machines

MDPI

Article

Trajectory Tracking of Underwater Hexapod Robot Based on Model Predictive Control

Ruiwei Liu [1], Jieyu Zhu [2], Manjia Su [1], Xianyan Gu [1], Shuohao Fang [3], Dehui Zheng [1] and Haoyu Yang [1,*]

[1] School of Intelligent Manufacturing, Guangzhou Maritime University, Guangzhou 510700, China; liuruiwei8124@163.com (R.L.); kylesu1@163.com (M.S.); 18318341895@163.com (X.G.); m17820223245@163.com (D.Z.)
[2] School of Future Transportation, Guangzhou Maritime University, Guangzhou 510700, China; zhujieyu9322@163.com
[3] School of Low Altitude Equipment and Intelligent Control, Guangzhou Maritime University, Guangzhou 510700, China; 19875195183@163.com
* Correspondence: yanghaoyu0532@126.com

Abstract

To achieve high-precision trajectory tracking control for an underwater hexapod robot, this paper proposes a hierarchical control architecture. Firstly, a multi-rigid-body dynamic model for the robot is established based on the Newton-Euler method and reasonably simplified. Secondly, a Central Pattern Generator (CPG) network with the Hopf oscillator as its core is designed to generate stable and coordinated crawling gaits. By introducing a steering parameter, a kinematic model connecting the CPG output is constructed. Furthermore, based on this dynamic and kinematic model, an upper-layer Model Predictive Controller (MPC) is designed. The optimized control quantities output by the MPC are mapped into the rhythmic parameters of the CPG network via a transfer function established by fitting experimental data, thus forming the complete MPC-CPG controller. Finally, the proposed method is validated through simulations of circular trajectory tracking. The results show that even in the presence of initial errors, the controller can converge rapidly, with trajectory position error consistently maintained within −0.1 m~0.1 m, and heading angle error confined to the range of −15~15°. The experiments fully demonstrate the effectiveness of the proposed MPC-CPG controller in ensuring trajectory tracking accuracy, motion smoothness, and system stability.

Keywords: hexapod robot; dynamics; parameter mapping; model predictive control; trajectory tracking

Check for updates

Academic Editor: Zhuming Bi

Received: 7 January 2026
Revised: 29 January 2026
Accepted: 29 January 2026
Published: 2 February 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Animals possess remarkable locomotion capabilities and adaptability to their environment, serving as the primary inspiration for developing advanced robotic controls [1,2]. By mimicking terrestrial creatures, various legged robots have been developed, demonstrating excellent locomotion performance [3–5]. However, their controllers are mostly based on traditional methods, such as the Central Control Hub method [6], Inverse Dynamics Model Control [7], and Optimization-based Control [8]. The performance of these control methods heavily relies on the accuracy of the robot model and environmental conditions, and optimizing control parameters is a complex task. Therefore, in recent years, control methods inspired by biological CPGs, which do not rely on precise robot and environment models, have gained widespread attention.

The rhythmic patterns of animal locomotion are generated by a neural network known as the Central Pattern Generator (CPG) [9,10]. CPGs are widely present in vertebrates and invertebrates, primarily responsible for generating coordinated, rhythmic movements such as crawling, flying, swimming, jumping, walking, and running [11]. Inspired by this, researchers have developed various CPG oscillator models [12–14] and applied them in engineering practice. To address issues like sinking, imbalance, and control complexity for hexapod robots walking on lunar soil, Shu et al. successfully achieved stable walking on lunar simulant by combining a biomimetic structure design with an umbrella-like foot-end expansion mechanism and generating rhythmic signals based on a CPG [15]. Mao et al. proposed a bionic soft rectal robot system, which innovatively adopts a double-layer sphincter-like actuator optimized by a multi-layer perceptron, achieving precise pressure control with high contraction rate and rapid response [16]. Sun et al. integrated vestibular reflexes with various CPGs to generate adaptive body postures, enabling the robot to walk on different sloped terrains [17]. Gong et al. proposed a control method for an underwater hexapod robot based on super-twisting sliding mode and fuzzy CPG. By designing heading/speed controllers and parameter fuzzy mapping, combined with LOS guidance, path tracking was achieved. Experiments verified that this method ensures motion coordination while maintaining high tracking accuracy [18]. Zhong et al. applied CPG to motion control and gait planning for hexapod robots, adjusting relevant robot motion parameters based on feedback information. Experiments proved that this method enables stable walking of hexapod robots on rough terrain, with good agility, stability, and environmental adaptability [19]. Fukui et al. used vestibular sensory feedback to modulate the CPG, allowing the robot to autonomously change its gait according to speed, thereby safely traversing terrain with unknown obstacles at various speeds [20]. Chen et al. proposed a lightweight control strategy based on reinforcement learning and CPG, enabling robotic fish tails to adaptively adjust under complex flow fields and structural variations. Through training in real flow environments and extensive multi-scenario experiments, the proposed method outperformed traditional PID control in terms of tracking accuracy, stability, and response speed, demonstrating strong robustness and adaptability [21]. Liu et al. developed a biomimetic tactile perception CPG controller based on an improved Matsuoka oscillator feedback mechanism, integrating high-level "brain" commands with low-level "reflex arc" tactile feedback. The controller regulates sensory inputs through both reinforcement learning and local reflex responses, successfully driving a soft snake-like robot to navigate flexibly in dense obstacle environments and narrow passages. Both simulation and real-world experiments validated its superior tactile perception and locomotion capabilities [22]. Chen et al. proposed a lightweight heading control strategy based on reinforcement learning by combining RL-based decision making with CPG-based execution, enabling physical training and deployment on low-computational-power platforms. The method was validated on a robotic fish trained in a recirculating water tank and demonstrated strong generalization ability, robustness, and adaptability under complex flow conditions and tail-fin damage scenarios. Its control accuracy and stability were significantly superior to those of conventional PID methods [23].

Although the introduction of CPG into robot motion control has improved steady-state motion performance and adaptive capability, in scenarios such as walking in narrow environments, detecting landmines, and avoiding obstacles, robots need to follow predefined trajectories, requiring accurate and robust trajectory tracking performance [24]. Trajectory tracking is a key advantage of autonomous robots, but there is currently limited research on this topic. During trajectory tracking, interactions exist between the robot's components and between the robot and the environment, making it challenging to develop effective, accurate, and robust controllers for trajectory tracking. Control methods commonly used

for trajectory tracking include Sliding Mode Control (SMC) [25,26] and Model Predictive Control (MPC) [27,28], among others. Based on a trajectory tracking model, Chen et al. established a non-singular terminal sliding mode control algorithm based on an NTSM reaching law, enabling precise tracking of the reference trajectory by a hexapod robot in simulation, and validated the effectiveness of the control method through experiments [24]. Mera Manuel et al. proposed a method based on sliding mode robust control for a unicycle robot kinematic model closer to reality, introducing a novel sliding surface to reduce chattering effects on the heading angle, thereby achieving trajectory tracking control for the unicycle robot under disturbances [29]. Huang et al. proposed a Linear Time-Varying Model Predictive Controller (LTV-MPC) based on an online estimator, solving the trajectory tracking problem for electro-hydraulic actuators in the presence of time-varying parameters, high-frequency external load disturbances, measurement noise, and some unmeasurable states. Experiments showed that the LTV-MPC controller has strong adaptive capability and robust trajectory tracking performance [30]. Liu et al. combined the A* algorithm with the dynamic model of a four-wheel independent steering robot for trajectory planning, then established a robot kinematic model with dynamic constraints and a model predictive controller, achieving high-speed trajectory tracking control for the robot. Simulations and experiments verified the effectiveness and practicality of the trajectory tracking method [31]. At present, there are few studies on trajectory tracking for hexapod robots, although trajectory tracking is one of the key functions for robots to accomplish operational tasks. Therefore, this paper mainly investigates the realization of trajectory tracking for a hexapod robot by using a high-level control strategy (e.g., MPC) to regulate the low-level motion controller (CPG), while simultaneously improving the robustness and adaptability of locomotion.

Model Predictive Control, due to its advantages such as convenient modeling, good system robustness, stability, and the ability to incorporate multiple constraints into rolling optimization, has been widely applied to motion control of various robots with good results [32–36]. The crawling motion of the hexapod robot is a nonlinear problem with multiple constraints, and frequent switching of crawling gaits is required during trajectory tracking to ensure tracking accuracy of the reference trajectory. Therefore, this paper employs a CPG network as the underlying motion controller to ensure smoothness during motion control, while introducing algorithms like Model Predictive Control as the upper-level controller to adjust the CPG network parameters in real-time, thereby accomplishing the trajectory tracking task.

The main academic contributions of this paper are summarized as follows. First, a dynamic model of the underwater hexapod robot is established. By introducing a steering parameter and a kinematic model, the linear and angular velocities of the underwater hexapod robot are related to the CPG parameters, and a conversion function is constructed through experimental fitting. Second, by integrating the output of the model predictive controller with the parameters of the CPG network via the conversion function, a novel CPG-based model predictive control approach (MPC–CPG controller) is proposed, in which the MPC serves as the planning layer to regulate the execution-layer CPG. Finally, the effectiveness of the proposed MPC–CPG controller in trajectory tracking is validated through simulation experiments.

The remaining part of this paper is arranged as follows: Section 2 presents the mathematical model of the underwater hexapod robot. Section 3 presents the detailed design framework of the MPC-CPG controller. The simulation and experimental verification were conducted in Section 4, and the conclusion is given in Section 5.

2. Underwater Hexapod Robot Design and Modeling

2.1. System Overall Design

The structure of the underwater hexapod robot studied in this paper is shown in Figure 1. The platform's hull is made of carbon fiber composite material, with main dimensions of 1.74 m × 1.07 m × 0.62 m and a weight of approximately 42.5 kg. The system is equipped with 6 mechanical legs, each containing 3 degrees of freedom (hip joint, knee joint, and ankle joint), all driven by independent servo motors. The core control system integrates an onboard computer running ROS and an STM32 microcontroller, responsible for sensor data fusion (e.g., depth sensor, IMU) and control command distribution. The main technical parameters of the platform are listed in Table 1.

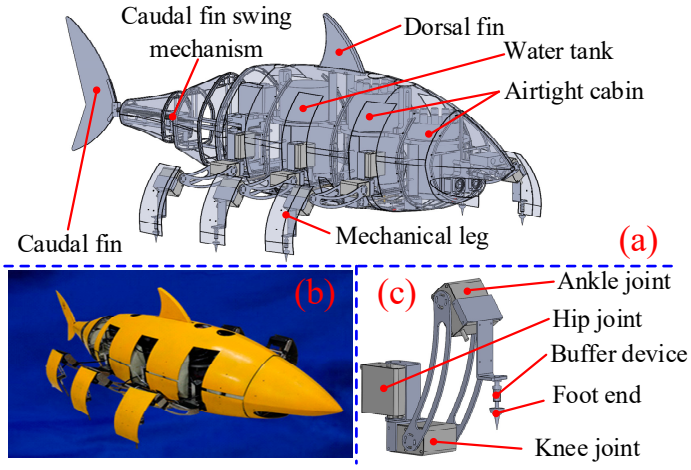


Figure 1. Model diagram and prototype of underwater hexapod robot. (**a**) the 3D model of the robot; (**b**) Robot prototype; (**c**) Mechanical leg structure.

Table 1. Underwater hexapod robot's technical parameters.

Item	Description
Size	1.74 m × 1.07 m × 0.62 m
Weight	42.5 kg
Control mode	Wireless or autonomous mode
Battery life	1.5 h
Voltage	24 V

2.2. Dynamics Modeling

Figure 2 defines the global fixed coordinate system and the body motion coordinate system of the robot. When performing dynamic analysis of the entire machine, the robot can be considered as a rigid body with uniformly distributed mass. Firstly, the translational motion under force and the rotational motion under torque are analyzed. Then, all external forces and moments acting on the robot are comprehensively analyzed, and its dynamic equations are derived by ensuring the balance of resultant forces and moments in all directions within the motion coordinate system. Referring to the method of establishing a six-degree-of-freedom dynamic model in the literature [37,38] and considering the force characteristics of the underwater hexapod robot, its simplified dynamic model is established as shown in Equation (1).

$$\begin{aligned} m(\dot{u} - vr - x_g r^2 - y_g \dot{r}) &= X_{\dot{u}}\dot{u} + X_{\dot{v}}\dot{v} + X_{\dot{r}}\dot{r} + X_u u + X_v v + X_r r + X_{|u|u}\left|u\right|u + X_{|v|v}\left|v\right|v + X_{|r|r}\left|r\right|r \\ &\quad -(W - B)\sin\theta + F_{Lx} + F_{Rx} \\ m(\dot{v} + ur - y_g r^2 + x_g \dot{r}) &= Y_{\dot{u}}\dot{u} + Y_{\dot{v}}\dot{v} + Y_{\dot{r}}\dot{r} + Y_u u + Y_v v + Y_r r + Y_{|u|u}\left|u\right|u + Y_{|v|v}\left|v\right|v + Y_{|r|r}\left|r\right|r \\ &\quad +(W - B)\cos\theta\sin\varphi + F_{Ly} + F_{Ry} \\ I_z\dot{r} + m[x_g(\dot{v} + ur) - y_g(\dot{u} - vr)] &= N_{\dot{u}}\dot{u} + N_{\dot{v}}\dot{v} + N_{\dot{r}}\dot{r} + N_u u + N_v v + N_r r + N_{|u|u}\left|u\right|u + N_{|y|v}\left|v\right|v \\ &\quad +N_{|r|r}\left|r\right|r + (x_g W - x_b B)\cos\theta\sin\varphi + (y_g W - y_b B)\sin\theta + (F_L - F_R)k \end{aligned} \tag{1}$$

where W and B represent the gravity and buoyancy experienced by the underwater hexapod robot, respectively. In the motion coordinate system, the center of gravity coordinate is (x_g, y_g, z_g), the center of buoyancy coordinate is (x_b, y_b, z_b). F_L and F_R are the forces applied by the foot ends of the left and right mechanical legs, respectively, k represents the perpendicular distance from the foot force action line to the body centerline, and $X_{(\cdot)}, Y_{(\cdot)}, N_{(\cdot)}$ are the hydrodynamic coefficients corresponding to the three degrees of freedom directions.

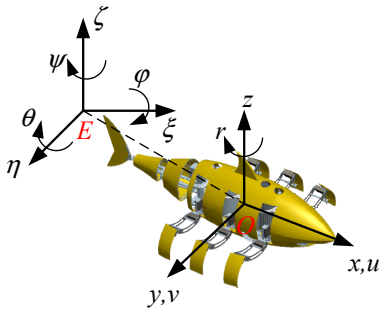


Figure 2. Fixed coordinate system and motion coordinate system.

Figure 3 shows the mechanism diagram of a single mechanical leg. The base coordinate system origin is set at the hip joint, the positive x-axis points in the forward direction of the robot, the z-axis is vertically upward, and the y-axis direction is determined according to the right-hand rule. The leg coordinate system established based on this is shown in Figure 3.

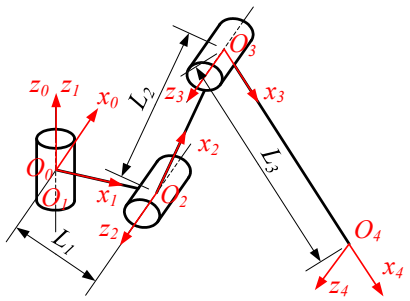


Figure 3. Mechanism diagram of the mechanical leg.

The homogeneous transformation matrix from the foot-end coordinate system to the base coordinate system is

$$ {}^0_4T = {}^0_1T \cdot {}^1_2T \cdot {}^2_3T \cdot {}^3_4T = \begin{bmatrix} c_1c_{23} & -c_1s_{23} & s_1 & L_1c_1 + L_2c_1c_2 + L_3c_1c_{23} \\ s_1c_{23} & -s_1s_{23} & -c_1 & L_1s_1 + L_2s_1c_2 + L_3s_1c_{23} \\ s_{23} & c_{23} & 1 & L_2s_2 + L_3s_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

Based on this, the forward kinematics solution from the foot end to the base coordinate system is

$$\begin{cases} x_0 = L_1c_1 + L_2c_1c_2 + L_3c_1c_{23} \\ y_0 = L_1s_1 + L_2s_1c_2 + L_3s_1c_{23} \\ z_0 = L_2s_2 + L_3s_{23} \end{cases} \tag{3}$$

where x_0, y_0, z_0 represent the position coordinates of the foot end point in the base coordinate system. Differentiating both sides of the above equation yields

$$\begin{bmatrix} \dot{x}_0 \\ \dot{y}_0 \\ \dot{z}_0 \end{bmatrix} = J \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \tag{4}$$

where J is the Jacobian matrix.

$$J = \begin{bmatrix} -(L_1s_1 + L_2s_1c_2 + L_3s_1c_{23}) & -(L_2c_1s_2 + L_3c_1s_{23}) & -L_3c_1s_{23} \\ L_1c_1 + L_2c_1c_2 + L_3c_1c_{23} & -(L_2s_1s_2 + L_3s_1s_{23}) & -L_3s_1s_{23} \\ 0 & L_2c_2 + L_3c_{23} & L_3c_{23} \end{bmatrix} \tag{5}$$

Given the foot-end force as $\boldsymbol{F} = [F_x, F_y, F_z]^{\mathrm{T}}$, the output torque of each joint $\tau = [\tau_1, \tau_2, \tau_3]^{\mathrm{T}}$ can be solved using the transpose of the Jacobian matrix

$$\boldsymbol{\tau} = \boldsymbol{J}^T \boldsymbol{F}_R \tag{6}$$

3. MPC-CPG Controller Design Based on Dynamics

3.1. Motion Modeling Based on CPG

Inspired by bionic principles and simulating biological central pattern generators, researchers have developed various mathematical models of CPG oscillators, commonly including recurrent neural oscillators, phase oscillators, and Hopf oscillators [39]. This paper selects the Hopf oscillator as the basic unit of the CPG network to construct the underlying motion control network. The mathematical model of the CPG network is expressed as follows

$$\begin{aligned} \dot{x}_i &= -\omega_i y_i + kx_i\left(H_i^2 - x_i^2 - y_i^2\right) + h_i(x_{i-1}\cos\varphi_i + y_{i-1}\sin\varphi_{i,i-1}) \\ \dot{y}_i &= \omega_i x_i + ky_i\left(H_i^2 - x_i^2 - y_i^2\right) + h_{i+1}(x_{i+1}\sin\varphi_{i+1} + y_{i+1}\cos\varphi_{i,i+1}) \end{aligned} \tag{7}$$

where x_i and y_i represent the state variables of the i-th oscillator, respectively. ω_i and H_i represent the oscillation frequency and amplitude of the i-th oscillator, respectively. h_i and φ_i represent the coupling factor and phase difference in the i-th oscillator, respectively, i = 1, 2, . . ., 6.

To analyze the existence and stability of the limit cycle of the Hopf oscillator, the following Lyapunov function is introduced:

$$V(x, y) = \frac{1}{2}(x^2 + y^2) \tag{8}$$

Taking the time derivative yields:

$$\dot{V} = x\dot{x} + y\dot{y} \tag{9}$$

Substituting the Hopf oscillator dynamics gives:

$$\dot{V} = \mu(x^2 + y^2) - (x^2 + y^2)^2 \tag{10}$$

It can be observed that when $\mu < 0$, $\dot{V} < 0$ holds for all $(x, y) \neq (0, 0)$, and the origin is a stable focus. When $\mu > 0$, the origin becomes an unstable focus and trajectories are repelled from the origin.

Let $r^2 = x^2 + y^2$. For sufficiently large r, $\dot{V} < 0$ holds, which implies that system trajectories are bounded and form a trapping region. According to the Bendixson criterion, there exists at least one limit cycle in this region. Furthermore, based on the Chepkac theorem, the system admits at most one stable limit cycle. Therefore, for $\mu > 0$, the Hopf oscillator possesses a unique and stable limit cycle, and all trajectories converge asymptotically to the same periodic orbit regardless of initial conditions.

This property guarantees that the CPG network can generate stable and robust rhythmic signals for locomotion control.

The CPG network structure constructed through coupling terms is shown in Figure 4. The output signals generated by the CPG network are appropriately scaled by mapping functions and then sent to the hip, knee, and ankle joints of the robot's six mechanical legs to drive coordinated motion of each joint. The upper-level controller adjusts the output of the CPG network in real time to change the foot-end force, ultimately achieving the trajectory tracking control objective.

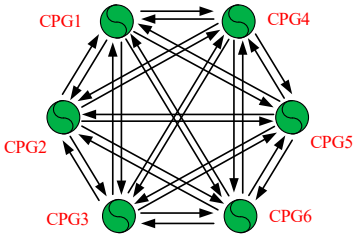


Figure 4. Crawling CPG network.

The robot's steering motion during crawling is achieved by adjusting the swing frequency and amplitude of the left and right mechanical legs to create a speed difference. To facilitate control of the speed difference between the two sides, a steering parameter $\delta \in [0, \pi/2]$ is introduced, and the parameter δ and the stride length A_i are set to satisfy the following relationship

$$A_i = \begin{cases} A_{max} \sin \delta, i = 1, 2, 3. \\ A_{max} \cos \delta, i = 4, 5, 6. \end{cases} \tag{11}$$

where A_{max} represents the maximum stride length of the mechanical leg, $A_{max} = L \sin H_{max}$, H_{max} is the maximum amplitude output by the CPG network, L is the projection length of the mechanical leg on the ground. Given that H_{max} is relatively small, to simplify the control algorithm, it can be approximated that $A_{max} = LH_{max}$. i represents the number of the mechanical leg, consistent with the oscillator number, $i = 1, 2, 3$ corresponds to the left legs, $i = 4, 5, 6$ corresponds to the right legs.

The velocities of the left and right legs are

$$\begin{aligned} v_L &= A_{max} f \sin \delta \\ v_R &= A_{max} f \cos \delta \end{aligned} \tag{12}$$

where v_L is the velocity of the left legs, v_R is the velocity of the right legs, and $f = \omega/2\pi$ is the swing frequency of the mechanical legs.

The velocity difference between the two sides Δv is

$$\Delta v = v_R - v_L = \sqrt{2} A_{max} f \sin(\frac{\pi}{4} - \delta) \tag{13}$$

Thus, it can be seen that when $\delta < \pi/4$, $\Delta v > 0$, the robot turns left; when $\delta > \pi/4$, $\Delta v < 0$, the robot turns right; when $\delta = \pi/4$, $\Delta v = 0$, the robot moves straight.

The moving velocity of the robot's center is

$$v = \frac{(v_R + v_L)}{2} = \frac{\sqrt{2}}{2} A_{max} f \sin(\frac{\pi}{4} + \delta) \tag{14}$$

As shown in Figure 5, when a velocity difference is generated between the left and right legs, the underwater hexapod robot begins to turn.

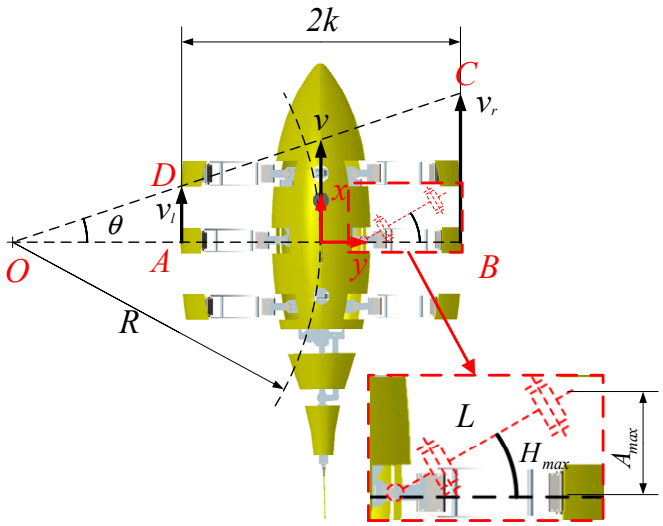


Figure 5. Speed of the underwater hexapod robot.

Where A_{max} is the maximum stride length of the mechanical leg, 2k is the width of the robot, θ is the yaw angle of the body. The angular velocity of the robot can be expressed as

$$\dot{\theta} = \omega = \frac{v}{R} = \frac{v_L - v_R}{2k} = \frac{\sqrt{2}}{2k} A_{max} f \sin(\frac{\pi}{4} - \delta) \tag{15}$$

Thus, the motion model of the underwater hexapod robot based on CPG is

$$\begin{cases} v = \frac{\sqrt{2}}{2} A_{max} f \sin(\frac{\pi}{4} + \delta) \\ \omega = \frac{\sqrt{2}}{2k} A_{max} f \sin(\frac{\pi}{4} - \delta) \end{cases} \tag{16}$$

3.2. MPC-CPG Controller Design

The motion control of the underwater hexapod robot is a typical nonlinear problem with multiple constraints. During trajectory tracking, frequent adjustment of the crawling gait is often required to ensure tracking accuracy. Therefore, this paper uses a CPG network as the underlying motion controller to ensure smoothness and coordination of the motion process; simultaneously, an upper-level Model Predictive Controller is introduced to optimize and adjust the parameters of the CPG network in real time to complete complex trajectory tracking tasks. Model Predictive Control has advantages such as intuitive modeling, strong robustness, good stability, and is particularly adept at handling multi-constraint optimization problems. It has been widely applied to motion control of various robots with good results.

Modeling the robot's crawling motion as a control system with input $\boldsymbol{u}(F_L, F_R)$ and state quantity $\boldsymbol{\chi}(x, y, r)$, its general form is

$$\dot{\boldsymbol{\chi}} = f(\boldsymbol{\chi}, \boldsymbol{u}) \tag{17}$$

Referring to the method described in the literature [40], the linear error model of this system can be established as follows

$$\dot{\tilde{\chi}} = \dot{\chi} - \dot{\chi}_r = \boldsymbol{A}(t)\dot{\tilde{\chi}} + \boldsymbol{B}(t)\dot{\tilde{u}} \tag{18}$$

where χ_r is the state quantity corresponding to the reference trajectory, $\chi_r = \begin{bmatrix} x_r & y_r & r_r \end{bmatrix}^T$, $\boldsymbol{u}_r = \begin{bmatrix} F_{Lr} & F_{Rr} \end{bmatrix}^T$, $\dot{\chi}_r = f(\chi_r, \boldsymbol{u}_r)$.

Discretizing the above continuous model yields the discrete-time linear error model

$$\tilde{\chi}(k+1) = \boldsymbol{A}_{k,t}\tilde{\chi}(k) + \boldsymbol{B}_{k,t}\tilde{\boldsymbol{u}}(k) \tag{19}$$

where $\boldsymbol{A}_{k,t} = I + T\boldsymbol{A}(t)$, $\boldsymbol{B}_{k,t} = T\boldsymbol{B}(t)$, T is the sampling time.

The objective function is established as follows

$$J(k) = \sum_{i=1}^{N_\mathrm{p}} ||\boldsymbol{\eta}(k+i|t) - \boldsymbol{\eta}_\mathrm{r}(k+i|t)||_Q^2 + \sum_{i=1}^{N_c-1} ||\Delta \boldsymbol{U}(k+i|t)||_R^2 \tag{20}$$

where $\mathrm{N_p}$ is the prediction horizon, $\mathrm{N_c}$ is the control horizon.

Converting Equation (13) into the form of a state space is

$$\begin{aligned} \boldsymbol{\xi}(k+1|t) &= \widetilde{\boldsymbol{A}}_{k,t}\boldsymbol{\xi}(k|t) + \widetilde{\boldsymbol{B}}_{k,t}\Delta \boldsymbol{U}(k|t) \\ \boldsymbol{\eta}(k|t) &= \widetilde{\boldsymbol{C}}_{k,t}\boldsymbol{\xi}(k|t) \end{aligned} \tag{21}$$

where $\widetilde{\boldsymbol{A}}_{k,t} = \begin{bmatrix} \boldsymbol{A}_{k,t} & \boldsymbol{B}_{k,t} \\ \boldsymbol{0}_{m\times n} & \boldsymbol{I}_m \end{bmatrix}$, $\widetilde{\boldsymbol{B}}_{k,t} = \begin{bmatrix} \boldsymbol{B}_{k,t} \\ \boldsymbol{I}_m \end{bmatrix}$, n is the dimension of the state quantity, and m is the dimension of the control quantity.

After derivation, the predicted output expression of the system is obtained

$$\boldsymbol{Y}(t) = \boldsymbol{\Psi}_t\boldsymbol{\xi}(t|t) + \boldsymbol{\Theta}_t\Delta \boldsymbol{U}(t) \tag{22}$$

It can be seen from Equation (16) that the predicted output is calculated based on the current state $\boldsymbol{\xi}(t|t)$ and control input.

The expression form of the control quantity constraint and the control increment constraint during the movement of the underwater hexapod robot are as follows

$$\begin{matrix} \boldsymbol{u}_{\min}(t+k) \leqslant \boldsymbol{u}(t+k) \leqslant \boldsymbol{u}_{\max}(t+k) \\ \Delta \boldsymbol{u}_{\min}(t+k) \leqslant \Delta \boldsymbol{u}(t+k) \leqslant \Delta \boldsymbol{u}_{\max}(t+k) \end{matrix} , k = 0, 1, \ldots, N_c - 1 \tag{23}$$

The stability constraint (Terminal equality constraint) is as follows

$$\left|\left|\boldsymbol{\eta}(k+N_p\middle|t) - \boldsymbol{\eta}_r(k+N_p\middle|t)\right|\right|_Q^2 = 0 \tag{24}$$

Combine the constraints into a matrix form as

$$\boldsymbol{U}_{\min} \leqslant \boldsymbol{K}\Delta \boldsymbol{U}_t + \boldsymbol{U}_t \leqslant \boldsymbol{U}_{\max} \tag{25}$$

where $\boldsymbol{U}_{\min}$, $\boldsymbol{U}_{\max}$ are the minimum and maximum values of the control variable in the control horizon. $\boldsymbol{U}_t = \boldsymbol{1}_{N_c} \otimes \boldsymbol{u}(k-\boldsymbol{1})$, $\boldsymbol{1}_{N_c}$ is the column vector with the number of rows $\mathrm{N_c}$, and $\otimes$ is the Kronecker product.

$$\boldsymbol{K} = \begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & 1 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \dots & 1 \end{bmatrix}_{N_c \times N_c} \otimes \boldsymbol{I}_m \tag{26}$$

Substituting Equation (16) into Equation (14), Equation (14) is solved under the constraints of Equation (19). When the terminal conditions are met, the entire optimization process is completed and the global optimal solution is obtained, $\Delta \boldsymbol{U}_t^* = [\Delta \boldsymbol{u}_t^*, \Delta \boldsymbol{u}_{t+1}^*, \dots, \Delta \boldsymbol{u}_{t+N_c-1}^*]^T$. According to the MPC principle, the first element of the control increment sequence is used as the actual control increment to act on the system, that is, $\boldsymbol{u}(t) = \boldsymbol{u}(t-1) + \Delta \boldsymbol{u}_t^*$.

Since the model predictive controller directly outputs the foot-end forces F_L, F_R, which cannot be directly used to adjust the CPG, a conversion function needs to be designed to map the MPC output to CPG parameters.

From Equation (7), it can be seen that both the CPG frequency f and the steering parameter δ can affect the robot's crawling speed and angular velocity. Therefore, the conversion function to be designed relates the functional relationship between the foot-end forces F_L, F_R and f, δ. As shown in Figure 6, the relationship between the hip joint torque τ_1 during robot crawling and f, δ was obtained through multiple experiments, and the experimental data were fitted.

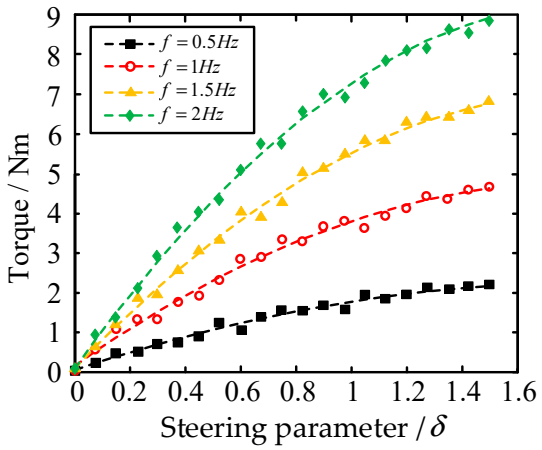


Figure 6. The relationship between hip joint torque τ_1 and f, δ.

From Figure 6, it can be seen that as f and δ increase, the output torque of the left leg's hip joint also increases. To simplify the calculation, during the robot's crawling control, the CPG frequency is fixed at $f = 1$ Hz. Therefore, the crawling trajectory tracking control of the robot can be achieved simply by adjusting the steering parameters δ.

When $f = 1$ Hz, the relationship between torque and steering parameter δ obtained by data fitting is

$$\tau_1 = -1.15\delta^2 + 4.65\delta + 0.15 \tag{27}$$

Combining the fitting function Equation (21) and the mechanical leg dynamics model Equation (6) yields the conversion function

$$\boldsymbol{F} = -1.15\delta^2 + 4.65\delta + 0.15(\boldsymbol{J}^T)^{-1} \tag{28}$$

By combining the model predictive controller with CPG through the transformation function, the control block diagram of the constructed MPC-CPG controller is shown in Figure 7. The model predictive controller calculates the desired foot-end force for the

next time step based on the error between the current reference trajectory and the actual trajectory. Then, the required torque for each joint is calculated through the mechanical leg dynamics model. Next, the conversion function is used to convert the calculated hip joint torque into the steering parameter δ, which is input into the CPG network for parameter adjustment. Finally, the CPG outputs rhythmic control signals for each joint. This process repeats until the entire trajectory tracking task is completed.

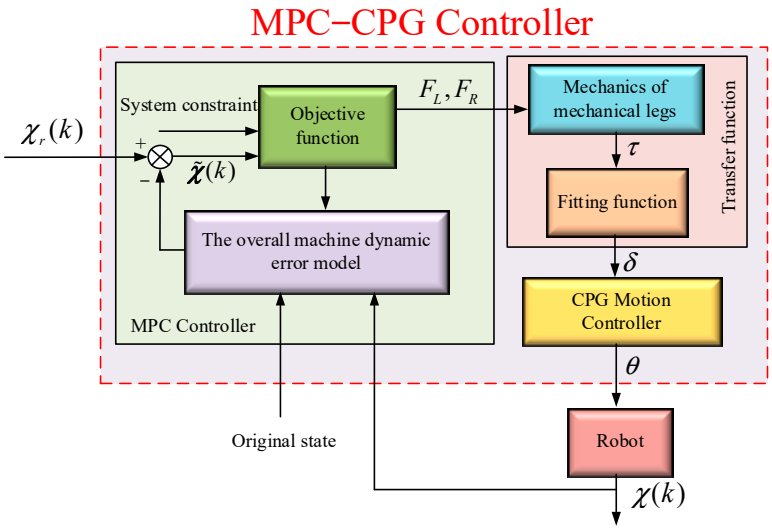


Figure 7. Control block diagram of the MPC-CPG controller.

4. Simulation Experiment

To comprehensively evaluate the adaptability and robustness of the MPC-CPG control architecture, simulation experiments were conducted for the bionic crawling characteristics of underwater hexapod robots, with a focus on testing key indicators such as the dynamic response characteristics and control accuracy of the system in circular and linear trajectory tracking tasks. The equation of the circular trajectory is $(x-2.5)^2+(y-3)^2=6.25$, the speed reference is $v=0.1$ m/s, $\omega=0.04$ rad/s. The linear trajectory equation is $y=3$, the speed reference is $v=0.1$ m/s, $\omega=0$ rad/s. The parameter settings of MPC and CPG are shown in Table 2.

Table 2. Parameter setting.

Parameter		Numerical Value
MPC parameter	Prediction horizon N	20
	Sampling time T	0.05
	Constraint condition	$\begin{bmatrix}0\\0\end{bmatrix} \leqslant \begin{bmatrix}F_L\\F_R\end{bmatrix} \leqslant \begin{bmatrix}15\\15\end{bmatrix}$ $\begin{bmatrix}-0.05\\-0.05\end{bmatrix} \leqslant \begin{bmatrix}\Delta F_L\\\Delta F_R\end{bmatrix} \leqslant \begin{bmatrix}0.05\\0.05\end{bmatrix}$
CPG parameter	Amplitude H_i	20
	Phase difference ϕ_i	π
	Coupling factor h_i	0.5
	Convergence factor k	15

The simulation results of circular trajectory tracking are shown in Figure 8. In terms of dynamic response characteristics, the system exhibits good transient performance under the MPC controller. The position error converges rapidly in the initial stage, entering the

allowable error range within 15 s. The position and heading angle errors exhibit exponential decay characteristics during tracking, with a decay time of 3.2 s, indicating good damping characteristics of the system. The error analysis indicates that after reaching the steady state, the position tracking error is strictly controlled within ±0.17 m (9.8% of the body length), and the heading angle error remains within the range of 15°. Compared with sliding mode control, MPC control has a faster convergence speed and higher tracking accuracy. The position control accuracy fully meets the operational requirements in complex environments.

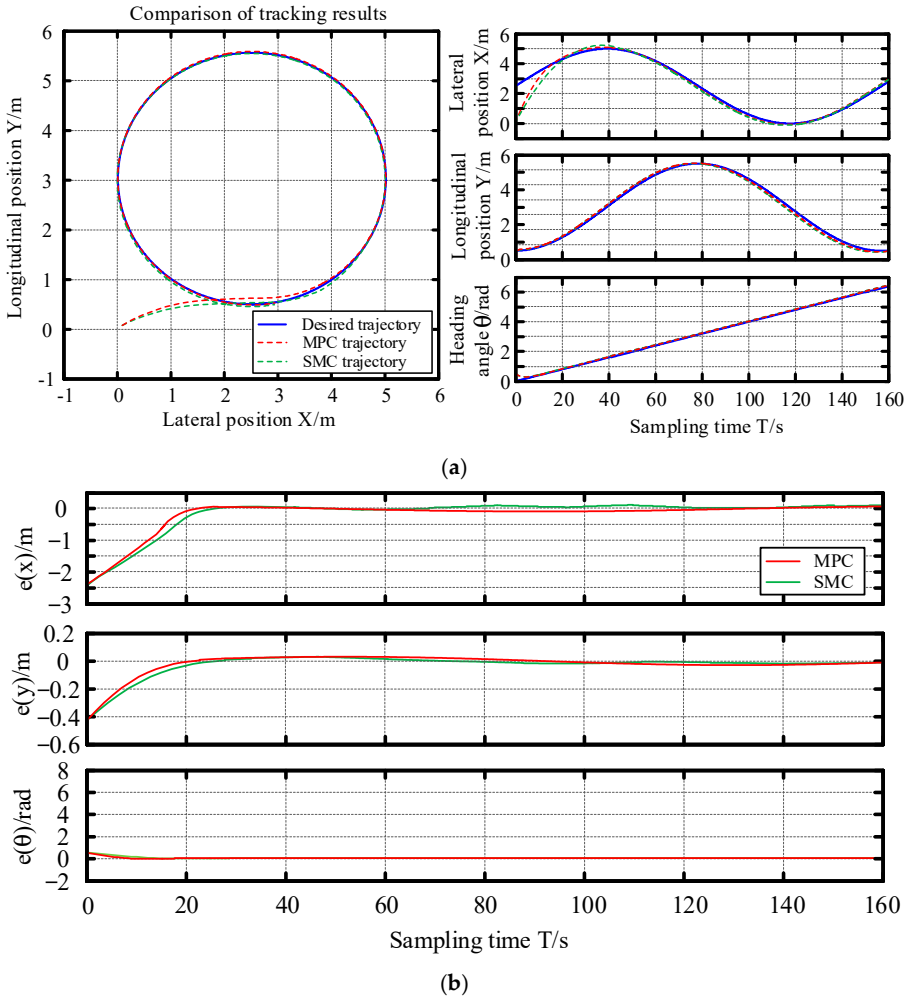


Figure 8. Simulation results of circular trajectory tracking: (**a**) Trajectory tracking; (**b**) Tracking error.

In terms of control allocation, Figure 9 reveals the foot-end force allocation strategy optimized by the MPC. In the initial stage of motion, the controller adopted an asymmetric force allocation scheme: the right foot-end force first decreased and then increased, reaching a minimum of 8.76 N at t = 13 s. It then increased rapidly, peaking at 11.1 N at t = 25 s and stabilizing, while the left foot-end force showed a symmetric change process opposite to the right, eventually converging with the right force value to a steady state. This force allocation strategy optimally corrected the initial heading deviation while ensuring system motion stability, demonstrating the advantage of MPC in handling constraints.

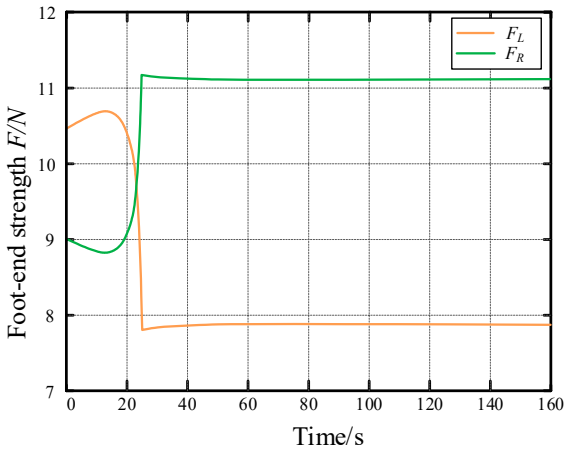


Figure 9. MPC controller output of circular trajectory tracking.

By mapping the control quantities output by the MPC to the CPG parameter space using the designed conversion function, the variation process of the steering parameter shown in Figure 10 was obtained. The parameter change shows obvious three-stage characteristics: in the initial adjustment stage from 0 to 13 s, the steering parameter rapidly increased from 45° to 52.4° to generate sufficient steering torque. In the fine adjustment stage from 13 to 23 s, the parameter is adaptively adjusted within the range of 34–52.4° to achieve precise tracking of the trajectory curvature. In the steady-state stage after 23 s, the parameter stabilized around 34°, where the system entered a constant-curvature circular motion mode and the motion state reached stability.

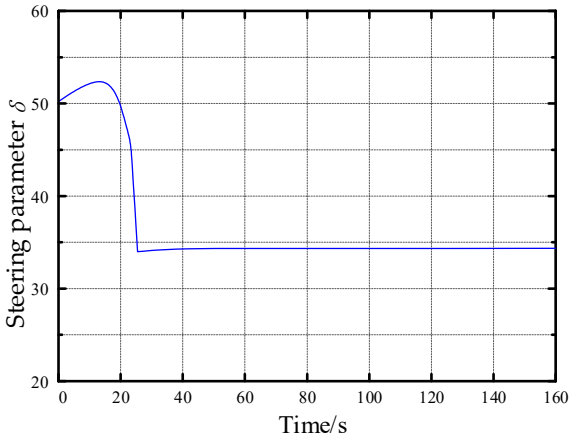


Figure 10. Steering parameter of circular trajectory tracking.

The output of the CPG network is shown in Figure 11. Analysis of the CPG network output characteristics reveals the superiority of the underlying control system. As shown, the system only requires 1.5 s to complete the transition from transient to steady state, establishing stable phase-locked relationships among the oscillators. The amplitude of the left hip joint oscillator finally stabilized at 17°, while the right stabilized at 25°. The amplitude difference between the two sides provided precise steering torque for the system, enabling the underwater hexapod robot to complete trajectory tracking at a speed of 0.1 m/s and an angular velocity of 0.04 rad/s. It is particularly worth emphasizing that although the upper-level MPC control quantity fluctuated by over 25% within 10 s, after the threshold limiting and smoothing effect of the CPG, the actual control signals for the actuators changed smoothly without jumps, significantly improving the service life of the actuators.

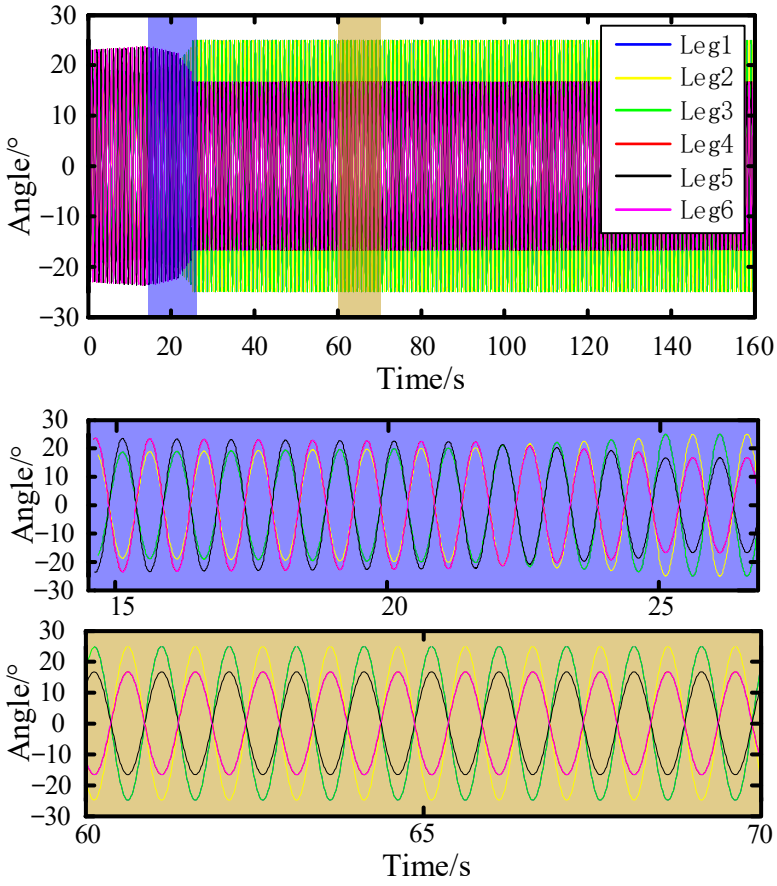
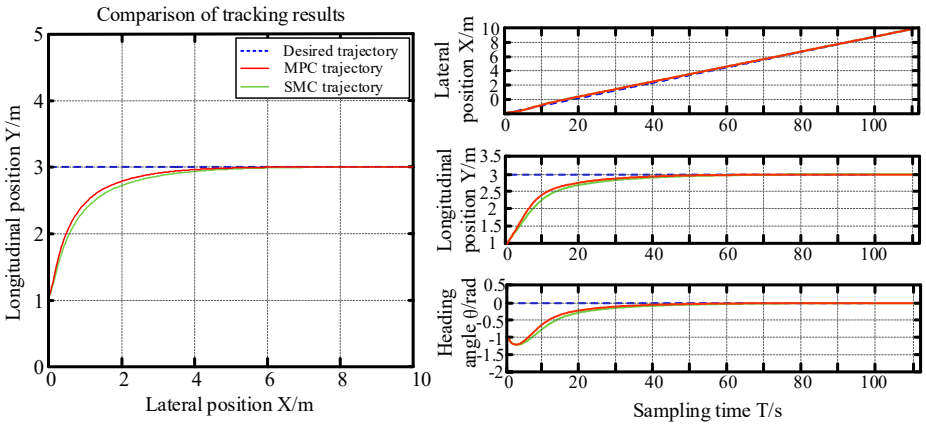


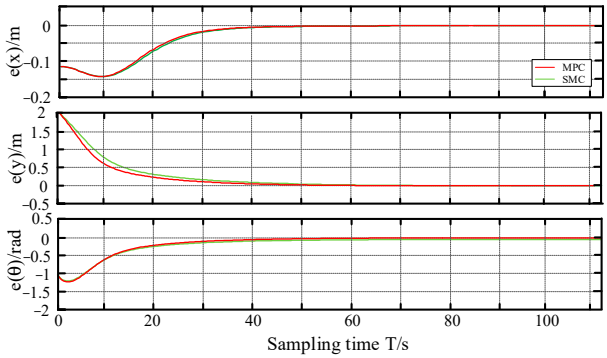
Figure 11. CPG output of circular trajectory tracking.

The linear trajectory tracking results are shown in Figure 12. Under MPC control, the system converges rapidly, and the position error enters the allowable range within 50 s. After reaching steady state, the tracking error is maintained within 0.12 m (6.9% of the body length), while the heading error remains within ±5°, which satisfies the operational requirements. Compared with sliding mode control, the MPC-based controller exhibits faster convergence speed and higher tracking accuracy.



(**a**)

Figure 12. *Cont*.



(**b**)

Figure 12. Simulation results of linear trajectory tracking: (**a**) Trajectory tracking; (**b**) Tracking error.

Figures 13 and 14 show that during the first 50 s, under MPC regulation, the steering parameter rapidly increases from 45° to 78° and then gradually decreases back to 45°. Meanwhile, the left foot-end force quickly rises to a peak value of 16.3 N and then decreases to 9.5 N, whereas the right foot-end force drops to a minimum of 2.8 N before increasing to 9.5 N. Through this asymmetric force distribution, the robot executes a right-turn maneuver to eliminate the initial position error. After 50 s, the foot-end forces on both sides stabilize at 9.5 N, and the robot successfully completes linear trajectory tracking.

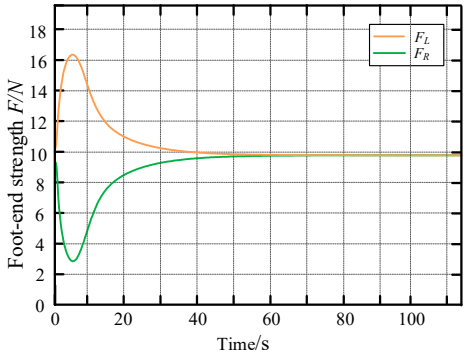


Figure 13. MPC controller output of linear trajectory tracking.

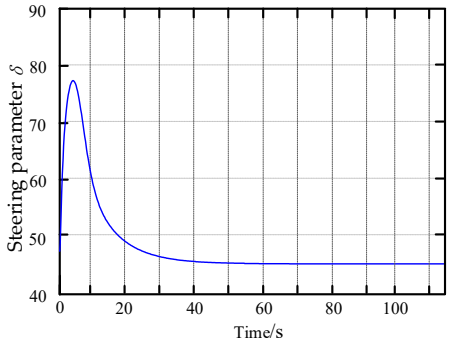


Figure 14. Steering parameter of linear trajectory tracking.

Figure 15 demonstrates the excellent performance of the lower-level control system. The system requires only 1.2 s to transition from the transient state to steady operation. Owing to the threshold limiting and smoothing effects of the CPG network, the actual actuator control signals vary smoothly without abrupt changes, and the maximum joint angular velocity is constrained within 30°/s, significantly improving actuator service life.

Ultimately, the underwater hexapod robot completes linear trajectory tracking at a stable velocity of 0.1 m/s.

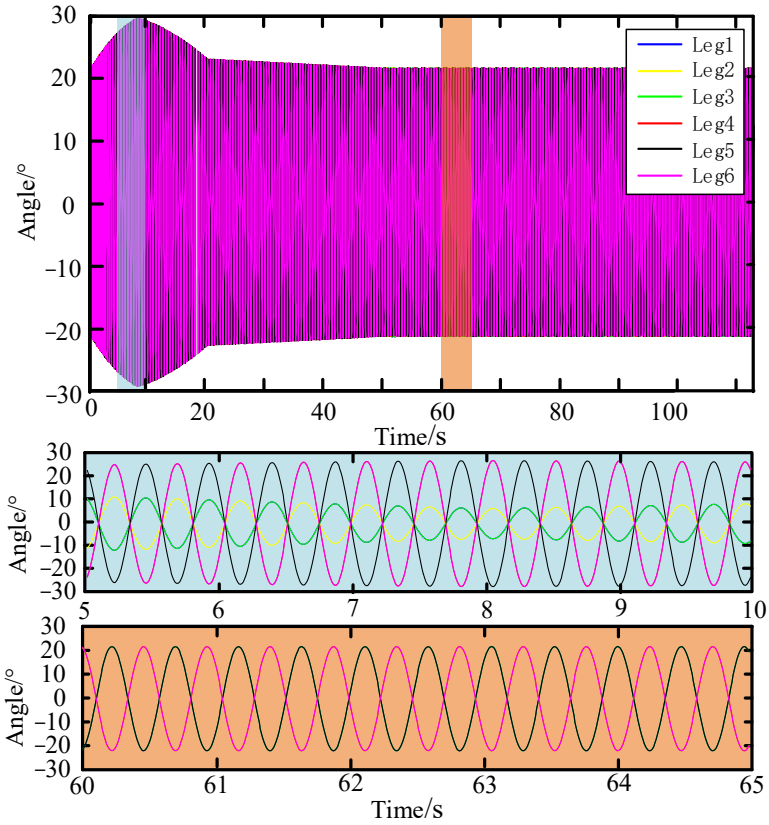


Figure 15. CPG output of linear trajectory tracking.

5. Conclusions

This paper proposed and validated an MPC-CPG hierarchical control architecture, achieving crawling trajectory tracking for an underwater hexapod robot in complex environments by establishing a control link from the decision layer to the execution layer. The MPC controller at the decision layer, based on the system's dynamic model, not only effectively handled practical issues such as system state constraints and actuator saturation but also achieved online compensation for uncertainties through a rolling optimization mechanism. By designing a conversion function as the transition layer, a bridge was established between high-level decision-making and low-level execution, transforming the optimization results of the MPC into the rhythmic parameters of the CPG. This not only ensured accurate transmission of control intent but also achieved smooth switching between different gaits through a parameter scheduling mechanism. The CPG motion control network, as the execution layer, provided the system with stable rhythmic motion patterns, achieving dynamic motion reconstruction through online parameter adjustment, significantly enhancing the naturalness and adaptability of the motion while ensuring system stability. Experiments showed that under conditions with external disturbances and model uncertainties, the MPC-CPG trajectory tracking controller maintained good control performance, with position tracking error less than 0.17 m and heading angle tracking error less than 15°.

6. Limitations and Outlook

Although the proposed MPC–CPG controller demonstrates good trajectory tracking performance, it still has certain limitations. First, when the prediction horizon is extended

or the model dimension is increased, the computational burden of MPC may impose challenges for real-time implementation. In addition, since the controller relies on a simplified dynamic model and an experimentally fitted parameter mapping, its performance may be affected under strong underwater turbulence or hydrodynamic uncertainties. Furthermore, the adopted dimensionality reduction strategy, which maps high-dimensional leg dynamics into a single steering parameter, inevitably reduces the control degrees of freedom and may neglect part of the leg dynamics. As a result, the proposed method is more suitable for low-speed, quasi-steady motion scenarios with planar trajectory tracking objectives.

Future work will focus on enhancing the adaptability and robustness of the control framework. Adaptive learning mechanisms will be introduced to improve model robustness, and multi-sensor fusion techniques will be incorporated to enhance state perception capability. Moreover, multi-parameter coordinated modulation of the CPG will be explored by jointly optimizing the phase, frequency, and amplitude of the oscillators in addition to the steering parameter, thereby enabling higher-dimensional gait regulation and stronger environmental adaptability.

Author Contributions: Writing, Conceptualization and Resources, R.L.; Conceptualization, Methodology and Resources, J.Z.; Writing and Visualization, M.S.; Software and Visualization, X.G.; Visualization, S.F.; Writing—Original Draft, D.Z.; Methodology and Conceptualization, H.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (No. 52105011), Youth S&T Talent Support Programme of Guangdong Provincial Association for Science and Technology (GDSTA) (No. SKXRC2025415), National Natural Science Foundation of China (No. 52305014), Guangdong Provincial Education Science Planning Project (No. 2025GXJK0505).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors have no conflicts of interest to declare that are relevant to the content of this article.

References

1. Sun, T.; Dai, Z.; Manoonpong, P. Distributed-force-feedback-based reflex with online learning for adaptive quadruped motor control. *Neural Netw.* **2021**, *142*, 410–427. [CrossRef] [PubMed]
2. Qiu, C.L.; Wu, Z.X.; Wang, J.; Tan, M.; Yu, J.Z. Locomotion Optimization of a Tendon-Driven Robotic Fish with Variable Passive Tail Fin. *IEEE Trans. Ind. Electron.* **2023**, *70*, 4983–4992. [CrossRef]
3. Chen, G.; Han, Y.; Li, Y.H.; Shen, J.T.; Tu, J.J.; Yu, Z.C.; Zhang, J.R.; Cheng, H.; Zhu, L.Y.; Dong, F. Autonomous gait switching method and experiments of a hexapod walking robot for Mars environment with multiple terrains. *Intell. Serv. Robot.* **2024**, *17*, 533–553. [CrossRef]
4. Xu, K.; Qin, R.P.; Chen, C.; Dong, G.Y.; Chen, J.W.; Ding, X.L. Design and multimodal locomotion plan of a hexapod robot with improved knee joints. *J. Field Robot.* **2024**, *41*, 1279–1296. [CrossRef]
5. Li, J.Y.; Yang, C.G.; Ding, L.; You, B.; Li, W.H.; Zhang, X.Y.; Gao, H.B. Trilateral shared control of a dual-user haptic tele-training system for a hexapod robot with adaptive authority adjustment. *IEEE Trans. Autom. Sci. Eng.* **2025**, *22*, 4366–4381. [CrossRef]
6. Hutter, M.; Gehring, C.; Lauber, A.; Gunther, F.; Bellicoso, C.D.; Tsounis, V.; Fankhauser, P.; Diethelm, R.; Bachmann, S.; Bloesch, M.; et al. Anymal-toward legged robots for harsh environments. *Adv. Robot.* **2017**, *31*, 918–931. [CrossRef]
7. Tao, Y.; Chen, S.; Liu, H.T.; Wan, J.H.; Wei, H.X.; Wang, T.M. Robot hybrid inverse dynamics model compensation method based on the BLL residual prediction algorithm. *Robotica* **2025**, *43*, 350–367. [CrossRef]
8. Fahmi, S.; Mastalli, C.; Focchi, M.; Semini, C. Passive whole-body control for quadruped robots: Experimental validation over challenging terrain. *IEEE Robot. Autom. Lett.* **2019**, *4*, 2553–2560. [CrossRef]
9. He, Y.; Xie, Y.; Pan, G. Depth and heading control of a manta robot based on S-plane control. *J. Mar. Sci. Eng.* **2022**, *10*, 1698. [CrossRef]
10. Meng, Y.; Wu, Z.; Chen, D. Development and 3-D path-following control of an agile robotic manta with flexible pectoral fins. *IEEE Trans. Cybern.* **2024**, *54*, 3227–3738. [CrossRef]

11. Tong, R.; Feng, Y.K.; Wang, J.; Wu, Z.X.; Tan, M.; Yu, J.Z. A survey on reinforcement learning methods in bionic underwater robots. *Biomimetics* **2023**, *8*, 168. [CrossRef] [PubMed]
12. Zhao, J.X.; Iwasaki, T. CPG Control for Harmonic Motion of Assistive Robot with Human Motor Control Identification. *IEEE Trans. Control. Syst. Technol.* **2020**, *28*, 1323–1336. [CrossRef]
13. Lu, Q.; Zhang, Z.C.; Yue, C. The programmable CPG model based on Matsuoka oscillator and its application to robot locomotion. *Int. J. Model. Simul. Sci. Comput.* **2020**, *11*, 2050018. [CrossRef]
14. Wang, B.R.; Zhang, K.; Yang, X.F.; Cui, X.H. The gait planning of hexapod robot based on CPG with feedback. *Int. J. Adv. Robot. Syst.* **2020**, *17*. [CrossRef]
15. Shu, B.M.; Guo, Y.Q.; Luo, W.H.; Xu, Z.D.; Xu, Q. Structure and Gait Design of a Lunar Exploration Hexapod Robot Based on Central Pattern Generator Model. *Actuators* **2024**, *13*, 79. [CrossRef]
16. Mao, Z.; Suzuki, S.; Nabae, H.; Miyagawa, S.; Suzumori, K.; Maeda, S. Machine learning-enhanced soft robotic system inspired by rectal functions to investigate fecal incontinence. *Bio-Des. Manuf.* **2025**, *8*, 482–494. [CrossRef]
17. Sun, T.; Xiong, X.; Dai, Z.; Manoonpong, P. Small-sized reconfigurable quadruped robot with multiple sensory feedback for studying adaptive and versatile behaviors. *Front. Neurorobotics* **2020**, *14*, 14. [CrossRef]
18. Gong, Q.S.; Zhang, W.; Su, Y.; Yang, H.Y. Guidance and control of underwater hexapod robot based on adaptive sliding mode strategy. *J. Bionic Eng.* **2025**, *22*, 118–132. [CrossRef]
19. Zhong, G.; Chen, L.; Jiao, Z.; Li, J.; Deng, H. Locomotion Control Gait Planning of a Novel Hexapod Robot Using Biomimetic Neurons. *IEEE Trans. Control. Syst. Technol.* **2018**, *26*, 624–636. [CrossRef]
20. Fukui, T.; Fujisawa, H.; Otaka, K.; Fukuoka, Y. Autonomous gait transition galloping over unperceived obstacles of a quadruped robot with CPGmodulated by vestibular feedback. *Robot. Auton. Syst.* **2018**, *111*, 1–19. [CrossRef]
21. Chen, X.; Wen, B.R.; Tian, X.L.; Sun, S.Y.; Wang, P.; Li, X. Reinforcement learning based CPG-controlled method with high adaptability and robustness: An experimental study on a robotic fishtail. *Ocean Eng.* **2023**, *289*, 116259. [CrossRef]
22. Liu, X.; Onal, C.D.; Fu, J. Integrating contact-aware CPG system for learning-based soft snake robot locomotion controllers. *IEEE Trans. Robot.* **2025**, *41*, 1581–1601. [CrossRef]
23. Chen, X.; Tian, X.L.; Chen, J.P.; Hu, Q.; Wen, B.R.; Liu, X.Y.; Chen, Y.; Tang, S.Y. A reinforcement learning-based control approach with lightweight feature for robotic fish heading control in complex environments: Real-world training. *Ocean Eng.* **2025**, *335*, 121667. [CrossRef]
24. Chen, G.; Jin, B.; Chen, Y. Accurate and robust body position trajectory tracking of six-legged walking robots with nonsingular terminal sliding mode control method. *Appl. Math. Model.* **2020**, *77*, 1348–1372. [CrossRef]
25. Li, B.; Gao, X.; Huang, H.; Yang, H. Improved adaptive twisting sliding mode control for trajectory tracking of an AUV subject to uncertainties. *J. Intell. Robot. Syst.* **2024**, *297*, 116204. [CrossRef]
26. Feng, Y.; Yu, X.; Man, Z. Non-singular terminal sliding mode control of rigid manipulators. *Automatica* **2002**, *38*, 2159–2167. [CrossRef]
27. Hu, Z.; Zhu, D.; Cui, C.; Sun, B. Trajectory Tracking and Re-planning with Model Predictive Control of Autonomous Underwater Vehicles. *J. Navig.* **2019**, *72*, 321–341. [CrossRef]
28. Zhou, L.; Wang, G.; Sun, K.; Li, X. Trajectory Tracking Study of Track Vehicles Based on Model Predictive Control. *Stroj. Vestn.-J. Mech. Eng.* **2019**, *65*, 329–342. [CrossRef]
29. Mera, M.; Rios, H.; Martinez, E.A. A sliding-mode based controller for trajectory tracking of perturbed Unicycle Mobile Robot. *Control. Eng. Pract.* **2020**, *102*, 104548. [CrossRef]
30. Huang, J.; An, H.; Yang, Y.; Wu, C.; Wei, Q.; Ma, H. Model Predictive Trajectory Tracking Control of Electro-Hydraulic Actuator in Legged Robot with Multi-Scale Online Estimator. *IEEE Access* **2020**, *8*, 95918–95933. [CrossRef]
31. Liu, X.; Wang, W.; Li, X.; Liu, F.; He, Z.; Yao, Y.; Ruan, H.; Zhang, T. MPC-based high-speed trajectory tracking for 4WIS robot. *ISA Trans.* **2021**, *123*, 413–424. [CrossRef] [PubMed]
32. Bruggemann, S.; Possieri, C. On the Use of Difference of Log-Sum-Exp Neural Networks to Solve Data-Driven Model Predictive Control Tracking Problems. *IEEE Control. Syst. Lett.* **2021**, *5*, 1267–1272. [CrossRef]
33. He, H.W.; Shi, M.; Li, J.W.; Cao, J.F.; Han, M. Design and experiential test of a model predictive path following control with adaptive preview for autonomous buses. *Mech. Syst. Signal Process.* **2021**, *157*, 107701. [CrossRef]
34. Zhu, J.; Ma, Y.; Yang, H.; Lv, C.; Zhang, Y.; Hao, S. Driver's Hand Trajectory-Guided Takeover Intention Inference in Conditionally Automated Driving. *IEEE Trans. Veh. Technol.* **2025**, *74*, 15331–15342. [CrossRef]
35. Tran, V.P.; Santoso, F.; Garratt, M.A.; Petersen, I.R. Fuzzy Self-Tuning of Strictly Negative-Imaginary Controllers for Trajectory Tracking of a Quadcopter Unmanned Aerial Vehicle. *IEEE Trans. Ind. Electron.* **2021**, *68*, 5036–5045. [CrossRef]
36. Zhu, J.; Lv, C.; Ma, Y.; Yang, H.; Zhang, Y. Quantitative estimation of driver cognitive workload: A dual-stage learning approach. *IEEE Trans. Intell. Transp. Syst.* **2024**, *25*, 20227–20239. [CrossRef]
37. Do, K.D.; Pan, J. *Control of Ships and Underwater Vehicles: Design for Underactuated and Nonlinear Marine Systems*; Springer: London, UK, 2009.

38. Yan, Z.; Hao, L.; Pi, Q.; Chen, T. Fractional-Order Sliding Mode Terrain-Tracking Control of Autonomous Underwater Vehicle with Sparse Identification. *Fractal Fract.* **2025**, *9*, 15. [CrossRef]
39. Yu, F.; Wu, Z.; Wang, J.; Yun, L.; Feng, Y.; Tan, M.; Yu, J. Learning from Fish: A Two-Stage Transfer Learning Method for a Bionic Robotic Fish. *IEEE Trans. Autom. Sci. Eng.* **2025**, *22*, 18796–18808. [CrossRef]
40. Li, S.; Wang, J.; Li, K. Soft Constraint Linear Model Predictive Control System Stability Method. *J. Tsinghua Univ. Nat. Sci. Ed.* **2010**, 1848–1852. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.