International Journal of Advanced Robotic Systems

OPEN ACCESS ARTICLE

CPG-based Locomotion Controller Design for a Boxfish-like Robot

Regular Paper

Wei Wang[1,*] and Guangming Xie[1]

1 Industrial Engineering & Management, Peking University, Beijing, China
* Corresponding author E-mail: wangweiw4y4@pku.edu.cn

Received 11 Feb 2014; Accepted 03 Apr 2014

DOI: 10.5772/58564

Abstract This paper focuses on a Central Pattern Generator (CPG)-based locomotion controller design for a boxfish-like robot. The bio-inspired controller is aimed at flexible switching in multiple 3D swimming patterns and exact attitude control of yaw and roll such that the robot will swim more like a real boxfish. The CPG network comprises two layers, the lower layer is the network of coupled linear oscillators and the upper is the transition layer where the lower-dimensional locomotion stimuli are transformed into the higher-dimensional control parameters serving for all the oscillators. Based on such a two-layer framework, flexible switching between multiple three-dimensional swimming patterns, such as swimming forwards/backwards, turning left/right, swimming upwards/downwards and rolling clockwise/counter-clockwise, can be simply realized by inputting different stimuli. Moreover, the stability of the CPG network is strictly proved to guarantee the intrinsic stability of the swimming patterns. As to exact attitude control, based on this open-loop CPG network and the sensory feedback from the Inertial Measurement Unit (IMU), a closed-loop CPG controller is advanced for yaw and roll control of the robotic fish for the first time. This CPG-based online attitude control for a robotic fish will greatly facilitate high-level practical underwater applications. A series of relevant experiments with the robotic fish are conducted systematically to validate the effectiveness and stability of the open-loop and closed-loop CPG controllers.

Keywords Robotic Fish, Open-loop CPG, Closed-loop CPG, Attitude Control, Sensory Feedback

1. Introduction

Locomotion control has been a durative hot research topic among the robotics community due to its fundamentality in mobile robot research. Conventional locomotion controllers need to model the dynamics of the robot, which, in most cases, is too complicated to be established. Moreover, deficiencies like limited flexibility, adaptability and stability make these model-based approaches incompetent at easily generating a smooth trajectory online and responding to the unknown environment rapidly. Therefore, more and more researchers have taken inspiration from animal locomotion, which has evolved to be efficient, rapid, adjustable and reliable under evolutionary pressures [1]. One class of the promising bio-inspired controllers is the CPG controller [2], which consists of coupled networks capable of producing coordinated oscillatory patterns of rhythmic activity while receiving simple adjustment signals from higher control centres [2–4]. It is desirable that the artificial CPG controllers possess the similar intrinsic properties of rhythm, coordination, variety and robustness to those naturally occurring CPGs in animals [2]. The artificial CPG controllers have been extensively studied and applied to various bio-inspired robots, such as bipedal-

or quadrupedal-legged robots [5–7], robotic fish [8–10], a salamander robot [11] and a snake-like robot [12].

Noticeably, a wide variety of CPG controllers have been designed for and applied to bio-inspired swimming robots which have drawn more and more attention over the past few decades. Research on the CPG-based locomotion controllers of swimming robots can be divided into two categories: open-loop CPG controllers and closed-loop CPG controllers. Open-loop CPG controllers, which have primarily been focused on in the existing literature, are always employed to generate multiple swimming behaviours and to realize smooth transitions between those related behaviours. Examples include Ijspeert et al.'s salamander robot [11] and fish-like robot [8], Yu et al.'s robotic fish [13], Kamimura et al.'s snake-like robot [12] and our group's robotic fish [10, 14, 15]. In contrast, closed-loop CPG controllers always integrate sensory feedback into the CPG network to regulate the CPG behaviours automatically, which is more like the real mechanism of animal CPG activities. According to the specific surroundings and tasks, closed-loop CPG controllers are able to realize automatic swimming gait transition [9], pattern synchronization [16], gait stability [17] and gait optimization [18].

Although open-loop and closed-loop CPG controllers have been extensively investigated on the swimming robots, most of CPG controllers need massive control parameters working simultaneously to generate a specific swimming pattern. Moreover, most of the CPG controllers only produce planar motions or partially typical swimming behaviours. It is unusual for CPG-based locomotion controllers to be designed to involve all typical swimming patterns for a certain swimming robot just as it is with its natural counterpart. More remarkably, exact attitude control of swimming robots within the CPG-based control framework was rarely been tackled in the literature. One example can be found in Seo et al. [16], where a closed-loop CPG method is proposed to control the attitude of the foil fins and altitude of the underwater vehicle by synchronizing the 8-DOF foil fins.

Motivated by the above discussion and based on our previous work on robotic fish design and locomotion control [10], this paper aims at designing a CPG-based locomotion controller, which endows the robot with abilities in flexible switching among multiple 3D swimming patterns and exact attitude control of yaw and roll of the robot. First, by importing a transition layer, we primarily improve the open-loop CPG network proposed in [8] to generate all typical 3D swimming patterns with only four control parameters. Moreover, the stability of the built CPG model is proved strictly to ensure the intrinsic stability of swimming patterns. By employing sensory feedback from the IMU, we propose a closed-loop CPG controller to realize automatic attitude control in the robot's yaw and roll angles while swimming. Using the suggested CPG-based locomotion controllers, typical swimming pattern transitions and attitude control are further validated on a robotic fish.

The rest of this paper is organized as follows: Section II gives an overview of the robotic fish. The open-loop and closed-loop CPG networks are formulated in Section III. Experiments are provided in Section IV to validate the proposed CPG-based networks. Section V concludes this paper with an outline of future work.

2. Prototype of the Boxfish-like Robot

With the purpose of designing manoeuvrable and fully autonomous 3D motions, we attempt to fuse the mechanical structure, functional characteristics and multiple sensors of the physical robot. That is, the conceived robot has as many equivalent functions as possible compared with its biological counterpart while remaining a compact mechanical structure. Therefore, the dorsal and anal fins of the natural boxfish are excluded in the robot design since they play less of a role compared with the paired pectoral fins and the caudal fin. Figure 1a shows the mechanical configurations of the boxfish robot consisting of a roughly rectangular main body, a pair of pectoral fins and a caudal fin. The rigid and waterproof main body housing rechargeable battery, sensors and actuators, is composed of two parts: the upper case body and the lower case body. The upper is made of transparent Fibre Reinforce Plastic (FRP) while the lower is made of lightened polyurethare. Both static and moving seals are adopted to ensure the robot is waterproof. Specifically, tailor-made O-rings and silicone adhesives are used between the two external casings, while seals and grease are used between the driving shaft and the body shell. Note that the maximum operating depth of the robot is only about one metre due to the technical limitations of the waterproofing in our laboratory condition. All three replaceable fins are made of Polyethylene with a non-uniform thickness to approximate the hardness and flexibility of real fins. The robot density has been calibrated close to that of water so that it can float in water. Mass distribution is well balanced by placing a clump weight in the tested position to achieve smooth rolling motions.

The robot is operated by an embedded Linux system and can work in two control modes: the manual mode and autonomous control mode. In the manual mode, the robot is dominated by a remote control server via a wireless communication network (WiFi). In the autonomous control mode, the robot is able to swim autonomously and interacts with its near environment by using its onboard sensors: the Inertial Measurement Unit (IMU) and camera

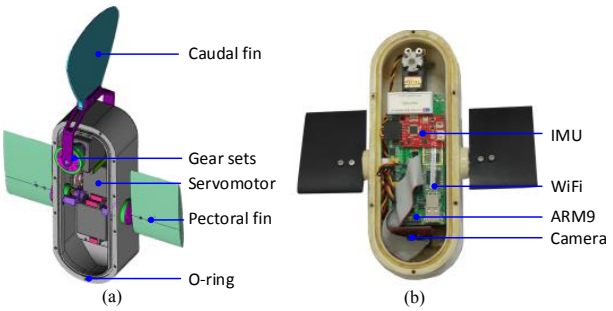

Figure 1. Mechatronics of the boxfish robot. (a) Mechanical configuration of the robot. (b) Electronics and sensors in the robot.

Items	Characteristics
Dimension(L×W×H)	∼330 mm× 80 mm × 90 mm
Total mass	∼1.40 kg
Drive mode	DC servermotors (9.4 kg·cm)
Controller	Camera and IMU
Onboard sensors	Samsung S3C2440A
Power supply	7.2 V rechargeable Ni-MH batteries
Operation time	∼ 1.5 h
Maximum forward speed	1.05BL/s

Table 1. Technical specifications of the boxfish robot prototype

[10], as shown in Figure 1b. Specifically, the IMU consisting of a triaxial accelerometer, a triaxial gyroscope and a triaxial electronic compass, is fixed parallel to the body principal axis to monitor the robot's pitch, yaw and roll angles. It can also measure the robot's accelerations for future developments. The sampling rate is set at 50 Hz for real time application. The vision sensor is placed centrally in the front of the main body to capture images of surroundings for obstacle avoidance and localization. Table 1 lists the basic technical specifications of the robot.

3. CPG-based Locomotion Control

To mimic the excellent locomotion capacity of natural fish, two kinds of CPG networks are designed in this section. First, we give a simplified dynamic analysis of ostraciiform robotic fish. Based on the locomotion analysis, we then propose a two-layer CPG controller with explicit expressions to produce multiple swimming patterns and flexible switching among these patterns. Next, we prove the stability of the built open-loop CPG model to ensure the intrinsic stability of the robot locomotion. More strikingly, by employing the sensory feedback from the IMU, we propose a closed-loop CPG controller to realize automatic attitude control of the robot while swimming. Automatic attitude control is one of the key abilities when the robot executes specific tasks in an unknown environment, such as passing through narrow spaces, trajectory tracking and multi-robot formation control.

3.1. Simplified Dynamic Analysis

A simplified dynamic analysis of robotic fish is provided to explain why the robot has various swimming behaviours in the water. As an ostraciiform robotic fish, the robot swims in the water by actively oscillating its bio-inspired fins while the rigid main body keeps to small passive oscillations. Therefore, the propulsive force is mainly provided by the robot's oscillating fins (the paired pectoral fins and caudal fin) which push the nearby water. According to the conservation of momentum, the water stream will give each oscillating fin a reaction force, which acts on the centre of pressure of the fin and pushes the fin with a direction parallel to the midplane of the oscillating fin. Since the rotating shafts of the fins are fixed on the robot, the reaction force will equivalently act on the robot. For the robotic fish, because the paired pectoral fins are able to rotate 360 degrees and the caudal fin is able to rotate about 120 degrees, a variety of reaction forces can be generated and then act on the robot by oscillating different combinations of these three fins. Therefore, the robot is able to generate multiple swimming patterns.

Figure 2 shows the force analysis on eight typical swimming patterns while other patterns can be analysed in a similar way. Note that the reference position for the pectoral fins is when the fins are turned backwards in a horizontal position while the reference position for the caudal fin is when that fin is in the sagittal plane, as illustrated in Figure 2a. Note also that the indicated forces F_l, F_r and F_t in Figure 2 are the average values during one beating period. Here, we analyse several swimming patterns for example: (i) backward swimming; by turning the pectoral fins forwards, two backward forces F_l and F_r are generated and therefore, the robot will swim backwards, as shown in Figure 2c. (ii) Rolling motion; by turning one pectoral fin upside and the other pectoral fin downside, the oscillating pectoral fins will produce opposite forces on the robot, which rotate the robot about its longitudinal body axis. Furthermore, we have built two three-dimensional dynamic models for the robotic fish, and the details can be found in [19, 20].

3.2. Two-layer Open-loop CPG Controller

With the purpose of simplifying the stability proof as well as maintaining the intrinsic properties of the CPG network,

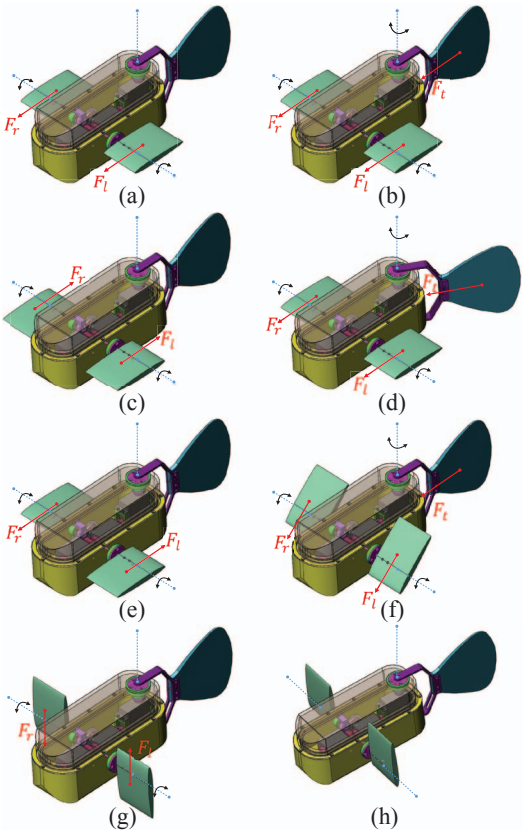

Figure 2. Simplified force analysis of the robot in typical locomotor patterns. (a) Swimming forwards [paired-fin (PF) mode], (b) swimming forwards [body-caudal fin (BCF)+PF mode], (c) swimming backwards, (d) turning, (e) turning on the spot, (f) pitching. (g) rolling, (h) braking.

we linearize the nonlinear phase oscillator proposed in [8] through the first-order Taylor expansion. Thus, the coupled linear oscillators take the forms

$$\dot{a}_i = \alpha_i(A_i - a_i) \tag{1a}$$
$$\dot{x}_i = \beta_i(X_i - x_i) \tag{1b}$$
$$\dot{\phi}_i = 2\pi f_i + \sum_{j\in T_i} \mu_{ij}(\phi_j - \phi_i - \varphi_{ij}) \tag{1c}$$
$$\theta_i = x_i + a_i \cos(\phi_i) \tag{1d}$$

where a_i, x_i, and ϕ_i are the state variables representing the amplitude, offset and phase of the i^{th} oscillator, and the variable θ_i is its output. The parameters f_i, A_i and X_i are the control parameters for the desired frequency, amplitude and offset of the oscillations. μ_{ij} and φ_{ij} determine the coupling weight and phase bias of the j^{th} oscillator to the i^{th} oscillator. α_i and β_i are structure parameters, representing the dynamic performance of i^{th} oscillator. T_i is the set of neighbours of oscillator i exerting couplings on the i^{th} oscillator. Note that the subscripts $i = 1, 2, 3$ in this paper represent the left pectoral fin, right pectoral fin and caudal fin of the robotic fish, respectively.

In order to reduce the complicatedness of swimming pattern switching, a two-layer CPG network is proposed, as illustrated in Figure 3. The input drive received from the upper controller can be divided into four command inputs, namely d_v, d_y, d_p and d_r. The command d_v determines the robot speed, while d_y, d_p and d_r regulate the robot yawing, pitching and rolling motions, respectively. In contrast with other CPG networks, a transition layer with only four control variables is defined explicitly to generate all the typical 3D swimming patterns, such as forward/backward swimming, turning, pitching and clockwise/counter-clockwise rolling. Below we give the expressions of the transition layer.

Inspired by the swimming characteristics of the boxfish where the swimming speed is roughly linear with the frequency and amplitude [21–23], we use a combination of f_i and A_i for speed control of the robot, which are individually determined by the input stimulus d_v. Moreover, with the input drive d_v, the robot performs two basic swimming modes, that is, PF model and PF+BCF mode, which can satisfy various speed demands as well as reduce energy consumption. Then, details of the functions are given as follows:

$$f_i = \begin{cases} c_{f,i}^{PF} d_v + f_{i,b}^{PF}, & \text{if } d_v^{i,\min} \le d_v < d_v^{trans}; \\ c_{f,i}^{PBC} d_v + f_{i,b}^{PBC}, & \text{if } d_v^{trans} \le d_v \le d_v^{i,\max}; \\ f_{i,sat}, & \text{otherwise.} \end{cases} \tag{2}$$

$$A_i = \begin{cases} c_{A,i}^{PF} d_v + A_{i,b}^{PF}, & \text{if } d_v^{i,\min} \le d_v < d_v^{trans}; \\ c_{A,i}^{PBC} d_v + A_{i,b}^{PBC}, & \text{if } d_v^{trans} \le d_v \le d_v^{i,\max}; \\ A_{i,sat}, & \text{otherwise.} \end{cases} \tag{3}$$

where $d_v^{i,\min}$ and $d_v^{i,\max}$ represent the lower and upper threshold drive of the i^{th} oscillator, respectively; d_v^{trans} is the critical transition drive that triggers the swimming mode from PF to PF+BCF . $[c_{fi}^{PF},\ f_{i,b}^{PF},\ c_{fi}^{PF+BCF},\ f_{i,b}^{PF+BCF}\ f_{i,sat}]$ and $[c_{Ai}^{PF},\ A_{i,b}^{PF},\ c_{Ai}^{PF+BCF},\ A_{i,b}^{PF+BCF},\ A_{i,sat}]$ are the frequency and amplitude coefficients, respectively.

Next, the combination of offset X_i's is used for the yaw, pitch and roll control of the robot, which are relevant to d_y, d_p and d_r. For the robot with the paired pectoral fins and one caudal fin, the input drive d_y turns the robot by providing offset of the caudal fin, d_p donates the attack angle of the paired pectoral fins and makes the robot pitch, and d_r rolls the robot by taking the paired pectoral fins with a phase difference of π. The functional forms of X_i's are given as

$$X_1 = \begin{cases} d_p + \text{sgn}\left(1 + (-1)^{\text{sgn}(d_p)}\right) d_r, \\ \text{if} d_p \in [-0.5\pi, 0] \cup [0, 0.5\pi] \cup \{\pi, -\pi\}, \\ d_r \in \{-0.5\pi, -\pi, 0.5\pi, \pi\}; \\ 0, \text{ otherwise.} \end{cases} \tag{4a}$$

$$X_2 = \begin{cases} d_p + \text{sgn}\left(1 + (-1)^{\text{sgn}(d_p)}\right)(d_r - \pi\text{sgn}(d_r)), \\ \text{if} d_p \in [-0.5\pi, 0] \cup [0, 0.5\pi] \cup \{\pi, -\pi\}, \\ d_r \in \{-0.5\pi, -\pi, 0.5\pi, \pi\}; \\ 0, \text{ otherwise.} \end{cases} \tag{4b}$$

$$X_3 = \begin{cases} d_y, & \text{if } d_y \in [-\pi/3, \pi/3]; \\ 0, & \text{otherwise.} \end{cases} \tag{4c}$$

where $\text{sgn}(\cdot)$ is a signum function and is defined as follows:

$$\text{sgn}(t) = \begin{cases} -1, & \text{if } t < 0; \\ 0, & \text{if } t = 0; \\ 1, & \text{otherwise.} \end{cases} \tag{5}$$

To explicitly express the relationship between the control parameter set $\{d_v, d_y, d_p, d_r\}$ and multiple swimming patterns, a mapping table is presented in Table 2. Moreover, by using the CPG network, the robot is able to switch smoothly between different swimming patterns. Figure 4 demonstrates a sequence of swimming pattern transitions from one to another. In the sequence, the swimming patterns are performed as follows: swimming forwards with PF mode ($0 \le t \le 4$ s), turning ($4 \le t \le 8$ s), swimming backwards ($8 \le t \le 12$ s), swimming

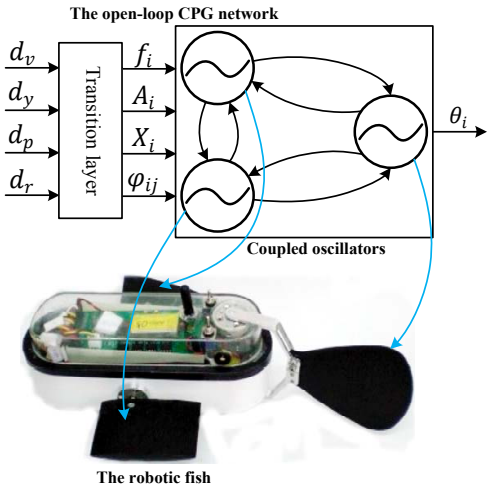

Figure 3. The proposed open-loop CPG model

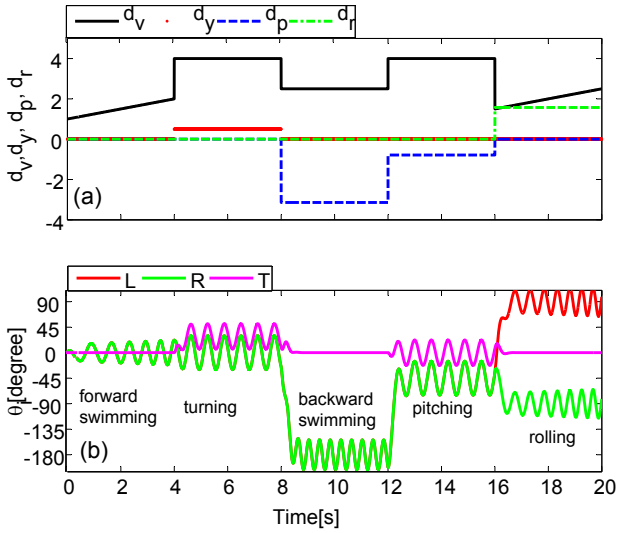

Figure 4. Illustration of the robot performing multiple swimming patterns. (a) Input drive d; (b) output signals θ_i for joint actuation.

upwards ($12 \leq t \leq 16$ s) and rolling counter-clockwise ($16 \leq t \leq 20$ s). We can see that the robot is able to transit smoothly and rapidly between different swimming behaviours. Thus, the proposed open-loop CPG controller is effective in switching multiple 3D swimming patterns online. The used CPG parameters are further listed in Table 3.

3.3. Stability Proof of Open-loop CPG Model

By using theory of multi-agent systems (MAS), we prove the stability of the employed open-loop CPG model. To facilitate the analysis, three reasonable assumptions are made. 1) Dynamic parameters α_i, β_i and μ_{ij} of each oscillator are set as follows: $\alpha_i = \alpha$, $\beta_i = \beta$ and $\mu_{ij} = \mu$ where $\alpha \in \mathbb{R}^+$, $\beta \in \mathbb{R}^+$ and $\mu \in \mathbb{R}^+$. 2) φ_{ij} can be expressed as $\varphi_{ij} = \varphi_j - \varphi_i$, where $_i \in \mathbb{R}$. 3) In most applications, the oscillating frequencies f_i are always set to be equal, therefore, here $f_i = f$.

First, Equations (1a) and (1b) explain the dynamics of amplitude and offset of oscillator i, and solutions can be easily derived

$$a_i(t) = A_i + (A_{i0} - A_i)e^{-\alpha(t-t_0)} \tag{6}$$

$$x_i(t) = X_i + (X_{i0} - X_i)e^{-\beta(t-t_0)} \tag{7}$$

It is obvious that $a_i(t)$ and $x_i(t)$ will exponentially converge to A_i and X_i from any initial states, respectively. Next, our focus is on the stability proof of Equation (1c) and (1d).

Letting $z_i = \phi_i - \varphi_i$, then Equation (1c) is expressed as follows:

$$\dot{z}_i = 2\pi f + \sum_{j \in T_i} \mu(z_j - z_i) \tag{8}$$

Next, using algebraic graph theory, we rewrite Equation (8) as the matrix form

$$\dot{\boldsymbol{z}} = -L\boldsymbol{z} + 2\pi f \mathbf{1} \tag{9}$$

where $\boldsymbol{z} = [z_1\ z_2\ \cdots\ z_N]^T$, $\mathbf{1}$ donates the $N \times 1$ column vector of all ones. $L = (l_{ij})_{N\times N}$ donating the Laplacian matrix of CPG network takes the form [24]

$$l_{ij} = \begin{cases} (N-1)\mu, & i = j; \\ -\mu, & i \neq j. \end{cases} \tag{10}$$

Because μ is real, then L is a real, symmetric and semi-definite positive matrix. It has N nonnegative eigenvalues: $\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_N$. Thus, we can diagonalize L as

$$T^T L T = \begin{bmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \cdots & \\ & & & \lambda_N \end{bmatrix} \tag{11}$$

where T is an orthogonal matrix satisfying $TT^T = I$, and I is the identity matrix. Then, letting $\boldsymbol{z} = T\boldsymbol{y}$, we rewrite Equation (9) as follows:

$$\dot{\boldsymbol{y}} = -T^{-1}LT\boldsymbol{y} + 2\pi f T^{-1}\mathbf{1} \tag{12}$$

Further, letting $T = [_1\ _2\ \cdots\ _N]$ where $_i$ is a column vector. Thus, Equation (13) can be expressed as

$$\dot{y}_i = -\lambda_i y_i + 2\pi f\, {}_i^T \mathbf{1} \tag{13}$$

Since every oscillator is affected by the other $N-1$ oscillators, the graph here is complete. By definition, the sum of every row of the Laplacion matrix is zero. Thus, the Laplacian matrix always has a zero eigenvalue, that is, $\lambda_1 = 0$. Thus, by combining the property of orthogonal matrix, we can get two main results: (1)$\mathbf{1} = \sqrt{N}_1$; (2) ${}_i^T 1 = 0$ where $i \neq 1$. According to these two results, we can solve the differential Equation (13)

$$y_i = \begin{cases} 2\pi f\sqrt{N}t + y_i(0), & i = 1; \\ e^{-\lambda_i t} y_i(0), & i = 2, 3, \cdots, N. \end{cases} \tag{14}$$

where $y_i(0)$ is the initial state of y_i. It is easy to see that $y_i (i \geq 2)$ will exponentially attenuate to zero as $t \to \infty$. Then, z_i will converge to $2\pi f t + z_1(0)$ where $z_1(0) = \frac{1}{\sqrt{N}} y_1(0)$. As a result,

$$\phi_i = z_i + \varphi_i \to 2\pi f t + z_1(0) + \varphi_i \tag{15}$$

$$\phi_j = z_j + \varphi_j \to 2\pi f t + z_1(0) + \varphi_j \tag{16}$$

Thus, $\phi_j - \phi_i \to \varphi_{ij}$, and Equation (1c) is stable.

3.4. Closed-loop CPG Model

Using sensing systems, animals always need to stabilize their motion states (such as attitude and position) in their daily activities (for instance, in prey and migration). To duplicate the ability of attitude stabilization in animals, we introduce sensory feedback into the CPG to form a closed-loop CPG network, which endows the robot with an automatic stabilizing ability while swimming. Because of the mechanical configurations of robotic fish, pitching angle is almost zero while the robot is swimming downwards or upwards in the 3D space. Therefore, only the attitude of yaw and roll angles are stabilized

Swimming pattern	Parameter set			
	d_v	d_y	d_p	d_r
Swimming forwards(PF)	[1,3)	0	0	0
Swimming forwards(PF+BCF)	(3,5]	0	0	0
Swimming backward	[1,3)	0	$\{\pi, -\pi\}$	0
Swimming on the spot	[1,3)	0	0	$\{\pi, -\pi\}$
Turning	(3,5]	$[-\pi/3, \pi/3]$	0	0
Pitching(PF)	[1,3)	0	$[-0.5\pi, 0) \cup (0, 0.5\pi]$	0
Pitching(PF+BCF)	(3,5]	0	$[-0.5\pi, 0) \cup (0, 0.5\pi]$	0
Pitching+Turning	(3,5]	$[-\pi/3, \pi/3]$	$[-0.5\pi, 0) \cup (0, 0.5\pi]$	0
Rolling(PF)	[1,3)	0	0	$\{-0.5\pi, 0.5\pi\}$
Rolling(PF+BCF)	(3,5]	0	0	$\{-0.5\pi, 0.5\pi\}$

Table 2. Mapping table between parameter set and swimming patterns

Parameters	Value	Units
α_i	10	None
β_i	10	None
μ_{ij}	$\mu_{ij} = 50$	None
c_{fi}^{PF}	1	None
$f_{i,b}^{PF}$	0	Hz
c_{fi}^{PBC}	0.6	None
$f_{i,b}^{PBC}$	-0.8	Hz
$f_{i,sat}$	0	Hz
c_{Ai}^{PF}	$c_{A1}^{PF} = c_{A2}^{PF} = \pi/24,\ c_{A3}^{PF} = 0$	None
$A_{i,b}^{PF}$	$A_{1,b}^{PF} = A_{2,b}^{PF} = \pi/24,\ A_{3,b}^{PF} = 0$	Degree
c_{Ai}^{PBC}	$c_{A1}^{PBC} = c_{A2}^{PBC} = 0,\ c_{A3}^{PBC} = \pi/24$	None
$A_{i,b}^{PBC}$	$A_{1,b}^{PBC} = A_{2,b}^{PBC} = \pi/6,\ A_{3,b}^{PF} = -\pi/24$	Degree
$A_{i,sat}$	0	Degree
d_v	$d_v^{i,\min} = 1,\ d_v^{i,\max} = 5,\ d_v^{trans} = 3$	None
[0.8pt]		

Table 3. CPG parameter values applied to the boxfish robot

in this paper. As shown in Figure 5, the closed-loop CPG network contains four parts: the coupled oscillators, feedback signal generator, PID controller and onboard IMU. Correspondingly, the set of input drive for the closed-loop CPG controller becomes $\{d_v, d_p, d_y^*, d_r^*\}$ where d_y^* and d_r^* denote the expected yaw angle and roll angle of the robot, respectively.

Specifically, the working process of the closed-loop CPG controller is described as follows. To stabilize the robot attitude with an expected yaw angle d_y^* and an expected roll angle d_r^*, the onboard IMU first records the actual yaw angle y_{IMU} and roll angle r_{IMU} of the robot in real time. Then, according to the attitude error between the actual and the expected, the PID controller generates two appropriate signals λ_y and λ_r serving as the inputs of the feedback signal generator. Based on the force analysis described in Section 3.1, the feedback signal generator produces feedback signals u_i and v_i which feed into the coupled linear oscillation network to regulate the behaviours of the CPG network. By regulating the behaviours of the CPG network, the attitude error always decreases to meet the demand of attitude stabilization. Next, we will give a detailed description of the proposed closed-loop CPG network.

First, the coupled oscillators containing the feedback signals are designed as follows:

$$\dot{a}_i = \alpha(A_i - a_i + u_i) \tag{17a}$$
$$\dot{x}_i = \beta(X_i - x_i + v_i) \tag{17b}$$
$$\dot{\phi}_i = 2\pi f_i + \sum_{j \in T_i} \mu_{ij}(\phi_j - \phi_i - \varphi_{ij}) \tag{17c}$$
$$\theta_i = x_i + a_i \cos(\phi_i) \tag{17d}$$

where u_i and v_i respectively stand for the feedback signals related to the i^{th} oscillator.

Then, based on the force analysis described in Section 3.1, the feedback signal generator are mathematically defined as follows:

$$u_1 = A_1(0.5 - \lambda_r) - A_1 \tag{18}$$
$$u_2 = A_2(0.5 + \lambda_r) - A_2 \tag{19}$$
$$u_3 = 0 \tag{20}$$
$$v_1 = 0.5\pi \text{sgn}\left(1 + (-1)^{\text{sgn}(d_p)}\right) - X_1 \tag{21}$$
$$v_2 = 0.5\pi \text{sgn}\left(1 + (-1)^{\text{sgn}(d_p)}\right) - X_2 \tag{22}$$
$$v_3 = \lambda_y - X_3 \tag{23}$$

where u_i and v_i are the outputs of feedback signal generator while λ_y and λ_r are the inputs of feedback signal

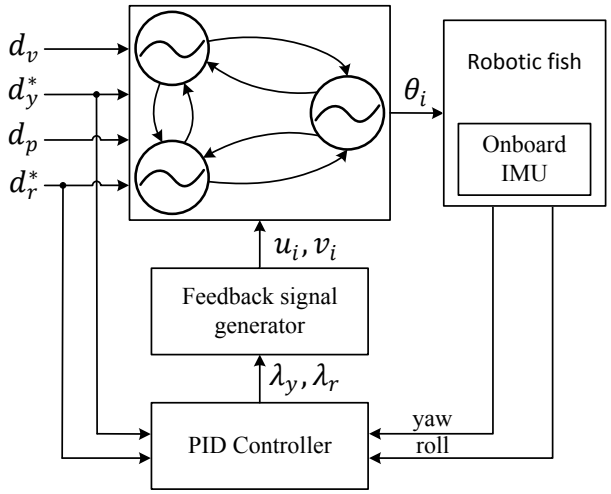

Figure 5. The proposed closed-loop CPG network

generator. Specifically, v_3 is used to stabilize the yaw angle of the robot and the combination of u_1, u_2, v_1 and v_2 is adopted to stabilize the roll angle of the robot.

Here we explain how the paired pectoral fins operate to stabilize the roll angle of the robot. First, to make the best of the force produced by the pectoral fins, the fin plane should be perpendicular to the transversal surface of the body, as shown in Figure 2g. Moreover, to maintain the roll angle at a specific value, the directions and the magnitude of the produced forces on two sides should be the same. Thus, the offsets of the paired fins are equal, as defined in Equation (21) and (22). Furthermore, the built dynamic model [25] and experimental validation [26] have shown that the thrust (speed) increases roughly and linearly with the amplitude. As a result, we use A_i as the control variable to stabilize the roll angle of the robot, as defined in Equation (18) and (19).

Moreover, the PID controller is used to generate the appropriate λ_y and λ_r according to $e_y = d_y^* - y_{IMU}$ and $e_r = d_r^* - r_{IMU}$. Then, $\lambda_y = -k_p^y e_y - k_i^y \int e_y - k_d^y \dot{e}_y$ and $\lambda_r = -k_p^r e_r - k_i^r \int e_r - k_d^r \dot{e}_r$ are the outputs of the PID controller. The ranges of λ_y and λ_r are respectively defined as $\lambda_y \in [-0.5, 0.5]$ and $\lambda_r \in [-0.5, 0.5]$ to avoid possible systematic divergence while the robot encounters large disturbances in the water. The output rate of e_y and e_r are set to be 15 Hz in the experiments. And since the robot functioning in the water is a large time-delay system, this processing rate is satisfactory in the experiments.

When the feedback is introduced into the CPG network, the obtained closed-loop dynamic system becomes a complicated nonlinear dynamic system. However, the stability analysis for the general nonlinear system is still unsolved, and we now also have not solved the stability proof of our closed-loop CPG network. Therefore, the mathematical proof of the closed-loop CPG network will be left as an open problem. Instead, extensive attitude control experiments will be performed on the robotic fish to experimentally validate the stability of the full model with feedback. The structure parameters of α and β, and the PID controller should be selected carefully. If the output rate of the PID controller is set to be high, α and β should also take large values to rapidly converge to the steady state. However, α and β values that are too large may result in an unsmooth swimming curve of the robot.

4. Experiments and Results

In this section, a series of relevant experiments with the robotic fish are conducted systematically to evaluate the proposed open-loop and closed-loop CPG controllers.

4.1. Multiple Swimming Patterns

To evaluate the ability of the proposed open-loop CPG controller, multiple swimming patterns, such as forward/backward swimming, turning, pitching and rolling motion, were first tested in a swimming tank (300 cm × 200 cm × 30 cm). Some scenarios of swimming patterns are illustrated in Figure 6. The 2D state estimation of the robot (position and speed) is analysed online using vision tracking software. While 3D state estimation is conducted offline by analysing the experimental videos. Due to the space limitation, only the pitching and rolling motions are specified below.

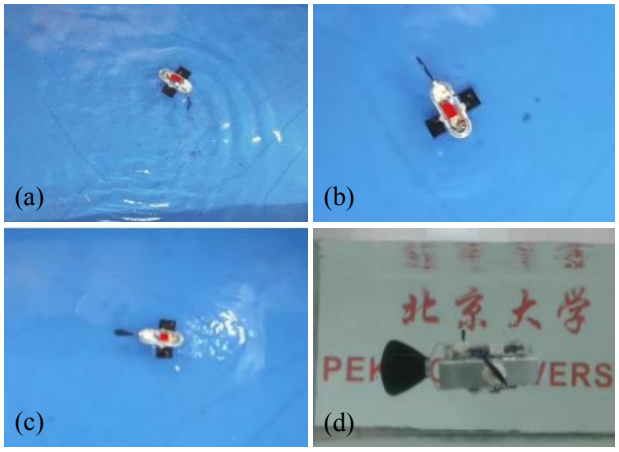

Figure 6. Snapshots of typical swimming patterns. (a) Turning, (b) turning on a spot, (c) swimming backwards, (d) swimming downwards.

4.1.1. Pitching Control

By modulating the attack angle d_p, the robotic fish can perform pitching motions in the 3D space, and this section evaluates the relationship between the attack angle d_p and the pitching speed.

Figure 7 compares time-averaged horizontal (v_x) and vertical (v_y) velocities with various attack angles d_p under a constant speed drive d_v = 2.5. During each test, the robot was configured to float in the water. The two orthometric speeds were measured offline. It is shown that v_y increases while v_x decreases roughly linearly with the increasing attack angle. However, because the transversal section of the robot body suffers larger drag force than that of the longitudinal section, the slope of v_y is smaller than v_x. Moreover, compared with the pitching motion in [9], the robotic fish is able to vertically move up and down when the angle of attack reaches 90°. This indicates that the robot

can swim in almost any direction under water, and this is quite important for path planning in an open environment with different kinds of obstacles.

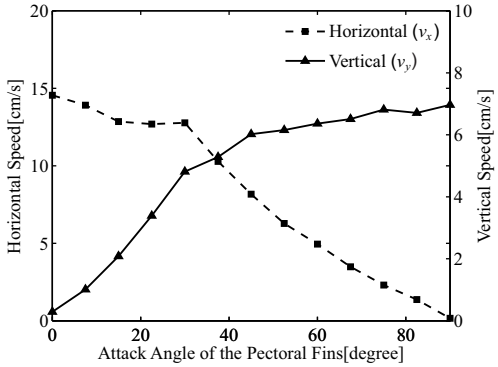

Figure 7. Relationship between the pitch speeds (horizontal speed and vertical speed) and the varied attack angle d_p while $d_v = 2.5$ and $d_y = d_r = 0$

4.1.2. Rolling Motion

This section validates the capability of the open-loop CPG in the rolling motion in the 3D space. Aquatic animals exhibit rolling manoeuvres in the wild [27]. They can be used to escape or to remove attachments from the animal's body surface. However, compared with yawing and pitching motions, the rolling motions of robotic fish have rarely been investigated in the literature, except regarding two kinds of robotic fish in Ijspeert's group [8] and Yu's group [9]. Thanks to the proposed CPG controller, we can just use input drive d_r to make the robot perform a rolling motion, as shown in Figure 8. Moreover, the angular rate of the rolling motion can be controlled by adjusting input drive d_v. Compared with the rolling motions in [8] and [9], observably, the boxfish-like robot needs fewer control parameters and it can accomplish both clockwise and counter-clockwise rolling behaviours, which will benefit upper controllers in real applications.

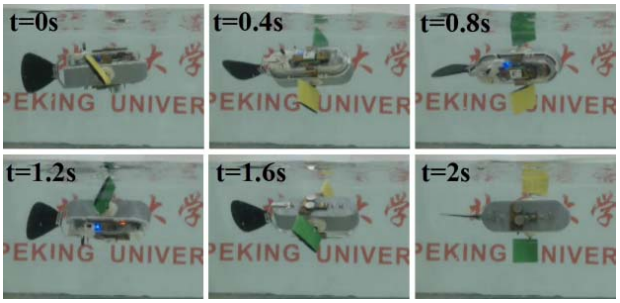

Figure 8. Snapshots of the robot rolling with the control drives $d_v = 2.5$, $d_y = 0$, $d_p = 0$ and $d_r = 0.25\pi$

4.2. Attitude Control

In this section, attitude control experiments were conducted to validate the effectiveness and stability of the closed-loop CPG controller. In the experiments, the real attitude measurements of the robot are recorded online by the onboard IMU. Specifically, three kinds of experiments are performed: yaw angle control, roll angle control and the united attitude control of yaw and pitch.

4.2.1. Yaw Angle Control

When the robot operates in an unknown environment, the ability of yaw angle control (i.e., orientation control) makes great sense. Therefore, under the control of the closed-loop CPG controller, the stabilizing performance of the yaw angle control of the robot is first evaluated. Figure 9 illustrates the tracking results when the reference yaw angles are constant. We can see that the estimated angles can rapidly converge around the corresponding reference angles. Because the offset of the beating tail is used to regulate the yaw angle of the robot, the experimental results exhibit as an oscillatory form. We did 20 similar experiments to evaluate the stability of the closed-loop controller. All of them are successful to track the reference yaw angles and the average tracking error is about 5° for arbitrary reference yaw angles. Note that the experimental errors are remarkably smaller than that of the yawing control results in [28], where a dynamic model is built and implemented on a robotic fish.

In addition, to evaluate the response speed of the closed-loop CPG controller, the following reference yaw angle is adopted

$$d_y^*(t) = \frac{\pi}{6}\sin(2\pi\omega t) \tag{24}$$

whose frequency ω was set to 0.05 Hz. As shown in Figure 10, the controller is able to track the sine-based $d_y^*(t)$ basically, although time lags exist between the real yaw angle and reference yaw angle.

4.2.2. Roll Angle Control

Possibly due to the contradiction between the complicated motion and the limited control parameters as well as the simple mechanical configuration of the robot, roll control of robotic fish has rarely been tackled in the literature. An example can be found in Barbera et al. [29], where a simplified Newton-Euler linearly dynamic model was built to track a designed roll angle. In this work, however, we use the model-free CPG-based controller to stabilize the roll angle of the robot. We did 15 experiments to evaluate the stability of the rolling control and all of them

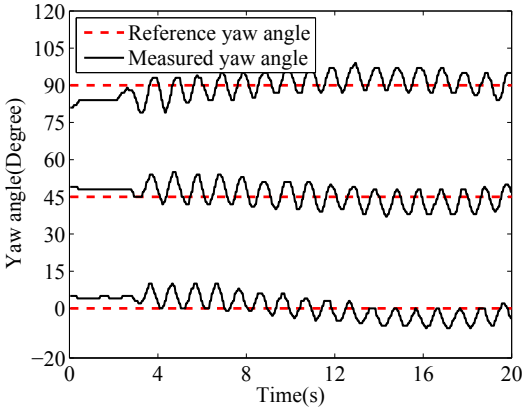

Figure 9. Data collected during three yaw stabilization experiments,respectively at $d_y^* = 0°$, $d_y^* = 45°$ and $d_y^* = 90°$. The other input drives $d_v = 3$, $d_p = 0$ and the roll angle of the robot, d_r^*, is not controlled, that is, u_1, u_2, v_1 and v_2 are all equal to 0.

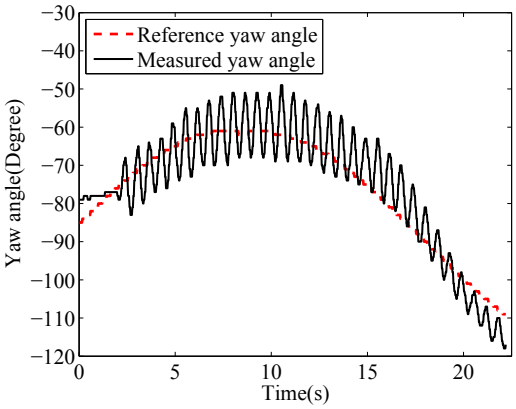

Figure 10. The robot is tracking a sine-based $d_y^*(t)$. The other input drives $d_v = 3$, $d_p = 0$ and the roll angle of the robot, d_r^*, is not controlled, that is, u_1, u_2, v_1 and v_2 are all equal to 0.

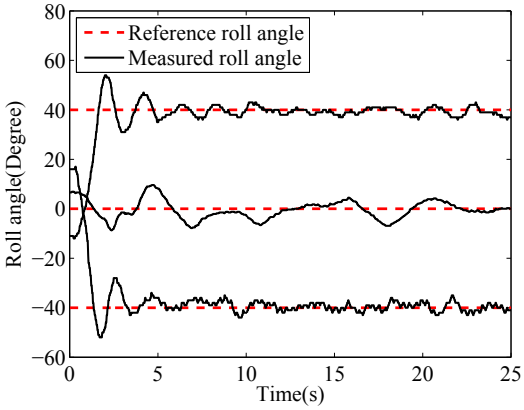

Figure 11. The measured roll angles in three different roll control experiments where $d_r^* = -40°$, $d_r^* = 0°$ and $d_y^* = 40°$. The other input drives $d_v = 3$, $d_p = 0$ and the yaw angle of the robot, d_y^*, is not controlled, that is, v_3 is equal to 0.

are successful at tracking the reference roll angles with the average tracking error being about 3° for the reference roll angle range $-45° \leq d_r^* \leq 45°$. Figure 11 shows three rolling control results. We can see that although the proposed closed-loop CPG controller is model-free, the results are comparable with those in [29]. Moreover, unlike the interferences between rolling and yawing control in [29], rolling and yawing are decoupled in our control strategy, which will be validated in the next section.

4.2.3. United Control of Yaw and Roll Angles

By coordinating the motions of the pectoral and caudal fins, the robot can stabilize its yaw angle and roll angle at the same time. This section validates the effectiveness of the united control of yaw and roll angles of the robotic fish by using the proposed closed-loop CPG network. To the best of our knowledge, the united yaw and roll angle control of the robotic fish within the CPG control framework is realized here for the first time. For instance, the expected roll angle and yaw angle are defined as follows:

$$d_r^*(t) = \frac{7}{36}\pi \sin(2\pi\omega t) \tag{25}$$

$$d_y^*(t) = -\frac{\pi}{2} \tag{26}$$

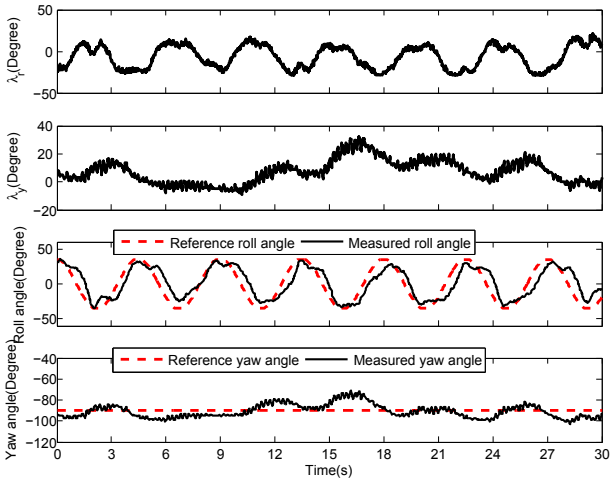

Figure 12. The united control of yaw and roll angle of the robot by using the closed-loop CPG controller. The other input parameters are $d_v = 4$ and $d_p = 0$.

whose frequency ω was set to 0.22 Hz. As illustrated in Fig. 12, we can see that both the roll and yaw angles of the robotic fish are tracked under the control of the closed-loop CPG model. Specifically, the real roll angle exhibits a time lag as compared with the reference roll angle d_r^*. This is probably due to the existence of a response time in mechanical systems. Furthermore, the tracked results are not as exact as the results where the yawing and the rolling are tracked alone, as shown in Figure 9 and 11. The reason for this may be that the forces produced by the paired fins and the caudal fin are interacting with each other to a certain degree, which disturbs the close-loop CPG controller.

5. Conclusion and Future Work

In this paper, two kinds of CPG-based controllers, that is, the open-loop controller and closed-loop controller, are proposed successively for the boxfish-like robot. Based on the CPG-based controller, the robot is able to flexibly switch between multiple 3D swimming patterns and control its attitude of yaw and roll exactly while swimming.

Specifically, a two-layer open-loop CPG model with only four control parameters is first proposed in an explicit expression to generate all the typical swimming patterns of the robotic fish, such as swimming forwards/backwards, turning left/right, swimming upwards/downwards and rolling clockwise/counter-clockwise. To ensure the intrinsic stability of swimming patterns, the stability of this open-loop CPG controller is proved strictly. Nevertheless, a more important contribution of this paper, besides improving the open-loop CPG network in theory and experiments, is that we introduce a novel closed-loop CPG-based control method, which can automatically stabilize the attitude of the robot without interference from the upper control centre. The CPG-based attitude control of the yaw and roll of the robotic fish is designed and realized for the first time in the literature. Systematical experiments have been conducted to validate

the effectiveness and stability of the proposed open-loop and closed-loop CPG controllers.

It is anticipated that the proposed CPG-based locomotion controllers will be a useful tool to optimize the swimming patterns of robots and assist them in accomplishing practical underwater tasks. For this purpose, interesting topics like control parameter optimization, adaptive learning and path planning are worthy of further investigation.

6. Acknowledgment

The authors would like to gratefully thank Yongnan Jia for the technical support in design and construction of the robot. They also thank Fayang Cao and Chengchao Shang for doing amounts of experiments.

7. References

[1] R. Pfeifer, M. Lungarella, and F. Iida, "Self-organization, embodiment, and biologically inspired robotics", *Science*, vol. 318, pp. 1088-1093, 2007.

[2] Ijspeert, A. J., "Central pattern generators for locomotion control in animals and robots: A review", *Neural Networks*, vol. 21, pp. 642-653, 2008.

[3] F. Delcomyn, "Neural basis of rhythmic behavior in animals", *Science*, vol. 210, pp. 492-498, 1980.

[4] S. Grillner, "Control of locomotion in bipeds, tetrapods, and fish", Handbook of Physiology II American Physiology Society, Bethesda, MD, 1981.

[5] G. Endo, J. Morimoto, T. Matsubara, J. Nakanishi, and G. Cheng, "Learning CPG-based biped locomotion with a policy gradient method: Application to a humanoid robot", *The International Journal of Robotics Research*, vol. 27, pp. 213-228, 2008.

[6] C. Liu, Q. Chen, and D. Wang, "CPG-inspired workspace trajectory generation and adaptive locomotion control for quadruped robots", *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 41, pp. 867-880, 2011.

[7] Carla M. A. Pinto, "Stability of quadruped robots' trajectories subjected to discrete perturbations", *Nonlinear Dynamics*, vol. 70, pp. 2089-2094, 2012.

[8] A. Crespi, D. Lachat, A. Pasquier, and A. J. Ijspeert, "Controlling swimming and crawling in a fish robot using a central pattern generator", *Autonomous Robots*, vol. 25, pp. 3-13, 2008.

[9] J. Yu, R. Ding, Q. Yang, and J. Tan, "Amphibious pattern design of a robotic fish with wheel-propeller-fin mechanisms", *Journal of Field Robotics*, vol. 3, pp. 702-716, 2013.

[10] W. Wang, J. Guo, Z. Wang, and G. Xie, "Neural controller for swimming modes and gait transition on an ostraciiform fish robot", In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM2013)*, pp. 1564-1569, 2013.

[11] A. J. Ijspeert, A. Crespi, D. Ryczko, and J. M. Cabelguen, "From swimming to walking with a salamander robot driven by a spinal cord model", *Science*, vol. 315, pp. 1416-1420, 2007.

[12] A. Kamimura, H. Kurokawa, S.and Tomita K. Yoshida, E.and Murata, and S. Kokaji, "Automatic locomotion design and experiments for a modular robotic system", *IEEE/ASME Transactions on Mechatronics*, vol. 10, pp. 314-325, 2005.

[13] J. Yu, R. Ding, Q. Yang, M. Tan, W. Wang, and J. Zhang, "On a bio-inspired amphibious robot capable of multimodal motion", *IEEE/ASME Transactions on Mechatronics*, vol. 17, pp. 847-856, 2012.

[14] Y. Hu, W. Zhao, L. Wang, and Y. Jia, "Neural-based control of modular robotic fish with multiple propulsors", In *47th IEEE Conference on Decision and Control (CDC 2008)*, pp. 5232-5237, 2008.

[15] C. Wang, G. Xie, L. Wang, and M. Cao, "CPG-based locomotion control of a robotic fish: Using linear oscillators and reducing control parameters via PSO", *International Journal of Innovative Computing Information and Control*, vol. 7, pp. 4237-4249, 2011.

[16] K. Seo, S. J. Chung, and JJE Slotine, "CPG-based control of a turtle-like underwater vehicle", *Autonomous Robots*, vol. 28, pp. 247-269, 2010.

[17] Santos-Victor J. Gay, S. and A. Ijspeert, "Learning robot gait stability using neural networks as sensory feedback function for central pattern generators", in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2013)*, pp. 194-201, 2013.

[18] I. B. Jeong, C. S. Park, K. I. Na, S. Han, and J. H. Kim, "Particle swarm optimization-based central patter generator for robotic fish locomotion", in *IEEE Congress on Evolutionary Computation (CEC2011)*, pp. 152-157, 2011.

[19] Y. Hu, W. Zhao, and L. Wang, "Vision-based target tracking and collision avoidance for two autonomous robotic fish", *IEEE Transactions on Industrial Electronics*, vol. 56, pp. 1401-1410, 2009.

[20] *, "Dynamic modeling of an ostraciiform robotic fish based on angle of attack theory", in *2014 IEEE World Congress on Computational Intelligence (IEEE WCCI)*, accepted, 2014.

[21] M. S. Gordon, J. R. Hove, P. W. Webb, and D. Weihs, "Boxfishes as unusually well-controlled autonomous underwater vehicles", *Physiological and Biochemical Zoology*, vol. 73, pp. 663-671, 2000.

[22] J. R. Hove, L. M. O??Bryan, M. S. Gordon, P. W. Webb, and D. Weihs, "Boxfishes (Teleostei: Ostraciidae) as a model system for fishes swimming with many fins: Kinematics", *Journal of Experimental Biology*, vol. 204, pp. 1459-1471, 2001.

[23] I.K. Bartol, M. Gharib, P.W. Webb, D. Weihs, and M.S. Gordon, "Body-induced vortical flows: a common mechanism for self-corrective trimming control in boxfishes", *Journal of Experimental Biology*, vol. 208, pp. 327-344, 2005.

[24] J. A. Fax and R. M. Murray, "Information flow and cooperative control of vehicle formations", *IEEE Transactions on Automatic Control*, vol. 49, pp. 1465-1476, 2004.

[25] R. Ding, J. Yu, Q. Yang, and M. Tan, "Dynamic modelling of a CPG-controlled amphibious biomimetic swimming robot", *International Journal of Advanced Robotic Systems*, vol. 10, pp. 1-11, 2013.

[26] W. Hu, Y.and Zhao and L. Xie, G.and Wang, "Development and target following of vision-based autonomous robotic fish", *Robotica*, vol. 27, pp. 1075-1089, 2009.

[27] F. E. Fish, S. A. Bostic, A. J. Nicastro, and J. T. Beneski, "Death roll of the alligator: mechanics of twist feeding in water", *Journal of Experimental Biology*, vol. 210, pp. 2811-2818, 2007.

[28] K. A. Morgansen, B. I. Triplett, and D. J. Klein, "Geometric methods for modeling and control of free-swimming fin-actuated underwater vehicles", *IEEE Transactions on Robotics*, vol. 23, pp. 1184-1199, 2007.

[29] G. Barbera, L. Pi, and Y. Deng, "Attitude control for a pectoral fin actuated bio-inspired robotic fish", in *IEEE International Conference on Robotics and Automation (ICRA2011)*, pp. 526-531, 2011.