

[25] Gong C, Tesch M, Rollinson D and Choset H 2014 Snakes on an inclined plane: learning an adaptive sidewinding motion for changing slopes *2014 IEEE/RSJ Int. Conf. Intelligent Robots and Systems* pp 1114–9 (IEEE)

[26] Tesch M, Lipkin K, Brown I, Hatton R, Peck A, Rembisz J and Choset H 2009 Parameterized and scripted gaits for modular snake robots *Adv. Robot.* **23** 1131–58

[27] Melo K, Paez L and Parra C 2012 Indoor and outdoor parametrized gait execution with modular snake robots *2012 IEEE Int. Conf. Robotics and Automation*

[28] Tanaka M and Tanaka K 2016 Shape control of a snake robot with joint limit and self-collision avoidance *IEEE Trans. Control Syst. Technol.* **99** 1–8

[29] Matsuno F and Suenaga K 2003 Control of redundant 3D snake robot based on kinematic model *IEEE Int. Conf. Robotics and Automation. Proc.* vol **2** pp 2061–6 (IEEE)

[30] Ute J and Ono K 2002 Fast and efficient locomotion of a snake robot based on self-excitation principle *7th Int. Workshop Advanced Motion Control* pp 532–9 (IEEE)

[31] Shugen M 2001 Analysis of creeping locomotion of a snake-like robot *Adv. Robot.* **15** 205–24

[32] Crespi A, Lachat D, Pasquier A and Ijspeert A J 2008 Controlling swimming and crawling in a fish robot using a central pattern generator *Auton. Robots* **25** 3–13

[33] Qiao H, Peng J, Xu Z B and Zhang B 2003 A reference model approach to stability analysis of neural networks *IEEE Transactions on Systems, Man, and Cybernetics, Part B* **33** 925–936

[34] Yu J, Ding R, Yang Q, Tan M and Zhang J 2013 Amphibious pattern design of a robotic fish with wheel-propeller-fin mechanisms *J. Field Robot.* **30** 702–16

[35] Ijspeert A J, Crespi A, Ryczko D and Cabelguen J-M 2007 From swimming to walking with a salamander robot driven by a spinal cord model *Science* **315** 1416–20

[36] Lu Z L, Ma S G, Li B and Wang Y 2008 Gaits-transferable CPG controller for a snake-like robot *Sci. China* F **51** 293–305

[37] Wu X and Ma S 2013 Neurally controlled steering for collision-free behavior of a snake robot *IEEE Trans. Control Syst. Technol.* **21** 2443–9

[38] Seo K, Chung S-J and Slotine J-J E 2010 CPG-based control of a turtle-like underwater vehicle *Auton. Robots* **28** 247–69

[39] Yu H, Guo W, Deng J, Li M and Cai H 2013 A CPG-based locomotion control architecture for hexapod robot *IEEE/RSJ Int. Conf. Intelligent Robots and Systems* pp 5615–21

[40] Liu G L, Habib M K, Watanabe K and Izumi K 2008 Central pattern generators based on Matsuoka oscillators for the locomotion of biped robots *Artif. Life Robot.* **12** 264–9

[41] de Pina Filho A C, Dutra M S and Raptopoulos L S C 2005 Modeling of a bipedal robot using mutually coupled Rayleigh oscillators *Biol. Cybern.* **92** 1–7

[42] Yu J, Tan M, Chen J and Zhang J 2014 A survey on CPG-inspired control models and system implementation *IEEE Trans. Neural Netw. Learn. Syst.* **25** 441–56

[43] Qiao H, Peng J and Xu Z-B 2001 Nonlinear measures: a new approach to exponential stability analysis for Hopfield-type neural networks *IEEE Transactions on Neural Networks* **12** 360–370

[44] Kopell N and Ermentrout G B 1988 Coupled oscillators and the design of central pattern generators *Math. Biosci.* **90** 87–109

[45] Bing Z, Cheng L, Huang K, Zhou M and Knoll A 2017 Smooth gait transition of body shape and locomotion speed based on CPG control for snake-like robot *IEEE Int. Conf. Robotics and Automation (June 2017)*

[46] Pearlmutter B A 1995 Gradient calculations for dynamic recurrent neural networks: a survey *IEEE Trans. Neural Netw.* **6** 1212–28

[47] Hinkel G *et al* 2017 A framework for coupled simulations of robots and spiking neuronal networks *J. Intell. Robot. Syst.* **85** 71–91

[48] Araújo R, Waniek N and Conradt J 2014 Development of a dynamically extendable spinnaker chip computing module *Int. Conf. Artificial Neural Networks* (Berlin: Springer) pp 821–8