Noname manuscript No.
(will be inserted by the editor)

Locomotion of Snake-like Robots Using Adaptive Neural Oscillators

Received: date / Accepted: date

Abstract This paper proposes a CPG-based control architecture using a frequency-adaptive oscillator for undulatory locomotion of snake-like robots. The control architecture consists of a network of neural oscillators that generates desired oscillatory output signals with specific phase lags. A key feature of the proposed architecture is a self-adaptation process that modulates the parameters of the CPG to adapt the motion of the robot to varying coefficients of body-ground friction. This process is based on the frequency-adaptation rule of the oscillator that is designed to learn the periodicity of sensory feedback signals. It has an important meaning of establishing a closed-loop CPG much more robust against environmental and/or system parameter changes. We verify the validity of the proposed locomotion control system employing a simulated snake-like robot moving over terrains with different friction coefficients with a constant velocity.

Keywords Undulatory Locomotion · Central Pattern Generator · Adaptive Oscillator · Frequency Adaptation

1 Introduction

Snakes have highly efficient methods of locomotion, easily adapting themselves to widely varying ground conditions. Mimicking their locomotor repertoire, snake-like robots have been developed for use in, *e.g.*, search and rescue tasks. Hirose [1] pioneered the research and development of snake-like robots in 1972. Since then, a variety of different mechanisms have been designed [2] [3] [4], and their trajectory planning and motion control have been discussed extensively [5] [6] [7].

Address(es) of author(s) should be given

This work is inspired by recent attempts imitating the vertebrate nervous system to generate rhythmic movements of robots. In the spinal cord of vertebrates, there exists the locomotor central pattern generator (CPG) that consists of neural oscillators producing rhythmic patterned outputs. One example is the undulatory locomotion of snakes that emerges as a stable limit-cycle system from a global entrainment between the nervous system and the body interacting with its environment. There are certain advantages relating to the use of CPGs in robotics for achieving natural rhythms of movements. Some of them can be summarized as follows: (1) They efficiently support a distributed control system, (2) they require only simple command signals to produce complex coordinated multi-dimensional output signals, and (3) it is easy to incorporate sensory feedback and absorb parameter variation uncertainties and external perturbations.

Previous studies suggest that CPG-based locomotion control appears promising. Conradt and Varshavskaya developed WormBot controlled by local CPGs that are scalable to much larger numbers of degrees of freedom [8]. Ekeberg created a neuro-mechanical model of swimming locomotion of lamprey using physiologically acquired knowledge about its neural structure [9]. The model has biological reality, but it is not easy to be applied to real robots due to its structural complexity. Ijspeert developed a neural mechanism and realized swimming and crawling motion in lamprey and salamander using Ekeberg's CPG model and a genetic algorithm [10] [11]. Inoue and Ma's group employed mutual inhibitory CPG neurons to construct a locomotion controller for snake-like robots [12] [13]. They recently attempted to configure a closed-loop control system by incorporating sensory feedback from the environment, where the CPG parameters were determined with a ge-

netic algorithm [14]. But their approach did not provide an evolutionary process of adaptation to changing environments, and the CPG parameters were adjusted with a particular set of frictions. Overall, most previous CPG-based control methods were open-loop that did not feed any sensory information to CPGs, and the potentials for environmental adaptability was not fully exploited.

Regarding the implementation of the CPG-based control, several issues need to be further considered. For robots with large degrees of freedom, appropriate adjustment of inter-CPG relation remains a challenge. It is also difficult to tune the adequate parameters to ensure that CPGs generate stable rhythmic patterns when sensory inputs of varying frequencies exist. What is more important is online modulation of the CPG parameters, which allows the robot to adapt itself to unknown environment without stopping its motion or requiring any human intervention. In this paper, we propose a new CPG-based control architecture to realize adaptive undulatory locomotion of snake-like robots under environmental uncertainties. The proposed architecture consists of a network of existing neural oscillators and a new frequency-adaptive neural oscillator. Each oscillator generates desired oscillatory signals with a specific phase shift, while the frequency-adaptive oscillator modulates the frequency and phase of the signals responding to sensory inputs. We verify the validity of the proposed control architecture, focusing on how the robot keeps its locomotion velocity over terrains with different ground frictions.

The rest of the paper is organized as follows. Section 2 describes our computer model to simulate the locomotion of the snake-like robot. Section 3 describes the details of the control architecture and its mathematical properties. Section 4 presents dynamic simulations performed to verify how the proposed architecture enhances locomotor adaptive function where varying friction conditions are encountered. Finally, a conclusion is drawn in Section 5.

2 Simulation model of snake locomotion

Snake locomotion can be widely classified into four distinct modes such as lateral undulation, rectilinear, sidewinding, and concertina [1]. Depending on the environmental conditions, snakes perform mixed and more complex gait, but the lateral undulation is the most commonly seen mode characterized by an S-shaped wave propagating from head to tail [15]. To create this lateral undulation, a low friction force is required in the direction of forward movement, and a high friction force is required in lateral directions, allowing snakes to avoid lateral slips and propel itself forward. Such difference in friction force is obtained by the particular structure of scales in snakes. Thus, a ratio between the friction forces in the forward and lateral directions affects the movement velocity of snakes. Snakes modulate their winding angle or change the locomotion mode to cope with varying friction or uneven ground conditions. Therefore, given environment conditions, many research efforts have been devoted to optimizing the configuration of the body segments such as the winding angle and the axis length [1] [12]. The winding angle is an angle between the winding body and the movement direction. The axis length is one quarter of a wave cycle, which is related to the wavelength of the motion. In this work, we concentrate our efforts on the lateral undulatory locomotion by controlling the oscillation frequency of the traveling wave. It is well understood that the oscillation frequency changes the locomotion velocity in a linear proportional manner [15].

Table 1 Parameters of the snake-like robot

Parameters	Values
$Segment\ number$	15
$Segment\ length[m]$	0.1052
$Segment\ mass[kg]$	0.5
$Segment\ inertia[kg \cdot m^2]$	0.016
$Wheel\ joint\ friction\ coefficient$	0.05

Specifically, we have enhanced the performance of lateral undulatory locomotion of snake-like robots by the proposed CPG-based control architecture. Considering the robot body dynamics and interaction with its environment as shown in Fig. 1, the simulated model was created with RecurDyn [16] and SolidWorks. The robot moving on a horizontal plane is composed of serially connected 15 segments. Between neighboring segments, a one-dimensional revolute joint rotating about the yaw axis is located. A tactile sensor and an acceleration sensor are located in the head segment. The friction force between the robot body and the terrain is assumed to be large in the lateral direction and small in the forward direction, which was realized by using passive wheels. Table 1 shows the detailed parameters of our simulation model.

3 Adaptive locomotion control architecture

In this section, we propose a novel CPG-based locomotion control architecture for snake-like robots. The architecture allows robots to self-modulate locomotion patterns by changing the CPG parameters adapting to the changes in the ground friction coefficient.

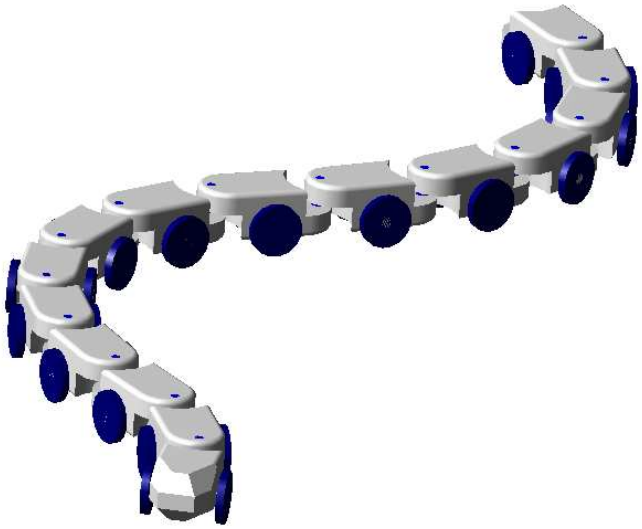


Fig. 1 Simulation model of snake-like robot

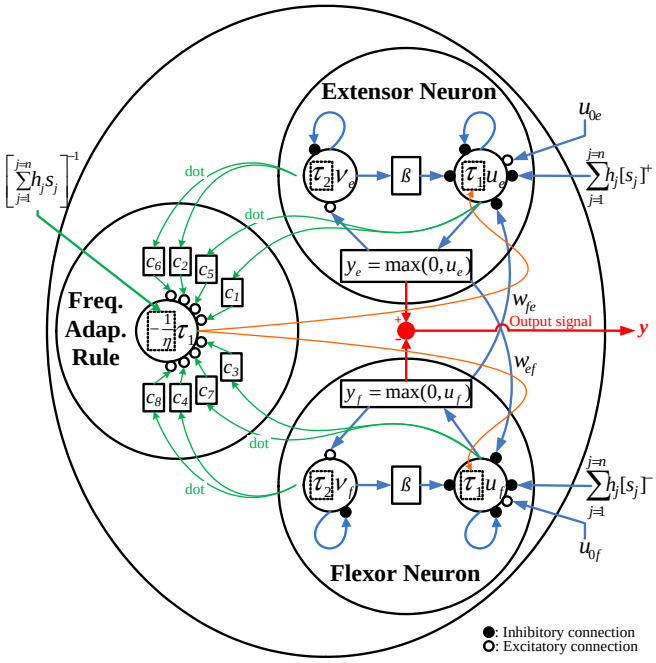


Fig. 2 Adaptive Matsuoka's oscillator

3.1 CPG model

We use the neural oscillator model proposed by Matsuoka [17] [18] [19] [20]. The neural model is a half-center oscillator which consists of two (extensor and flexor) neurons, having mutual inhibitory interactions. The model can be described by the following set of differential equations.

$$\tau_1 \dot{u_e} = u_{0e} - u_e - \beta \nu_e - w_{ef}[u_f]^+ - \sum_{j=1}^{j=n} h_j [s_j]^+ \quad (1)$$

$$\tau_2 \dot{\nu_e} = -\nu_e + [u_e]^+ \quad (2)$$

$$\tau_1 \dot{u_f} = u_{0f} - u_f - \beta \nu_f - w_{fe}[u_e]^+ - \sum_{j=1}^{j=n} h_j [s_j]^- \quad (3)$$

$$\tau_2 \dot{\nu_f} = -\nu_f + [u_f]^+ \quad (4)$$

$$y_i = [u_i]^+ = max(u_i, 0), [u_i]^- = min(u_i, 0) \quad (5)$$

$$y_{out} = [u_e]^+ - [u_f]^+ = y_e - y_f \quad (6)$$

where u_e, ν_e, u_f, and ν_f are the internal states of the oscillator. y_{out} is the output of the oscillator. β is the adaptation coefficient. w and h are the weights of the inter-oscillator and the input signal, respectively. s is the input from other oscillators or sensory systems. τ_1 and τ_2 are the positive time constants that determine the envelope and frequency of the output. $u_{0e,0f}$ is the positive tonic input that modulates the amplitude of the output. Specifically, to maintain a stable oscillation, we set the ratio between the time constants to a value falling within the range of 0.1-0.5 [20]. In this work, we assume that τ_1 is the only parameter that controls the intrinsic frequency of the oscillator. τ_2 is automatically determined by the predetermined ratio.

3.2 Frequency-adaptive oscillator

If the input frequency is far away from the oscillator frequency, no synchronization may occur between them. Therefore in such cases, the parameters of the oscillator need to be adjusted appropriately to change the frequency and phase of the oscillator signal. It allows the oscillator to entrain input signals over a wide range of frequencies. Few methodologies have yet emerged for adjusting oscillator parameters online. Some existing approaches required *a priori* knowledge of the dynamics of oscillators. Moreover, they were limited to simple classes of oscillators [22] [23] [24]. Recently, Righetti [21] proposed a frequency learning rule based on the phase dynamics of the limit cycle system, and extended the rule to different oscillators. But the frequency change was too slow for most real-time control problems. Héliot [25] presented an observer-based method to estimate oscillator state variables, and computed the phase of the oscillator with a given periodic sensory input. But a phenomenological model of the oscillator needs to be built and its isochrones have to be defined in advance. In this work, we propose an evolutionary approach that explores interrelationships among potential factors associated with the process of frequency adaptation for all types of oscillators. It does not require any *a priori* knowledge about the oscillator dynamics. The proposed

approach is based on the phase dynamics of the limit-cycle system [26] and an artificial evolution method [27].

Neural oscillators generate sustained, periodic patterned outputs. Given various initial states, their trajectory settles into a limit cycle in the phase plane. If the input signal is periodic, the phase of oscillator is delayed or advanced according to the difference between the oscillator and input frequencies. If the input frequency is close to the oscillator frequency and the input amplitude is large, the difference in frequency is tightly locked. We now need to devise a rule that control the adaptation of the oscillator frequency to deal with a wide range of input frequencies. The rule is assumed to be a linear combination of the state variables as follows:

$$\dot{\tau}_1(t) = sgn \times \eta S_{external} \times (w_1 u_e + w_2 \nu_e + w_3 u_f + w_4 \nu_f + w_5 \dot{u}_e + w_6 \dot{\nu}_e + w_7 \dot{u}_f + w_8 \dot{\nu}_f) \quad (7)$$

$$\tau_2(t) = \tau_1(t)/R \quad (8)$$

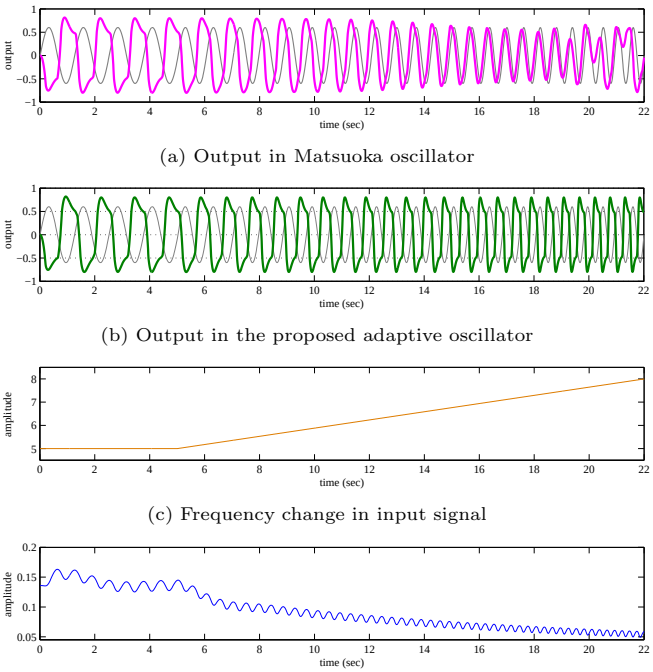
where $\eta < 1$ is the constant that determines the learning rate. w_1 through w_8 are the weighting factors. The amplitude of the periodic input signal $S_{external}$ is multiplied to ensure that the frequency adaptation process can be activated only when the input signal exists. The sgn is determined by the rotation direction of the oscillator trajectory in the phase plane. R is the time constant ratio.

The form of frequency-adaptation rule given above is based on the state feedback control system. We employ a genetic algorithm (GA) to discover the interrelationships among the state variables. Specifically, the GA determines the weighting factors of the proposed rule. In order to evaluate the fitness of the weighting factors, we design an objective function that quantifies the convergence time in frequency adaptation and the degree of correlation in frequency/phase between the input and oscillator output signals given by

$$V_F = MF - (\alpha T_{elapsed} + (1-\alpha)(F_d + P_d)) \quad (9)$$

where MF is the maximum fitness that is set to 20. α is the factor that determines the relative weight between the convergence time in frequency adaptation and the correlation degree in frequency/phase. $T_{elapsed}$ means the elapsed time until the frequency of the oscillator converges into the input frequency. F_d is the difference in frequency between the oscillator output and input signals. P_d is the difference in phase when the oscillator output is synchronized with the input.

A real-coded GA was implemented in this work as follows: first, the initial population is determined. Then, two individuals are randomly selected to produce offspring fivefold. We used the UNDX crossover and no mutation. After the production, the offspring is evaluated by the objective function given in Eq. 9. Two best individuals in the offspring replace the selected individuals in the population. This updating model is the Minimal Gap Generation (MGG) model. This process iterates until the termination condition is satisfied. Each loop of the algorithm is referred to as a generation. Finally, we can obtain an optimal adaptation rule of the oscillator.



(a) Output in Matsuoka oscillator

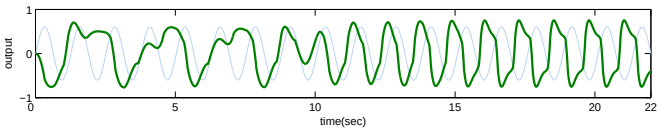
(b) Output in the proposed adaptive oscillator

(c) Frequency change in input signal

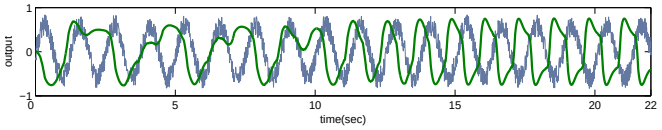
(d) Variation in τ_1 by the frequency adaptation rule

Fig. 3 Frequency adaptation of the proposed adaptive oscillator

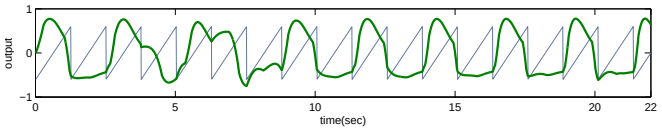
We performed simulations to confirm whether the frequency adaptive oscillator can adapt its phase to an input signal with varying frequency. We set the parameters of the oscillator as follows: $\tau_{1_{initial}}$=0.3, R=0.5, β=2.5, h=0.6, w_{ef}=w_{fe}=2.5, u_{0e}=u_{0f}=1.0, and η=0.01. The input signal is represented by a sinusoidal function $\sin 5t$. We encoded only four weighting factors (w_1, w_2, w_5, and w_6) of the extensor neuron in genotype because the extensor and flexor neurons have mutually inhibitory interactions. The population size is 50 and the total generation is 200. Fig. 3 shows that when the frequency of the input signal increases, Matsuoka's oscillator fails to entrain the signal, but the frequency-adaptive oscillator entrains the signal keeping a specific phase difference. Thus, an adaptive CPG can be realized using the proposed oscillator that autonomously adapt itself to changes in feedback signals. Moreover, through the adaptation process, we can obtain the adequate parameters that produce a desired oscillator output. Fig. 4 shows how the proposed oscillator entrains a wide variety of different types of input signals of varying frequency.



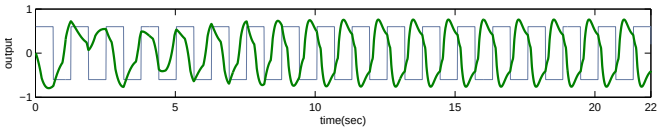
(a) Sinusoidal type



(b) Sinusoidal type with noise



(c) Sawtooth type



(d) Square type

Fig. 4 Frequency adaptation of adaptive oscillator to various types of input signals: ($\tau_{1_{initial}}$=0.35, η=0.05)

3.3 Stability of the frequency-adaptive oscillator

In this section, we prove the stability of the frequency-adaptive oscillator employing Matsuoka's oscillator. Since Matsuoka's oscillator is a half-center oscillator, which consists of two mutually inhibiting neurons (See Section 3.1, [17]), we can consider only one neuron.

$$\dot{u_i} = \frac{1}{\tau_1}(u_{0i} - u_i - \beta\nu_i - w_{ij}[u_j]^+ - h_i[s_i]^+) \tag{10}$$

$$\dot{\nu_i} = \frac{R}{\tau_1}(-\nu_i + [u_i]^+) \tag{11}$$

$$\dot{\tau_1} = -\eta s_i(w_1 u_i + w_2\nu_i + w_3\dot{u}_i + w_4\dot{\nu}_i) \tag{12}$$

$$c(x) = [x]^+ = max(0, x) \tag{13}$$

where R falls within the range of 0.1 to 0.5, assumed to remain constant [20].

Now the first step is to check whether the solutions of Eqs. 10 and 11 will be bounded regardless of any variation of $\tau_1(t)$ when periodic input signals are fed into the oscillator. Then we verify the stability of stationary states of Eqs. 10, 11, and 12 using Lyapunov's stability theory.

Theorem 1 (boundedness) *A solution of the frequency-adaptive Matsuoka oscillator (FAMO) with a periodic input $s_i(t)$ exists for any initial state (regardless of any variation of $\tau_1(t)$ that is assumed to be nonnegative) and is bounded for $t \geq 0$.*

Proof. *We extend the proof in [17] to include the effect of the frequency-adaptation rule given in Eq. 12. First, we assume that $\tau_1(t)$ is not negative for $t \geq 0$, and $\beta \geq 0$, $h_i > 0$, $w_{ij} \geq 0$, $u_{0i} > 0$, $0 < \eta < 1$.*

To verify the boundedness of the FAMO, we determine the maximum and minimum values for initial oscillator parameters and initial states. Integrating Eq. 11 with respect to ν_i, we get

$$\nu_i(t) = \nu_i(0)e^{-h_2} + Re^{-h_2}\int_0^t e^{h_2}\frac{1}{\tau_1(T)}c(u_i(T))dT,$$

$$h_2 = R\int_0^t \frac{1}{\tau_1(t)}dt \tag{14}$$

Since $c(u_i(T))$ is not negative and $\tau_1(t) > 0$

$$\nu_i(t) \geq -|\nu_i(0)| \qquad (t \geq 0) \tag{15}$$

Also, integrating Eq. 10 with respect to u_i, we get

$$\begin{aligned} u_i(t) = u_i(0)e^{-h_1} + u_{0i}e^{-h_1}\int_0^t e^{h_1}\frac{1}{\tau_1(T)}dT \\ -\beta e^{h_1}\int_0^t e^{h_1}\frac{1}{\tau_1(T)}\nu_i(T)dT \\ -w_{ij}e^{-h_1}\int_0^t e^{h_1}\frac{1}{\tau_1(T)}c(u_j(T))dT \\ -h_i e^{-h_1}\int_0^t e^{h_1}\frac{1}{\tau_1(T)}c(s_i(T))dT, \\ h_1 = \int_0^t \frac{1}{\tau_1(t)}dt \end{aligned} \tag{16}$$

Since $c(u_j(T))$, $c(s_i(T))$ is not negative and $\tau_1(t) > 0$, applying Eq. 15 to Eq. 16, we get

$$u_i(t) \leq |u_i(0)| + \frac{1}{\tau_1(0)}u_{0i} + \frac{1}{\tau_1(0)}\beta|\nu_i(0)| \tag{17}$$

Applying Eq. 17 to Eq. 14 gives similarly

$$\begin{aligned} \nu_i(t) \leq |\nu_i(0)| + \frac{R}{\tau_1(0)}(|u_i(0)| + \frac{1}{\tau_1(0)}u_{0i} \\ + \frac{1}{\tau_1(0)}\beta|\nu_i(0)|) \end{aligned} \tag{18}$$

Applying Eqs. 17 and 18 again to Eq. 16, we obtain

$$\begin{aligned} u_i(t) \geq -|u_i(0)| - u_{0i}\frac{1}{\tau_1(0)} - \beta\frac{1}{\tau_1(0)}(|\nu_i(0)| \\ + \frac{R}{\tau_1(0)}|u_i(0)| + \frac{1}{\tau_1(0)}u_{0i} + \frac{1}{\tau_1(0)}\beta|\nu_i(0)|) \\ -w_{ij}(\frac{1}{\tau_1(0)}|u_j(0)| + \frac{1}{\tau_1(0)}u_{0j} + \frac{1}{\tau_1(0)}|\nu_j(0)|) \\ -h_i c\frac{1}{\tau_1(0)}(s_i(0) \end{aligned} \tag{19}$$

Since $s_i(t)$ is a bounded signal, $s_{min} \leq s_i(t) \leq s_{max}$, from Eqs. 15, 17, 18, and 19, we can conclude that the neuron state is bounded for $t \geq 0$ regardless of any variation of $\tau_1(t)$.

Theorem 2 (Stability) *There is at least one stable limit cycle in FAMO.*

Proof. *We assume that the input s_i is non-negative. We also assume that $\tau_1(t) > 0$, $R > 0$, $\beta \geq 0$, $h_i > 0$, $w_{ij} \geq 0$, $u_{0i} > 0$, and $0 < \eta < 1$. (For details, refer to [17], [20])*

In case that $u_i > 0$, let us define a Lyapunov function candidate as

$$V(u_i, \nu_i, \tau_1) = \frac{1}{2}u_i^2 + \frac{1}{2R}\beta\nu_i^2 + \frac{1}{2\eta}\tau_1^2 \tag{20}$$

which represents a measure of distance from the limit cycle. We take the derivative of the above function given by

$$\dot{V} = u_i\dot{u}_i + \frac{1}{R}\beta\nu_i\dot{\nu}_i + \frac{1}{\eta}\tau_1\dot{\tau}_1 \tag{21}$$

From Eq. 10,

$$u_i\dot{u}_i = u_i(-\frac{1}{\tau_1}u_i - \frac{1}{\tau_1}\beta\nu_i + a) = -\frac{1}{\tau_1}u_i^2 - \frac{1}{\tau_1}\beta u_i\nu_i + au_i \tag{22}$$

where $a = 1/\tau_1(u_{0i} - w_{ij}[y_j]^+ - h_i[s_i]^+)$.

From Eq. 11,

$$\frac{1}{R}\beta\nu_i\dot{\nu}_i = \frac{1}{R}\beta\nu_i(\frac{R}{\tau_1}\nu_i + \frac{R}{\tau_1}c(u_i)) = -\frac{1}{\tau_1}\beta\nu_i^2 + \frac{1}{\tau_1}\beta\nu_i c(u_i) \tag{23}$$

From Eq. 12,

$$\begin{aligned}\frac{1}{\eta}\tau_1\dot{\tau}_1 = \frac{1}{\eta}\tau_1(-\eta s_i w_1 u_i - \eta s_i w_2\nu_i - \eta s_i w_3\dot{u}_i \\ -\eta s_i w_4\dot{\nu}_i) = -s_i((\tau_1 w_1 - w_2 + Rw_4)u_i + \\ (\tau_1 w_2 - \beta w_3 - Rw_4)\nu_i + w_3 a)\end{aligned} \tag{24}$$

Now $\dot{V}$ can be written in the following form.

$$\begin{aligned}\dot{V} = -\frac{1}{\tau_1}u_i^2 + au_i - \frac{1}{\tau_1}\beta\nu_i^2 - s_i((\tau_1 w_1 - w_2 \\ +Rw_4)u_i + (\tau_1 w_2 - \beta w_3 - Rw_4)\nu_i + w_3 a)\end{aligned} \tag{25}$$

Thus $\dot{V}$ is negative semi-definite, if the following conditions are satisfied:

$$\begin{aligned}a < 0, (\tau_1 w_1 - w_2 + Rw_4) > 0, w_3 < 0, \\ (\tau_1 w_2 - \beta w_3 - Rw_4) > 0\end{aligned} \tag{26}$$

From Theorem 2, we can conclude that there must exist one stable limit cycle, if the parameters of the oscillator and the weight factors of Eq. 12 are suitably determined. Note that the varying parameter τ_1 is controlled by Eq. 12 whose weighting factors are determined by employing our genetic algorithm. In practice, Eq. 26 is implicitly satisfied, since the weighting factor candidates that do not generate a stable limit cycle may not be selected for reproduction. If the weighting factor candidates do not satisfy Eq. 26, they do not achieve a good fitness value for the given objective function that includes the degree of correlation such as F_d and P_d. This stability condition could be also explicitly incorporated in designing an objective function.

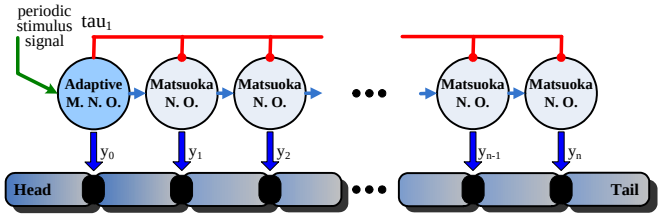


Fig. 5 Adaptive CPG network used in simulations

3.4 Adaptive CPG network for snake-like robot

CPG-based locomotion controllers may exhibit a certain degree of robustness to external perturbations, which relies on their limit cycle behavior. Snakes can adapt continuously their locomotion to a wide variety of environmental conditions. Here we describe how this self-modulating, adaptive function of snakes can be implemented with our adaptive CPG-based approach to achieve higher degrees of adaptation.

We construct a novel adaptive CPG network that produces undulatory locomotion patterns and modulates the oscillation frequency of the traveling wave of the snake-like robot according to the ground friction. One adaptive oscillator and thirteen Matsuoka's oscillators are coupled to each of the robot joints from the head to the tail with constant coupling weights as shown in Fig. 5. Each oscillator produces rhythmic motor patterns for corresponding robot joints. The oscillation of the traveling wave is propagated from the head to the tail with a phase lag. The adaptive oscillator, connected to the head joint, is synchronized with a periodic stimulus signal which may vary in frequency in time. Also propagating the time constant to the following thirteen Matsuoka's oscillators, the entire CPG network can be kept synchronized continuously to the periodic stimulus signal. Therefore, the locomotion of the snake-like robot can be adapted to different environments through sensory feedback.

3.5 Overall control architecture

Fig. 6 shows the locomotion control architecture that consists of the signal modulator and the adaptive CPG.

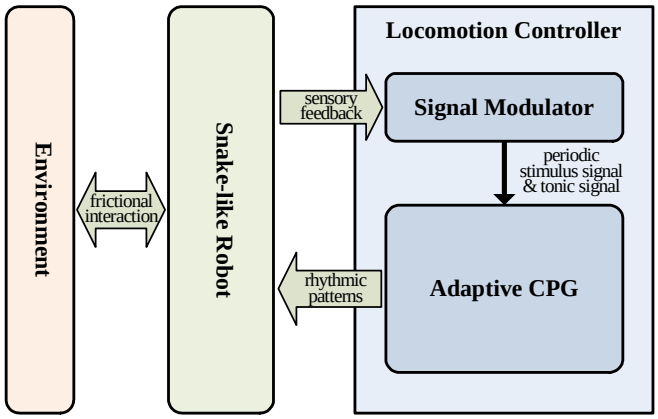


Fig. 6 Adaptive locomotion control architecture

The adaptive CPG generates rhythmic patterns for the undulatory locomotion, which can be synchronized to sensory feedback signals. Sensory signals are incorporated into the adaptive CPG via the signal modulator. The signal modulator determines the tonic signal $u_{0e,0f}$ to control the traveling direction of the robot or the traveling wave's amplitude, and generates periodic stimulus signals to increase/decrease the traveling wave's frequency. Considering a practical way of implementing sensory feedback, we assume that the head joint is equipped with an accelerometer to measure the traveling velocity. We employ a Hopf oscillator to generate periodic signals using the measured feedback signals. More specifically, to keep the robot moving at a constant velocity irrespective of changes in friction, periodic stimulus signals can be obtained by the following equations.

$$\dot{\omega} = k_1 e + k_2 \dot{e} \tag{27}$$

$$\dot{x_1} = (\mu - r^2)x_1 - \omega x_2 \tag{28}$$

$$\dot{x_2} = (\mu - r^2)x_2 - \omega x_1, \tag{29}$$

where k_1 and k_2 are gains, e and $\dot{e}$ mean the error and its derivative between the desired and current traveling velocity. x_1 and x_2 are the states of the oscillator, and $r = \sqrt{x_1^2 + x_2^2}$. $\omega > 0$ is the parameter that controls the oscillator frequency, μ is a damping coefficient, and x_1 is the periodic stimulus signal, which is equivalent to $S_{external}$ in Eq. 7, to be fed to the adaptive CPG. The Hopf oscillator can also be used as a filter that suppresses the measurement noise [26]. Eventually, entraining the periodic stimulus signal x_1, the adaptive CPG can modulate its frequency and keep the robot's traveling velocity constant under varying friction conditions.

Table 2 Parameters of the adaptive CPG network

Parameters	Values
u_{0e}	1
u_{0f}	1
τ_1	0.595
τ_2	$\tau_1/0.5$
β	3.5
w_{ef}	2.0
w_{fe}	2.0
$h_{input\ signal}$	0.6

4 Numerical simulation

We performed simulations of undulatory motion by the proposed control architecture through co-simulation with Matlab and RecurDyn. Specifically, we aimed to verify the self-modulation of the CPG parameters relevant to the traveling wave's frequency, assuming that the robot locomotion velocity needs to remain unchanged under varying or unknown environments.

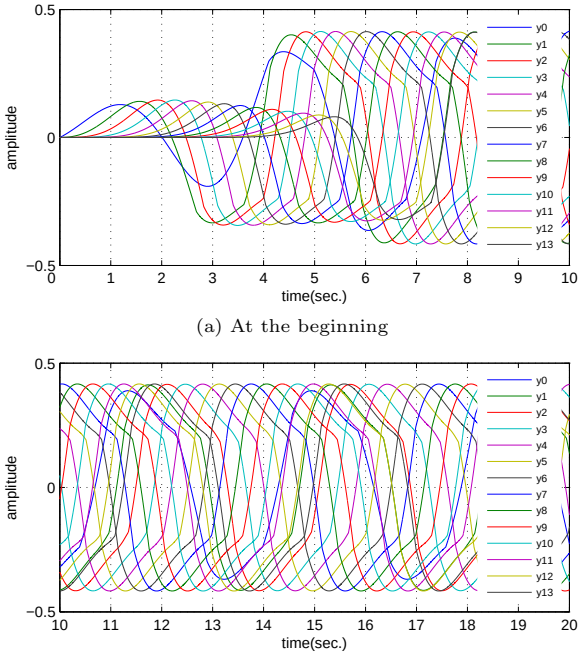
4.1 Undulatory locomotion by the adaptive CPG

First, we tested snake-like robot locomotion using the adaptive CPG network with parameters shown in Table 2. In this simulation, no sensory signal was fed back to the CPG. Fig. 7 shows that the CPG network can make sustained rhythmic patterns for joint input signals. The robot successfully exhibited undulatory locomotion as shown in Fig. 8. The time step between the snapshots is 0.6 seconds.

Now we investigate how the oscillator parameters affect the locomotion behavior of the robot. The phase lag depends mainly on the adaptation coefficient, β. If β is set to a large value, the phase lag will be large. This corresponds to the number of S-shapes along the robot body. The frequency of oscillation can be controlled by the time constant, τ_1. If τ_1 is set to a large value, the frequency becomes high. This corresponds to the locomotion velocity. The positive tonic inputs of each neuron, u_{0e} and u_{0f}, modulate the amplitude of S-shaped traveling wave, and the difference between the inputs makes a left or right turn. Fig. 9 shows the CPG output and left turning motions controlled by the difference of the tonic inputs.

4.2 Frequency adaptation under different friction conditions

In order to verify the validity of the proposed frequency adaptation, we investigate how the frictional force exerted on the robot body affects its locomotion veloc-



(a) At the beginning

(b) After convergence

Fig. 7 Output pattern of CPG

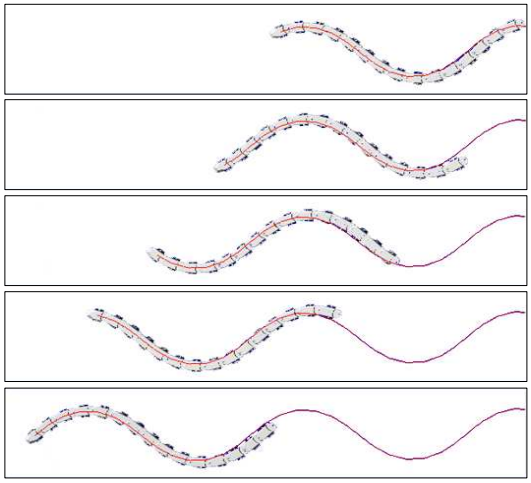
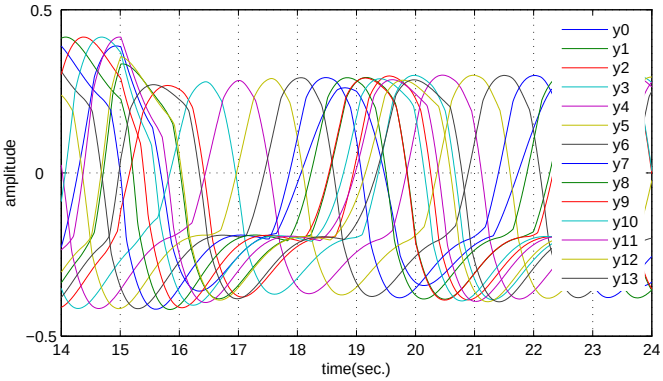
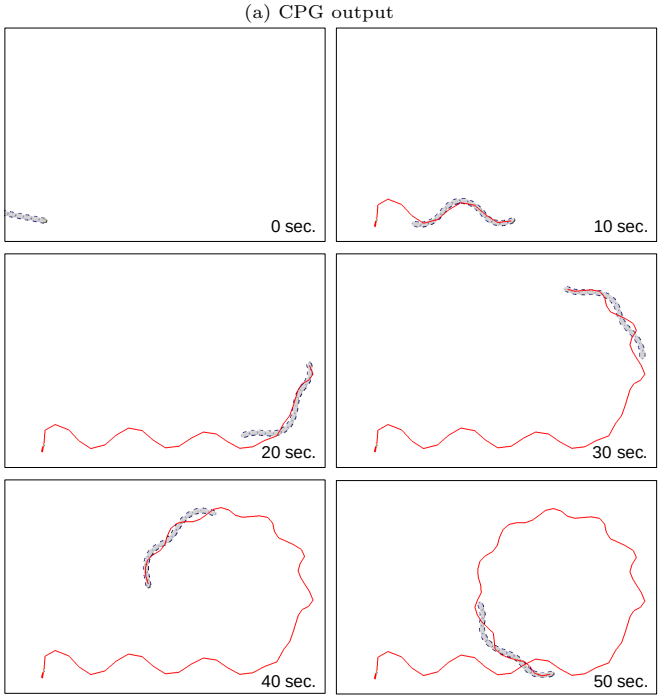


Fig. 8 The shape of the robot during forward locomotion



(a) CPG output



(b) Snake-like robot's motion

Fig. 9 Left turning motion (at 15 sec., u_{0e}: 1→0.7, u_{0f}: 1→1)

ity. Fig. 10 shows the variation in forward locomotion velocity when the coefficient of friction changes. Note that the locomotion velocity is affected by the ground friction, but when the coefficient of friction reaches a certain threshold value, the velocity remains almost unchanged. The threshold level above which the robot can effectively get a propulsive force will be determined by the coefficient of friction between the passive wheels and the ground as well as the shape of the oscillating robot body. It can be observed from the figure that when the coefficient of friction is high enough, since all segments line up straightly before starting an undulating motion, the initial velocity rapidly increases.

We now compare two cases where there is frequency adaptation or not. In both cases, the periodic stimulus signal obtained from sensory feedback is provided (See Eqs. 27, 28, and 29). Fig. 11 shows the case where there is no frequency adaptation. The robot moves over different types of terrain (ground A ⇒ ground B). One can notice that the robot can not keep its desired head velocity after the coefficient of friction changes at the instant of 30 sec. as shown in Fig. 11-(a). The robot can not modulate its pattern of coordinated locomotion adapting to changes in stimulus signal. Fig. 11-(b, c, d) indicate that the phase difference between the head and tail joints is not kept constant. Finally, the robot can not move forward, but wiggles in place. Fig. 12 shows

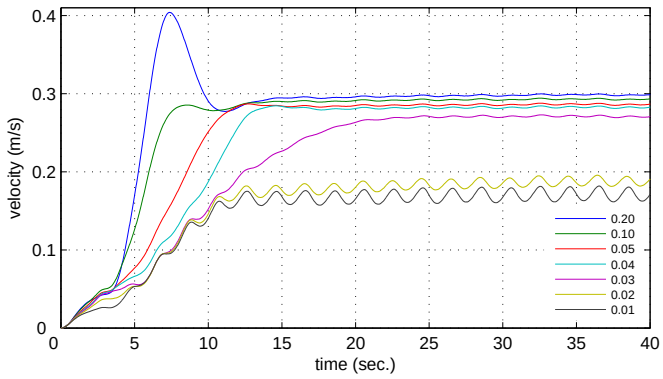
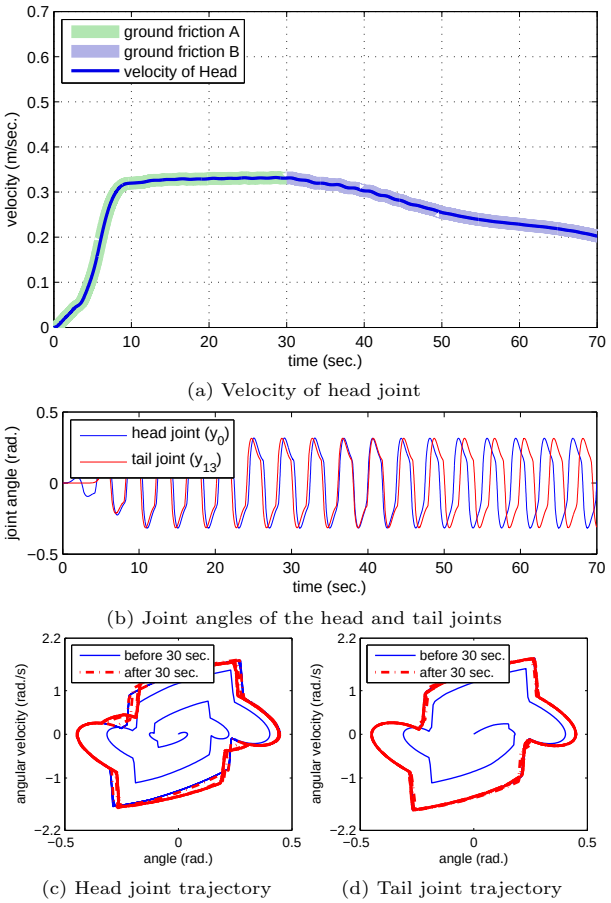


Fig. 10 Effect of the ground friction variation on the velocity



(a) Velocity of head joint

(b) Joint angles of the head and tail joints

(c) Head joint trajectory

(d) Tail joint trajectory

Fig. 11 Adapting to terrains with different friction coefficients through constant frequency (friction coefficient A: 0.1 ⇒ friction coefficient B: 0.04 at 30 sec.)

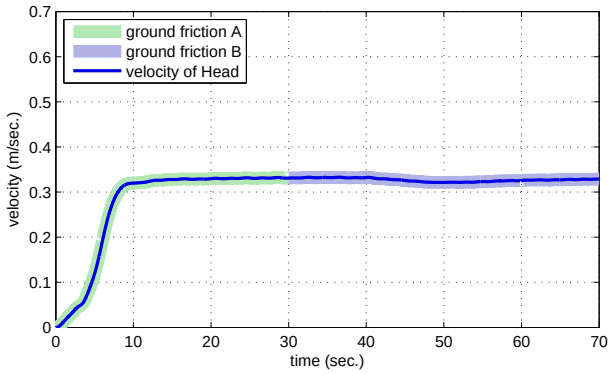
the case where our frequency adaptation rule is employed. The snake-like robot can adapt to a different ground friction self-modulating its CPG parameter to maintain the desired head velocity. Fig. 12-(b, c, d) indicate that the phase difference between the head and tail joints remains unchanged. It can be also observed from Fig. 12-(c, d) that the orbital shape enlarges along the vertical direction (symmetrically with respect to the horizontal axis) according to the change in ground friction, when compared to Fig. 11-(c, d). It is apparent that the frequency of the proposed CPG network becomes higher to maintain the desired locomotion velocity.

5 Conclusion

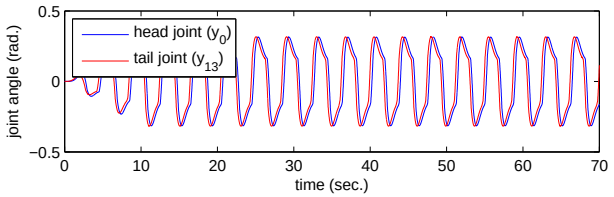
In this paper, we proposed a CPG-based control architecture to produce stable undulatory locomotion patterns for snake-like robots by modulating the CPG parameters according to changes in ground frictions. This self-modulation function controlled eventually the locomotion velocity of the robot through modulating the traveling wave's frequency while keeping a specific phase difference between each joint of the robot body segment. For this, we employed our frequency-adaptive oscillator to build the CPG network, which can learn and adapt to the changes in the frequency of external signals. It facilitated the configuration of a closed-loop CPG much more robust against environmental and/or system parameter changes. In order to verify the validity of the self-modulation function of the proposed control architecture, we carried out dynamic simulations with a simulated snake-like robot under different ground friction conditions. Employing the frequency-adaptation capability of our adaptive CPG, the snake-like robot could keep its locomotion velocity, and showed the capability to maintain its locomotion patterns and phases against changes in the ground friction. Our future study will attempt to investigate the direct feedback from sensory systems to the oscillators distributed in each joint. This will allow us to achieve a more efficient locomotion of the snake-like robot especially where terrain slopes exist.

References

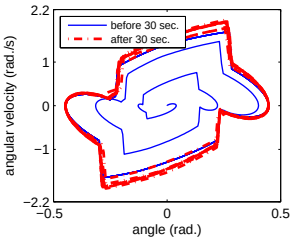
1. S. Hirose, Biologically inspired robots - snake-like locomotors and manipulators, Oxford university press: Oxford (1993)
2. T. Lee, T. Ohm, and S. Hayati, A highly redundant robot system for inspection, Conf. on Intelligent Robotics in the Field, Factory, Service, and Space, pp. 142-149 (1994)
3. N. Takanashi, K. Aoki, and S. Yashima, A gait control for the hyper-redundant robot o-ro-chi, Proc. JSME Annual Conference on Robotics and Mechatronics, pp. 79-80 (1996)
4. K. L. Paap, M. Dehlwisch, and B. Klaassen, GMD-Snake: A Semi-Autonomous Snake-like Robot, Int. Symp. on Distribted Autonomous Robotic Systems, pp. 29-31 (1996)
5. J. Burdick, J. Radford, and G. Chirikjian, A 'siderwinding' locomotion gait for hyper-redundant robots, Adavanced Robotics, vol. 9, no. 3, pp. 195-216 (1995)
6. J. Ostrowski and J. Burdick, Geometric perspectives in the mechanics and control of robotic locomotion, Proc. Int. Symp. Robotics Research, pp. 487-504 (1995)



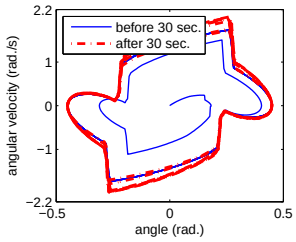
(a) Velocity of head joint



(b) Joint angles of the head and tail joints



(c) Head joint trajectory



(d) Tail joint trajectory

Fig. 12 Adapting to terrains with different friction coefficients through frequency-modulation (friction coefficient A: 0.1 ⇒ friction coefficient B: 0.04 at 30 sec.)

7. Z. Y. Bayraktarouglu, F. Butel, P. Blazevic, and V. Pasqui, A geometrical approach to the trajectory planning of a snake-like mechansim, Proc. IEEE Int. Conf. on Intelligent Robots and Systems, pp. 1322-1327 (1999)
8. J. Conradt and P. Varshavskaya, Distributed central pattern generator control for a serpentine roboto, Proc. Joint Int. Conf. on Artificial Neural Networks and Neural Information Processing, pp. 338-341 (2003)
9. O. Ekeberg, A combined neuronal and mechanical model of fish swimming, Biological Cybernetics, vol. 69, pp. 363-374 (1993)
10. A. J. Ijspeert and J. Kodjabachian, Evolution and development of a central pattern generator for the swimming of a lamprey, Artificial Life, vol. 5, no. 3, pp. 247-269 (1999)
11. A. J. Ijspeert, A connectionist central pattern generator for the aquatic and terrestrial gaits of a simulated salamander, Biological Cybernetics, vol. 84, no. 5, pp. 33-348 (2001)
12. K. Inoue, S. Ma, and C. Jin, Neural oscillator network-based controller for meandering locomotion of snake-like robots, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 5064-5069 (2004)
13. Z. Lu, S. Ma, B. Li, and Y. Wang, 3D locomotion of a snake-like robot controlled by cyclic inhibitory CPG model, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 3897-3902 (2006)
14. K. Inoue, T. Sumi, and S. Ma, CPG-based Control of a Simulated Snake-like Robot Adaptable to Changing Ground Friction, Proc. IEEE Int. Conf. on Robotics and Automation, pp. 1957-1962 (2007)
15. A. Crespi and A. J. Ijspeert, Amphibot II: an amphibious snake-like robot that crawls and swims using a central pattern generator, Proc. Int. Conf. on Climbing and Walking Robots, pp. 19-27 (2006)
16. http://www.functionbay.co.jp/
17. K. Matsuoka, Sustained oscillations generated by mutually inhibiting neurons with adaptation, Biological Cybernetics, vol. 52, pp. 97-111 (1985)
18. H. Kimura, Y. Y. Fukuoka, and A. H. Cohen, Biologically inspired adaptive dynamic walking of a quadruped robot, Proc. Int. Conf. on the Simulation of Adaptive Behavior, pp. 201-210 (2003)
19. G. Taga, A model of the neuron-musclo-skeletal system for human locomotion: I. emergence of basic gait, Biological Cybernetics, vol. 73, no. 2, pp. 97-111 (1995)
20. M. Williamson, Neural control of rhythmic arm movements, Neural Networks, vol. 11, no. 7-8, pp. 1379-1394 (1998)
21. L. Righetti, J. Buchli, and A. J. Insjpeert, Dynamic hebbian learning in adaptive frequency oscillators, Physica D, vol. 216, pp. 269-281 (2006)
22. B. Ermentrout and N. Kopell, Learning of phase lags in coupled oscillators, Neural Computation, vol. 6, pp. 225-241 (1994)
23. R. S. Sutton and A. G. Barto, Toward a modern theory of adaptive networks: expectation and prediction, Psychological Review, vol. 88, no. 2, pp. 135-170 (1981)
24. J. Nishii, A learning model for oscillatory networks, Neural Networks, vol. 11, pp. 249-257 (1998)
25. R. Héliot and B. Espiau, Online generation of cyclic leg trajectories synchronized with sensor measurement, Robotics and Autonomous Systems, vol. 56, pp. 410-421 (2008)
26. A. Pikovsky, M. Rosenblum, and J. Kurths, Synchronization - a universal concept in nonlinear sciences, Cambridge (2001)
27. X. Yao, Evolving artificial neural networks, Proc. of the IEEE, vol. 87, no. 9, pp. 1423-1447 (1999)