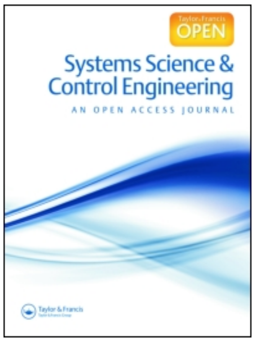

Systems Science & Control Engineering

An Open Access Journal

ISSN: (Print) 2164-2583 (Online) Journal homepage: https://www.tandfonline.com/loi/tssc20

Motion control research of underwater spherical robot

Peilong Li, Lei Wang & Wenwen Zhang

SYSTEMS SCIENCE & CONTROL ENGINEERING: AN OPEN ACCESS JOURNAL
2018, VOL. 6, NO. 1, 430–437
https://doi.org/10.1080/21642583.2018.1526136

Taylor & Francis
Taylor & Francis Group

OPEN ACCESS Check for updates

Motion control research of underwater spherical robot

Peilong Li[a], Lei Wang[b] and Wenwen Zhang[a]

[a]College of Electronic and Information Engineering, Tongji University, Shanghai, China; [b]Sino-German College for Graduate Studies, Tongji University, Shanghai, China

ABSTRACT

Underwater spherical robot motion control can be used to simulate satellite motion control under weightlessness. In this paper, firstly the motion equation of spherical robot is established by force analysis. Then, to solve the nonlinear and control channel coupling problems, the CMAC-PID control algorithm with neural network decoupling and genetic algorithm optimization is proposed. In the algorithm, CMAC-PID compound control is designed to generate control signals, and neural network decoupler is designed to decouple multi-channel control into single-channel control, and genetic algorithm is adopted to obtain the best PID parameters. The simulation results show that the GA can obtain the appropriate parameters quickly, and the decoupler can adapt to the change of the attitude of the controlled object. Compared with conventional PID control method, the proposed algorithm shows fast dynamic response and high steady-state accuracy, which can be used for reference of neutral buoyancy experimental object motion control.

ARTICLE HISTORY
Received 24 May 2018
Accepted 17 September 2018

KEYWORDS
Robotics; neural networks; feedback control; genetic algorithms

1. Introduction

Underwater motion control experimental simulation can be used to simulate weightlessness in space environment by submerging the tested object in the water completely and making the gravity of the object be equal with the buoyancy it subjected by accurately adjusted counterweight (Churchill, Akin, & Howard, 1993).

There have been numerous studies about modelling the experimental object. Most modelling takes Newton –Euler approach. Isa and Arshad (2013) established the motion equation of the underwater vehicle in the form of the vector on the basis of the Newton-Euler method, which is concise and common in form, and easy to carry out theoretical analysis. According to the characteristics of the autonomous underwater vehicle (short for AUV) with single vectored thruster. Wang et al. (2014) established the 6-DOF kinematic model and dynamic model of the AUV by Newton-Euler method. Isa and Arshad (2012) addressed mathematical model for a buoyancy-driven underwater glider on the basis of the AUV characteristics. And the recursive Newton-Euler approach is adopted to develop the direct dynamic function.

There have been numerous studies about controlling the experimental object. The underwater vehicle generally adopts PID control but also adopts other complicated control methods. Khodayari and Balochian (2015) combined adaptive methods and dual PID controllers to solve the uncertainty challenge in the PID parameters and AUV modelling uncertainty. Yang and Ma (2011) developed a nonlinear feedforward and feedback controller to improve the inherent efficiency. The method adopted a new stable inversion technique and a quadratic optimal control method. Keehong, Chung, and Slotine (2010) presented biologically inspired control strategies for an autonomous underwater vehicle propelled by flapping fins that resemble the paddle-like fore limbs of a sea turtle.

The control algorithm mentioned above is complicated, the calculation is not fast enough, and the delay effect can reduce the control precision. At the same time, the constant variation of the coupling degree between the hydrodynamic parameters and the propeller makes the above algorithms not ideal for adaptive performance.

To solve these problems, the multidimensional PID and Cerebellar Model Articulation Controller (short for CMAC) compound control algorithm, which is based on the dynamic decoupling and genetic algorithm, is proposed. In the algorithm, CMAC-PID compound control are used to generate control signals, and a two-layer neural network decoupler is used to decouple multichannel control into single-channel control, and genetic algorithm is used to optimize PID parameters.

CONTACT Peilong Li ✉ peilongli@qq.com

2. Dynamics model of the robot

2.1. Coordinate system and coordinate system transformation

As shown below in Figure 1, the origin of specified reference coordinate system $(E-\xi\eta\zeta)$ was designated in the zero point of the measurement coordinate system, with the $E\xi$, $E\eta$ and $E\zeta$ pointing to the geographical east, geographical south and geocentric, respectively. The moving coordinate system $(O-xyz)$ was fixed on the plant, and the origin O was the center of mass of the plant with Ox axis pointing to the right front of the plant, Oy axis pointing to the right side of the plant and Oz axis pointing to the bottom of the plant.

The fixed coordinate system is superimposed with the moving coordinate system after three times of rotation, so the formula of the coordinate system transformation is shown as follows:

$$\begin{bmatrix} \xi \\ \eta \\ \zeta \end{bmatrix} = \mathbf{T} \cdot \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \tag{1}$$

where the rotation matrix T is

$$\mathbf{T} = \begin{bmatrix} \cos\psi\cos\theta & \cos\psi\sin\theta\sin\varphi - \sin\psi\cos\varphi & \cos\psi\sin\theta\cos\varphi + \sin\psi\sin\varphi \\ \sin\psi\cos\theta & \sin\psi\sin\theta\sin\varphi + \cos\psi\cos\varphi & \sin\psi\sin\theta\cos\ \varphi - \cos\psi\sin\varphi \\ -\sin\theta & \cos\theta\sin\varphi & \cos\theta\cos\varphi \end{bmatrix}, \tag{2}$$

where φ is roll angle, θ is pitch angle and ψ is yaw angle.

2.2. Gravity and moment of gravity

The gravity and the moment of gravity of the plant are projected into the moving coordinate system, expressed as follow:

$$\begin{bmatrix} F_{\mathrm{G}} \\ M_{\mathrm{G}} \end{bmatrix} = G[\ -\sin\theta \quad \cos\theta \quad \sin\varphi \quad 0 \quad 0 \quad 0\ ]^{\mathrm{T}}, \tag{3}$$

where F_{G} is gravity vector of three directions in moving coordinate system, M_{G} is gravity moment vector of three directions in moving coordinate system, G is The gravity of the experimental plant, φ is roll angle, θ is pitch angle and ψ is yaw angle.

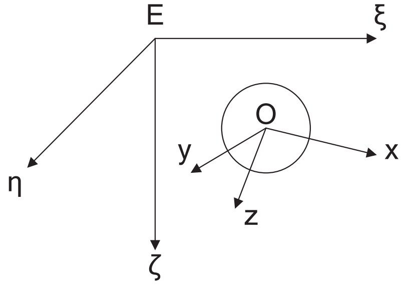

Figure 1. Fixed coordinate system and moving coordinate system.

2.3. Buoyancy and buoyancy moment

The buoyancy of the plant subjected is projected into the moving coordinate system, expressed as follow:

$$\begin{bmatrix} F_{\mathrm{B}} \\ M_{\mathrm{B}} \end{bmatrix} = -B \begin{bmatrix} -\sin\theta \\ \cos\theta\sin\varphi \\ \cos\theta\cos\varphi \\ z_b\cos\theta\sin\varphi - y_b\cos\theta\cos\varphi \\ x_b\cos\theta\cos\varphi + z_b\sin\theta \\ -y_b\sin\theta - x_b\cos\theta\sin\varphi \end{bmatrix}, \tag{4}$$

where F_{B} is buoyancy vector of three directions in moving coordinate system, M_{B} is buoyancy moment vector of three directions in moving coordinate system, B is the buoyancy the plant subjected, and the $(\ x_b \quad y_b \quad z_b\ )^T$ is the projection of the buoyancy center in the moving coordinate system, φ is roll angle, θ is pitch angle and ψ is yaw angle.

2.4. Hydrodynamic force and torque

In general, the hydrodynamics are related to a variety of factors, including the scale and shape of the plant, the physical state of the plant, the physical and geometric properties of the flow field, and the control elements of the propeller and so on. Since the plant is a symmetric sphere with slow moving speed, the predominant hydrodynamic force of the fluid inertia is considered only in order to facilitate the design of control law. In terms of six degrees of freedom, its forces and moment of forces are expressed as follow:

$$\begin{bmatrix} X_f \\ Y_f \\ Z_f \\ K_f \\ M_f \\ N_f \end{bmatrix} = \begin{bmatrix} -\lambda_{11}\dot{u} - \lambda_{33}wq + \lambda_{22}vr \\ -\lambda_{22}\dot{v} - \lambda_{11}ur + \lambda_{33}wp \\ -\lambda_{33}\dot{w} - \lambda_{22}vp + \lambda_{11}uq \\ -\lambda_{44}\dot{p} \\ -\lambda_{55}\dot{q} \\ -\lambda_{66}\dot{r} \end{bmatrix}, \tag{5}$$

where$(\lambda_{11} \quad \lambda_{22} \quad \lambda_{33})$ is the additional mass, and the $(\lambda_{44} \quad \lambda_{55} \quad \lambda_{66})$ is the additional moment of inertia, (u, v, w) is the speed in the moving coordinate system, (p, q, r) is angular velocity in the moving coordinate system.

2.5. Thrust of the propeller and its thrust torque

The plant with six degrees of freedom is a symmetrical sphere, being equipped with three pairs of propellers,

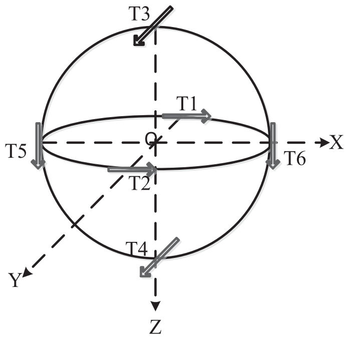

Figure 2. Position and direction of propellers.

6 sets of propellers in total. Every pair of propellers is deployed along each of the three axes on the surface of the plant, as shown in Figure 2.

The resultant force and moment generated by the propellers in the moving coordinate system are

$$\begin{bmatrix} Tx \\ Ty \\ Tz \\ Mx \\ My \\ Mz \end{bmatrix} = \begin{bmatrix} T_1 + T_2 \\ T_3 + T_4 \\ T_5 + T_6 \\ R_3T_3 + R_4T_4 \\ R_5T_5 + R_6T_6 \\ R_1T_1 + R_2T_2 \end{bmatrix}, \tag{6}$$

where R_i is the vertical distance between the ith propeller and the axis parallel with the propeller, T_i is the force of ith propeller.

2.6. Kinematics equation

The transformation formulas of the velocity and angular velocity between the fixed coordinate system and moving coordinate system are shown as follow:

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \\ \dot{\varepsilon} \\ \dot{\eta} \\ \dot{\zeta} \end{bmatrix} = \begin{bmatrix} p + q\tan\theta\sin\varphi + r\tan\theta\cos\varphi \\ q\cos\varphi - r\sin\varphi \\ q\sin\varphi\cos\theta + r\cos\varphi\sec\theta \\ u\cos\psi\cos\theta + v(\cos\psi\sin\theta\sin\varphi - \sin\psi\cos\varphi) \\ +w(\cos\psi\sin\theta\sin\varphi - \sin\psi\sin\varphi) \\ u\sin\psi\cos\theta + v(\sin\psi\sin\theta\sin\varphi + \cos\psi\cos\varphi) \\ +w(\sin\psi\sin\theta\cos\varphi - \cos\psi\sin\varphi) \\ v\cos\theta\sin\varphi + w\cos\theta\cos\varphi - u\sin\theta \end{bmatrix}. \tag{7}$$

2.7. Kinetic equation

According to the centre of mass motion theorem and the moment of momentum theorem, and the relationship between the absolute derivative and the relative derivative, the kinetic equation is obtained, as shown below:

$$\begin{cases} m(\dot{u} + qw - rv) = X \\ m(\dot{v} + ru - pw) = Y \\ m(\dot{w} + pv - qu) = Z \\ I_x\dot{p} + (I_z - I_y)qr = K \\ I_y\dot{q} + (I_x - I_z)rp = M \\ I_z\dot{r} + (I_y - I_z)pq = N \end{cases} \tag{8}$$

where m is mass of the robot, I_x, I_y, I_z is moment of inertia of the robot.

3. Design of CMAC-PID controller

The CMAC takes advantage of the associative memory and look-up table technology to achieve fast learning speed online with strong nonlinear approximation ability, so it can be used effectively for real-time dynamics control (Lin & Chen, 2001).

The overall control structure of the experimental body is shown in Figure 3 as below. It is a six closed-loop feedback control system. The actual location and attitude information is compared with the set value to generate the error signal. The error signal is processed by the controller and the decoupling device to drive six propellers to change the position and attitude of the subject. Each channel consists of a CMAC-PID controller which has fast learning ability and nonlinear approximation ability.

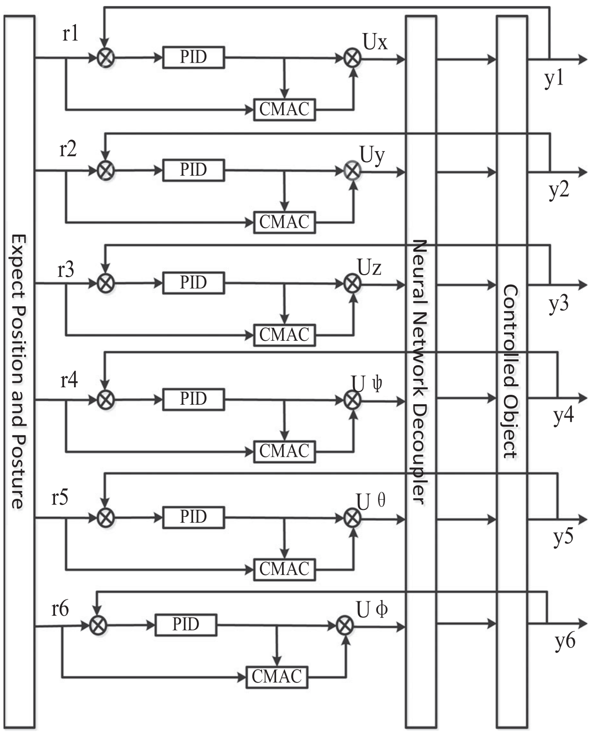

Figure 3. System control structure.

The implementation process of single loop CMAC and PID compound control algorithm is shown as follows.

The value range of input space S is $[S_{min}, S_{max}]$, which is divided into N+2C fractions:

$$\begin{cases} v_1 = v_2 = \cdots = v_c = S_{min} \\ v_i = v_{i-1} + \Delta v_i (i = c+1, c+2, \ldots, c+N), \\ v_{N+c+1} = \cdots = v_{N+2c} = S_{max} \end{cases} \tag{9}$$

where N is the quantize levels, c is the fraction generalization parameter.

The formula of the actual CMAC mapping method is described as follows:

$$a_i = \begin{cases} 1, & S_i \in [v_i, v_{i+c}], i = c+1, c+2, \ldots, c+N \\ 0, & \text{others} \end{cases} \tag{10}$$

The CMAC weights $w = [w_1, w_2, \ldots, w_N]$ is adjust according to gradient descent method, as shown as follow:

$$\begin{aligned} E(k) &= \frac{1}{2}(u_n(k) - u(k))^2 \cdot \frac{a_i}{c}, \\ \Delta w(k) &= -\eta \frac{\partial E}{\partial w} = \eta \frac{u(k) - u_n(k)}{c} a_i, \\ w(k) &= w(k-1) + \Delta w(k) + \alpha \cdot (w(k) - w(k-1)), \end{aligned} \tag{11}$$

where $\eta (0 < \eta < 1)$ is the learning rate of the CMAC neural network, $\alpha (0 < \alpha < 1)$ is the inertia coefficient.

The output of CMAC is

$$u_n(k) = \sum_{i=1}^{c} w_i a_i \tag{12}$$

The total control output is superposition of CMAC and PID:

$$u(k) = u_n(k) + u_p(k). \tag{13}$$

4. Design of neural network decouple method

There is not only the coupling between the six control channels, but also the coupling degree will change with the changing of the attitude of the experimental object (Vega, Chocron, & Benbouzid, 2016). So a two-layer linear neural network decouple method is designed. The decoupling principle is shown below as Figure 4.

The input and output relation of the neural network decoupler can be described by $V = MU$ as

$$\begin{bmatrix} v1 \\ v2 \\ v3 \\ v4 \\ v5 \\ v6 \end{bmatrix} = \begin{bmatrix} m_{11} & \cdots & m_{16} \\ \vdots & \ddots & \vdots \\ m_{61} & \cdots & m_{66} \end{bmatrix} \begin{bmatrix} u1 \\ u2 \\ u3 \\ u4 \\ u5 \\ u6 \end{bmatrix} \tag{14}$$

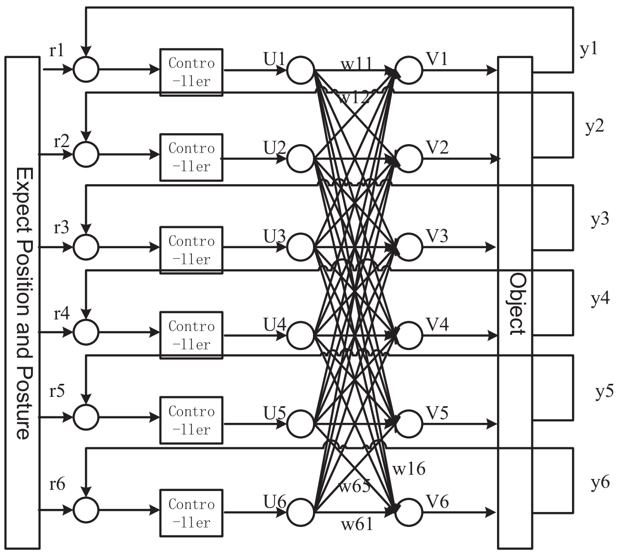

Figure 4. Neural network decoupler.

The input and output relations of the object can be described by $Y = GV$ as

$$\begin{bmatrix} y1 \\ y2 \\ y3 \\ y4 \\ y5 \\ y6 \end{bmatrix} = \begin{bmatrix} g_{11} & \cdots & g_{16} \\ \vdots & \ddots & \vdots \\ g_{61} & \cdots & g_{66} \end{bmatrix} \begin{bmatrix} v1 \\ v2 \\ v3 \\ v4 \\ v5 \\ v6 \end{bmatrix} \tag{15}$$

So the input and output relations of the decoupler and the object can be described by $Y = GMU$ as below:

$$\begin{bmatrix} y1 \\ y2 \\ y3 \\ y4 \\ y5 \\ y6 \end{bmatrix} = \begin{bmatrix} g_{11} & \cdots & g_{16} \\ \vdots & \ddots & \vdots \\ g_{61} & \cdots & g_{66} \end{bmatrix} \begin{bmatrix} m_{11} & \cdots & m_{16} \\ \vdots & \ddots & \vdots \\ m_{61} & \cdots & m_{66} \end{bmatrix} \begin{bmatrix} u1 \\ u2 \\ u3 \\ u4 \\ u5 \\ u6 \end{bmatrix} \tag{16}$$

The total system error can be defined as

$$E(k) = \sum_{s=1}^{6} e_s(k) = \sum_{s=1}^{6} (r_s(k) - y_s(k)) \tag{17}$$

The goal of the decoupler is to transform the matrix GM into a diagonal matrix. For the transfer matrix G is hard to obtain, the method used here is to reduce the total system error by adjusting the weight matrix M by gradient descent method as follows:

$$\begin{aligned}\Delta m_{ij}(k) &= -\eta\frac{\partial E(k)}{\partial m_{ij}(k)}\\ &= -\eta\sum_{s=1}^{6}\frac{\partial e_s(k)}{\partial m_{ij}(k)}\\ &= -\eta\sum_{s=1}^{6}\frac{\partial(r_s(k)-y_s(k))}{\partial m_{ij}(k)}\\ &= \eta\sum_{s=1}^{6}\frac{\partial y_s(k)}{\partial m_{ij}(k)}\\ &= \eta\sum_{s=1}^{6}\frac{\partial y_s(k)}{\partial v_i(k)}\frac{\partial v_i(k)}{\partial w_{ij}(k)}\\ &= \eta\sum_{s=1}^{6}\frac{\partial y_s(k)}{\partial v_i(k)}u_j(k)\\ &= \eta\sum_{s=1}^{6}\operatorname{sgn}\frac{y_s(k)-y_s(k-1)}{v_i(k)-v_i(k-1)}u_j(k)\\ &= \eta u_j(k)\sum_{s=1}^{6}\operatorname{sgn}\frac{y_s(k)-y_s(k-1)}{v_i(k)-v_i(k-1)}\end{aligned} \tag{18}$$

where the value of the $\partial y_s(k)/\partial v_i(k)$ could not be achieved for the specific parameters is unknown, however, its sign can be used as an approximation.

So the neural network decoupler weight adjustment method is

$$\begin{cases} m_{ij}(k) = m_{ij}(k-1) + \eta u_j(k)\sum\limits_{s=1}^{6}\operatorname{sgn}\frac{y_s(k)-y_s(k-1)}{v_i(k)-v_i(k-1)},\ i \neq j\\ m_{ij}(k) = 1\,,\quad i = j\end{cases} \tag{19}$$

5. Optimization of control parameters based on genetic algorithm.

The PID not only serves as the primary controller in the initial stage but also serves as the tutor signal for the CMAC. Therefore, the selection of appropriate PID parameters is of great significance to the whole control system. Genetic algorithms can solve complex linear and nonlinear problems, and it also can be used in multidimensional space optimization problems (Gupta, Agarwal, & Kumar, 2013). So genetic algorithm is selected to optimize PID parameters in this paper.

First of all, the real value encoding is selected in this paper. In the genetic algorithm, fitness function is the main index to describe the individual performance. According to the fitness level, the individual performs the survival of the fittest. Fitness is the driving force of the genetic algorithm.

To achieve better dynamic characteristics in the transition process, the systematic error e (t) is introduced into the formula. To limit the control energy, the controller output u (t) is introduced. In addition, to avoid overshoot, the penalty mechanism is introduced. That is, when *ey* (*t*) = *y* (*t*) −*y* (*t*−1) < 0 happens, it will be introduced in the optimal indicator. So, the optimal indicator can be shown as follows:

$$J = \begin{cases}\int\limits_0^{\infty}(t|e(t)| + t|u(t)|)dt, & \textit{if } ey(t) \geq 0\\ \int\limits_0^{\infty}(t|e(t)|+t|u(t)|+t|ey(t)|)dt, & \textit{if } ey(t) < 0\end{cases} \tag{20}$$

Therefore, the fitness function can be obtained as follows:

$$f = 1/J \tag{21}$$

The selection of fitness function directly affects the convergence speed of genetic algorithm and the ability to find the optimal solution. So the design of fitness function should be as simple as possible to minimize the time complexity of the calculation. The fitness ratio method is used to select. The greater the individual fitness, the higher the probability of selection, and vice versa. That is

$$P_{si} = f_i/\sum_{j=1}^{n} f_j, \tag{22}$$

where P_{si} is the *i*th individual selection probability, f_i is the *i*th individual fitness value, and *n* is the population number.

Crossover refers to the operation of generating new individuals by substituting the partial structure of the two parent individuals. By crossing, the search ability of genetic algorithm can be improved. The single-point crossover operator is used in the method. In this case, the adaptive mutation probability is used, that is to say, the variation probability is related to fitness, the less fitness, the greater the probability of mutation, and vice versa.

$$p_c = \begin{cases}\frac{k_1(f_{\max}-f_b)}{f_{\max}-f_{avg}}, & f_b \geq f_{avg}\\ k_2\,, & f_b < f_{avg}\end{cases} \tag{23}$$

$$p_m = \begin{cases}\frac{k_2(f_{\max}-f)}{f_{\max}-f_{avg}}\,, f \geq f_{avg}\\ k_4\,, f < f_{avg}\end{cases} \tag{24}$$

where p_c is the crossover ratio, p_m is the mutation ratio, $f_{\max}$ is the maximum individual fitness in the population, f_{avg} is the average individual fitness in the population, f_b is the greater fitness of two individuals involved in the crossover, and f is the individual fitness $k_1 = 0.65, k_2 = 0.85, k_3 = 0.05, k_4 = 0.15$.

6. Simulation analysis

The algorithm proposed in this paper is adopted. In the algorithm, CMAC-PID compound control are used to

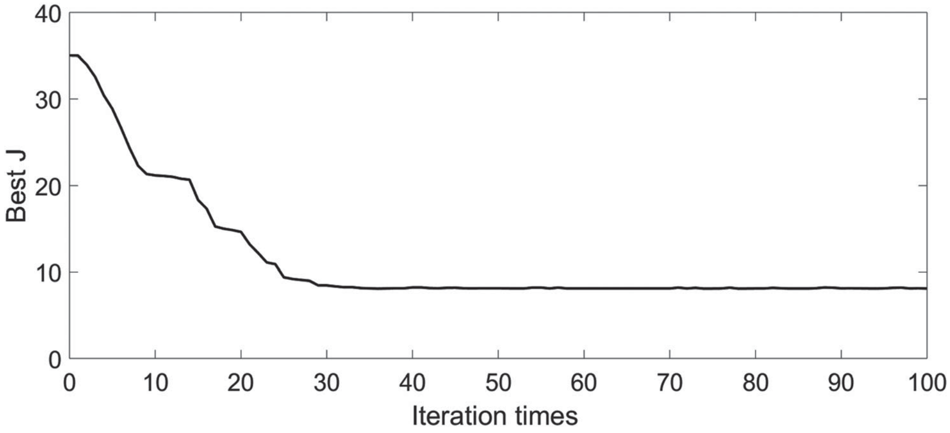

Figure 5. The adjustment process of J.

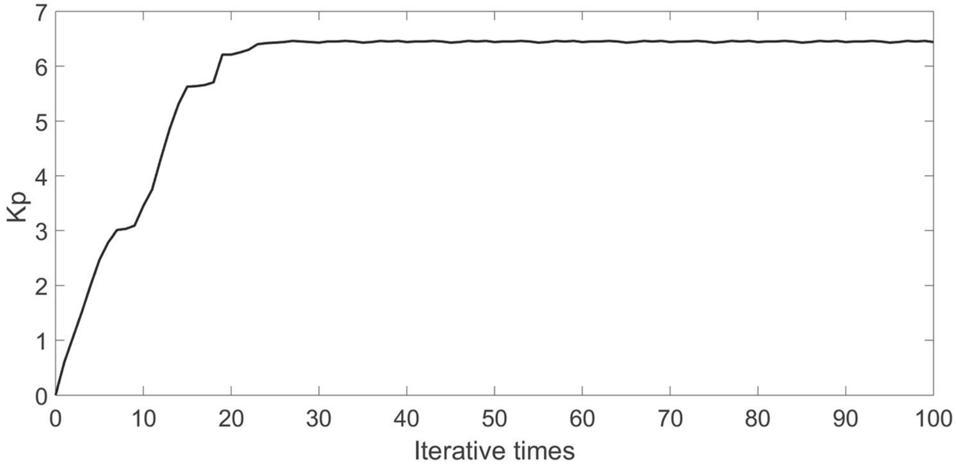

Figure 6. The adjustment process of Kp.

generate control signals, and two layers of neural networks decoupler are used to decouple multi-channel control into single-channel control, and genetic algorithm is used to optimize PID parameters.

The spherical controlled object can be simplified to a sphere with a mass of 40.32 kg and a diameter of 0.59 m. The centre of gravity and the centre of buoyancy of the experimental object on the three axes of *xyz* cannot be completely consistent, but the deviation is less than 0.02 m by accurate balancing.

The initial position and attitude angle of the controlled object is (0 m, 0 m, 0 m, 0°,0°,0°). The initial velocity and acceleration of controlled object is zero. The target position and attitude angle is (1 m, 1 m, 1 m, 0°, 0°, 30°). The simulation is performed in the MATLAB, with duration of 100 s and step of 1 ms.

Genetic algorithm is used with 200 samples and adaptive mutation probability. Crossover and mutation probability parameter KP values in the range of [0, 20]. Kd values in the range of [0.1, 10]. Real number encoding is adopted. After evolving over 100 generations, the optimization of the parameters are obtained: KP $= 6.54$, kd $= 0.73$. The optimization process of the cost function J and Kp are shown in the figure below.

As shown in Figure 5, the cost function J drops rapidly in 34 iteration times. Then the adjustment speed slows down, and finally stabilizes around 8.12, showing a good evolution effect.

Figure 6 as below is the change curve of Kp in PID parameter adjustment process. It can be seen that the optimal value of the Kp can be obtained after 35 iteration times.

The initial value of the two-layer neural network decoupling matrix is a diagonal matrix. With the change of the attitude of the experimental body, the weight of the decoupled matrix will be changed to reduce the coupling degree. The following Figure 7 shows the decoupler weight matrix at a certain moment in the calculation process.

The simulation results of the control algorithm proposed by this paper and the conventional PID control are shown as follow.

As shown in Figure 8 below, the black solid curve is the position adjustment curve in the $E-\xi$ direction of the

$$M = \begin{bmatrix} 0.972 & 0.025 & -0.154 & 0.184 & 0.023 & -0.079 \\ 0.093 & -0.904 & 0.101 & -0.045 & 0.078 & 0.103 \\ -0.102 & 0.099 & 0.913 & 0.103 & -0.121 & -0.094 \\ 0.109 & 0.096 & -0.119 & 0.922 & 0.016 & -0.012 \\ -0.086 & -0.098 & 0.019 & 0.102 & 0.986 & 0.087 \\ 0.139 & 0.108 & -0.052 & 0.029 & -0.017 & 0.912 \end{bmatrix}$$

Figure 7. Weight matrix of NN decoupler.

fixed coordinate system. From the figure, it is observed that the plant moved fast in the early phase, and it reached the designated position at the time of 57 s after fine tuning, and then it kept stable within the error limit of 0.02 m, so it had a high steady-state accuracy. The blue broken curve is the position curve in the $E - \xi$ direction when the conventional PID is used to control the plant. From the figure, it is known that the moving speed of the plant is relatively slow, the process of the plant reaching the steady-state costs more time.

As shown in Figure 9 below, the black solid curve is the yaw angle adjustment curve when the proposed algorithm is adopted, and the blue broke curve is the yaw angle adjustment curve when the conventional PID is adopted. The goal angle is 30 degrees. From the figure, it is observed that the black solid curve performs smaller overshoot and can get to the stable state faster than the blue broken curve. In addition, this method avoids the trouble of manually adjusting PID parameters.

When the interference of water flow exists, the change of running track is simulated. As shown in Figure 10 below, the black solid curve is the position adjustment curve in the $E - \xi$ direction of the fixed coordinate system in the absence of external interference. In order to simulate the water fluctuation in the water tank, sinusoidal signal with amplitude of 0.01 m and a frequency of 0.5 rad/s was used in the simulation. The blue broken curve

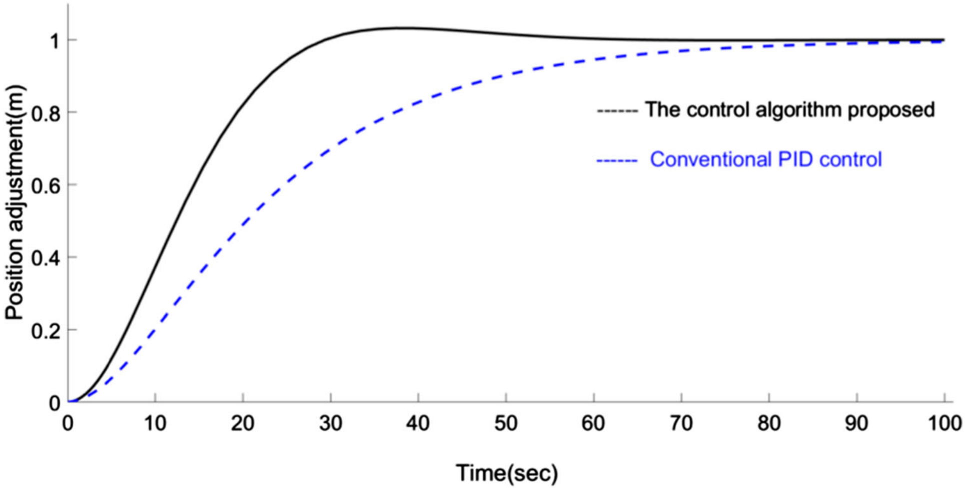

Figure 8. Position adjustment process.

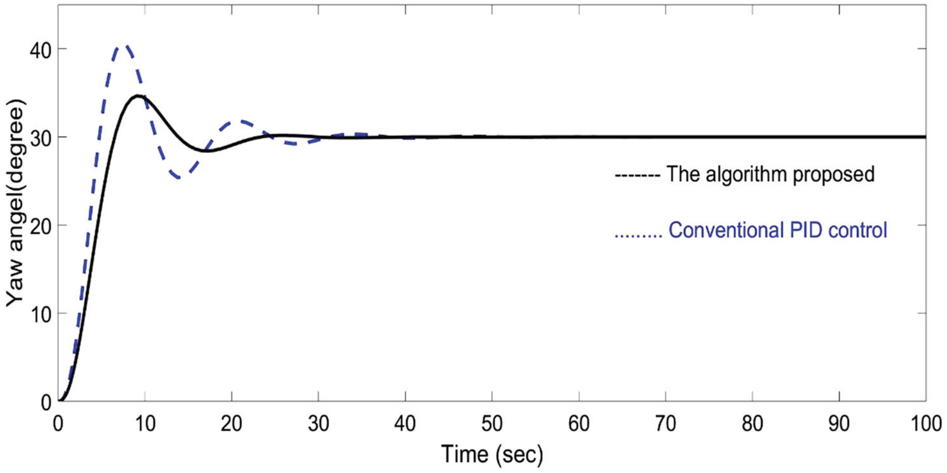

Figure 9. Angle adjustment process.

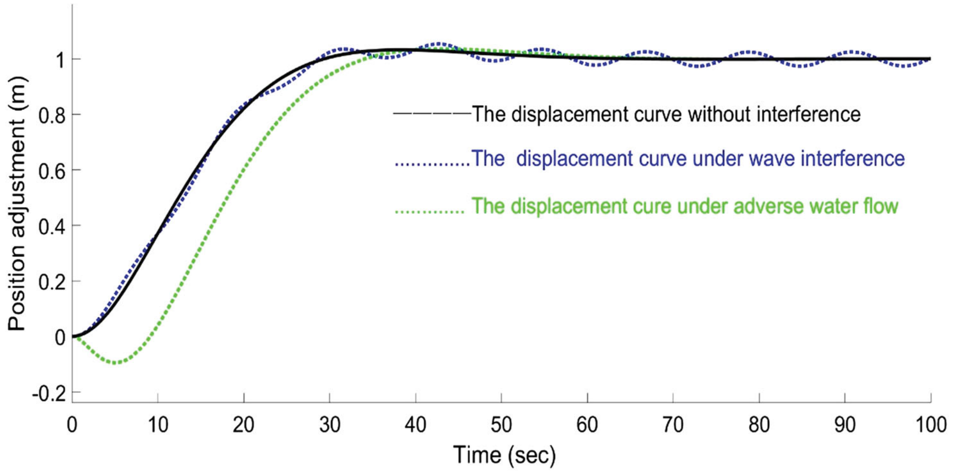

Figure 10. Position adjustment curve under different disturbances.

is the position adjustment under water fluctuation. From the figure, it is observed that the robot can reach the designated position in 34 s and then keep fluctuating around the designated position with the amplitude about 0.05 m. The green broken line is the position adjustment curve when the robot is at upstream with velocity 0.3 m/s. As can be seen from the figure, under the action of counter-current, the robot first moves backward, then moves toward the designated position, and finally stabilizes at the designated position in about 37 s. From the figure, it is observed that the algorithm proposed can good resist the flow from one direction. However, when the robot is in a stable state, the suppression of fluctuation is not very good, which requires further research. In conclusion, the result shows that the algorithm proposed by this paper is much better than the conventional PID control method.

7. Conclusion

In this paper, the motion control system of a six degrees of freedom spherical controlled object in the tank under neutral buoyancy is studied. Firstly, the motion equation of the controlled object was established by force analysis. Then, a control method based on CMAC is designed, and neural network decoupling and genetic algorithm parameter optimization are adopted. The calculation process shows that the decoupling matrix can adapt to changes in the body posture, the genetic algorithm can get good mentor parameters in a short time. The simulation result shows that the control algorithm proposed in this paper, with better dynamic characteristics and steady-state accuracy, is much better than conventional PID control for the proposed algorithm. Therefore, this algorithm proposed is a useful method in motion control under neutral buoyancy.

Disclosure statement

No potential conflict of interest was reported by the authors.

References

Churchill, P., Akin, D., & Howard, R. (1993). Neutral buoyancy simulation of space telerobotics operations. *Advances in the Astronautical Sciences*, *81*, 149–150.

Gupta, S., Agarwal, G., & Kumar, V. (2013). An efficient and robust genetic algorithm for multiprocessor task scheduling. *International Journal of Computer Theory and Engineering*, *5*(2), 377–382.

Isa, K., & Arshad, M. R. (2012). Buoyancy-driven underwater glider modelling and analysis of motion control. *Indian Journal of Geo-Marine Sciences*, *41*(6), 516–526.

Isa, K., & Arshad, M. R. (2013). Modeling and motion control of a hybrid-driven underwater glider. *Indian Journal of Geo-Marine Sciences*, *42*(8), 971–979.

Keehong, S., Chung, S. J., & Slotine, J. (2010). CPG-based control of a turtle-like underwater vehicle. *Autonomous Robots*, *28*(3), 247–269.

Khodayari, M. H., & Balochian, S. (2015). Modeling and control of autonomous underwater vehicle (AUV) in heading and depth attitude via self-adaptive fuzzy PID controller. *Journal of Marine Science and Technology*, *20*(3), 559–578.

Lin, C. C., & Chen, F. C. (2001). On a New CMAC control scheme, and its comparisons with the PID controllers. *Proceedings of the 2001 American Control Conference*, *2*, 769–774.

Vega, E. P., Chocron, O., & Benbouzid, M. (2016). A flat design and a validated model for an AUV reconfigurable magnetic coupling thruster. *IEEE/ASME Transactions on Mechatronics*, *21*(6), 2892–2901.

Wang, Y., Lin, X. T., Song, S. J., Liu, Y. H., Zhang, H. W., & Wang, S. X. (2014). Dynamic modeling and simulation of autonomous underwater vehicle with vectored thruster. *Journal of Tianjin University Science and Technology*, *47*(2), 143–148.

Yang, H., & Ma, J. (2011). Nonlinear feedforward and feedback control design for autonomous underwater glider. *Journal of Shanghai Jiaotong University (Science)*, *16*(1), 11–16.