Mathematical Biosciences and Engineering

MBE, 19 (1): 738–758.
DOI: 10.3934/mbe.2022033
Received: 10 August 2021
Accepted: 08 November 2021
Published: 19 November 2021

http://www.aimspress.com/journal/MBE

Research article

Reinforcement learning-based optimization of locomotion controller using multiple coupled CPG oscillators for elongated undulating fin propulsion

Van Dong Nguyen[1], Dinh Quoc Vo[3], Van Tu Duong[1,2,3], Huy Hung Nguyen[3,4,*] and Tan Tien Nguyen[1,2,3,*]

[1] Faculty of Mechanical Engineering, Ho Chi Minh City University of Technology (HCMUT), 268 Ly Thuong Kiet, District 10, Ho Chi Minh City, Vietnam
[2] Vietnam National University Ho Chi Minh City, Linh Trung Ward, Thu Duc District, Ho Chi Minh City, Vietnam
[3] National Key Laboratory of Digital Control and System Engineering (DCSELab), HCMUT, 268 Ly Thuong Kiet, District 10, Ho Chi Minh City, Vietnam
[4] Faculty of Electronics and Telecommunication, Saigon University, Vietnam

* **Correspondence:** Email: nhhung@dcselab.edu.vn, nttien@hcmut.edu.vn.

Abstract: This article proposes a locomotion controller inspired by black Knifefish for undulating elongated fin robot. The proposed controller is built by a modified CPG network using sixteen coupled Hopf oscillators with the feedback of the angle of each fin-ray. The convergence rate of the modified CPG network is optimized by a reinforcement learning algorithm. By employing the proposed controller, the undulating elongated fin robot can realize swimming pattern transformations naturally. Additionally, the proposed controller enables the configuration of the swimming pattern parameters known as the amplitude envelope, the oscillatory frequency to perform various swimming patterns. The implementation processing of the reinforcement learning-based optimization is discussed. The simulation and experimental results show the capability and effectiveness of the proposed controller through the performance of several swimming patterns in the varying oscillatory frequency and the amplitude envelope of each fin-ray.

Keywords: reinforcement learning; undulating fin; biomimetic robot; Hopf oscillator

1. Introduction

The oceans account for more than three-quarters of the earth, and the ocean seafloor has the considerable potential to recover the great benefit that may benefit humanity. Therefore, ocean exploration is recognized as an essential field in ocean science [1]. Ocean exploration identifies two primary devices called remotely operated underwater vehicles (ROV), an autonomous underwater vehicle (AUV). Almost all conventional AUVs adopt water pumps, air-jet engines, or single propellers as the propulsion system [2] that cause a loud noise affecting the organism's life on the seabed. In addition, the topological structure of conventional AUVs has been recognized that are not able to perform maneuverability and stability [3]. The propeller can also be stuck by sediment and seaweed in the operation of AUVs on the seafloor [4–6]. A bionic underwater robot equipped with a biomimetic fin mechanism is well-suited for ocean exploration [7] to overcome the drawbacks mentioned above. Many approaches studied about bio-fish robots concerned the diversity of fish species [6–30]. These studies pointed out that many significant factors affect the hydrodynamic of bio-fish robots. One such factor is the swimming pattern that enables the bio-fish robots to perform complex operations such as turning, swaying, twisting, and curving. Several studies utilized a sinusoidal-based kinematic equation to generate the undulating oscillatory motion for the bio-fish robots [31–36] to address this research field. This locomotion control strategy can provide various swimming patterns by predefining the amplitude envelope, oscillatory frequency, and phase lag regarded as the kinematic parameters of the sinusoidal generator. However, this does not feature a flexible transition swimming pattern, as well as it does not enable tuning online kinematic parameters to adapt to the environmental changes [8,31].

To achieve efficient locomotion, earlier studies have been proposed a central pattern generator (CPG) based locomotion controllers for widely application fields [11,27,39–45]. In terms of governing the locomotion of bio-fish robots, the authors early synthesized a locomotion controller using a Proportional-Integral-Derivative (PID) controller integrated with CPG for a prototype of the fish robot in 3D [24]. In 2008, Wang et al. [19] employed a modified Matsuoka oscillator to build a CPG-based locomotion controller for a prototype of an undulating fins propulsion system with ten fin-rays. Simulation and experimental results showed that the variable model of the weight matrix is consistent with the thrust propulsion generated by the prototype of the propulsion system. In 2011, a CPG-based controller of the proposed propulsion system was integrated with the rotary position sensors to improve the locomotion of undulating fin more flexibly [28]. In addition, this study also introduced two control levels with a high-level controller for commanding operation and a low-level controller for driving actuators. In 2012, Zhou et al. [39] developed a manta ray robot with two wide flexible pectoral fins. This robot used a CPG model to achieve rhythmic biomimetic movement. Simulation and experimental results showed that the yaw angle is stabilized, but the response time is slow. In 2014, Chunlin Zhou et al. [29] adopted a genetic algorithm to achieve a better conversion efficiency to optimize the CPG-based controller for the fish robot according to the thrust generation. To validate the CPG-based control approach for undulating fins propulsion, in 2015, Michael Sfakiotakis et al. [32] performed the CPG denominations using the conversion of single amplitude parameters and simultaneous transformation. The authors adopted a CPG model to achieve the undulating motion pattern for finding the critical factor which affects the propulsion. A fish robot prototype using the CPG model for swimming motion was inspired by cuttlefish [22]. This study presented the effect of the various kinematic parameters of the undulating fin and the validity of a fluid drag model used to estimate the generated thrust. Another study [8] dealt with the utilization of CPG for undulating biological fins with six degrees of freedom

to perform the replicated fish-like swimming robot by changing the parameters of the CPG model. Various parameters of the CPG model can be adjustable to generate undulating motion to produce the propulsion force, such as amplitude envelope, oscillatory frequency, and swimming patterns. Thus, Yong Cao et al. [46] predefined the undulation frequency and the undulation amplitude as constant parameters while governing CPG neuron output's phase angle to achieve various swimming patterns. It can be concluded that these earlier studies related to CPG have been successfully applied for locomotion control of biomimetic robots. However, most of these researches rely on trial-and-error data fitting to adjust a control parameter of the CPG model called convergence rate. Increasing the convergence rate can reduce the processing time for achieving the limit cycle; however, this can raise an oscillatory error defined as the difference between the intrinsic amplitude of CPG and the maximum amplitude envelope of the CPG's output. This issue is still a challenge for researchers with the lack of optimization for the convergence rate of CPG.

In terms of parameter optimization, several studies used particle swarm optimization (PSO) algorithm to seek the CPG parameters in order to minimize the difference between the desired oscillatory waveform and the generated output of the CPG [47], to reduce the control parameters [48] and to refine the feature parameters of the CPG [49]. In comparison to a genetic algorithm (GA), PSO is similar to GA as so to search for optimal solutions through iterations of a population, but PSO proved to be faster computed and easier implemented than GA [50]. However, PSO exhibits that it is susceptible to trap in local minima [51]. Reinforcement learning (RL) is known as an alternative strategy for optimization that has been applied recently in various applications such as robotic control, transportation, and energy supervision [52–58]. RL generates a series of sequence actions to obtain the maximum numerical rewards in the interaction with environments. RL can be categorized as model-based RL method, which attempts to model the environment known as Markov Decision Process (MDP) [59], and model-free RL method, which does not require the explicit of the environment. One such model-free RL method is Q-Learning which is recognized as a well-suited method for optimization to trade-off the performance time and the effectiveness [55,60–62]. According to these above studies, Q-learning can be feasible to implement in real-time on programmable devices. For the application of biomimetic robots, Y. Nakamura et al. [63,64] utilized a reinforcement learning model for the CPG-based motion controller, namely CPG-actor-critic, to learn the selection of motion patterns for biped robots. An actor observes the state of the biped robot and outputs a parameter of the motion controller. Then the motion controller with the selected parameter produces the control signal.

The above-aforementioned studies regarding CPG-based bio-fish robots have not conducted optimization for the convergence rate. Inspired from the studies concerned with applying RL for CPG, this paper proposes a reinforcement learning-based optimization of locomotion controller using CPG network for an elongated undulating fin. The elongated undulating fin comprises sixteen oblique fin-rays interconnected with a membrane known as a flexible surface that is controlled by the proposed CPG-based locomotion controller coupled with sixteen neural oscillators to generate the locomotor corresponding to sixteen fin-rays of the elongated undulating fin. The advantages of this control method in comparison to the sinusoidal kinematic equation are discussed. This paper, differentiating from the previous studies, utilizes a Q-learning with discrete state/action to optimize the convergence rate of the CPG controller. The actor observes the undulating signal of the CPG-based locomotion controller and outputs a value of the convergence rate. The locomotion controller with the chosen convergence rate produces the control signal. The proposed controller is promised that it can be implemented on a microcontroller due to its simplicity. The simulation and experimental results are

carried out to evaluate the performance and effectiveness of the proposed control method.

2. Elongated undulating fin

The elongated undulating fin comprises sixteen oblique adjacent fin-rays interconnected with a flexible membrane. Each fin-ray is driven by an RC servo motor that enables the fin-ray to sway around a rotary joint fixed to a supporting frame illustrated in Figure 1. The elongated undulating fin is built with a length of 775 mm, a width of 90 mm, and a height of 290 mm.

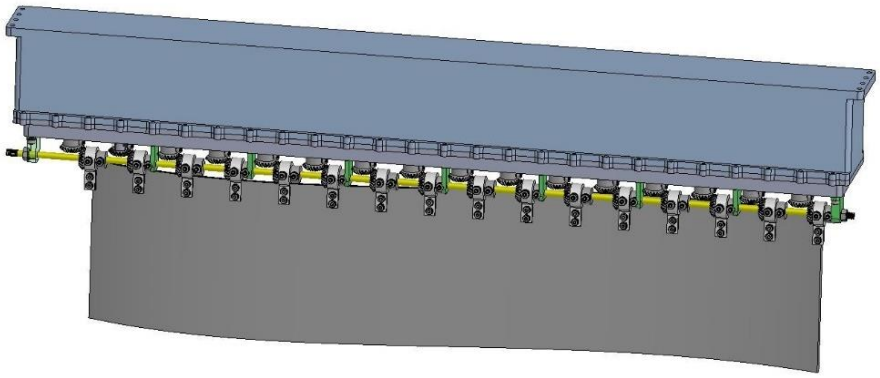

Figure 1. Overview of elongated undulating fin.

Accordingly, each fin-ray reacts as a shaker bar with a limited angle, and the phase difference between two adjacent fin-rays is regarded as a phase lag angle. By changing one of the kinematic parameters such as amplitude envelope, oscillatory frequency, and swimming pattern, the magnitude of the propulsive force can be adjustable. To perform forwarding/reversing motion, the elongated undulating fin might change the sign of the phase lag angle. Additionally, to avoid the counter-torque of the elongated undulating fin, the number of oscillation wavelengths should be an even number. Traditionally, the sinusoidal oscillatory equation employed for generating the undulating motion of bio-fish robots is given by [23]:

$$\theta^i(t) = \theta^i_{max}\sin(2\pi f t + \phi_i) \tag{1}$$

where θ^i is the sway angle for i^{th} fin-ray; θ^i_{max} is the maximum sway angle for each fin-ray; f is the oscillatory frequency; ϕ_i is the phase lag angle of each fin-ray.

The utilization of the sinusoidal oscillatory equation-based gait control can successfully generate the bio-fish robots' locomotion motion. However, high-performance aquatic locomotion requires swimming adaptability to environments of the bio-fish robots. The sinusoidal oscillatory equation might hardly achieve this feature because the abruptly changing of amplitude envelope, oscillatory frequency, or swimming pattern might cause the discontinuity and instability of the undulating motion. We simulated the sinusoidal swimming locomotion to illustrate this situation in Figure 2.

We make an abrupt change in the amplitude envelope referring to Figure 2a and the oscillatory frequency referring to Figure 2b at an arbitrary time t^*. It can be easy to observe that the output of the sinusoidal generator is discontinued at the arbitrary time t^*.

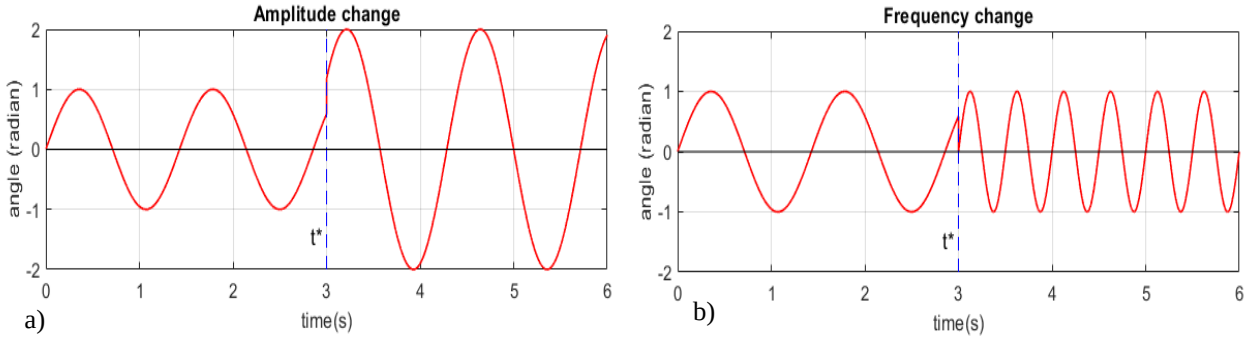

Figure 2. Output of sinusoidal equation in abrupt change of amplitude and frequency.

3. Reinforcement learning-based optimization for CPG locomotion controller

3.1. Hopf oscillator

The CPG is a circuit network of oscillators that can produce rhythmic patterns for biomimetic robots. Several kinds of oscillators such as Van der Pol, Wilson-Cowan, Kuramoto, Matsuoka, Amplitude-Controlled Phase, Rowat-Selverston, and Hopf have been applied successfully to generate the walking/swimming/flapping gaits of biomimetic robots. However, it seems that the Van der Pol oscillator is better for producing an electrocardiogram signal; most of the above oscillators are well-suited for generating the rhythmic movement of arm/legged robots with two moving phases. Therefore, this research employs a Hopf oscillator, which can realize a nonharmonic sine waveform, to construct a modified CPG for generating the rhythm locomotion of the elongated undulating fin. A typical structure of the Hopf oscillator is shown in Figure 3.

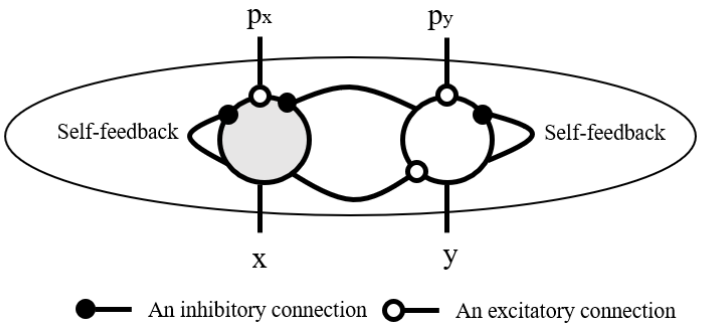

Figure 3. Typical structure of Hopf oscillator.

The dynamic of the Hopf oscillator is expressed by the following differential equation:

$$\begin{aligned}\dot{u}(t) &= k\left(A^2 - u^2(t) - v^2(t)\right)u(t) - 2\pi f v(t)\\ \dot{v}(t) &= k\left(A^2 - u^2(t) - v^2(t)\right)v(t) + 2\pi f u(t)\end{aligned} \tag{2}$$

where u, v are time-variant state variables of the oscillator; A is the intrinsic amplitude; f is the intrinsic frequency; k is the convergence rate to the limit cycle ($k > 0$).

For comparison to the traditionally sinusoidal generator, a simulation of a single Hopf oscillator is conducted in the same manner illustrated in Figure 4.

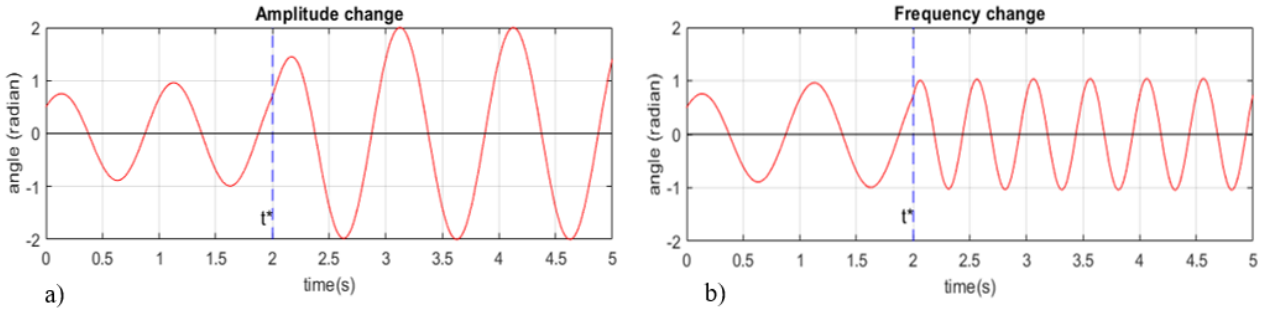

Figure 4. Output of Hopf oscillator in abrupt change of intrinsic amplitude and frequency.

It can be observed from Figure 4 that the oscillatory output generated by the Hopf oscillator can introduce the smooth transition when the abrupt changes of both intrinsic amplitude and oscillatory frequency are conducted at the arbitrary time t^*. In addition, the Hopf oscillator of Eq 2 also features the quick convergence to the limit cycle. Even though starting from different arbitrary initial points, the output of the Hopf oscillator converges to a stable limit cycle with the intrinsic amplitude A. The convergence rate can be tuned by adjusting k of the Eq 2. The Hopf oscillator output converges to the limit cycle more rapidly with an increasing k, regardless of the abrupt changes of intrinsic amplitude and intrinsic frequency. A simulation result of the Hopf oscillator with eight different initial points for each scenario is illustrated in Figure 5a). It can also be seen from Figure 5b) that the output of the Hopf oscillator can converge to the limit cycle rapidly, approximately 2 seconds.

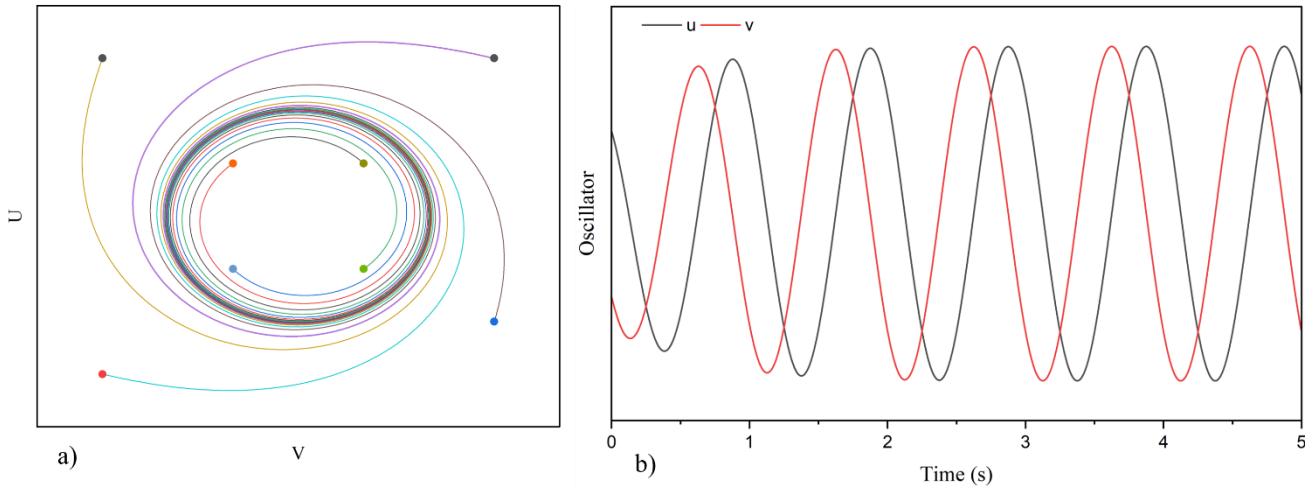

Figure 5. Convergence to limit cycle of Hopf oscillator.

3.2. Modified CPG with multi coupled Hopf oscillators

In both invertebrate and vertebrate organisms, there are several topological couplings between the joints to allow the muscle to work perfectly, which represent the role of stimulus and inhibition. The actual CPGs of animal brains are complicated networks that have abundant neurons. In order to

replicate the CPG for controlling biomimetic robots, it is necessary to simplify the coupling connections and categorize them into four main topological structures: chain coupling, radial coupling, ring coupling, and fully connected coupling [31]. Each topological structure of the coupling connection has appropriate property corresponding to the biological characteristic of each species. For instance, the chain coupling is mainly applied to stimulate the locomotion of swimmers, whereas the fully connected coupling is usually applied for rhythm generation of legged robots because all legs must be coupled to perform smooth motion against the environmental change.

The biological structure of the elongated undulating fin features a series of fin-rays. The abnormal movement of each arbitrary fin-ray due to environmental influences affects only its adjacent fin-ray. To generate the undulate motion for the elongated undulating fin, this research constructs the chain coupling of sixteen oscillators with bi-directional perturbation depicted in Figure 6. Each oscillator is employed to stimulate each fin-ray. The reflection of each fin-ray to its adjacent fin-ray is performed through the bi-directional perturbation. The pair of intrinsic amplitude and intrinsic frequency is an independent entity for each oscillator of the modified CPG network.

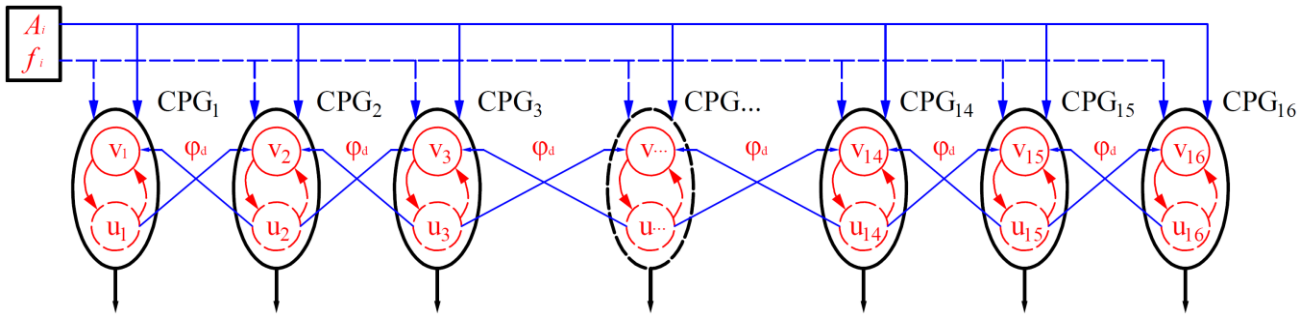

Figure 6. Structure of modified CPG network with chain coupling of sixteen oscillators in bi-directional perturbation.

In the modified CPG network, there are two terminal oscillators that are not affected by the adjacent oscillators. However, without loss of generality, the nonlinear function illustrating the modified CPG network shown in Figure 6 is given as follow:

$$\dot{X}_i = F(X_i) + P_i = \begin{bmatrix} k(A_i^2 - u_i^2 - v_i^2)u_i - 2\pi f v_i \\ k(A_i^2 - u_i^2 - v_i^2)v_i + 2\pi f u_i \end{bmatrix} + \begin{bmatrix} p_{u,i} \\ p_{v,i} \end{bmatrix} \tag{3}$$

where $X_i \triangleq [u_i\ v_i]^T$ is the state vector of the i-th oscillator; $F(X_i)$ represents a nonlinear function; $P_i \triangleq [p_{u,i}\ \ p_{v,i}]^T$ is a perturbation vector.

To clarify Eq 3 for the terminal oscillators, it is necessary to consider the coupling connection of three adjacent oscillators as shown in Figure 7:

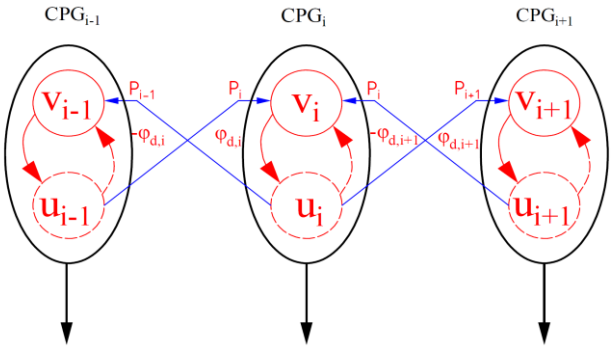

Figure 7. Coupling connection of three adjacent oscillator.

For the first oscillator ($i = 1$), there is only perturbation from the second oscillator ($i + 1$); thus, the perturbation of the first oscillator is given by:

$$P_1 = \begin{bmatrix} 0 \\ \beta(v_2 cos\varphi_d - u_2 sin\varphi_d) \end{bmatrix} \tag{4}$$

where β is the coupling strength; φ_d is the phase lag angle of two adjacent oscillators.
In the same manner, the sixteenth oscillator is only affected by the perturbation from the fifteenth oscillators:

$$P_{16} = \begin{bmatrix} 0 \\ \beta(u_{15} sin\varphi_d + v_{15} cos\varphi_d) \end{bmatrix} \tag{5}$$

For i-th oscillators, the perturbation vector is given by the following:

$$P_i = \begin{bmatrix} 0 \\ \beta(u_{i-1} sin\varphi_d + v_{i-1} cos\varphi_d - u_{i+1} sin\varphi_d + v_{i+1} cos\varphi_d) \end{bmatrix} \tag{6}$$

Corresponding to various intrinsic amplitudes A_i, the modified CPG network can provide different swimming patterns for the elongated undulating fin, it thus can produce different propulsive forces.

3.3. Reinforcement learning-based optimization

It should be noted from Eq 3 that the convergence rate k is chosen by a trial-and-error method to obtain the limit cycle as quickly as possible. A large value of k can reduce the transient-state time, which is defined as a period from the beginning to the moment that the output of 16th CPG starts the first cycle; meanwhile, it might cause the oscillatory error of the modified CPG network output. Thus, it is necessary to optimal this significant parameter. On the other hand, Q-learning is a part of reinforcement learning that is the value-based learning algorithm to obtain a higher reward for each episode. This paper employs a Q-learning with discrete action because it costs a duration for the CPG to generate the oscillatory output corresponding to each chosen action before taking the following step. Furthermore, this algorithm does not require high computational time, enabling the onboard implementation. Accordingly, the state variables $s_t \in S$ (with S is the state variable compact set) are the oscillatory error s_t^1 and the transient-state time s_t^2 with $s_t^1 \in S^1, s_t^2 \in S^2,$ and $S^1, S^2 \subset S$. The shifting of the convergence rate is chosen as the action variable $a_t \in A$. The interaction of the agent and the environment of RL is shown in Figure 8.

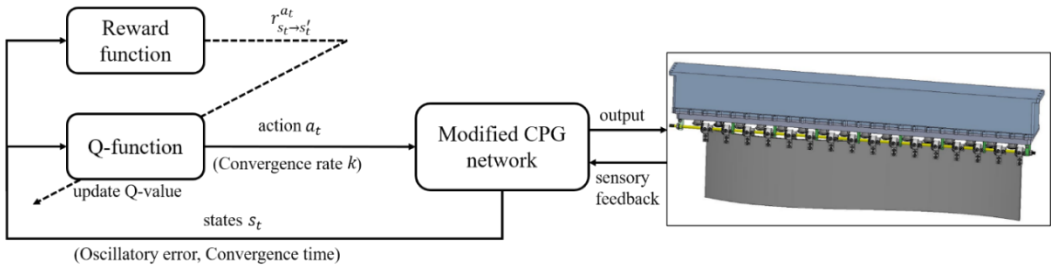

Figure 8. Interaction of agent and environment.

The reward function is proposed to trade-off between the transient-state time and the oscillatory error that the mathematical proposed reward function is given the following:

$$r^{a_t}_{s_t \to s'_t} = L^u r_1(s_t^1) + L^l r_2(s_t^2) \tag{7}$$

In Eq 7, s'_t is the next state variable, and L^u, L^l are reward constants set arbitrarily such that the condition holds $L^u \gg L^l$ to emphasis that the minimization of the oscillatory error is more significant than that of the transient-state time. Thus, L^u, L^l are respectively set to 100 and 10 in this case. The reward subfunctions $r_i(s_t^i)$ with $i = 1, 2$ are given by the following:

$$r_i(s_t^i) = \begin{cases} R_{max} & |s_t^i| < \min(S^i) \\ R_{min} & |s_t^i| = \min(S^i) \\ 0 & |s_t^i| > \min(S^i) \end{cases} \tag{8}$$

where R_{max}, R_{min} are the maximum reward and the minimum reward set to 1 and 0.1, respectively.

As well, the terminal state s_T known as the condition for complete an episode holds the constraint $s_T \coloneqq \{s_t \in S | \delta \triangleq (L^u|s_t^1| + L^l s_t^2) \leq \min_{\delta}(\Delta_e)\}$ with Δ_e is the compact set of δ of each episode.

The Q-value (action-value) function is updated by the simple Temporal Difference (TD) method:

$$Q_t(s_t, a_t) = Q_{t-1}(s_t, a_t) + \alpha \left(r^{a_t}_{s_t \to s'_t} + \gamma \max_{a'_t} Q_{t-1}(s'_t, a'_t) - Q_{t-1}(s_t, a_t) \right) \tag{9}$$

where α is the learning rate ($0 \leq \alpha < 1$); γ is the discount factor ($0 \leq \gamma < 1$); a'_t is the next action variable; $Q_{t-1}(\blacksquare)$ denotes the current Q-value; $Q_t(\blacksquare)$ denotes the new Q-value;

The next policy $\pi'(a_t, s_t)$ is implemented by ε-Greedy strategy which is given by:

$$\pi'(s_t, a_t) = \begin{cases} \operatorname{argmax}_{a_t} Q_{t-1}(s_t, a_t) & q < 1 - \varepsilon \\ \operatorname{rand}(Q_{t-1}(s_t, a_t)) & \text{otherwise} \end{cases} \tag{10}$$

where q is the uniform random number.

The optimal convergence rate can be determined by the optimal action-value:

$$a_t^* = \operatorname*{argmax}_{a_t} Q(s_t, a_t) \tag{11}$$

The pseudo-code of the Q-learning optimization for the convergence rate is illustrated in Table 1. The impact of the transient-state time and the oscillatory error on the convergence rate is depicted in Figure 9a. As well, the distribution of the Q-value on the state variable and the action variable is illustrated in Figure 9b).

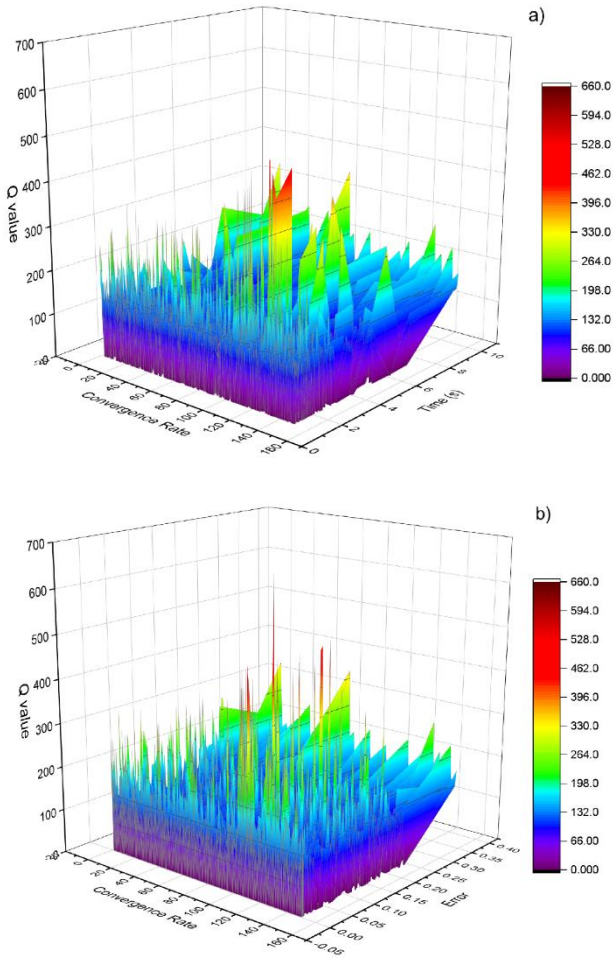

Figure 9. a) Impact of transient-state time and oscillatory error on the convergence rate. b) Distribution of Q-value on state variable and action variable.

Table 1. Pseudo-code of the Q-learning optimization

Algorithm: Q-learning based optimization of the convergence rate

1. Initialize $\alpha, \gamma, \varepsilon$
2. Initialize $Q_{t-1}(s_t, a_t) = [0], s_t = rand(S)$, and episode n
3. Repeat for each step of the episode:
   4. Choose $a_t = \underset{a_t}{\operatorname{argmax}}\, Q(s_t, a_t)$ if uniform random number $< 1 - \varepsilon$
   5. Choose $a_t = \operatorname{rand}(Q(s_t, a_t)$ if otherwise
   6. Take the action a_t (traveling the convergence rate k to the modified CPG network)
   7. Observe s_t', $r_{s_t \to s_t'}^{a_t}$ (perceiving the oscillatory error and the transient-state time, calculating the reward value by Eqs 7,8.
   8. Update Q-value by Eq 9.
   9. The next state is assigned as the next state $(s_t \leftarrow s_t')$
   10. Until the current state is the terminal state $(s_t \equiv s_T)$
11. Take the optimal action $a_t^* = \underset{a_t}{\operatorname{argmax}}\, Q(s_t, a_t)$

According to the implementation of the Q-learning based optimization for the convergence rate with the discount factor $\gamma = 0.75$, the learning rate $\alpha = 0.95$, the ε – greedy of 0.7, and the episode number $n = 2000$, the optimal Q-value achieved the approximate value $Q^*(s_t, a_t) = 658279$ with respect to the optimal action of $a_t^* = 96$, which is used for simulation/experimental studies in the next section.

4. Results and discussion

In this research, the simulation study of the modified CPG network is conducted through MATLAB with the aim that is to evaluate the flexible transition gait of the elongated undulating fin relevant to the swimming pattern, intrinsic amplitude, oscillatory frequency, and the number of waveforms. The swimming patterns utilized in this research are illustrated in Figure 10. The simulation results also demonstrate the affection of the convergence rate on the transient-state time and the oscillator error of the modified CPG network.

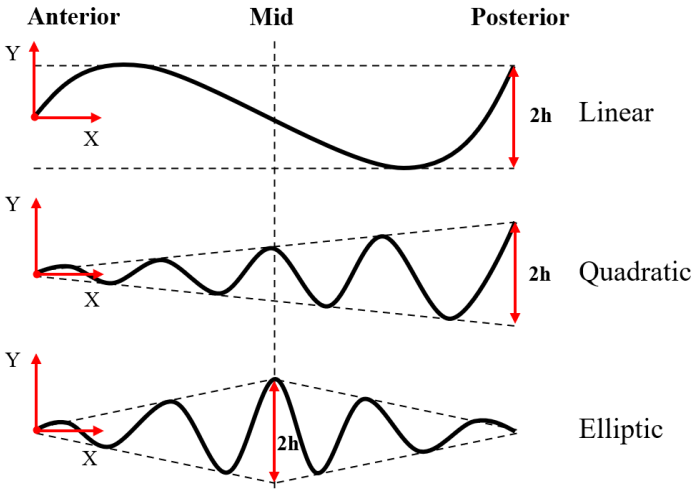

Figure 10. Swimming patterns of elongated undulating fin propulsion.

4.1. Characteristic of convergence rate

The modified CPG parameters are given for this study as $A_i = 1(\text{with } i = 1 \div 16)$, $f = 1, \varphi_d = -\pi/3$, $\beta = 0.8$ to allow the fin-rays to perform the cuttlefish-like swimming pattern. Figure 12 depicts the output of a single oscillator with k chosen arbitrarily around the optimal value of 96 for comparison. As can be seen, with $k = 86$, the transient-state time is nearly obtained as 1.45 seconds, whereas that of the case $k = 96$ is approximately value of 1.41 seconds compared to the case of $k = 106$ as 1.36 seconds. It is easy to note that the larger amount of k will result in the reducing of the transient-state time due to the modified CPG output converged to the limit cycle. Nevertheless, increasing the convergence rate k will cause the larger oscillatory error of the modified CPG output illustrated in Figure 11, which might affect the performance of the actuators powered for fin-rays. Therefore, the oscillator error is recognized as the more significant factor than the transient-state time.

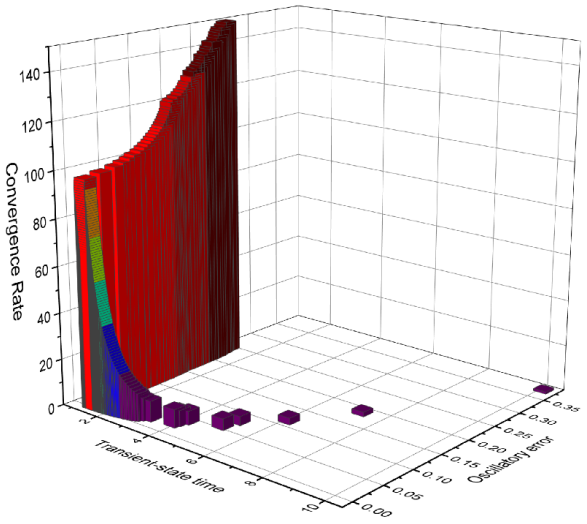

Figure 11. The relative convergence rate concerning transient-state time and oscillatory error.

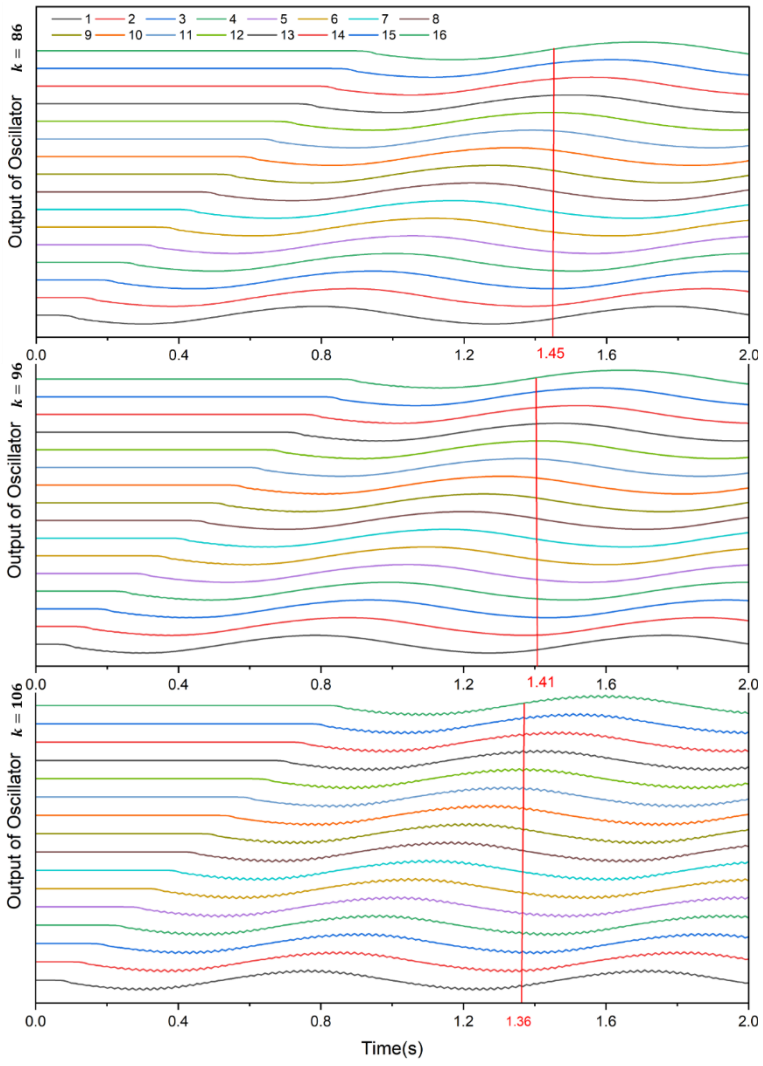

Figure 12. The output of a single oscillator with $k = 86, k = 96, k = 106$.

4.2. Transition gait

This simulation study aims to clarify several aspects as smooth accelerating/decelerating with no jerk by changing the oscillatory frequency f, flexible transition swimming pattern by changing the intrinsic amplitude A_i, the transition between forwarding and backward swimming by changing the phase lag angle φ_d, and transition of waveform number. It can be seen from Figure 13a, the modified CPG network initially generates a nonharmonic swimming pattern with the linear waveform to mimic the cuttlefish-like gait for 2.5 seconds. Afterward, the oscillatory frequency gradually increased from 1 Hz to 2 Hz, and the oscillatory output became faster to enable the elongated undulating fin to accelerate. During the time 5–7.5 seconds, the elongated undulating fin performs the quadratic swimming pattern. After 7.5 seconds, the swimming pattern is forced to change into the ecliptic waveform. In Figure 13b, the elongated undulating fin performs the waveform with the elliptical waveform to mimic the stingray-like swimming pattern for the first 5 seconds with the phase lag angle of $\varphi_d = -\pi/3$ for each fin-ray. At the time of 5 seconds, the swimming pattern abruptly change the phase lag angle into $\varphi_d = \pi/3$ to enable the elongated undulating fin to perform backward swimming. It can be seen that the modified CPG network can perform better smooth transition gait than the kinematic sinusoidal generator. During the time 5–20 seconds, the elongated undulating fin performs the backward swimming. Afterward, the phase lag angle is again changed into $\varphi_d = -\pi/3$ to force the elongated undulating fin to perform the forward swimming. This study scenario also reveals that a lower convergence rate endows the shorter transient-state time when the phase lag angle is changed to switch the swimming direction (see Figure 14).

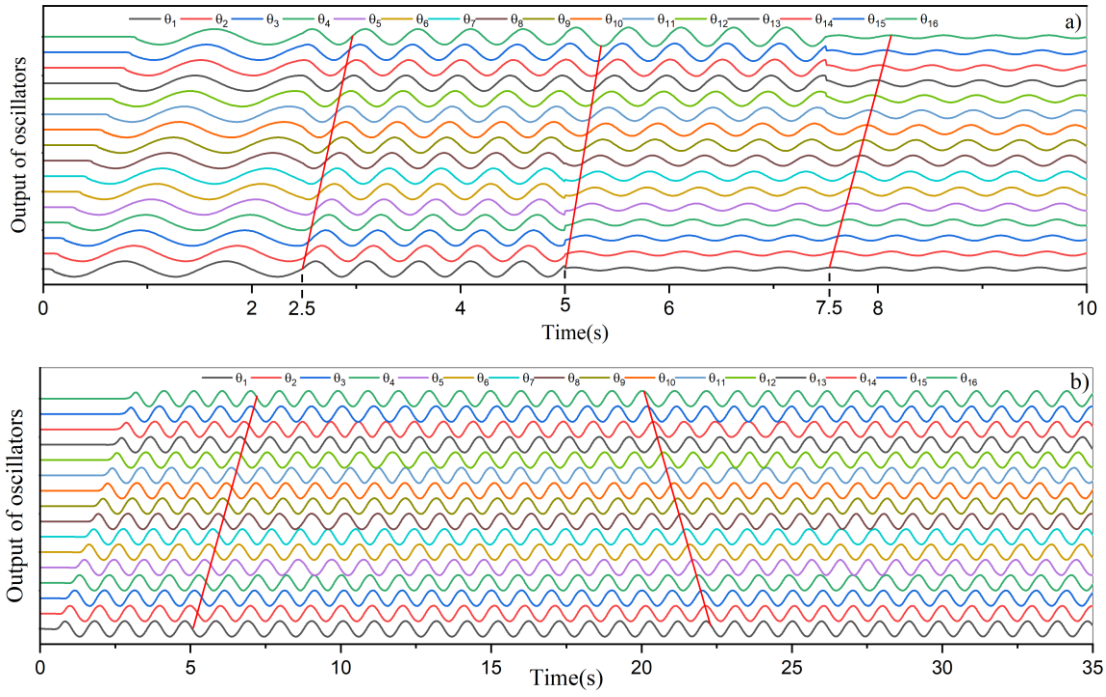

Figure 13. (a) Output of sixteen oscillators with changes of swimming pattern, oscillatory frequency, and waveform number – (b) Output of sixteen oscillators with changes of phase lag angle enabling for reverse swimming direction.

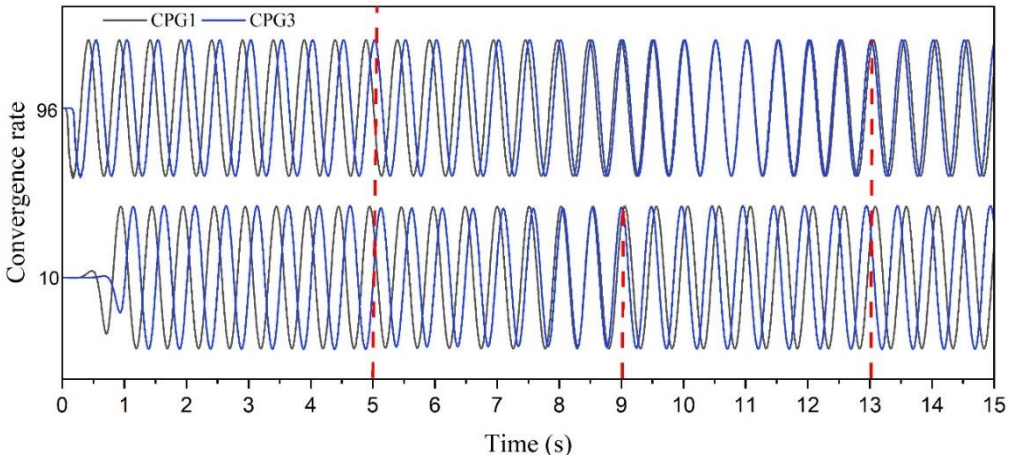

Figure 14. Relation of transient-state time with respect to convergence rate.

Figure 14 shows the CPG's outputs in the cases with the convergence rate of $k = 96$ and $k = 10$. For the sake of distinguishing, we take the undulating signals of the first and third CPGs. During the time 0–5 seconds, the CPGs perform the undulating waveform with the phase lag angle of $\varphi_d = -\pi/3$. It can be recognized by the fact that the output phase of 1st CPG leads that of 3rd CPG. At the time of 5 seconds, the CPGs are commanded to change into the phase lag angle of $\varphi_d = \pi/3$. It can be seen from the lower side of Figure 14 that the CPG's outputs take 4 seconds to change the swimming direction in the case with the convergence rate of 10. The reverse swimming direction can be recognized by the fact that the output phase of 1st CPG lags that of 3rd CPG. However, the CPG's outputs take 8 seconds to change the swimming direction in the case with the convergence rate of 96, as shown in the upper side of Figure 14. This implies that the convergence rate should be switched into a smaller sufficient value before the CPGs are commanded to the swimming direction.

4.3. Experimental study

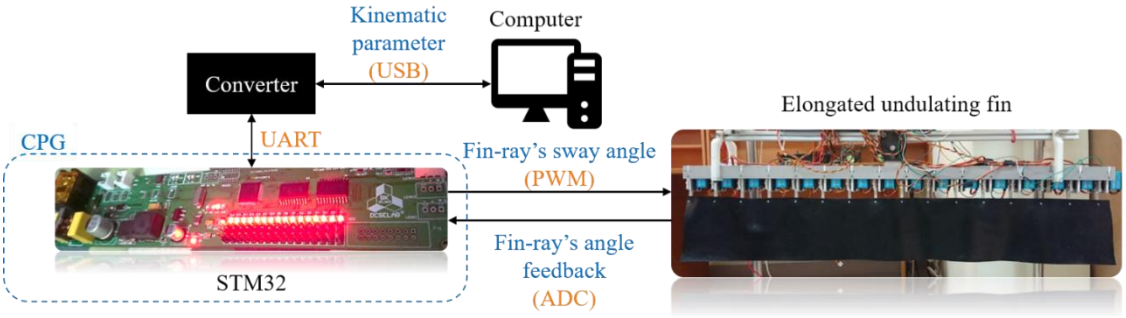

Figure 15. Experimental configuration.

A configuration of the experimental setup depicted in Figure 15 is employed to validate the applicability of the modified CPG network. A customized STM32F103RET6 microcontroller-based board is utilized to implement the modified CPG network to drive sixteen fin-rays through the 50Hz PWM signals. In order to perceive the fin-rays angle, all RC servos are modified to sense their rotary angle for the perturbation of the modified CPG network. A computer is utilized to operate the swimming parameter as well as to compute the Q-learning algorithm. The elongated undulating fin is

validated in a pre-test stage without water immersion. The kinematic parameters of the modified CPG network are given as $f = 1$ Hz, $k = 10$, $\varphi_d = -\pi/3$, $\beta = 0.8$, and the sampling time of 0.01 seconds. To match the required amplitude envelope, the output of oscillators is calculated by the following:

$$\theta_i = G_i u_i \tag{12}$$

where u_i is the output of each oscillator neural; G_i is the maximum sway angle of each fin-ray which is determined by $G_i = \arcsin(Y_i)/L$ with Y_i defined as the amplitude envelope of each fin-ray along to laterally, and L is the length of fin-ray, for this case $L = 150$ mm.

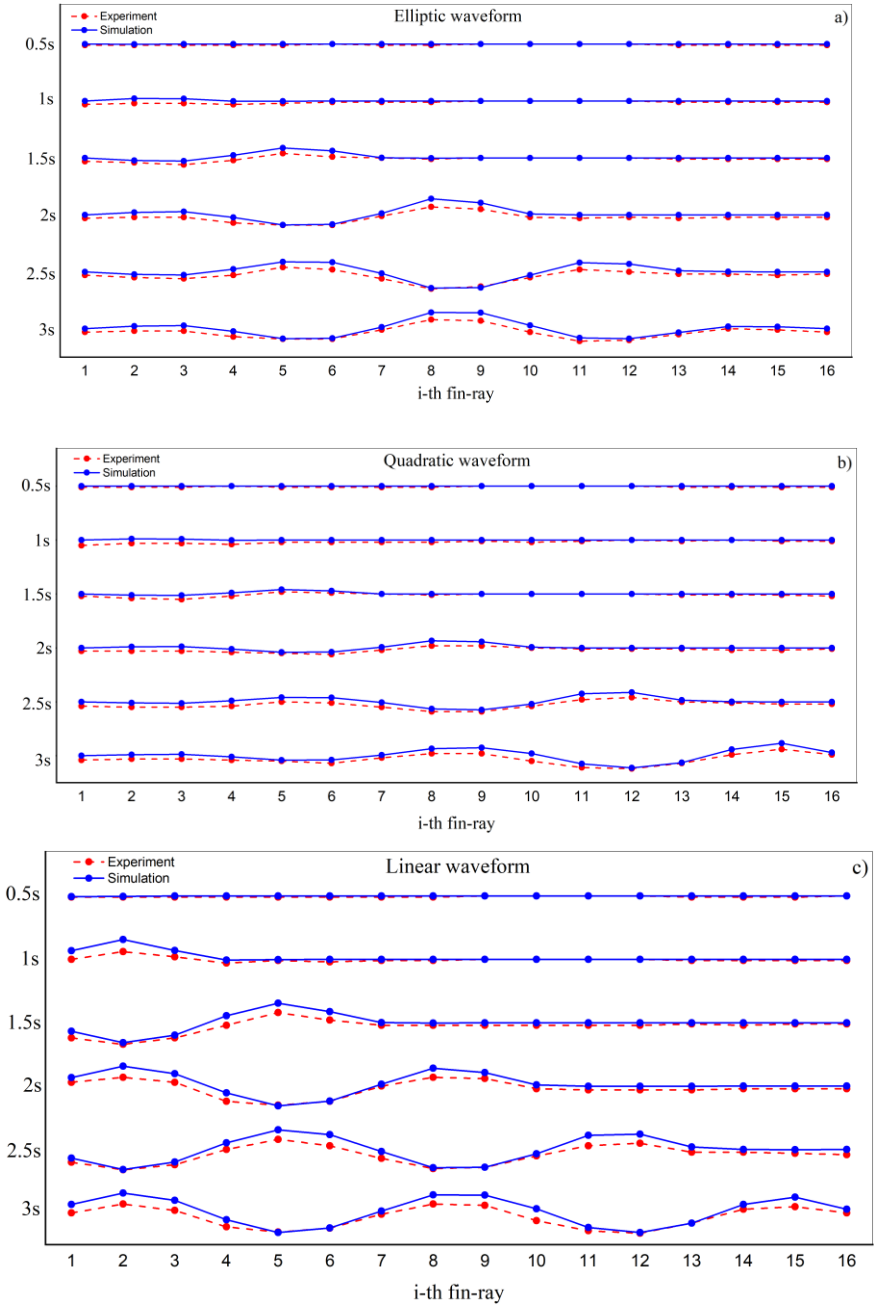

Figure 16. Experimental results of various swimming patterns.

The elongated undulating fin performs the elliptic waveform depicted in Figure 16a with the amplitude envelope Y_i for each fin-ray as {0, 5.7, 11.43, 17.14, 22.85, 28.57, 34.28, 40, 40, 34.28,

28.57, 22.85,17.14, 11.43, 5.7, 0} mm. Figure 16b shows the quadratic waveform of the elongated undulating fin with the amplitude envelope Y_i chosen as {0, 2.57, 5.33, 8, 10.67, 13.33, 16, 18.67, 21.33, 24, 26.67, 29.33, 32, 34.57, 37.33, 40} mm. The linear waveform with the constant amplitude envelope of 40 mm is illustrated in Figure 16c. The experimental data is denoted in a dashed-dot line, whereas the simulation result is denoted in a solid-dot line. As can be seen from Figure 16, the sway angles of sixteen fin-rays are gradually formed during the period 0.5–3 seconds. Throughout the formation stages of all fin-rays' oscillation, the amplitude envelope of the elongated undulating fin in the case of the experiment is smaller than that of the case of the simulation. This might be because of the limitation of the actuators' response.

5. Conclusions

This paper has presented the modified CPG network for generating the rhythm for the elongated undulating fin with sixteen fin-rays to mimic the fish's swimming patterns. Accordingly, the modified CPG network is composed by chain coupling sixteen oscillators with bidirectional perturbation because each fin-ray is only affected by its two adjacent oscillators. Both simulation and experimental results show that the modified CPG network seems to be very promising to perform the rhythm for a fish robot. It allows changing the kinematic parameters abruptly with no jerk of oscillation. Additionally, this paper has also investigated the intrinsic parameter of the CPG known as the convergence rate, which has not been considered before, usually using the trial-and-error method for this issue. The simulation results have revealed that the large convergence rate can reduce the transient-state time; however, it might cause the oscillator error worse. Therefore, the tunning of the convergence rate is to trade-off between the transient-state time and the oscillatory error. To deal with this issue, the Q-learning algorithm is appropriate to find the optimal convergence rate. To obtain smooth oscillation avoiding damage to the RC servo motor, the reward function of the Q-learning is defined with more significant oscillatory error than the transient-state time. The optimal convergence rate found by the Q-learning can provide the short transient-state time and the appropriate oscillatory error in the simulation/experimental results with the abrupt change of kinematic parameters such as amplitude envelope, oscillatory frequency, and waveform number. Especially, we have found that the transient-state time is longer in the case of using the large convergence rate when the phase lag angle is changed into the opposite value for reverse swimming. However, a change of the convergence rate while the limit cycle of the CPG is obtained does not affect the CPG output. Thus, this might raise a piece-wise switching function to change the convergence rate according to the swimming operation. Consequently, the convergence rate should be changed from the optimal value into a smaller appropriate value before the phase lag angle is changed to switch forward swimming into backward swimming and vice versa. Afterward, the convergence rate is again changed into the optimal value to obtain the short transient-state time.

From the perspective of science, this paper has only provided the experimental results in the pre-test stage with no water immersion. This is due to the impact of the COVID-19 epidemic, which terminated all of our laboratory activities at the research facility. The widespread impact and severity of the pandemic show no sign of ending. Therefore, this paper has admitted to lack series of experimental results with the elongated undulating fin submerged into a water tank. For further potential research direction, the kinematic parameters are required to trade-off for optimization of the energy consumption and the generated thrust force. A model-based reinforcement learning which tries to model the operation environment of the fish robot, is also interest to conduct in the future.

Conflict of interest

The authors declare there is no conflict of interests.

Acknowledgments

This research is supported by DCSELAB and funded by Vietnam National University Ho Chi Minh City (VNU-HCM) under grant number TX2021-20b-01. We acknowledge the support of time and facilities from Ho Chi Minh City University of Technology (HCMUT), VNU-HCM for this study.

References

1. J. Yuh, Design and Control of Autonomous Underwater Robots: A Survey, *Auton. Robot.,* **8** (2000), 7–24. doi: 10.1023/A:1008984701078.
2. K. H. Low, Maneuvering of biomimetic fish by integrating a bouyancy body with modular undulating fins, *Int. J. Humanoid Robot.*, **4** (2007), 671–695. doi: 10.1142/S0219843607001217.
3. C. Ren, X. Zhi, Y. Pu, F. Zhang, A multi-scale UAV image matching method applied to large-scale landslide reconstruction, *Math. Biosci. Eng.,* **18** (2021), 2274–2287. doi: 10.3934/MBE.2021115.
4. C. I. Sprague, O. Ozkahraman, A. Munafo, R. Marlow, A. Phillips, P. Ogren, Improving the Modularity of AUV Control Systems using Behaviour Trees, *AUV 2018 - 2018 IEEE/OES Auton. Underw. Veh. Work. Proc.*, Nov. 2018, doi: 10.1109/AUV.2018.8729810.
5. G. Ferri, A. Munafo, K. D. LePage, An Autonomous Underwater Vehicle Data-Driven Control Strategy for Target Tracking, *IEEE J. Ocean. Eng.*, **43** (2018), 323–343. doi: 10.1109/JOE.2018.2797558.
6. G. Salavasidis, A. Munafò, C. A. Harris, T. Prampart, R. Templeton, M. Smart, et al., Terrain-aided navigation for long-endurance and deep-rated autonomous underwater vehicles, *J. F. Robot.*, **36** (2019), 447–474. doi: 10.1002/ROB.21832.
7. W. Zhao, Y. Hu, L. Wang, Construction and Central Pattern Generator-Based Control of a Flipper-Actuated Turtle-Like Underwater Robot, *Adv. Robot.*, **23** (2009), 19–43. doi: 10.1163/156855308X392663.
8. C. Zhou, K. H. Low, Kinematic modeling framework for biomimetic undulatory fin motion based on coupled nonlinear oscillators, in *2010 IEEE/RSJ Int. Conf. Intel. Robots Syst.*, 2010, 934–939. doi: 10.1109/IROS.2010.5651162.
9. J. Yu, K. Wang, M. Tan, J. Zhang, Design and control of an embedded vision guided robotic fish with multiple control surfaces, *Sci. World J.*, **2014** (2014), 631296. doi: 10.1155/2014/631296.
10. A. J. Ijspeert, A. Crespi, Online trajectory generation in an amphibious snake robot using a lamprey-like central pattern generator model, *Proc. - IEEE Int. Conf. Robot. Autom.*, (2007), 262–268. doi: 10.1109/ROBOT.2007.363797.
11. D. Korkmaz, G. Ozmen Koca, G. Li, C. Bal, M. Ay, Z. H. Akpolat, Locomotion control of a biomimetic robotic fish based on closed loop sensory feedback CPG model, *J. Mar. Eng. Technol.*, **20** (2021), 125–137. doi: 10.1080/20464177.2019.1638703.
12. J.-K. Ryu, N. Chong, B.-J. You, H. Christensen, Locomotion of snake-like robots using adaptive neural oscillators, *Intell. Serv. Robot.*, **3** (2009), 1–10. doi: 10.1007/s11370-009-0049-4.
13. M. Ikeda, K. Watanabe, I. Nagai, Propulsion movement control using CPG for a Manta robot, in

The 6th Int. Conf. Soft Comput. Intel. Syst., and The 13th Int. Sympo. on Adv. Intel. Syst., 2012, 755–758. doi: 10.1109/SCIS-ISIS.2012.6505174.

14. L. Shang, S. Wang, M. Tan, Fuzzy Logic PID Based Control Design for a Biomimetic Underwater Vehicle with Two Undulating Long-fins, in *India Conf. (INDICON) 2015 Annual IEEE*, 2015, 1–6.
15. J. Zhang, Multimodal swimming control of a robotic fish with pectoral fins using a CPG network, *Chinese Sci. Bull.*, **57** (2012), 1209–1216.
16. K. Inoue, S. Ma, C. Jin, Neural oscillator network-based controller for meandering locomotion of snake-like robots, in *IEEE Int. Conf. Robot. Autom., 2004. Proc.. ICRA '04. 2004*, **5** (2004), 5064–5069. doi: 10.1109/ROBOT.2004.1302520.
17. C. Zhou, Modeling and control of swimming gaits for fish-like robots using coupled nonlinear oscillators, Nanyang Technological University, 2012.
18. V. D. Nguyen, D. K. Phan, C. A. T. Pham, D. H. Kim, V. T. Dinh, T. T. Nguyen, Study on Determining the Number of Fin-Rays of a Gymnotiform Undulating Fin Robot, *Lect. Notes Electr. Eng.*, **465** (2018), 745–752. doi: 10.1007/978-3-319-69814-4_72.
19. X. Dong, S. Wang, Z. Cao, M. Tan, CPG Based Motion Control for an Underwater Thruster with Undulating Long-Fin, *IFAC Proc. Vol.*, **41** (2008), 5433–5438. doi: 10.3182/20080706-5-KR-1001.00916.
20. A. Crespi, D. Lachat, A. Pasquier, A. J. Ijspeert, Controlling swimming and crawling in a fish robot using a central pattern generator, *Auton. Robots*, **25** (2008), 3–13. doi: 10.1007/s10514-007-9071-6.
21. M. Sfakiotakis, A. Manolis, N. Spyridakis, J. Fasoulas, M. Arapis, Development and Experimental Evaluation of an Undulatory Fin Prototype, in *Proceedings of the RAAD 2013 22nd Int. Workshop on Robot. Alpe-Adria-Danube Region*, 2013, no. May 2014, 1–8.
22. M. Sfakiotakis, R. Gliva, M. Mountoufaris, Steering-plane motion control for an underwater robot with a pair of undulatory fin propulsors, in *2016 24th Mediterranean Conf. Control Autom. (MED)*, 2016, 496–503, doi: 10.1109/MED.2016.7535989.
23. V. H. Nguyen, V. D. Nguyen, V. T. Duong, H. H. Nguyen, T. T. Nguyen, Experimental Study on Kinematic Parameter and Undulating Pattern Influencing Thrust Performance of Biomimetic Underwater Undulating Driven Propulsor, *Int. J. Mech. Mechatronics Eng.*, **20** (2020), 7.
24. W. Zhao, J. Yu, Y. Fang, L. Wang, Development of Multi-mode Biomimetic Robotic Fish Based on Central Pattern Generator, *2006 IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2006, doi: 10.1109/IROS.2006.281800.
25. X. Wu, S. Ma, CPG-based control of serpentine locomotion of a snake-like robot, *Mechatronics*, **20** (2010), 326–334. doi: 10.1016/j.mechatronics.2010.01.006.
26. R. Gliva, M. Mountoufaris, N. Spyridakis, M. Sfakiotakis, Development of a Bio-Inspired Underwater Robot Prototype with Undulatory Fin Propulsion, in *9th Int. Conf. on New Horiz. Ind. Bus. Edu. (NHIBE'15)*, 2015, 1–6.
27. Z. Lu, S. Ma, B. Li, Y. Wang, 3D Locomotion of a Snake-like Robot Controlled by Cyclic Inhibitory CPG Model, *2006 IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2006, doi: 10.1109/IROS.2006.281801.
28. M. Wang, J. Yu, M. Tan, G. Zhang, A CPG-based sensory feedback control method for robotic fish locomotion, in *Proceedings of the 30th Chinese Control Conf.*, 2011, 4115–4120.
29. C. Zhou, K. H. Low, On-line Optimization of Biomimetic Undulatory Swimming by an

Experiment-based Approach, *J. Bionic. Eng.*, **11** (2014), 213–225. doi: 10.1016/S1672-6529(14)60042-1.
30. M. Sfakiotakis, J. Fasoulas, R. Gliva, A. Yannakoudakis, Model-based fin ray joint tracking control for undulatory fin mechanisms, *Int. Congr. Ultra Mod. Telecommun. Control Syst. Work.*, **2016** (2016), 158–165. doi: 10.1109/ICUMT.2015.7382421.
31. C. Zhou and K. H. Low, Design and locomotion control of a biomimetic underwater vehicle with fin propulsion, *IEEE/ASME Trans. Mechatronics*, **17** (2012), 25–35. doi: 10.1109/TMECH.2011.2175004.
32. M. Sfakiotakis, J. Fasoulas, M. M. Kavoussanos, M. Arapis, Experimental investigation and propulsion control for a bio-inspired robotic undulatory fin, *Robotica*, **33** (2015), 1062–1084. doi: 10.1017/S0263574714002926.
33. P. M. Özturan, A. Bozanta, B. Basarir-Ozel, E. Akar, M. Coşkun, A roadmap for an integrated university information system based on connectivity issues: Case of Turkey, *Int. J. Manag. Sci. Inf. Technol.*, **17** (2015), 1–23. doi: 10.14313/JAMRIS.
34. K. H. Low, A. Willy, Biomimetic motion planning of an undulating robotic fish fin, *JVC/Journal Vib. Control*, **12** (2006), 1337–1359. doi: 10.1177/1077546306070597.
35. R. Ruiz-Torres, O. M. Curet, G. V. Lauder, M. A. Maciver, Erratum: Kinematics of the ribbon fin in hovering and swimming of the electric ghost knifefish (Journal of Experimental Biology 216, (823-834)), *J. Exp. Biol.*, **217** (2014), 3765–3766. doi: 10.1242/jeb.113670.
36. K. H. Low, Modelling and parametric study of modular undulating fin rays for fish robots, *Mech. Mach. Theory*, **44** (2009), 615–632. doi: 10.1016/j.mechmachtheory.2008.11.009.
37. I. English, H. Liu, O. M. Curet, Robotic device shows lack of momentum enhancement for gymnotiform swimmers, *Bioinspir. Biomim.,* **14** (2019), 024001. doi: 10.1088/1748-3190/aaf983.
38. I. D. Neveln, R. Bale, A. P. S. Bhalla, O. M. Curet, N. A. Patankar, M. A. MacIver, Undulating fins produce off-axis thrust and flow structures, *J. Exp. Biol.*, **217** (2014), 201–213. doi: 10.1242/jeb.091520.
39. M. Ikeda, S. Hikasa, K. Watanabe, I. Nagai, A CPG design of considering the attitude for the propulsion control of a Manta robot, in *IECON 2013 - 39th Ann. Conf. IEEE Ind. Electron. Soc.*, 2013, 6354–6358. doi: 10.1109/IECON.2013.6700181.
40. C. Liu, Q. Chen, D. Wang, CPG-inspired workspace trajectory generation and adaptive locomotion control for quadruped robots, *IEEE Trans. Syst. man, Cybern. Part B, Cybern. a Publ. IEEE Syst. Man, Cybern. Soc.*, **41** (2011), 867–880. doi: 10.1109/TSMCB.2010.2097589.
41. C. M. A. Pinto, D. Rocha, C. P. Santos, Hexapod robots: New CPG model for generation of trajectories, *J. Numer. Anal. Ind. Appl. Math.*, **7** (2012), 15–26.
42. T. Wang, W. Guo, M. Li, F. Zha, L. Sun, CPG Control for Biped Hopping Robot in Unpredictable Environment, *J. Bionic Eng.*, **9** (2012), 29–38. doi: 10.1016/S1672-6529(11)60094-2.
43. S. Inagaki, H. Yuasa, T. Arai, CPG model for autonomous decentralized multi-legged robot system—generation and transition of oscillation patterns and dynamics of oscillators, *Rob. Auton. Syst.*, **44** (2003), 171–179. doi: 10.1016/S0921-8890(03)00067-8.
44. M. Mokhtari, M. Taghizadeh, M. Mazare, Hybrid Adaptive Robust Control Based on CPG and ZMP for a Lower Limb Exoskeleton, *Robotica*, **39** (2021), 181–199. doi: 10.1017/S0263574720000260.
45. X. Wu, L. Teng, W. Chen, G. Ren, Y. Jin, H. Li, CPGs with continuous adjustment of phase difference for locomotion control, *Int. J. Adv. Robot. Syst.*, **10** (2013), 1–13. doi: 10.5772/56490.

46. Y. Cao, Y. Lu, Y. Cai, S. Bi, G. Pan, CPG-fuzzy-based control of a cownose-ray-like fish robot, *Ind. Robot Int. J. Robot. Res. Appl.*, **46** (2019), 779–791. doi: 10.1108/IR-02-2019-0029.
47. I. B. Jeong, C. S. Park, K. I. Na, S. Han, J. H. Kim, Particle swarm optimization-based central patter generator for robotic fish locomotion, *2011 IEEE Congr. Evol. Comput. CEC 2011*, (2011), 152–157, doi: 10.1109/CEC.2011.5949612.
48. M. C. Chen Wang, G. Xie, L. Wang, CPG-based locomotion control of a robotic fish: Using linear oscillators and reducing control parameters via PSO, *Int. J. Innov. Comput. Inf. Control*, **7** (2011), 4237–4249.
49. J. Yu, Z. Wu, M. Wang, M. Tan, CPG Network Optimization for a Biomimetic Robotic Fish via PSO, *IEEE Trans. Neural Networks Learn. Syst.*, **27** (2016), 1962–1968. doi: 10.1109/TNNLS.2015.2459913.
50. J. Lee, S. Lee, S. Chang, B.-H. Ahn, A Comparison of GA and PSO for Excess Return Evaluation in Stock Markets, *Lect. Notes Comput. Sci.*, **3562** (2005), 221–230. doi: 10.1007/11499305_23.
51. C. Niehaus, T. Röfer, T. Laue, Gait Optimization on a Humanoid Robot using Particle Swarm Optimization, 2007.
52. Y. Zou, T. Liu, D. Liu, F. Sun, Reinforcement learning-based real-time energy management for a hybrid tracked vehicle, *Appl. Energy*, **171** (2016), 372–382. doi: 10.1016/j.apenergy.2016.03.082.
53. T. Liu, Y. Zou, D. Liu, F. Sun, Reinforcement learning-based energy management strategy for a hybrid electric tracked vehicle, *Energies*, **8** (2015), 7243–7260. doi: 10.3390/en8077243.
54. R. C. Hsu, C. T. Liu, D. Y. Chan, A reinforcement-learning-based assisted power management with QoR provisioning for human-electric hybrid bicycle, *IEEE Trans. Ind. Electron.*, **59** (2012), 3350–3359. doi: 10.1109/TIE.2011.2141092.
55. H. Lee, C. Kang, Y. Il Park, N. Kim, S. W. Cha, Online data-driven energy management of a hybrid electric vehicle using model-based Q-learning, *IEEE Access*, **8** (2020), 84444–84454. doi: 10.1109/ACCESS.2020.2992062.
56. T. Liu, X. H, S. E. Li, D. Cao, Reinforcement Learning Optimized Look-Ahead Energy Management of a Parallel Hybrid Electric Vehicle, *IEEE/ASME Trans. Mechatronics*, **22** (2017), 1497–1507. doi: 10.1109/TMECH.2017.2707338.
57. Y. Lu, R. He, X. Chen, B. Lin, C. Yu, Energy-efficient depth-based opportunistic routing with q-learning for underwater wireless sensor networks, *Sensors (Switzerland)*, **20** (2020), 1–25. doi: 10.3390/s20041025.
58. R. Plate, C. Wakayama, Utilizing kinematics and selective sweeping in reinforcement learning-based routing algorithms for underwater networks, *Ad Hoc Networks*, **34** (2015), 105–120. doi: 10.1016/j.adhoc.2014.09.012.
59. Y. He, L. Xing, Y. Chen, W. Pedrycz, L. Wang, G. Wu, A Generic Markov Decision Process Model and Reinforcement Learning Method for Scheduling Agile Earth Observation Satellites, *IEEE Trans. Syst. Man. Cybern. Syst.*, 1–12, 2020. doi: 10.1109/tsmc.2020.3020732.
60. Z. Jin, Y. Ma, Y. Su, S. Li, X. Fu, A Q-learning-based delay-aware routing algorithm to extend the lifetime of underwater sensor networks, *Sensors (Switzerland)*, **17** (2017), 1–15. doi: 10.3390/s17071660.
61. D. Zhang, Z. H. Ye, P. C. Chen, Q. G. Wang, Intelligent event-based output feedback control with Q-learning for unmanned marine vehicle systems, *Control Eng. Pract.*, **105** (2020), 104616. doi: 10.1016/j.conengprac.2020.104616.
62. Z. Chen, B. Qin, M. Sun, Q. Sun, Q-Learning-based parameters adaptive algorithm for active

disturbance rejection control and its application to ship course control, *Neurocomputing*, **408** (2020), 51–63. doi: 10.1016/j.neucom.2019.10.060.

63. Y. Nakamura, T. Mori, S. Ishii, Natural Policy Gradient Reinforcement Learning for a CPG Control of a Biped Robot, *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, **3242** (2004), 972–981. doi: 10.1007/978-3-540-30217-9_98.
64. T. Mori, Y. Nakamura, M. A. Sato, S. Ishii, Reinforcement learning for a CPG-driven biped robot, *Proc. Natl. Conf. Artif. Intell.*, (2004), 623–630.

AIMS Press