Biomimetic Intelligence and Robotics 2 (2022) 100075

Contents lists available at ScienceDirect

Biomimetic Intelligence and Robotics

journal homepage: www.elsevier.com/locate/birob

NA-CPG: A robust and stable rhythm generator for robot motion control

Ru Tong [a,b], Changlin Qiu [a,b], Zhengxing Wu [a,b,*], Jian Wang [a,b], Min Tan [a,b], Junzhi Yu [a,c,1]

[a] State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
[b] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China
[c] Department of Advanced Manufacturing and Robotics, College of Engineering, Peking University, Beijing 100871, China

ARTICLE INFO

Keywords:
Normalized asymmetric CPGs
Rhythm generator
Motion control
Robotic fish

ABSTRACT

Central pattern generators (CPGs) have been widely applied in robot motion control for the spontaneous output of coherent periodic rhythms. However, the underlying CPG network exhibits good convergence performance only within a certain range of parameter spaces, and the coupling of oscillators affects the network output accuracy in complex topological relationships. Moreover, CPGs may diverge when parameters change drastically, and the divergence is irreversible, which is catastrophic for the control of robot motion. Therefore, normalized asymmetric CPGs (NA-CPGs) that normalize the amplitude parameters of Hopf-based CPGs and add a constraint function and a frequency regulation mechanism are proposed. NA-CPGs can realize parameter decoupling, precise amplitude output, and stable and rapid convergence, as well as asymmetric output waveforms. Thus, it can effectively cope with large parameter changes to avoid network oscillations and divergence. To optimize the parameters of the NA-CPG model, a reinforcement-learning-based online optimization method is further proposed. Meanwhile, a biomimetic robotic fish is illustrated to realize the whole optimization process. Simulations demonstrated that the designed NA-CPGs exhibit stable, secure, and accurate network outputs, and the proposed optimization method effectively improves the swimming speed and reduces the lateral swing of the multijoint robotic fish by 6.7% and 41.7%, respectively. The proposed approach provides a significant improvement in CPG research and can be widely employed in the field of robot motion control.

1. Introduction

Central pattern generators (CPGs) are originally found in animals as a neural circuit that spontaneously generates rhythmic neural activity without receiving any rhythmic inputs [1,2]. They provide rhythms for basic movements, such as breathing, digestion, swimming, and walking. More importantly, CPGs provide interesting features, including distributed control, the ability to handle redundancy, fast control loops, and the ability to modulate motion through simple control signals [3]. When these properties are converted to mathematical models, CPGs become useful building blocks to control robot motion. Through changing the simplified parameters of the CPG model, complex and variable rhythmic signals are easily generated to regulate the multimodal locomotion of robots.

Various mathematical CPG models that can generate periodic oscillation signals to meet the needs of rhythmic movement have been established based on the principles of biological CPGs. The classic CPG models can be divided into neuron-based CPG models and nonlinear oscillator-based CPG models. In particular, neuron-based CPG models such as the Matsuoka model [4] and the Kimura model [5] have clear biological significance but involve many complicated parameters and dynamic characteristic analysis, whereas nonlinear oscillator-based CPG models such as Kuramoto phase oscillators [6] and Hopf harmonic oscillators [7] have fewer parameters and mature mathematical expressions. The essence of an oscillator is a set of coupled differential equations, usually implemented by numerical computation on a microcontroller or processor. Most oscillators have a stable output waveform under the given parameters; thus, when applying CPGs in robot control, CPG parameters must be set according to the specific control scenario.

Recently, CPGs have been widely applied to legged robots, amphibious robots, underwater swimming robots, and so on. Fig. 1 illustrates CPGs applied in various types of robots [8–24]. For legged robots, CPGs are employed to generate gait control rhythms and then coordinate the movement of multiple legs. For example, Liu et al. designed a Matsuoka oscillator-based gait rhythm generator to produce the walking patterns of biped robots, the parameters of which were tuned to achieve flexible trajectories [9]. Li et al. developed a ring-type CPG model using a Hopf oscillator for a new type of hexapod robot with

* Corresponding author.
E-mail address: zhengxing.wu@ia.ac.cn (Z. Wu).

[1] Given his role as Managing Guest Editor of this journal, Junzhi Yu had no involvement in the peer-review of this article and had no access to information regarding its peer-review. Full responsibility for the editorial process for this article was delegated to Prof. Li Wen.

https://doi.org/10.1016/j.birob.2022.100075
Received 29 July 2022; Received in revised form 6 September 2022; Accepted 19 September 2022
Available online 7 October 2022

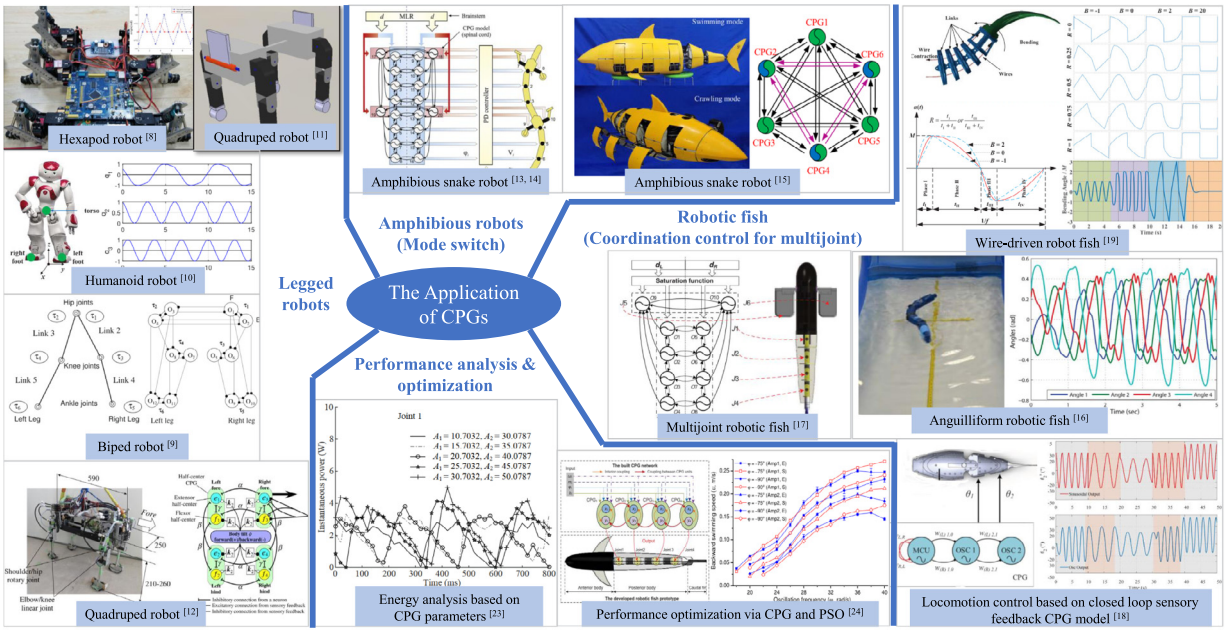

Fig. 1. Applications of CPGs in robotics.

a four-bar linkage mechanism [8]. The designed CPG network outputs stable and smooth signals with steady phase relationships, achieving smooth robot walking under the triangular gait mode. Fukui et al. verified the effectiveness and practicability of vestibular feedback to CPGs employed for the locomotion of quadruped robots [11]. As for amphibious robots, switching between gaits under different modes is achieved by CPGs with smooth transition characteristics. Ijspeert et al. proposed a CPG model for an amphibious salamander-like robot, which is composed of 20 amplitude-controlled phase oscillators and outputs the desired joint angle position [13]. Yan et al. designed a navigation and crawl underwater unmanned vehicle (NCUUV) and studied a smooth switching strategy for two motion modes based on CPGs [15]. Specifically, the CPG network has two configurations, a triangular fully symmetrical network and a hexagonal fully symmetrical network, which control the swimming and crawling of the NCUUV, respectively, and are switched according to the instructions of the upper controller. Similarly, CPGs play an important role in the coordinated flapping of multiple fins of biomimetic swimming robots [16–18]. Wu et al. realized the coordinated motion control of the multiple joints of the robotic dolphin based on CPGs [20]. Xie et al. developed improved CPGs for wire-driven robotic fish that has two special parameters, a time ratio and a shape parameter, to adjust the frequency of four flapping stages and the shape of rhythmic signals [19].

In addition, CPGs are used for performance evaluation, optimization, and redundancy/fault tolerance control of robots. For example, CPG parameters are used to analyze the energy performance of a miniature robotic fish and help optimize energy consumption [23]. Furthermore, rhythm control based on parametric CPGs can effectively improve locomotion performance based on mathematical optimization methods. Alessandro et al. adopted a gradient-free optimization method and Powell's method to improve the CPG-governed locomotion controller [14]. The feedback information can also be utilized to make the robot withstand external disturbances, while the feedforward compensator speeds up the convergence of the overall control systems [18]. Although CPGs are widely applied in robot motion control, the existing CPG models do not fully satisfy the requirements for the actual use cases of robots. First, CPG-based locomotion control relies more on empirical testing and requires a certain amount of time and effort. Second, the CPG network parameters (learning rate, coupling coefficient, etc.) greatly impact the convergence of the network and the cooperative movement of multiple joints. Third, in recent years, CPG networks have been widely studied in terms of locomotion learning and intelligent control [25,26]. During locomotion learning, CPG networks often output violent oscillations or even divergence due to large changes in parameters (i.e., amplitude and frequency), making it difficult to cope with parameter uncertainty when applied in intelligent control algorithms, such as reinforcement learning, and it is difficult to transition between parameters smoothly and safely.

In this work, to realize smooth gait transition during the modification of CPG parameters and further improve the diversity of gait control and CPG stability under the intelligent learning framework, an improved CPG network, namely normalized asymmetric CPGs (NA-CPGs), is proposed. Specifically, the improvements are concluded in three aspects: First, the Hopf-based CPG network is normalized to improve the network's tracking accuracy of the set amplitude. Second, a constraint function is added to limit the abrupt mutation of the oscillator to transition smoothly between different parameters and thus improve network stability. Finally, an adjustment parameter is added to change the network frequency to realize asymmetric rhythmic output and in turn, increase network diversity. Furthermore, to optimize CPG parameters, a reinforcement-learning-based online optimization method for a multijoint robotic fish is proposed for stable forward high-speed swimming. Simulations and experiments proved that the proposed NA-CPGs provide a stable, accurate, and adjustable rhythmic output. Meanwhile, the online optimization method for NA-CPGs achieved obvious performance improvements demonstrated by a 6.7% increase in the swimming speed and a 41.7% reduction in the lateral oscillations of the robotic fish. The proposed method offers high control accuracy and stability and can be applied in robot motion control.

The remainder of this paper is structured as follows: The normalized asymmetric CPG model is proposed in Section 2. Section 3 introduces the online optimization method based on NA-CPGs in detail. Results and analyses are discussed in Section 4. Finally, concluding remarks are provided in Section 5.

2. Normalized asymmetric CPGs

In nature, biological nerve centers can produce rhythmic signals to maintain normal activities, such as walking, swimming, and breathing. With the development of bionic engineering, CPGs capable of producing rhythmic signals have been increasingly applied to construct locomotion controllers for bionic robots.

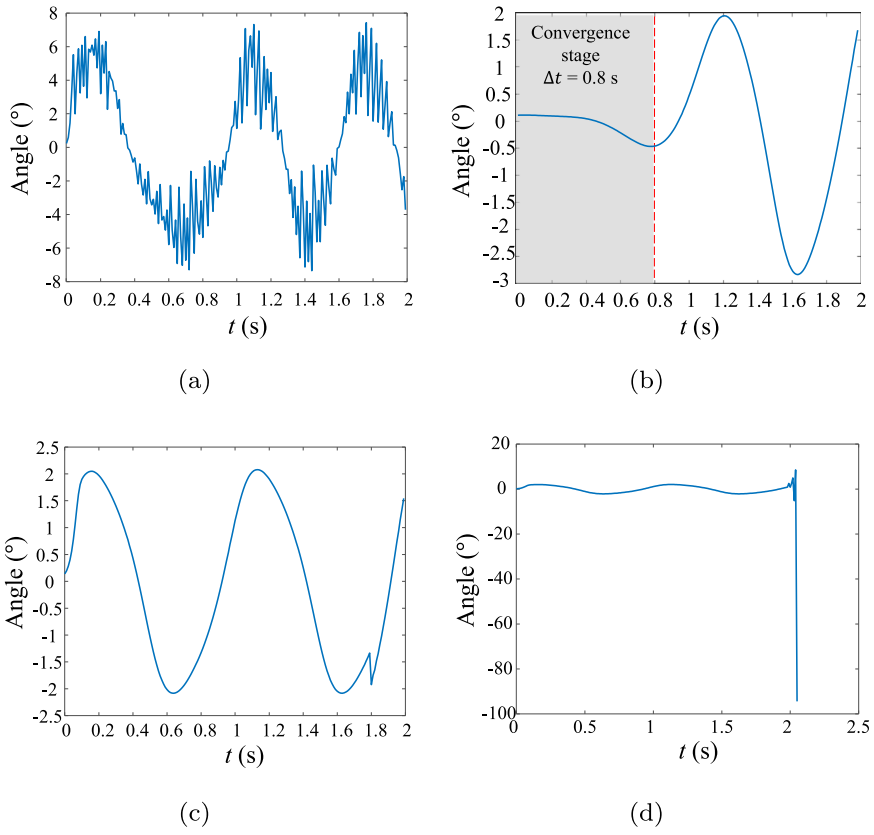

Fig. 2. Unexpected conditions of original Hopf-based CPGs. (a) Output oscillations caused by a large learning rate. (b) Slow convergence caused by a small learning rate. (c) Spikes caused by drastic changes in CPG parameters. (d) Divergence caused by drastic changes in CPG parameters.

2.1. Hopf-based CPG model

Existing CPG networks are constructed via nonlinear oscillations, including the Kuramoto phase oscillator and the Hopf oscillator. Benefiting from a stable limit cycle, the Hopf oscillator can quickly converge under an appropriate learning rate, α. The original Hopf-based CPG model is expressed as follows:

$$\begin{aligned}\begin{bmatrix}\dot{x}_i\\ \dot{y}_i\end{bmatrix} &= \begin{bmatrix}\alpha\left(A_i^2-r_i^2\right) & -\omega_i\\ \omega_i & \alpha\left(A_i^2-r_i^2\right)\end{bmatrix}\begin{bmatrix}x_i\\ y_i\end{bmatrix}\\ &\quad+\sum_{j=i\pm1} h\begin{bmatrix}\cos\left(\Delta\beta_{ij}\right) & -\sin\left(\Delta\beta_{ij}\right)\\ \sin\left(\Delta\beta_{ij}\right) & \cos\left(\Delta\beta_{ij}\right)\end{bmatrix}\begin{bmatrix}x_j\\ y_j\end{bmatrix}\\ \begin{bmatrix}x_i\\ y_i\end{bmatrix} &= \begin{bmatrix}x_i\\ y_i\end{bmatrix}+\begin{bmatrix}\dot{x}_i\\ \dot{y}_i\end{bmatrix}\mathrm{d}t\end{aligned} \tag{1}$$

where i refers to the biological neuron, x_i and y_i are CPG outputs, $r_i^2 = x_i^2 + y_i^2$, α denotes the learning rate, $\Delta\beta_{ij}$ is the phase difference between neuron i and neuron j, ω_i indicates the angular frequency, A_i is the amplitude, $\mathrm{d}t$ is the control period, and h is the coupling coefficient.

Using the original CPG model, only symmetric signals can be generated. However, in nature, asymmetric motion is widespread and indispensable. Therefore, this CPG model cannot effectively reflect all-real nervous control signals. As for bionic robots, single output signals of the CPG model greatly limit performance improvement and the exploration of motion modes. Besides, the optimal learning rate α often does not have the same value under different amplitudes in the original CPG model. If both α and amplitudes are large, the outputs oscillate strongly, as shown in Fig. 2(a). On the contrary, if they are both small, the outputs converge slowly, as shown in Fig. 2(b). Obviously, the optimal learning rate of the CPG network should not be fixed. In addition, when the CPG parameters (frequency or amplitude) are changed sharply, the oscillator may produce a large $\dot{x}_i$ and $\dot{y}_i$ for quickly converging, leading to a bad spike, even divergence, for the joint angle curve of the robot, as shown in Figs. 2(c) and 2(d).

2.2. Normalized asymmetric CPGs

To cope with the problems mentioned above, we propose an updated NA-CPG model, i.e., Eqs. (2)–(6), which has a single-chain topology, as shown in Fig. 3. Generally, two strategies are applied to obtain stable and multimodal rhythmic outputs. First, the oscillator amplitude is normalized to unit 1, and a constraint function is added to improve the stability and precision of CPGs. Second, an asymmetric parameter is utilized to adjust the velocity of the limit cycle at different positions to generate asymmetric signals. More details are introduced in the following:

$$\begin{aligned}\begin{bmatrix}\dot{x}_i\\ \dot{y}_i\end{bmatrix} &= \begin{bmatrix}\alpha\left(1-r_i^2\right) & -\frac{1}{\zeta_i}\omega_i\\ \frac{1}{\zeta_i}\omega_i & \alpha\left(1-r_i^2\right)\end{bmatrix}\begin{bmatrix}x_i\\ y_i-b_i\end{bmatrix}\\ &\quad+\sum_{j=i\pm1} h\begin{bmatrix}\cos\left(\Delta\varphi_{ij}\right) & -\sin\left(\Delta\varphi_{ij}\right)\\ \sin\left(\Delta\varphi_{ij}\right) & \cos\left(\Delta\varphi_{ij}\right)\end{bmatrix}\begin{bmatrix}x_j\\ y_j-b_j\end{bmatrix}\end{aligned} \tag{2}$$

$$\zeta_i = 1 - h_{a,i}\frac{\dot{x}_{i,old}}{\left|\dot{x}_{i,old}\right|} \tag{3}$$

$$\begin{bmatrix}\dot{x}_i\\ \dot{y}_i\end{bmatrix} \in \left[\begin{bmatrix}\dot{x}_{i,\mathrm{old}}\\ \dot{y}_{i,\mathrm{old}}\end{bmatrix} - \mathrm{CF}\left(w_i\right), \begin{bmatrix}\dot{x}_{i,\mathrm{old}}\\ \dot{y}_{i,\mathrm{old}}\end{bmatrix} + \mathrm{CF}\left(w_i\right)\right] \tag{4}$$

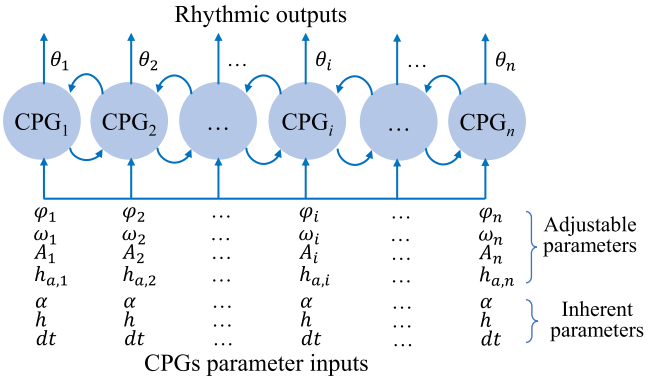

Fig. 3. The single-chain topology of NA-CPGs.

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} \dot{x}_i \\ \dot{y}_i \end{bmatrix} \mathrm{d}t \tag{5}$$

$$\theta_i = A_i \times y_i \tag{6}$$

where i refers to the biological neuron ($i = 1, \ldots, n$), x_i and y_i are NA-CPG outputs, whose amplitudes are set to 1, α is the learning rate, $r_i^2 = x_i^2 + (y_i - b_i)^2$, A_i is the amplitude, θ_i is the output angle, b_i is the offset ratio of the rhythmic signals, $\Delta\varphi_{ij}$ is the phase difference between i and j, ω_i is the angular frequency, ζ_i is the adjustment parameter, $\mathrm{d}t$ is the control period, and h is the coupling coefficient.

The stability and precision of the CPG network are effectively improved owing to amplitude normalization. Since the amplitude is normalized to unit 1, the gain should be compensated in the following calculation of the control angles, as shown in (6). Note that if the amplitudes are suddenly modified, the system states x_i, y_i must be divided by the old amplitudes, which indicates that the initial position of the limited cycle of Hopf oscillators is altered. Thereafter, the new value can be assigned to the amplitude. Thus, CPGs can generate smooth control signals even if the amplitude parameters are modified. To constrain $\dot{x}_i$ and $\dot{y}_i$, a constraint function, $\mathrm{CF}(w_i)$, is employed, as illustrated in (4). It is fitted by the maximum values of $\ddot{x}_i$ and $\ddot{y}_i$ at different angular frequencies. Meanwhile, an adjustment parameter, ζ_i, is applied to obtain asymmetric rhythmic signals, which can be calculated by (3). Note that ζ_i only changes the velocity of the limit cycle at different positions but not the trajectory of the limit cycle. When $h_{a,i}$ is positive, the extra acceleration is applied when $\dot{\alpha}_i > 0$, and the extra deceleration is applied in the other phase. Thus, with the acceleration and deceleration in different phases, the output of NA-CPGs becomes asymmetric. Since the acceleration and deceleration of the limit cycle are symmetric, the outputs of NA-CPGs can remain constant in frequency and amplitude steadily. In this way, the proposed NA-CPGs can achieve different speeds in one period, and the degree of the asymmetric oscillation can be adjusted by $h_{a,i}$. Combined with the structural improvement and the asymmetric parameter, the developed NA-CPGs can produce stable and diverse rhythmic signals. In conclusion, the NA-CPGs have the following characteristics:

(1) The optimal learning rate matching unit amplitude can ensure the transition of CPGs to the next convergence state as quickly as possible when other parameters are modified.
(2) The error between the actual and set amplitudes can be kept within a fixed range through adjusting the learning rate.
(3) The constraint function ensures a smooth transition even when the parameters change drastically.
(4) The adjustment parameter ζ_i can regulate the velocity of the limit cycle at different positions to generate asymmetric rhythmic signals.

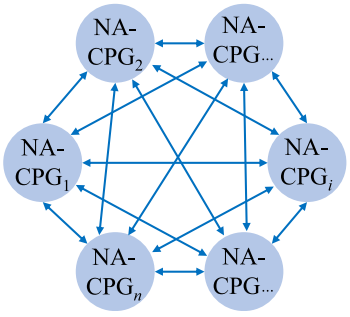

Fig. 4. Fully-connected NA-CPGs topology.

2.3. Matrix form of NA-CPGs

When performing numerical calculations, we take the matrix form of NA-CPGs. The matrix form of NA-CPGs has higher computational efficiency and is more advantageous in numerical calculations, especially in large-scale intelligent algorithms. To universalize the NA-CPG model, we propose the following fully connected mathematical model of NA-CPGs (Fig. 4) and establish a matrix form to be widely applied to robot motion control. Assuming there are n joints, i.e., n oscillators, the phases of each oscillator are φ_1, φ_2, ..., φ_n, and the $\Delta\varphi_{ij}$ in the coupling term is $\varphi_i - \varphi_j$. The coupling term is obtained by the weighted superposition of the membrane potential and the adjustment potential of all oscillators connected to the ith oscillator at the sine and cosine values of the phase difference. The analysis of the ith oscillator is as follows:

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \end{bmatrix} = K(x_i, y_i) \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \sum_{j=1}^{n} R(\Delta\varphi_{ij}) \begin{bmatrix} x_j \\ y_j \end{bmatrix}, i = 1, 2, \ldots, n \tag{7}$$

where

$$K_{ij} = K(x_i, y_i) = \begin{bmatrix} \alpha \cdot (1 - r_i^2) & -\frac{1}{\zeta_i}\omega_i \\ \frac{1}{\zeta_i}\omega_i & \alpha(1 - r_i^2) \end{bmatrix} \tag{8}$$

$$R_{ij} = R(\Delta\varphi_{ij}) = \begin{bmatrix} \cos(\Delta\varphi_{ij}) & -\sin(\Delta\varphi_{ij}) \\ \sin(\Delta\varphi_{ij}) & \cos(\Delta\varphi_{ij}) \end{bmatrix}, \quad \Delta\varphi_{ij} = \varphi_i - \varphi_j \tag{9}$$

$$r_i^2 = x_i^2 + y_i^2 \tag{10}$$

For the oscillator itself, $\Delta\varphi_{ii} = 0$, $R_{ii} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, the polynomials to the right of (7) are merged to give the matrix form of NA-CPGs:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \\ \ldots \\ \dot{x}_i \\ \dot{y}_i \\ \ldots \\ \dot{x}_n \\ \dot{y}_n \end{bmatrix}_{2n\times 1} = \begin{bmatrix} K_{11} & R_{12} & \ldots & R_{1i} & \ldots & R_{1n} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ R_{i1} & R_{i2} & \ldots & K_{ii} & \ldots & R_{in} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ R_{n1} & R_{n2} & \ldots & R_{ni} & \ldots & K_{nn} \end{bmatrix}_{2n\times 2n} \begin{bmatrix} x_1 \\ y_1 - b_1 \\ \ldots \\ x_i \\ y_i - b_i \\ \ldots \\ x_n \\ y_n - b_n \end{bmatrix}_{2n\times 1} \tag{11}$$

$$\begin{bmatrix} \dot{x}_i \\ \dot{y}_i \end{bmatrix} \in \left[\begin{bmatrix} \dot{x}_{i,\mathrm{old}} \\ \dot{y}_{i,\mathrm{old}} \end{bmatrix} - \mathrm{CF}(w_i), \begin{bmatrix} \dot{x}_{i,\mathrm{old}} \\ \dot{y}_{i,\mathrm{old}} \end{bmatrix} + \mathrm{CF}(w_i) \right] \tag{12}$$

2.4. Selection of inherent network parameters

Generally, CPG networks have two types of parameters: inherent parameters and adjustable parameters. Inherent parameters such as learning rate, coupling rate, and time interval, which are generally set to fixed values. Adjustable parameters related to the output rhythm, such as frequency, amplitude, and phase relationship, which are generally changed with the mode of motion. Inherent parameters are critical to the dynamic performance of the CPG network. For example, the

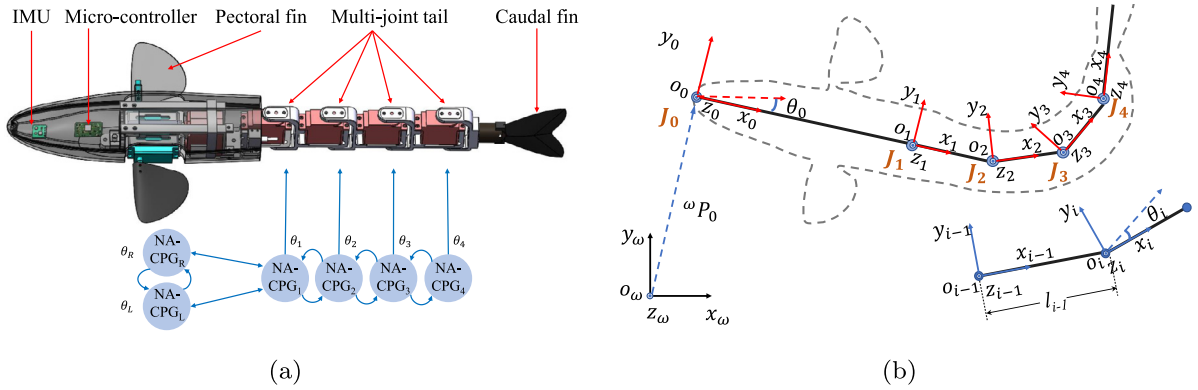

Fig. 5. Experimental platform. (a) Prototype of the robotic fish. (b) Coordinate systems of the robotic fish.

learning rate affects the vibration speed. The larger the learning rate is, the faster the vibration. However, if the learning rate is too large, the oscillator may become unstable. As mentioned above, the proposed NA-CPGs enhance network adaptability to parameter changes at a fixed optimal learning rate owing to amplitude normalization. Notably, the network can also realize performance optimization by adjusting inherent parameters, e.g., learning rate, in the absence of amplitude normalization.

Time interval also impacts the numerical calculations of CPGs. The smaller the interval is, the smoother the output rhythm is. As the time interval decreases, the learning rate can be increased appropriately, and the network can converge to a stable rhythmic output more quickly. In the proposed NA-CPGs network, the time interval is set to 0.01 s, and correspondingly, the learning rate is set to 30 because the time interval of 0.01 can meet the computing requirements of CPG networks at high frequencies, and it can be easily obtained from the timer peripherals in the STM32 chip.

The amplitude of CPGs' output is precise relative to the set value in a single Hopf oscillator or in the condition that all oscillators have the same amplitude in a multi-oscillator-coupled CPG system. However, when oscillators have different amplitudes, the amplitudes of rhythmic outputs are affected by phase coupling and deviate from the set amplitude. There are two ways to address the effect of strong coupling between joints on the output amplitude: One is to weaken the coupling term and the interaction between the joints; however, this method needs to compensate for the amplitude, and the output accuracy of the amplitude cannot be guaranteed. The other is to normalize the amplitude in topological coupling and gain the desired amplitude at the output rhythm so that all oscillators have the same amplitude of 1; this method reduces the influence of the coupling term on the amplitude and is adopted in NA-CPGs.

In summary, the normalized network greatly weakens the coupling between adjustable parameters and inherent parameters so that the network obtains optimal convergence performance under fixed inherent parameters, more smoothly copes with random changes in adjustable parameters, and more robustly adapts to different combinations of motion parameters.

3. Online optimization method based on NA-CPGs

In this section, we propose an online optimization method for the constructed NA-CPGs to improve the motion performance of the developed robot. In particular, a multijoint robotic fish is illustrated as the optimization object. In the following, a robotic fish platform is first built based on the hydrodynamic analysis. Subsequently, in the constructed robotic environment, an optimization algorithm based on reinforcement learning is proposed to improve the speed performance of the robotic fish.

3.1. Robotic fish platform setup

The employed robotic fish was developed in our previous work [27], which consisted of a four-joint tail, a hard-shelled head, and two pectoral fins with 2 degrees of rotational freedom. Its mechanical structure and prototype are illustrated in Fig. 5(a), and the tail joints are driven by four servomotors. The control signal comes from a STM32 microcontroller, which obtains the motion information of the robotic fish from an IMU module and a depth sensor and communicates with the experimental console through the radio frequency module. This robotic fish can not only obtain comprehensive sensor information but also swim flexibly with the cooperation of the tail joints. The real-time communication and control capabilities enable online optimization.

Considering the multijoint robotic fish as a multilink mechanism, the head speed can be calculated by the Newton–Euler equation using the kinematic analysis, and the kinematic data of each tail joint are calculated by the converting matrix of the coordinate system of the multilink. The recursive relationship of the velocity vector is built as follows:

$$\begin{aligned} \boldsymbol{V}_i &= \begin{bmatrix} \boldsymbol{U}_i \\ \boldsymbol{\Omega}_i \end{bmatrix} = \boldsymbol{H}_{i-1}^{i} \boldsymbol{V}_{i-1} + \dot{\theta}_i \boldsymbol{K}_i, \\ \boldsymbol{H}_{i-1}^{i} &= \begin{bmatrix} \boldsymbol{R}_{i-1}^{i} & \boldsymbol{R}_{i-1}^{i} \hat{\boldsymbol{P}}_{i-1}^{i} \\ 0_{3\times3} & \boldsymbol{R}_{i-1}^{i} \end{bmatrix}, \boldsymbol{K}_i = \begin{bmatrix} \boldsymbol{0}_{3\times1} \\ \boldsymbol{k}_i \end{bmatrix} \\ \boldsymbol{R}_i^{i-1} &= \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 \\ \sin\theta_i & \cos\theta_i & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad \boldsymbol{P}_i^{i-1} = \begin{bmatrix} l_{i-1} \\ 0 \\ 0 \end{bmatrix} \end{aligned} \tag{13}$$

where $\boldsymbol{H}_{i-1}^{i}$, $\boldsymbol{R}_i^{i-1}$ and $\boldsymbol{P}_i^{i-1}$ are the converting matrix, rotation matrix, and position vector, respectively; $\boldsymbol{V}_i$ is the ith joint's velocity vector; θ_i is the ith joint angle; $\boldsymbol{k}_i$ is the vertical upward unit vector at each joint.

With regard to dynamic analysis, the thrust of the robotic fish is mainly generated at the last joint of the tail. The force can be converted forward in turn through the coordinate system converting matrix of the multilink, and finally, the force of the head can be obtained. Thus, the kinematic data such as speed and attitude can be computed by the Newton–Euler law. The recurring relationship of the interaction forces between adjacent joints is as follows:

$$\boldsymbol{G}_{i+1,i}^{i} = \boldsymbol{H}_{i+1}^{i} \boldsymbol{G}_{i+1,i}^{i+1} = \boldsymbol{H}_{i+1}^{i} \boldsymbol{G}_{i,i+1}^{i+1} \tag{14}$$

where $\boldsymbol{G}_{i,j}^{k}$ is the representation of the force of the ith joint on the jth joint in the kth joint coordinate system.

The force analysis of each joint is obtained based on hydrodynamic theory [28]. Each joint is mainly subject to lift $\boldsymbol{L}_i$ and drag $\boldsymbol{D}_i$ of the tail fin, additional mass forces $\boldsymbol{f}_{ad,i}$, resistance $\boldsymbol{f}_{dr,i}$ from water, Corioli forces, and conceding forces $\boldsymbol{\gamma}_i$. According to the Newton–Euler

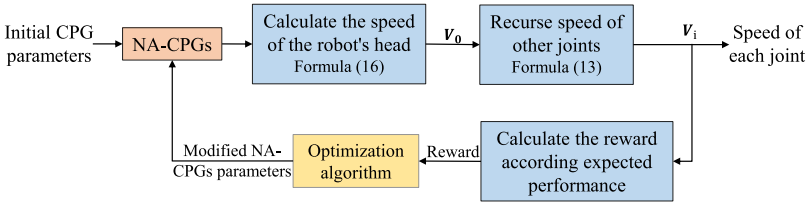

Fig. 6. Schematic diagram of data flow in the robotic fish environment.

equation, the relationship between the force and the acceleration of the joints is as follows:

$$\boldsymbol{G}^{i}_{i-1,i} - \boldsymbol{G}^{i}_{i+1,i} + \boldsymbol{L}_i + \boldsymbol{D}_i + \boldsymbol{f}_{ad,i} + \boldsymbol{f}_{dr,i} = \boldsymbol{M}_i \boldsymbol{V}_i + \boldsymbol{\gamma}_i \tag{15}$$

where $\boldsymbol{f}_{ad,i} = -\boldsymbol{M}_{ad,i}\boldsymbol{V}_i - \boldsymbol{\gamma}_{ad,i}$; $\boldsymbol{f}_{ad,i}$ is the additional mass force of each joint; $\boldsymbol{\gamma}_{ad,i}$ is the Corioli force and conceding force caused by additional mass.

The speed is recursive from the head to the tail, and the force is recursive from the tail to the head. In the head, the speed and the force are connected by (13)–(15) so that the explicit dynamic equation is obtained as follows:

$$\begin{aligned}\dot{\mathbf{V}}_0 = &-\left[\sum_{i=0}^{n}\left(\mathbf{H}_i^0\left(\mathbf{M_i}+\mathbf{M}_{ad,i}\right)\mathbf{H}_0^i\right)\right]^{-1} \\ &\left[\sum_{i=1}^{n}\mathbf{H}_i^0\left(\dot{\mathbf{H}}_{i-1}^i\mathbf{V}_{i-1}+\ddot{\theta}_i\mathbf{K}_i\right)\sum_{j=i}^{n}\mathbf{H}_j^i\left(\mathbf{M}_j+\mathbf{M}_{ad,j}\right)\mathbf{H}_i^j\right. \\ &\left.-\sum_{i=0}^{n}\mathbf{H}_i^0\left(\mathbf{L}_i+\mathbf{D}_i-\gamma_{ad,i}-\gamma_i+\mathbf{f}_{dr,i}\right)\right]\end{aligned} \tag{16}$$

Note that the hydrodynamic parameters in the model were obtained by performing aquatic experiments, and more details on the dynamic model can be found in [17].

To clearly demonstrate the relationship between the robotic fish platform and the optimization algorithm for NA-CPGs, the calculation process and data flow are summarized, as shown in Fig. 6. Until now, the multijoint robotic fish environment has been established.

3.2. Optimization algorithm based on reinforcement learning

The optimization algorithm is used to build an original RL-Agent for the adaptive update of NA-CPG parameters and then obtain a set of optimized NA-CPG parameters. Owing to the deep network's ability to fit complex functions, the learning problem in a multi-dimensional action space can be solved based on a deep Q network (DQN) method. In this work, a six-layer neural network was designed to serve as an approximator of the action value Q function. For the multijoint robotic fish environment, the action space can be constrained by the prior knowledge of NA-CPG parameters, and the globally optimal NA-CPG parameters can be recorded throughout the training process to reset the NA-CPG parameters in the next episode. Prior knowledge and global data can improve the convergence speed of training to a certain extent.

The learning algorithm involves the following five aspects:

- **Action space.** It is composed of 15 discrete actions. In detail, 8 actions control the increase or decrease of the amplitude of four-joint angles, 6 actions control the increase or decrease of the phase difference between the four joints, and the last action does not change any CPG parameter. The execution of the action is based on the prior knowledge of the CPG parameters of the robotic fish system. The prior knowledge is reflected in that the amplitude of each joint increases from front to back and that the phase differences between the joints are positive, that is, the phase of the next joint is behind the previous joint. Due to mechanical limits, the effective amplitude range is set to $[0,\ 60°]$, and the phase difference range is set to $[0,\ 60°]$. A nonzero exploration rate ε is configured to accelerate the exploration of the action space.
- **State space.** It is an 8×1 vector, as shown in (17). $\overline{V}_x$ is the average speed in one period after the speed stabilizes.

$$State = \left[\boldsymbol{A}_{4\times1}, \boldsymbol{\Delta\varphi}_{3\times1}, \overline{V}_x\right] \tag{17}$$

- **Reward function.** Its design is related to the expected swimming performance. The desired performance in this work was high swimming speed and small head swing. The small head swing is reflected in the small fluctuation range of the lateral velocity. As a result, the reward function was designed using the forward speed V_x and the speed interval V_{range} in one period. After several attempts, a feasible reward function was designed as (18). The reward function consists of two parts, i.e., kinetic energy reward and fluctuation range penalty, respectively. a and b are the weights of the two parts.

$$Reward = a \times \overline{V}_x^2 - b \times V_{range} \tag{18}$$

- **Termination condition.** It is mainly determined by two factors. The one is the maximum executions. It is expected that the trained agent can obtain optimized parameters within 30 executions of the action. The other one is the abnormal condition. It is never expected that the forward speed is negative. Therefore, when the speed reaches a negative value, a large penalty is given, and consequently, the episode ends.
- **Environment reset.** When the environment is reset, the CPG parameters are reset with the recorded global optimal ones, which are used as the initial state of the next episode, effectively speeding up training.

4. Results and discussion

In this section, the results of the experimental work conducted to validate the improved performance of the designed NA-CPGs model and the effectiveness of the proposed online optimization method.

4.1. Testing on the performance of NA-CPGs

The first simulation focuses on the diversity and stability of the developed NA-CPG model. By adjusting some key parameters, the NA-CPGs can output various control rhythmic signals for the robot. For example, h_a decides the asymmetry of the output rhythmic signals. Fig. 7(a) shows the asymmetric oscillation pattern of the NA-CPGs under different h_a. The output signals have a symmetric pattern when $h_a = 0$; however, when h_a is 0.4, the signals obviously become asymmetric. These asymmetric control signals are very important for robotic control. As for the robotic fish, high flapping speed can generate a large hydrodynamic force to swim. When tail flapping becomes asymmetric, the asymmetric force acts on the robotic fish to improve its steering ability. Besides, the developed NA-CPGs have superior accuracy and stability. Relying on an appropriate learning rate, the error between the actual amplitude and the set amplitude can be kept within a fixed range. By adjusting the learning rate according to control requirements, the error magnitude can be controlled. As shown in Fig. 7(b), in the case of $\alpha = 30$, the amplitude error is only $(1.053 - 1) \div 1 \approx 5.3\%$. Based on the experiments under different parameters, the statistical value of the amplitude error is $4.85\%(\pm 3.14\%)$. In the case of the normalized amplitude,

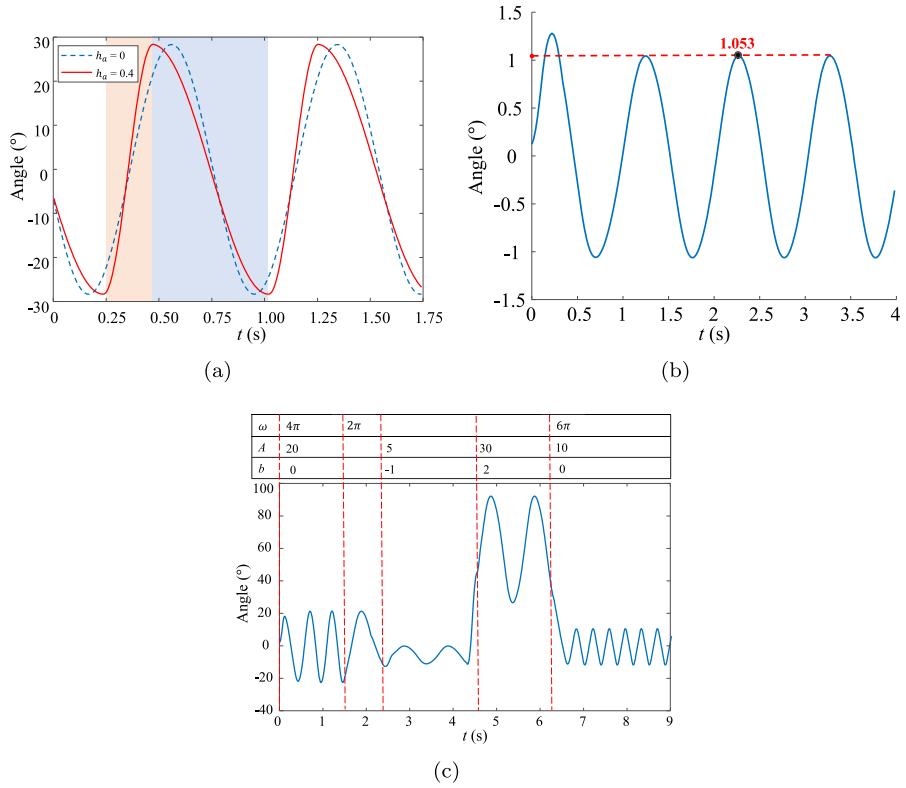

Fig. 7. Output characteristics of the NA-CPG model. (a) Asymmetric output. (b) Precise rhythmic output. (c) Smooth outputs of NA-CPGs with parameter variation.

the optimal learning rate matching unit amplitude is selected to ensure that CPGs can be transited to the next convergence state as quickly as possible when the CPG parameters are modified. When dealing with drastically changing network parameters, NA-CPGs behave as shown in Fig. 7(c). NA-CPGs have a smooth transition when the parameters change significantly, offering two benefits: The smooth output rhythm greatly improves the control safety of the robot's end, and NA-CPGs can still ensure a stable output of the control signal when applied to algorithms with black boxes, such as reinforcement learning.

Fig. 8 shows the comprehensive performance of NA-CPGs. Within 12 s of decreasing the frequency ω, adjusting the asymmetric parameters h_a, increasing the amplitude A_2, setting the offset parameters b, and reducing the phase difference $\Delta\varphi_{12}$ sequentially, the curve can all be smoothly and quickly transitioned. In the outputs of multiple coupled oscillators, the amplitude of each joint is maintained at the set value, the offset and asymmetric settings can be smoothly implemented, and the change of phase difference can be easily achieved. The rhythmic outputs of the comprehensive experiment verified the stability, accuracy, safety, and diversity of NA-CPGs.

4.2. RL-based online optimization of the robotic fish

Within the algorithm framework presented in Section 3, the RL-Agent was trained. According to previous experiences, the initial NA-CPG parameters were set to $\boldsymbol{A} = [10, 15, 20, 25]$, $\boldsymbol{\Delta\varphi} = [30, 15, 15]$. The frequency of NA-CPGs was a fixed value, 2π rad/s, that is, the tail flaps once per second. The weight a and b were set to 200 and 100, respectively. The exploration rate ε was set to 0.1.

After training for 60 episodes, the training curve was converged, as shown in Fig. 9(a). From the convergence curve, after training 30 episodes, the original RL-Agent and optimized NA-CPGs parameters were obtained.

To highlight the effects of the optimized NA-CPG parameters, the initial parameters before training and the optimized parameters after training were used to control the robotic fish. The curves of forward speed under two sets of parameters are illustrated in Fig. 9(b). The blue curve is the speed under the initial parameters, while the red curve is the speed under the optimized parameters. At 1 Hz, the forward speed increased from 0.30 m/s to 0.32 m/s and the head swing amplitude reduced from 0.24 m/s to 0.14 m/s, verifying the effectiveness of the optimization algorithm. After obtaining the RL-Agent, we can load the RL-Agent on the robotic fish. The RL-Agent no longer needs the exploration of random actions; thus, the ε is set to zero. The optimized NA-CPG parameters are used as the initial parameters of the robotic fish. The RL-Agent can optimize the parameters online through the sensor data and improve its intelligence. Through online learning during swimming, the robotic fish gradually learns to adjust tail flapping parameters more robustly. As a result, irrespective of the environment the robotic fish swims in, the optimized parameters can be used as the initial parameters.

The optimization algorithm is not limited to the swimming speed performance. By designing appropriate reward functions according to the intended performance, the optimization algorithm based on NA-CPGs can be adopted for other tasks, such as high acceleration, fast turning, and high energy efficiency, to name a few.

5. Conclusions and future work

Based on the classic Hopf-based oscillator, we proposed a novel rhythm generator, NA-CPGs, which normalized the network amplitude and added a constraint function and adjustment parameters. In particular, normalized networks set the optimal learning rate to obtain better convergence performance and more accurate amplitude output. The constraint function ensured a safe and smooth transition of the NA-CPGs when the network parameters were switched, improving system stability. Meanwhile, the adjustment parameters provided asymmetric factors for the network, increasing the diversity of the network. Based on the proposed NA-CPGs, we designed an online NA-CPGs parameter

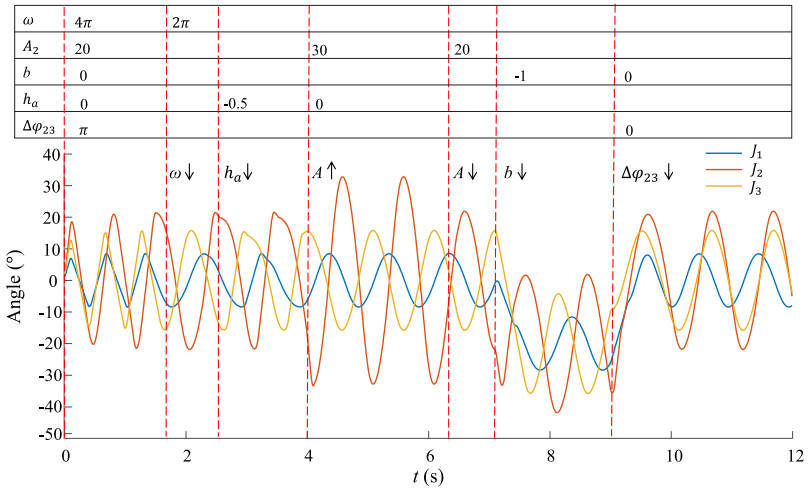

Fig. 8. Comprehensive performance of NA-CPGs, where $A_1 = 8, A_3 = 15, \Delta\varphi_{12} = \pi/2$.

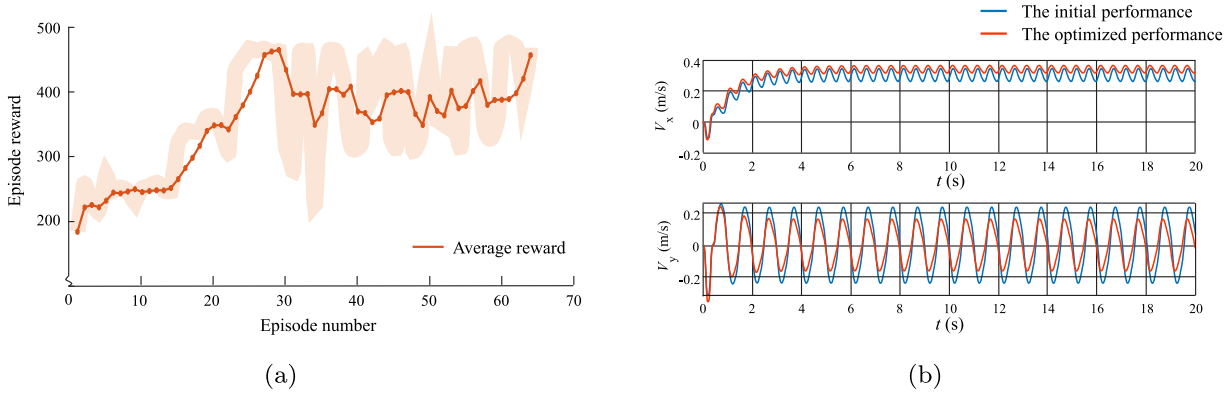

Fig. 9. Results of the optimization methods. (a) Plot of training convergence curve. (b) Speed performance comparison under NA-CPG parameters before and after training. Note: V_x is along the direction of x_ω, the forward swimming speed of the robotic fish. V_y is along the axis y_ω, and the range of V_y refers to the V_{range} in formula (18).

optimization method based on reinforcement learning for a biomimetic robotic fish to improve its swimming performance, and trained an online optimizer that can adjust NA-CPG parameters. The simulation results proved that NA-CPGs can generate a stable, diverse, and safe rhythmic output, and the intelligent optimization method for NA-CPGs improves the swimming speed (6.7%), reduces the lateral swing (41.7%), and optimizes the motion control of the robotic fish.

In the future, we will expand the output waveform of NA-CPGs and conduct optimization training based on NA-CPGs to improve steering maneuverability, etc., and further verify the effectiveness of optimization in actual experiments.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (61836015, U1909206, 62022090, and 62033013).

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.birob.2022.100075.

References

[1] F. Delcomyn, Neural basis for rhythmic behaviour in animals, Science 210 (1980) 492–498.
[2] P.S.G. Stein, J. Smith, Neural and iomechanical control strategies for different forms of vertebrate hindlimb motor tasks, in: Neurons, Networks and Motor Behavior, 1997, pp. 61–73.
[3] A.J. Ijspeert, Central pattern generators for locomotion control in animals and robots: A review, Neural Netw. 21 (4) (2008) 642–653.
[4] J. Buchli, A.J. Ijspeert, Distributed central pattern generator model for robotics application based on phase sensitivity analysis, in: International Workshop on Biologically Inspired Approaches to Advanced Information Technology, Springer, 2004, pp. 333–349.
[5] H. Kimura, S. Akiyama, K. Sakurama, Realization of dynamic walking and running of the quadruped using neural oscillator, Auton. Robots 7 (3) (1999) 247–258.
[6] Y. Kuramoto, Collective behavior of coupled phase oscillators, in: The Handbook of Brain Theory and Neural Networks, 2003, pp. 223–226.
[7] M. Golubitsky, I. Stewart, P.L. Buono, J.J. Collins, A modular network for legged locomotion, Physica D 115 (1–2) (1998) 56–72.
[8] X. Li, H. Liu, X. Wu, R. Li, X. Wang, Improved CPG model based on hopf oscillator for gait design of a new type of hexapod robot, in: International Conference on Intelligent Robotics and Applications, Springer, 2019, pp. 72–83.
[9] G. Liu, M.K. Habib, K. Watanabe, K. Izumi, Central pattern generators based on matsuoka oscillators for the locomotion of biped robots, Artif. Life Robot. 12 (1) (2008) 264–269.

[10] Y. Wang, X. Xue, B. Chen, Matsuoka CPG with desired rhythmic signals for adaptive walking of humanoid robots, IEEE Trans. Cybern. 50 (2) (2018) 613–626.
[11] P. Arena, P. Sueri, S. Taffara, L. Patanè, MPC-based control strategy of a neuro-inspired quadruped robot, in: IEEE International Joint Conference on Neural Networks, IJCNN, 2021, pp. 1–8.
[12] T. Fukui, H. Fujisawa, K. Otaka, Y. Fukuoka, Autonomous gait transition and galloping over unperceived obstacles of a quadruped robot with CPG modulated by vestibular feedback, Robot. Auton. Syst. 111 (2019) 1–19.
[13] A.J. Ijspeert, A. Crespi, Online trajectory generation in an amphibious snake robot using a lamprey-like central pattern generator model, in: Proceedings of the 2007 IEEE International Conference on Robotics and Automation, 2007, pp. 262–268.
[14] A. Crespi, A.J. Ijspeert, Online optimization of swimming and crawling in an amphibious snake robot, IEEE Trans. Robot. 24 (1) (2008) 75–87.
[15] Z. Yan, H. Yang, W. Zhang, Q. Gong, F. Lin, Research on motion mode switching method based on CPG network reconstruction, IEEE Access 8 (2020) 224871–224883.
[16] X. Niu, J. Xu, Q. Ren, Q. Wang, Locomotion learning for an anguilliform robotic fish using central pattern generator approach, IEEE Trans. Ind. Electron. 61 (9) (2013) 4780–4787.
[17] J. Yu, M. Wang, Z. Su, M. Tan, J. Zhang, Dynamic modeling of a CPG-governed multijoint robotic fish, Adv. Robot. 27 (4) (2013) 275–285.
[18] D. Korkmaz, G.O. 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 (2) (2021) 125–137.
[19] F. Xie, Z. Li, Y. Ding, Y. Zhong, R. Du, An experimental study on the fish body flapping patterns by using a biomimetic robot fish, IEEE Robot. Autom. Lett. 5 (1) (2019) 64–71.
[20] Z. Wu, J. Yu, J. Yuan, M. Tan, Towards a gliding robotic dolphin: Design, modeling, and experiments, IEEE/ASME Trans. Mechatronics 24 (1) (2019) 260–270.
[21] J. Yu, Z. Wu, X. Yang, Y. Yang, P. Zhang, Underwater target tracking control of an untethered robotic fish with a camera stabilizer, IEEE Trans. Syst. Man Cybern.: Syst. 51 (10) (2020) 6523–6534.
[22] G.N. Carryon, J.L. Tangorra, The effect of sensory feedback topology on the entrainment of a neural oscillator with a compliant foil for swimming systems, Bioinspiration Biomim. 15 (4) (2020) 046013.
[23] J. Yu, S. Chen, Z. Wu, X. Chen, M. Wang, Energy analysis of a CPG-controlled miniature robotic fish, J. Bionic Eng. 15 (2) (2018) 260–269.
[24] J. Yu, Z. Wu, M. Wang, M. Tan, CPG network optimization for a biomimetic robotic fish via PSO, IEEE Trans. Neural Netw. Learn. Syst. 27 (9) (2015) 1962–1968.
[25] X. Niu, J. Xu, Q. Ren, Q. Wang, Locomotion learning for an anguilliform robotic fish using central pattern generator approach, IEEE Trans. Ind. Electron. 61 (9) (2013) 4780–4787.
[26] S. Yan, Z. Wu, J. Wang, M. Tan, J. Yu, Efficient cooperative structured control for a multijoint biomimetic robotic fish, IEEE/ASME Trans. Mechatronics 26 (5) (2020) 2506–2516.
[27] J. Yu, M. Tan, Design and control of a multi-joint robotic fish, in: Robot Fish, Springer, 2015, pp. 93–117.
[28] F. El-Hawary, The Ocean Engineering Handbook, Crc Press, 2000.