MDPI

Spiral Dive Control of Underactuated AUV Based on a Single-Input Fractional-Order Fuzzy Logic Controller

Zhiyu Cui [1,2]**, Lu Liu** [1,2,]*[iD]**, Boyu Zhu** [1]**, Lichuan Zhang** [1,2][iD]**, Yang Yu** [1,2]**, Zhexuan Zhao** [1]**, Shiyuan Li** [1] **and Mingwei Liu** [1]

1   Research & Development Institute, Northwestern Polytechnical University, Shenzhen 518057, China
2   School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an 710072, China
*   Correspondence: liulu12201220@nwpu.edu.cn

Abstract: Autonomous underwater vehicles (AUVs) have broad applications owing to their ability to undertake long voyages, strong concealment, high level of intelligence and ability to replace humans in dangerous operations. AUV motion control systems can ensure stable operation in the complex ocean environment and have attracted significant research attention. In this paper, we propose a single-input fractional-order fuzzy logic controller (SIFOFLC) as an AUV motion control system. First, a single-input fuzzy logic controller (SIFLC) was proposed based on the signed distance method, whose control input is the linear combination of the error signal and its derivative. The SIFLC offers a significant reduction in the controller design and calculation process. Then, a SIFOFLC was obtained with the derivative of the error signal extending to a fractional order and offering greater flexibility and adaptability. Finally, to verify the superiority of the proposed control algorithm, comparative numerical simulations in terms of spiral dive motion control were conducted. Meanwhile, the parameters of different controllers were optimized according to the hybrid particle swarm optimization (HPSO) algorithm. The simulation results illustrate the superior stability and transient performance of the proposed control algorithm.

Keywords: fractional calculus; autonomous underwater vehicle; fuzzy logic control; particle swarm optimization algorithm

1. Introduction

Autonomous underwater vehicles (AUVs) are a crucial technical platform for ocean information acquisition and autonomous operation. They have extensive application prospects, such as marine environment observation, marine resources exploration and security defense. Nevertheless, motion control systems for AUV have become very challenging due to their high nonlinearity, strong coupling, model parameter uncertainties and external disturbances. In addition, an AUV system is usually designed to be underactuated to save cost and improve propulsion efficiency.

With regard to the motion control of underactuated AUV, a variety of control algorithms are available, including proportional-integral-derivative (PID) control, backstepping control, fuzzy logic control, and sliding mode control [1–7]. In [4], a single-input fuzzy logic controller (SIFLC) was proposed for AUV depth control. Simulation results show that the SIFLC gives an identical response as Mamdani and T-S type FLC to the same input sets, while its execution time is more than two orders of magnitude faster than the conventional FLC. In [6], a switching control algorithm based on active disturbance rejection control (ADRC) and fuzzy logic control was applied to the depth control of a self-developed AUV. Numerical simulations showed that the proposed method is more efficient in suppressing external disturbances and inner signal transmission disturbance than PID controller.

Fractional calculus is an extension of traditional integral calculus, it describes the fractal dimension of space. In recent years, its applications in the control field, such as

fractional-order model [8,9], fractional-order control algorithm [10–12], fractional-order optimization algorithm [13,14], have attracted significant research attention. Furthermore, stability analysis for fractional-order control systems have been proposed in several studies [15–17]. A fractional-order, proportional–integral–derivative (FOPID) controller has been proposed by Podlubny. Their proposed controller has two additional parameters: integral order and differential order compared with a PID controller [18]. In [19], an optimized FOPID controller for improved transient control performance was applied to an AUV yaw control system. In addition, a fractional-order Mamdani fuzzy logic controller has been proposed for vehicle nonlinear active suspension, which effectively improves ride comfort and handling stability [20]. However, there has been no report on the application of fractional-order fuzzy logic control in AUV motion.

In this paper, a single-input fractional-order fuzzy logic controller (SIFOFLC) is proposed and applied to an AUV motion control system. Its control input was simplified to a single variable known as distance variable by applying the signed distance method [21], which aims to reduce the computation burden and complex parameter tuning process. Furthermore, a fractional calculus operator was applied to the enhanced FLC due to its recognized ability to increase the controller's flexibility and adaptability. With respect to the controller parameters, we developed and applied a hybrid particle swarm optimization (HPSO) algorithm to obtain optimal control performance. Unlike a conventional PSO algorithm, this includes the local optimal particle term to avoid falling into local optimal region, and the fitness value function includes both steady-state performance and transient performance of an AUV motion control system. To verify the effectiveness of SIFOFLC, we conducted comparative numerical simulations of spiral dive motion control. The object of study, REMUS-100 AUV, was developed by Woods Hole Oceanographic Institution [22], while the simulation was performed using the marine systems simulator (MSS) by Fossen and Perez [23]. Simulation results show that, compared with a FOPID controller and conventional T-S FLC, the SIFOFLC is more efficient in reducing angular velocity oscillations, shortening settling time and improving control accuracy.

The remainder of this paper is organized as follows. Section 2 discusses the six degrees of freedom nonlinear motion equations of AUV. The SIFOFLC design is introduced in Section 3, along with its advantages compared with traditional T-S FLC. In Section 4, the HPSO algorithm is described and is applied to various control systems to obtain optimal parameters. To verify the effectiveness of the proposed method, simulations and numerical comparisons are carried out in Section 5. Finally, some concluding remarks are presented in Section 6.

2. Kinematic and Dynamic Modeling of AUV

Six degrees of freedom motion equations of AUV can be described using the earth-fixed coordinate system and body-fixed coordinate system shown in Figure 1, both of which are right-handed. The earth-fixed coordinate system $O - xyz$ has its origin O fixed to the earth, and the body-fixed coordinate system $O_b - x_b y_b z_b$ is a moving reference frame with its origin O_b fixed to AUV center of buoyancy.

The general motion of a vehicle in six degrees of freedom can be described with the following vectors:

$$\eta_1 = \begin{bmatrix} x & y & z \end{bmatrix}^T \quad \eta_2 = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T$$
$$v_1 = \begin{bmatrix} u & v & w \end{bmatrix}^T \quad v_2 = \begin{bmatrix} p & q & r \end{bmatrix}^T$$
$$\tau_1 = \begin{bmatrix} X & Y & Z \end{bmatrix}^T \quad \tau_2 = \begin{bmatrix} K & M & N \end{bmatrix}^T$$

where η describes the position and orientation of the vehicle with respect to the earth-fixed reference frame, v denotes the linear and angular velocities with respect to the body-fixed reference frame, and τ describes the total forces and moments acting on the vehicle in the body-fixed reference frame.
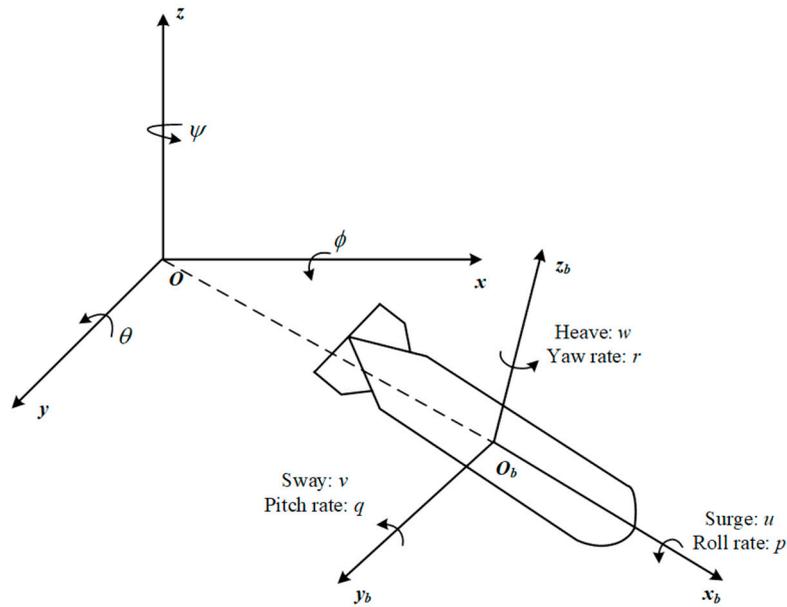
Figure 1. Coordinate systems of AUV.

The coordinate transformation of the translational velocity between earth-fixed and body-fixed coordinate systems can be expressed as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = J_1 \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{1}$$

where

$$J_1 = \begin{bmatrix} \cos\psi\cos\theta & -\sin\psi\cos\phi + \cos\psi\sin\theta\sin\phi & \sin\psi\sin\phi + \cos\psi\sin\theta\cos\phi \\ \sin\psi\cos\theta & \cos\psi\cos\phi + \sin\psi\sin\theta\sin\phi & -\cos\psi\sin\phi + \sin\psi\sin\theta\cos\phi \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix}$$

The coordinate transformation relates rotational velocity between two coordinate systems and can be described as

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = J_2 \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2}$$

where

$$J_2 = \begin{bmatrix} 1 & \sin\phi\tan\theta & \cos\phi\tan\theta \\ 0 & \cos\phi & -\sin\phi \\ 0 & \sin\phi/\cos\theta & \cos\phi/\cos\theta \end{bmatrix}$$

The locations of the AUV centers of gravity and buoyancy are defined in the body-fixed coordinate system as follows:

$$r_G = \begin{bmatrix} x_g & y_g & z_g \end{bmatrix}^T \quad r_B = \begin{bmatrix} x_b & y_b & z_b \end{bmatrix}^T$$

Based on the theory of rigid body dynamics and the analysis of total forces and moments acting on AUV, the nonlinear motion equations for the REMUS vehicle in six degrees of freedom can be expressed as follows [24]:

$$m[\dot{u} - vr + wq - x_g(q^2 + r^2) + y_g(pq - \dot{r}) + z_g(pr + \dot{q})] = X_{HS} + X_{u|u|}u|u| +$$
$$X_{\dot{u}}\dot{u} + X_{wq}wq + X_{qq}qq + X_{vr}vr + X_{rr}rr + X_{prop}$$
$$m[\dot{v} - wp + ur - y_g(r^2 + p^2) + z_g(qr - \dot{p}) + x_g(pq + \dot{r})] = Y_{HS} + Y_{v|v|}v|v| + Y_{r|r|}r|r| +$$
$$Y_{\dot{v}}\dot{v} + Y_{\dot{r}}\dot{r} + Y_{ur}ur + Y_{wp}wp + Y_{pq}pq + Y_{uv}uv + Y_{uu\delta_r}u^2\delta_r$$
$$m[\dot{w} - uq + vp - z_g(p^2 + q^2) + x_g(rp - \dot{q}) + y_g(rq + \dot{p})] = Z_{HS} + Z_{w|w|}w|w| + Z_{q|q|}q|q| +$$
$$Z_{\dot{w}}\dot{w} + Z_{\dot{q}}\dot{q} + Z_{uq}uq + Z_{vp}vp + Z_{rp}rp + Z_{uw}uw + Z_{uu\delta_s}u^2\delta_s$$
$$I_{xx}\dot{p} + (I_{zz} - I_{yy})qr + m[y_g(\dot{w} - uq + vp) - z_g(\dot{v} - wp + ur)] = K_{HS} + K_{p|p|}p|p| + K_{\dot{p}}\dot{p} + K_{prop}$$
$$I_{yy}\dot{q} + (I_{xx} - I_{zz})rp + m[z_g(\dot{u} - vr + wq) - x_g(\dot{w} - uq + vp)] = M_{HS} + M_{w|w|}w|w| + M_{q|q|}q|q| +$$
$$M_{\dot{w}}\dot{w} + M_{\dot{q}}\dot{q} + M_{uq}uq + M_{vp}vp + M_{rp}rp + M_{uw}uw + M_{uu\delta_s}u^2\delta_s$$
$$I_{zz}\dot{r} + (I_{yy} - I_{xx})pq + m[x_g(\dot{v} - wp + ur) - y_g(\dot{u} - vr + wq)] = N_{HS} + N_{v|v|}v|v| + N_{r|r|}r|r| +$$
$$N_{\dot{v}}\dot{v} + N_{\dot{r}}\dot{r} + N_{ur}ur + N_{wp}wp + N_{pq}pq + N_{uv}uv + N_{uu\delta_r}u^2\delta_r \tag{3}$$

where m is AUV's mass, I_{xx}, I_{yy}, I_{zz} are the moments of inertia of AUV to three coordinate axes, $X_{HS}, Y_{HS}, Z_{HS}, K_{HS}, M_{HS}, N_{HS}$ are hydrostatics, $X_{u|u|}, Y_{v|v|}, Y_{r|r|}, Z_{w|w|}, Z_{q|q|}, K_{p|p|}, M_{w|w|}, M_{q|q|}, N_{v|v|}, N_{r|r|}$ are hydrodynamic drag coefficients, $Y_{uv}, Y_{uu\delta_r}, Z_{uw}, Z_{uu\delta_s}, M_{uw}, M_{uu\delta_s}, N_{uv}, N_{uu\delta_r}$ are lift coefficients and lift moment coefficients of body and control fins, respectively, X_{prop}, K_{prop} are propeller thrust and torque, respectively, and δ_r, δ_s are rudder angle and stern plane angle, respectively. The remaining coefficients are added mass coefficients.

Separate the acceleration terms from the other terms in the equations of AUV motion so that the equations can be summarized in matrix form as follows:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \\ \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 & 0 & mz_g & -my_g \\ 0 & m - Y_{\dot{v}} & 0 & -mz_g & 0 & mx_g - Y_{\dot{r}} \\ 0 & 0 & m - Z_{\dot{w}} & my_g & -mx_g - Z_{\dot{q}} & 0 \\ 0 & -mz_g & my_g & I_{xx} - K_{\dot{p}} & 0 & 0 \\ mz_g & 0 & -mx_g - M_{\dot{w}} & 0 & I_{yy} - M_{\dot{q}} & 0 \\ -my_g & mx_g - N_{\dot{v}} & 0 & 0 & 0 & I_{zz} - N_{\dot{r}} \end{bmatrix}^{-1} \begin{bmatrix} \sum X \\ \sum Y \\ \sum Z \\ \sum K \\ \sum M \\ \sum N \end{bmatrix} \tag{4}$$

where $\sum X \cdots \sum N$ refer to the sum of terms without acceleration. So far, six degrees of freedom nonlinear motion equations of AUV can be obtained by combining (4) with (1) and (2).

3. Design of SIFOFLC

3.1. Fundamentals of Fractional Calculus

Fractional calculus, essentially the non-integer order calculus, has the same history as integer order calculus. The three frequently used definitions of fractional calculus are the Grunwald–Letnikov definition, the Riemann–Liouville definition and the Caputo definition [25].

We consider the Caputo definition in this study because of its wide applications in engineering problems, the fractional integral and derivative by Caputo definition are as follows:

$$_{t_0}D_t^{-\alpha}f(t) = \frac{1}{\Gamma(\alpha)}\int_{t_0}^{t}\frac{f(\tau)}{(t-\tau)^{1-\alpha}}d\tau \tag{5}$$

$$_{t_0}D_t^{\alpha}f(t) = \frac{1}{\Gamma(m-\alpha)}\int_{t_0}^{t}\frac{f^{(m)}(\tau)}{(t-\tau)^{1+\alpha-m}}d\tau \tag{6}$$

where $_{t_0}D_t^{\alpha}$ represents the fractional calculus operator, $f(t)$ is a continuous function and t_0 denotes the initial time. α represents the fractional-order, $m = \lceil\alpha\rceil$. $\Gamma(\cdot)$ denotes the Gamma function as in (7).

$$\Gamma(z) = \int_0^{\infty} e^{-t}t^{z-1}dt \tag{7}$$

In the numerical simulations, we adopt the standard Oustaloup approximation method to obtain the consistent frequency characteristics as fractional differential operator. A rational

transfer function in the form of zero-pole type is described according to the Oustaloup method, that N is the order of filter, $[\omega_b, \omega_h]$ is the selected frequency bound. The zero and pole are defined as ω'_k and ω_k, which divide the frequency band into $2N + 1$ intervals.

The Oustaloup rational approximation is described as

$$s^\alpha \approx G(s) = K \prod_{k=1}^{N} \frac{s + \omega'_k}{s + \omega_k} \tag{8}$$

where $\omega'_k = \omega_b \omega_u^{(2k-1-\alpha)/N}$, $\omega_u = \sqrt{\omega_h/\omega_b}$, $\omega_k = \omega_b \omega_u^{(2k-1+\alpha)/N}$, $K = \omega_h^\alpha$.

3.2. Structure Design of SIFOFLC

A fuzzy logic controller has four components: knowledge base, inference engine, fuzzification interface, and defuzzification interface [26]. The basic structure of a two-dimensional fuzzy logic controller is described as in Figure 2, where the continuous input signal, e and $\dot{e}$, convert to the membership degree vector of the fuzzy variables through a fuzzification interface, the inference engine carries out rule inference and actual output signal is obtained through defuzzification interface. The data base denotes membership functions of the total input and output variables and the rule base is performed using a collection of fuzzy if–then rules by expert experience, both of which make up the knowledge base.
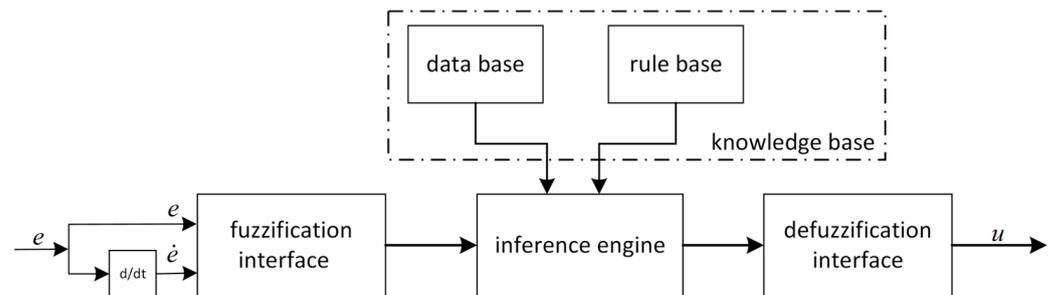

Figure 2. Block diagram of two-dimensional FLC.

Compared with the linear controller, fuzzy logic control method is more robust and suitable for complex control requirements, while its complex decision-making process brings a challenge for real-time operation. Thus, we aim to adjust the controller structure to reduce computation burden and achieve better performance.

Typically, a fuzzy logic controller has two control inputs, namely error (e) and its derivative ($\dot{e}$). It is common for its rule table to have the same output membership in a diagonal direction, something known as the Toeplitz structure, as shown in Table 1 [4]. In addition, each position on a diagonal line has the same distance from the main diagonal line of rule table. Thus, instead of using two-variable input sets ($e, \dot{e}$), the corresponding control output can be obtained using the distance between input signal and the main diagonal line. This finding was first proposed by Choi et al. and is known as the signed distance method [21]. To derive the distance, d, a two-dimensional space of e and $\dot{e}$ is established as shown in Figure 3.

Table 1. Rule table with the Toeplitz structure.

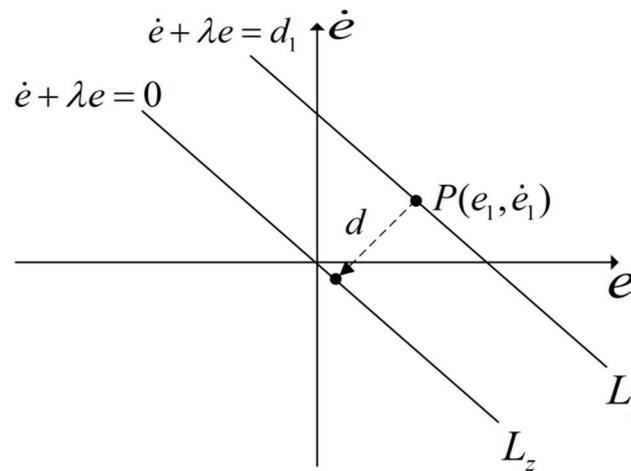| $\dot{e}$ \ e	PL	PM	PS	Z	NS	NM	NL
NL	Z	NS	NM	NL	NL	NL	NL
NM	PS	Z	NS	NM	NL	NL	NL
NS	PM	PS	Z	NS	NM	NL	NL
Z	PL	PM	PS	Z	NS	NM	NL
PS	PL	PL	PM	PS	Z	NS	NM
PM	PL	PL	PL	PM	PS	Z	NS
PL	PL	PL	PL	PL	PM	PS	Z

Figure 3. Distance variable.

The main diagonal line of the rule table is presented as a straight line crossing over the origin, whose function is $L_z : \dot{e} + \lambda e = 0$. In this case, the distance from point $P(e_1, \dot{e}_1)$ to L_z can be obtained as $d = \frac{\dot{e}_1 + \lambda e_1}{\sqrt{1+\lambda^2}}$.

Furthermore, in order to achieve better control performance, we extend the derivative of error signal to fractional order, thus the distance variable is described as

$$d = \frac{D_t^\alpha e + \lambda e}{\sqrt{1+\lambda^2}} \tag{9}$$

where α is the fractional order of error signal and it further increases degrees of freedom and complexity of the controller.

Based on the above analysis, the overall structure of SIFLC can be depicted as in Figure 4. The distance variable is obtained through linear combination of error signal and its fractional derivative, the inner fuzzy logic controller is single input single output (SISO) and the final output u is obtained by multiplying u_0 with the scale factor, which is denoted as r. We set λ to 1. Thus, in addition to the membership functions, there are two adjustable parameters in the SIFOFLC, namely α and r.
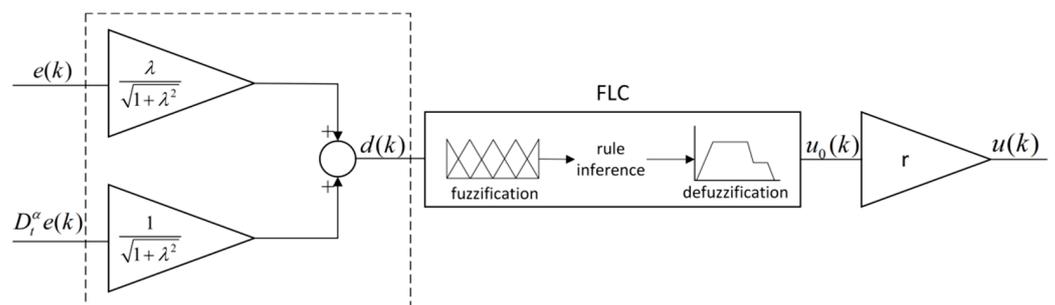

Figure 4. The SIFLC control structure.

3.3. Characteristics of SIFOFLC

In this section, the SIFOFLC and T-S FLC used in the simulation research of Section 4 are proposed and the superiority of SIFOFLC is presented through comparison.

To the SIFOFLC, the fuzzy sets of d and u_0 are both {NL, NM, NS, Z, PS, PM, PL} and the membership functions are shown in Figure 5. The membership functions of d include both S shape and triangular shape membership functions, with a singleton value membership function for u_0. In this case, the rule table is reduced to the one-dimensional vector as shown in Table 2.
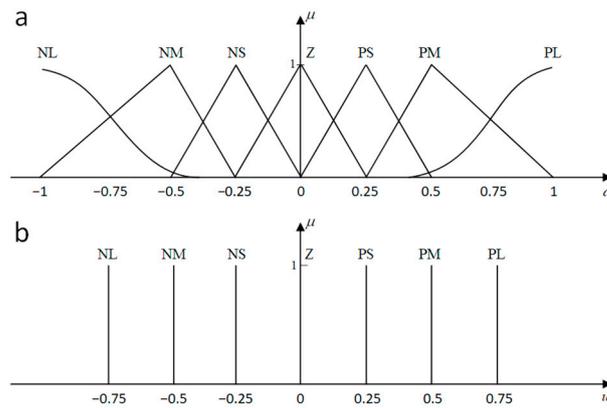
Figure 5. (**a**) Input membership function. (**b**) Output membership function.

Table 2. Rule table for the above SIFOFLC.

d	NL	NM	NS	Z	PS	PM	PL
u_0	NL	NM	NS	Z	PS	PM	PL

Based on the work mentioned above, the proposed SIFOFLC has the following advantages compared to conventional FLC:

(1)   Simplified design process.

For the two-dimensional FLC, the membership functions for error input and its derivative are required simultaneously, which means a lengthy complex tuning process. With the only input, d, the parameter tuning process for SIFOFLC is significantly reduced. Further, in a two-dimensional FLC, the number of fuzzy rules to be inferred is the square of n, which is the size of the fuzzy set. The distinguishing feature of SIFLC is that it requires only n rules, which is an exponential decay. Typically, better performance can be obtained with more a complicated control algorithm, such as the increase of fuzzy sets and rules, which further reveals the superiority of SIFOFLC.

(2)   Reduced computation burden.

The control surfaces for the SIFOFLC and conventional T-S FLC are respectively shown in Figures 6 and 7. Compared with the complex curved surface of T-S FLC, the control surface of SIFOFLC has been simplified to a linear with different slopes. Thus, the computation burden of controller operation, which includes fuzzification, fuzzy inference and defuzzification, has been significantly reduced.
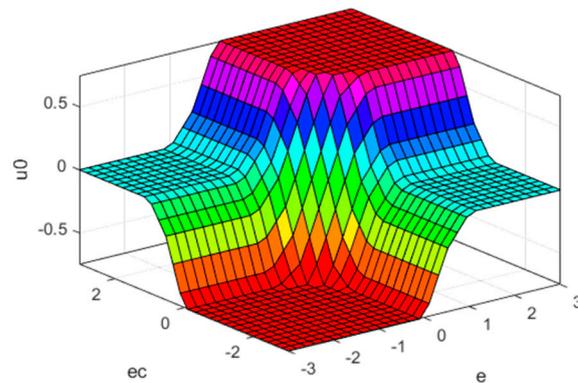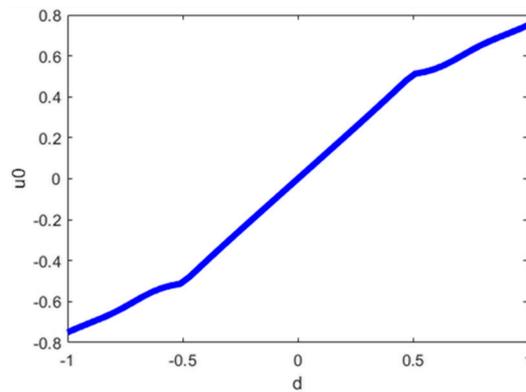

Figure 6. T-S FLC control surface.

Figure 7. SIFLC control surface.

4. Parameter Optimization for Three Controllers with HPSO Algorithm

In this section, the HPSO algorithm is developed and applied to optimize the controller parameters. Except for the SIFOFLC, the T-S FLC and FOPID control systems are optimized for further comparison in Section 5, as the SIFOFLC algorithm essentially originates from the T-S FLC, and because the FOPID controller is the most extensively used fractional order control algorithm.

It is necessary to mention that our optimization and research scenario is the spiral dive motion control of REMUS-100 AUV. It is performed using Marine Systems Simulator (MSS), which is a MATLAB toolbox and offers a set of tools for marine engineering researchers [23]. Some research details are as follows: The target depth of AUV motion linearly increases from 0 m to 30 m and the target yaw angle from 0° to 720°. The simulation time is 300 s and the input constraint of fins is set to 13.8° in accordance with [24]. Furthermore, the external disturbances are ignored.

Figure 8 illustrates the block diagram of an AUV motion control system. The propeller speed is fixed at 1500 r/m so that the cruise speed maintains a constant value. In this case, the target trajectory is obtained with heading controller and depth controller.
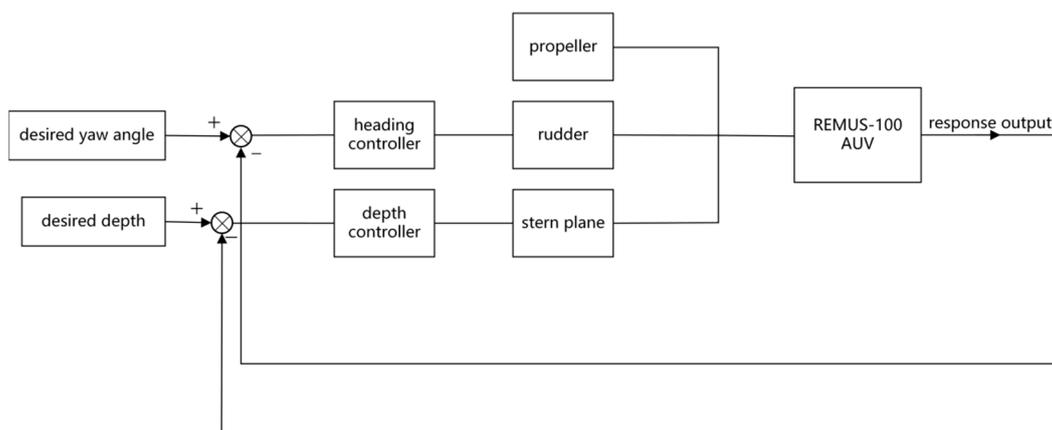

Figure 8. REMUS-100 AUV control system.

4.1. HPSO Algorithm

Because of the uncertainty and nonlinearity of the control system, adjusting parameters of the controller by manual experience is usually hard, thus optimization algorithms are applied to obtain the optimal or suboptimal solution. PSO is a typical swarm intelligence algorithm developed in 1995, stemming from research on the foraging behavior of birds [27]. Studies in [28] have shown that it lacks the capability to achieve sustainable development and the swarm becomes stagnant after a certain number of iterations. To improve this, the

term of local optimal particle is introduced in the HPSO algorithm and the velocity and position of particles are updated according to the following two equations:

$$V_i^{K+1} = \omega \cdot V_i^K + c_1 rand(p_i^K - X_i^K) + c_2(q \cdot rand(S_i^K - X_i^K) + (1-q) \cdot rand(p_{iL}^K - X_i^K)) \quad (10)$$

$$X_i^{K+1} = X_i^K + V_i^{K+1} \cdot dt \quad (11)$$

where K is the current number of evolution and at the K-th evolution, V_i^K denotes the velocity of the i-th particle; X_i^K represents the position of the i-th particle; p_i^K depicts the best position of i-th particle; S_i^K is the best position of the particle swarm; p_{iL}^K is the position of the local optimal particle for the i-th particle, and the introduction of p_{iL}^K avoids falling into the local optimal region. ω is inertia weight, which decreases as evolution unfolds and aids in global search in the early stage and local optimization in the late stage. c_1 and c_2 are learning factors, q represents social factor, and dt denotes time interval coefficient.

Instead of single-object optimization, we consider multiple objects in our study, including the steady-state performance and transient performance of AUV motion control system. And the objective function is defined as follows:

$$
\begin{aligned}
F = {} & \log(1 + \tfrac{p\_vibration}{p\_vibration0}) + \log(1 + \tfrac{q\_vibration}{q\_vibration0}) + \log(1 + \tfrac{r\_vibration}{r\_vibration0}) + \log(1 + \tfrac{p\_ts}{p\_ts0}) \\
& + \log(1 + \tfrac{q\_ts}{q\_ts0}) + \log(1 + \tfrac{r\_ts}{r\_ts0}) + \log(1 + \tfrac{z\_ITAE}{z\_ITAE0}) + \log(1 + \tfrac{\psi\_ITAE}{\psi\_ITAE0})
\end{aligned}
\quad (12)
$$

where $(\cdot)\_vibration$ represents oscillation times of AUV angular velocity; $(\cdot)\_ts$ is the settling time in which the angular velocity is kept within a $\pm 5\%$ range of the steady state value; $(\cdot)\_ITAE$ describes the ITAE index of track error; $p\_(vibration0)$ which denotes the expected value and is set artificially, all of the expected values are presented in Appendix A.

The flowchart of optimization is shown in Figure 9 and the implementation procedure of the HPSO algorithm is summarized as follows:

Step 1: Specify the population size and the maximum number of evolutions, as well as other coefficients that can be noted from Equations (10)–(12). Determine the value range of controller parameters and initialize a population of particles with random positions.

Step 2: Evaluate the fitness value of all particles according to Equation (12), let p_i^K of each particle equal to its current position and let S_i^K equal the position of the best initial particle. After that, the local optimal position of each particle, p_{iL}^K, is obtained in accordance with Appendix A.

Step 3: Update the particles' velocities and positions in terms of Equations (10) and (11). Compare the fitness value of each particle with its own best fitness value, if the current one is better, update the p_i^K. Similarly, compare the best fitness value of the new generation with the fitness value of the global best position S_i^K and decide whether it has to be updated. The local optimal position of each particle is also updated.

Step 4: The optimization is terminated when it reaches the maximum number of evolutions. Then the global best position is output, which is the target parameters of the controller. The convergence curve of optimal fitness value is also printed.
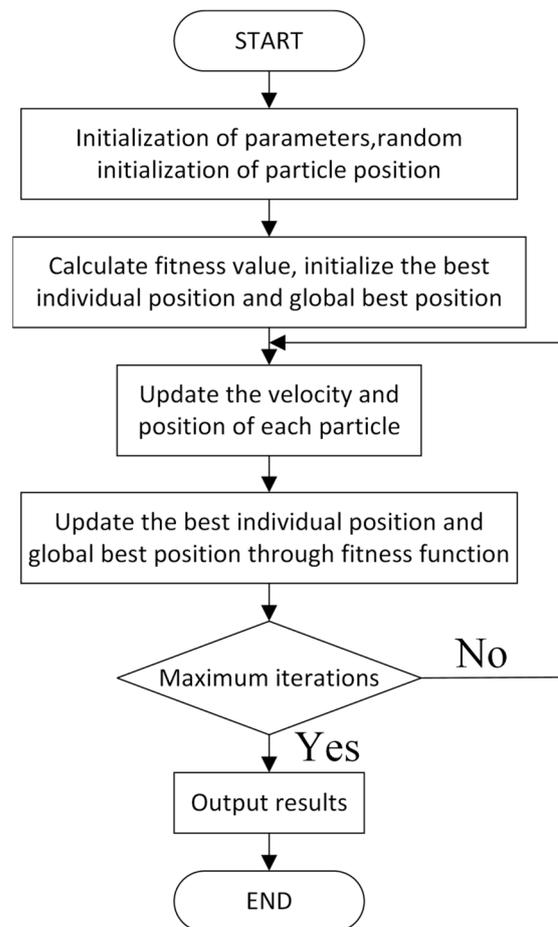
Figure 9. Flowchart of HPSO algorithm.

4.2. Optimization Experiments and Result Analysis

Here we conduct optimization of three control systems by using the HPSO algorithm mentioned above. To the SIFOFLC system, the particle dimension is four with the fractional order of the error signal and the scale factor of the depth controller and heading controller. We adopted the standard Oustaloup filter module deriving from FOTF toolbox [29] to perform fractional derivative operator, the frequency band was set to [0.001,1000] and the filter order was 4. As for the FOPID control system, the dimension of particle is 10, so that each FOPID controller has five unknown parameters. For the T-S FLC control system, only two scale factors are to be optimized. The membership functions and fuzzy rules of T-S FLC were determined through empirical approach, and its control surface is shown in Figure 6.

The coefficients of the HPSO algorithm are set as follows: the particle size is 10 and the maximum number of evolution is 20. The limit of the inertia weight, ω, is set to between 1 and 0.7. The learning factors c_1 and c_2 are set as value 2, the social factor q is set as value 0.7 and the time interval coefficient dt is set as value 0.5. The searching range for parameters are presented in Appendix A. Furthermore, we implement multiple optimizations and adopt the optimal result to solve randomness.

Figures 10–12 respectively show the convergence curve of optimal fitness of three control systems and the obtained target controller parameters are illustrated in Table 3. It can be observed that all the curves tend to decline through evolution, which illustrates the effectiveness of optimization. Actually, the optimal fitness of three control systems respectively decreases by 19.8%, 37.9% and 9.6%. The FOPID controller clearly outperforms the other two controllers as it has the highest degrees of freedom. The optimization effect of the SIFOFLC system is twice as good as the T-S FLC system and its ultimate fitness is significantly less than the others. These results demonstrate that the introduction of a

fractional differential operator not only increases degrees of freedom and the flexibility of the controller but also improves the control performance, as is particularly demonstrated in the next section.

Table 3. Controller parameters optimized by HPSO algorithm.

Control System	Parameter	Heading Controller	Depth Controller
SIFOFLC	α	1.08	1.21
	r	131.02	-102.50
FOPID	K_p	19.88	-15.50
	K_i	18.69	0
	K_d	2.96	-50
	λ	0.10	1.5
	μ	1.11	1.12
T-S FLC	r	33.20	-140.43
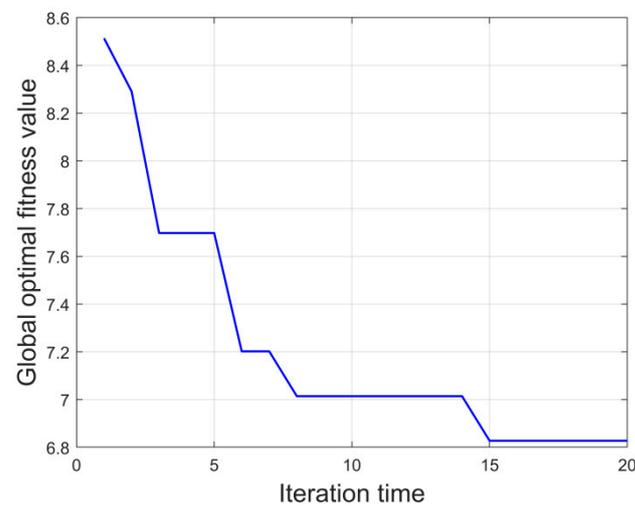
Figure 10. Convergence curve of SIFOFLC system.
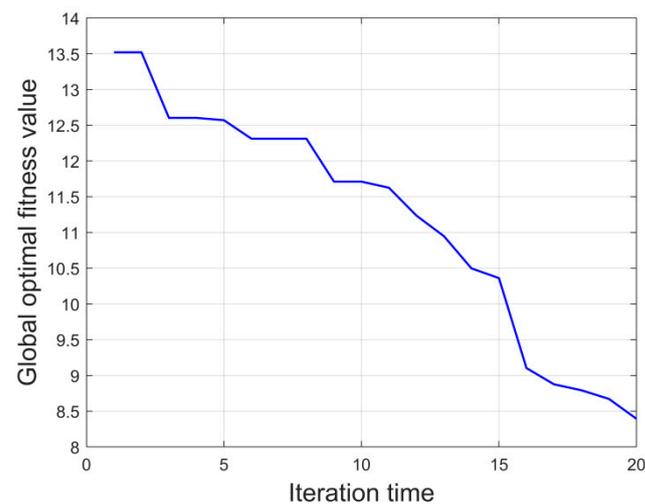
Figure 11. Convergence curve of FOPID control system.
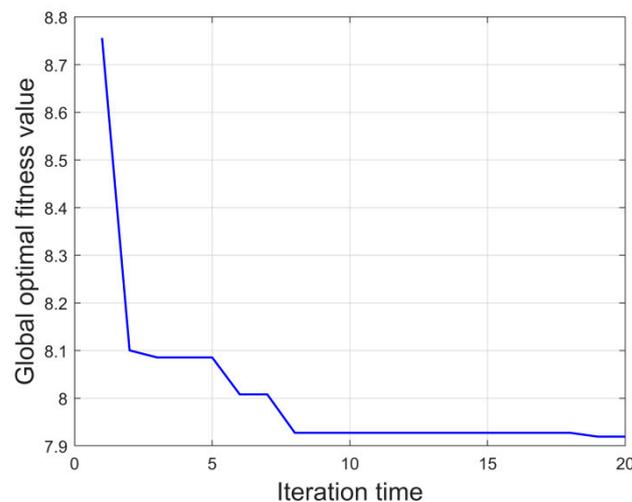
Figure 12. Convergence curve of T-S FLC system.

5. Simulation and Analysis

In this section, we conduct comparative simulations to verify the effectiveness of the proposed SIFOFLC algorithm for AUV motion control. The optimized controllers expressed in Table 3 are applied and the target trajectory of AUV does not change.

Figure 13 shows the trajectory of AUV under SIFOFLC and the state of the response is depicted in Figure 14. Simulation results reveal that the trajectory tracking is accurate and rapid with SIFOFLC. Actually, the steady state error of depth is within 0.01 m in 6.5 s and gradually decreases to approximately 0.002 m in 7 s. Correspondingly, the maximum error of yaw angle is 2.5° and it gradually decreases to 0.01° in 13 s. The cruise speed maintains 1.54 m/s and the peak overshoot of pitch angle is 10°.

Figures 15 and 16 respectively illustrate the angular velocity of an AUV using FOPID controller and T-S FLC. Furthermore, the oscillation and settling times of angular velocity, the ITAE index of track error and the optimal fitness value are all presented in Table 4. The responses with SIFOFLC algorithm have much shorter settling times and steady state errors. Compared with the SIFOFLC, the control performance obtained via the FOPID controller and T-S FLC requires a much longer settling time, and they also oscillate considerably in the beginning, which may lead to an unstable performance of the controlled system. Simulation results clearly reveal the superior stability and transient control performance of the proposed SIFOFLC.
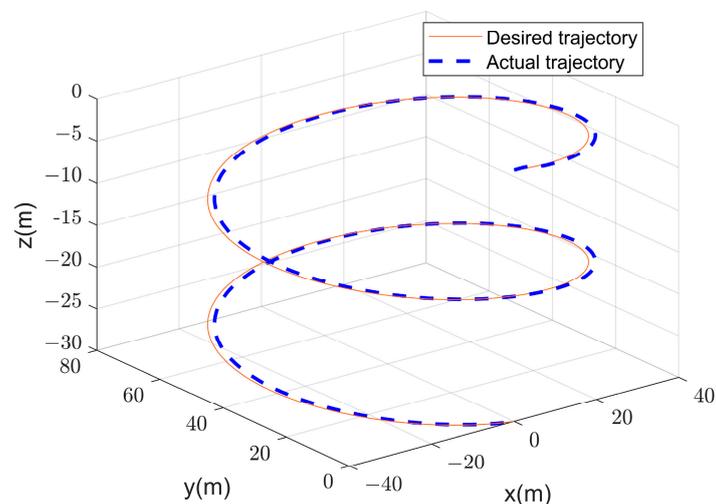

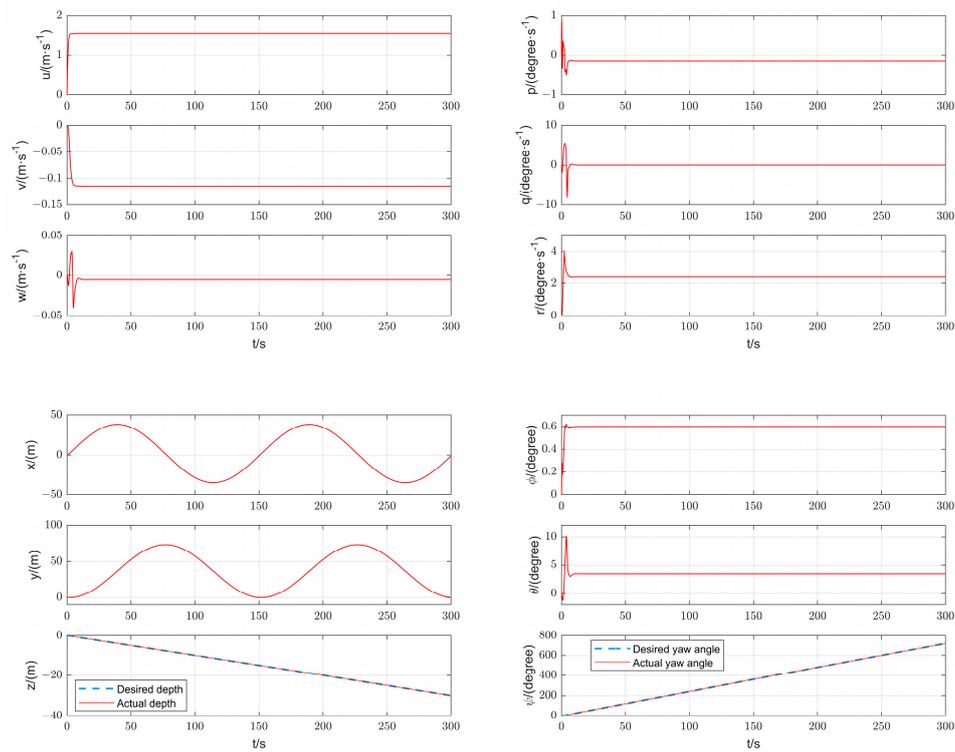**Figure 13.** Comparison of desired trajectory and actual trajectory with SIFOFLC.

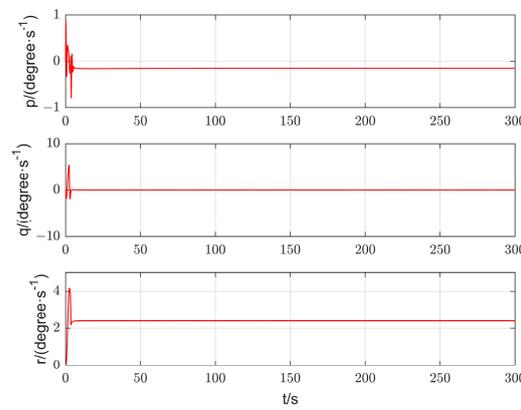**Figure 14.** State response of AUV with SIFOFLC.

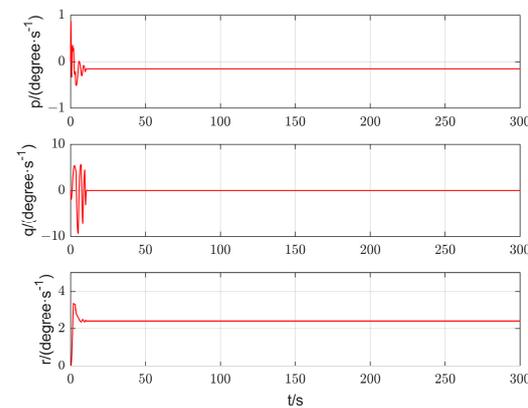**Figure 15.** Angular velocity curves with FOPID.

Figure 16. Angular velocity curves with T-S FLC.

Table 4. Quantitative analysis of system performance.

Characteristic	Control System		
	SIFOFLC	FOPID	T-S FLC
p_vibration	9	12	12
q_vibration	5	5	9
r_vibration	1	3	1
p_ts	8.15	10.62	10.17
q_ts	14.71	18.44	13.53
r_ts	4.97	3.87	5.43
z_ITAE	2.69	7.70	2.90
ψ_ITAE	12.21	14.84	20.38
Fitness value	6.83	8.39	7.92

6. Conclusions

In this study, we proposed a SIFOFLC algorithm for an AUV motion control system. Unlike a conventional FLC algorithm, this is reduced to a SISO controller by using the signed distance method, which provides a significant reduction to parameter tuning and computation burden. In addition, a fractional derivative operator was applied to increase degrees of freedom of the controller, hence the proposed control algorithm is more flexible and adaptive to the AUV motion control system. Furthermore, we developed an HPSO algorithm and applied it to optimize the controller parameters. The simulation results show that the proposed controller enhances the stability and transient performance of the controlled AUV motion system, which manifests in less oscillations of angular velocity, shorter dynamic settling time, and higher control accuracy. In future studies, we will perform experiments using the proposed controller to verify its practicability.

Appendix A

Table A1 shows the searching area of parameters, they are established through trial and error. The target values of characteristics of AUV motion control system, which are defined in Equation (12), are depicted in Table A2. The MATLAB function code for particle distance is given as follows:

```
function DM = Distance_Matrix (X,xmin,xmax)
n = size (X,1);
DM = zeros (n,n);
Normal = xmax-xmin;
for t = 1:n
Dif = (X (t,:)-X)./Normal;
 Dis = sum (Dif.^2,2);
DM (t,:) = Dis;
end
```

where X represents the parameters of particle swarm, xmin and xmax denote the searching bound of parameters. DM is a square matrix and DM (i,j) is the distance between the i-th particle and j-th particle. By the way, if the distance is less than 1, we consider them as local particles.

Table A1. Searching range of parameters for three control systems.

Control System	Parameter	Heading Controller	Depth Controller
SIFOFLC	α	[0.1, 1.5]	[0.1, 1.5]
	r	[5, 150]	[−150, −5]
FOPID	K_p	[0, 20]	[−20, 0]
	K_i	[0, 20]	[−20, 0]
	K_d	[0, 10]	[−50, 0]
	λ	[0.1, 1.5]	[0.1, 1.5]
	μ	[0.1, 1.5]	[0.1, 1.5]
T-S FLC	r	[0, 150]	[−150, 0]

Table A2. Target performance of control system.

Characteristics of Control System	Target Value
p_vibration0	6
q_vibration0	4
r_vibration0	2
p_ts0	5
q_ts0	5
r_ts0	5
z_ITAE0	5
ψ_ITAE0	5

References

1. Ma, Y.T.; Zheng, R. Autonomous underwater vehicle deepening control based on transiting target value nonlinear PID. *Control Theory Appl.* **2018**, *35*, 1120–1125.
2. Kumar, A.; Sharma, R. A Genetic Algorithm based Fractional Fuzzy PID Controller for Integer and Fractional order Systems. *Int. J. Intell. Syst. Appl.* **2018**, *10*, 23–32. [CrossRef]
3. Xu, J.; Wang, M.; Qiao, L. Backstepping-based controller for three-dimentional trajectory of underactuated unmanned underwater vehicles. *Control Theory Appl.* **2014**, *31*, 1589–1596.
4. Ishaque, K.; Abdullah, S.S.; Ayob, S.M.; Salam, Z. A simplified approach to design fuzzy logic controller for an underwater vehicle. *Ocean Eng.* **2011**, *38*, 271–284. [CrossRef]
5. Qiao, L.; Zhang, W. Adaptive Second-Order Fast Nonsingular Terminal Sliding Mode Tracking Control for Fully Actuated Autonomous Underwater Vehicles. *IEEE J. Ocean. Eng.* **2019**, *44*, 363–385. [CrossRef]
6. Wan, L.; Zhang, Y.H. Depth Control of Underactuated AUV Under Complex Environment. *J. Shanghai Jiao Tong Univ.* **2015**, *49*, 1849–1854.

7.   Fan, Q.Y.; Wang, D.S.; Xu, B. H∞ Codesign for Uncertain Nonlinear Control Systems Based on Policy Iteration Method. *IEEE Trans. Cybern.* **2021**, 1–10. [CrossRef]

8.   Fatmawati; Yuliani, E.; Alfiniyah, C. On the Modeling of COVID-19 Transmission Dynamics with Two Strains: Insight through Caputo Fractional Derivative. *Fractal Fract.* **2022**, *6*, 346. [CrossRef]

9.   Monika, S.; Bharat, S.R. Development of Fractional Order Modeling of Voltage Source Converters. *IEEE Access* **2020**, *8*, 131750–131759.

10.   Liu, L.; Zhang, L.C.; Pan, G. Robust yaw control of autonomous underwater vehicle based on fractional-order PID controller. *Ocean Eng.* **2022**, *257*, 111493. [CrossRef]

11.   Ivo, P. Novel Fractional-Order Model Predictive Control: State-Space Approach. *IEEE Access* **2021**, *9*, 92769–92775.

12.   Zheng, W.J.; Huang, R.Q.; Luo, Y. A Look-Up Table Based Fractional Order Composite Controller Synthesis Method for the PMSM Speed Servo System. *Fractal Fract.* **2022**, *6*, 47. [CrossRef]

13.   Liu, L.; Zhang, S.; Zhang, L.C. Multi-Auv Dynamic Maneuver Decision-Making Based on Intuitionistic Fuzzy Counter-Game and Fractional-Order Particle Swarm Optimization. *Fractals-Complex Geom. Patterns Scaling Nat. Soc.* **2021**, *29*, 2140039. [CrossRef]

14.   Liu, L.; Wang, J.; Zhang, L.C. Multi-AUV Dynamic Maneuver Countermeasure Algorithm Based on Interval Information Game and Fractional-Order DE. *Fractal Fract.* **2022**, *6*, 235. [CrossRef]

15.   Zhang, S.; Liu, L.; Xue, D.Y. Nyquist-based stability analysis of non-commensurate fractional-order delay systems. *Appl. Math. Comput.* **2020**, *377*, 125111. [CrossRef]

16.   Zhang, S.; Liu, L.; Xue, D.Y. Stability and resonance analysis of a general non-commensurate elementary fraction-al-order system. *Fract. Calc. Appl. Anal.* **2020**, *23*, 183–210. [CrossRef]

17.   Dai, C.P.; Ma, W.Y. Lyapunov Direct Method for Nonlinear Hadamard-Type Fractional Order Systems. *Fractal Fract.* **2022**, *6*, 405. [CrossRef]

18.   Podlubny, I. Fractional-order systems and PI/sup/spl lambda//D/sup /spl mu//-controllers. *IEEE Trans. Autom. Control* **2002**, *44*, 208–214. [CrossRef]

19.   Zhao, R.; Xu, J.; Wang, M. Heading control of AUV based on GA and fractional order technology. *Chin. J. Ship Res.* **2018**, *13*, 87–93.

20.   Gao, Y.; Fan, J.W. Fractional-order Fuzzy Control Method for Vehicle Nonlinear Active Suspension. *China Mech. Eng.* **2015**, *26*, 1403–1408.

21.   Choi, B.J.; Kwak, S.W.; Kim, B.K. Design and stability analysis of single-input fuzzy logic controller. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* **2000**, *30*, 303–309. [CrossRef] [PubMed]

22.   Allen, B.; Stokey, R. REMUS: A small, low cost AUV.; System Description, Field Trials and Performance Results. In Proceedings of the Ocean '97. MTS/IEEE Conference Proceedings, Halifax, NS, Canada, 6–9 October 1997.

23.   Marine Systems Simulator (MSS). Available online: https://github.com/cybergalactic/MSS (accessed on 8 January 2022).

24.   Prestero, T. Verification of a six-degree of freedom simulation model for the REMUS autonomous underwater vehicle. Master's Thesis, Massachusetts Institute of Technology, Cambridge, UK, 2001.

25.   Xue, D.Y. *Advanced Applied Mathematical Problem Solutions with MATLAB*, 4th ed.; Tsinghua University Publishing House: Beijing, China, 2018; pp. 436–444.

26.   Johnston, R. Fuzzy logic control. *Microelectron. J.* **1995**, *26*, 481–495. [CrossRef]

27.   Kennedy, J.; Eberhart, R. Particle Swarm Optimization. In Proceedings of the Icnn95-international Conference on Neural Networks, Perth, WA, Australia, 27 November–1 December 1995.

28.   Xie, X.F.; Zhang, W.J.; Yang, Z.L. Dissipative particle swarm optimization. In Proceedings of the 2002 Congress on Evolutionary Computation, Honolulu, HI, USA, 12–17 May 2002.

29.   FOTF Toolbox. Available online: https://ww2.mathworks.cn/matlabcentral/fileexchange/60874-fotf-toolbox (accessed on 30 December 2021).