Ocean Engineering 307 (2024) 118171

Contents lists available at ScienceDirect

Ocean Engineering

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

An adaptive PID controller for path following of autonomous underwater vehicle based on Soft Actor–Critic☆

Yuxuan Wang [a], Yaochun Hou [a], Zhounian Lai [b], Linlin Cao [a], Weirong Hong [a], Dazhuan Wu [a,*]

[a] College of Energy Engineering, Institute of Process Equipment, Zhejiang University, Hangzhou 310027, China
[b] Huzhou Institute of Zhejiang University, Huzhou, 313000, China

ARTICLE INFO

Keywords:
Autonomous underwater vehicle
Proportional–Integral–Derivative (PID)
Path following control
Reinforcement learning
Soft Actor–Critic

ABSTRACT

In recent years, autonomous underwater vehicles (AUVs) have witnessed rapid development, and its motion control has garnered increasing attention. Meanwhile, in industries, PID controllers are still wildly employed by most AUVs due to their simplicity, ease of deployment, and a certain level of robustness. However, they are facing significant challenges in parameter tuning, especially when dealing with various control missions and changing external environments. Deep reinforcement learning, as a data-driven approach, has gradually made its impact in AUV control. However, its lack of interpretability has hindered its deployment in relevant experiments. To address these issues, this paper proposed an adaptive PID controller for path following of AUVs based on the Soft Actor–Critic (SAC). This controller combines the interpretability of PID with the intelligence of reinforcement learning. A simulation platform was established and compared with other typical control methods, demonstrating the superiority of the proposed controller. Finally, the feasibility of the proposed SAC-PID controller was validated by lake trials. The results showed that the SAC-PID controller significantly outperformed the PID and Proximal Policy Optimization (PPO) PID controllers in multiple dimensions, such as control precision and convergence speed.

1. Introduction

In recent years, there has been a burgeoning interest in autonomous underwater vehicles (AUVs) across diverse domains. Their high level of autonomy makes them ideal choices for applications in marine science research, underwater geology exploration, and marine ecosystem conservation. AUVs possess the potential to make substantial contributions to human understanding and the sustainable utilization of the underwater world. AUVs face high levels of nonlinearity, coupling, and time-varying dynamics during underwater motion. Thus, ensuring their stable and robust path following remains a crucial research topic in the control community (Kong et al., 2022).

In the literature, many control methods have been emerging, such as Proportional–Integral–Derivative (PID) Control (Park et al., 2009; Bingul and Gul, 2023), backstepping technique (Yu et al., 2019; He et al., 2020; Zhang et al., 2021; Dong et al., 2022; Sedghi et al., 2023; Chen et al., 2023), sliding mode control (SMC) (Elmokadem et al., 2016; Su et al., 2021; An et al., 2022; Zhang et al., 2023), model predictive control (MPC) (Shen et al., 2017; Wei et al., 2019; Shen and Shi, 2020; Yang et al., 2022; Bhat et al., 2022) and so on. However, in the industry, the most commonly used method remains PID control, due to its simplicity, efficiency, ease of understanding and implementation, and clear physical significance. For motion control of AUVs based on PID, the typical approach involves first establishing an as accurate as possible physical model. Then, appropriate PID parameters are tuned on this model. However, during experiments, due to the complexity of AUV systems and the inevitability of modeling errors, the initially tuned parameters often perform poorly. Therefore, further tuning during experiments is necessary to ultimately obtain a more suitable PID controller. Meanwhile, PID parameters finely tuned for a specific operating condition may not be suitable for a wide variety of other conditions.

The rapid advancement of data-driven intelligent methods has garnered widespread attention across various domains, with particular significance in the field of control. These approaches, when confronted

☆ This work was supported in part by the Key Research and Development Program of Zhejiang Province, China under Grant 2022C01047, in part by the Eyas Program Incubation Project of Zhejiang Provincial Administration for Market Regulation, China under Grant CY2022226, and in part by the Fundamental Research Funds for the Central Universities, China under Grant 226-2022-00208.

* Corresponding author.
E-mail addresses: wyx1118@zju.edu.cn (Y. Wang), 12027055@zju.edu.cn (Y. Hou), laizn@hizju.org (Z. Lai), caolinlin@zju.edu.cn (L. Cao), hongwr@zju.edu.cn (W. Hong), wudazhuan@zju.edu.cn (D. Wu).

https://doi.org/10.1016/j.oceaneng.2024.118171
Received 28 February 2024; Received in revised form 10 May 2024; Accepted 11 May 2024
Available online 20 May 2024
0029-8018/© 2024 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

with dynamic environments and diverse system requirements, have the ability to learn and refine control strategies from real-world observations. An adaptive fuzzy control approach was proposed for the control of marine vehicles, introducing a retractable fuzzy approximator, which can globally approximate unknown model dynamics (Wang et al., 2017). In Yan et al. (2019), the radial basis function neural network was used to compensate for the motion uncertainties of the AUV.

In this context, deep reinforcement learning (DRL) emerges as a quintessential machine learning approach, serving as a potent tool for designing intricate controllers in the absence of a priori information (Mnih et al., 2015). In Huang et al. (2023), a general reward function for AUV was designed, capable of accommodating various motion control tasks. The position tracking controller proposed, utilizing the deep deterministic policy gradient (DDPG) algorithm, incorporated roll control for AUVs (Fang et al., 2022). In Anderlini et al. (2019), a comparison was made between DDPG and deep Q network (DQN) in the context of AUV docking control. The study also provided evidence of their rapid deployability in practical applications. A policy network based on an attention mechanism was employed to enhance the extraction of hidden information, and a meta-reinforcement learning framework was introduced to tackle the time-varying dynamics of AUV (Jiang et al., 2021). A reinforcement learning-based obstacle avoidance path tracking controller was proposed, enabling real-time evasion of moving obstacles (Zhang et al., 2022). In order to improve data efficiency in AUV depth control, prioritized experience replay was introduced in Wu et al. (2018). Despite the promising performance of DRL in the simulations of AUV control tasks, its policy's black-box nature can make it vulnerable to unexpected states in complex and dynamic real-world environments. This may lead to catastrophic decisions, such as the AUV sailing too deep, leaping out of the water, or causing collisions. Moreover, understanding the underlying causes of these anomalies can be quite challenging. As a result, there is currently a significant lack of practical experimentation with DRL.

PID controllers have a long-standing history of practical application in the industrial domain, and they are known for their clear principles, stability, and robustness. In the realm of AUV motion control, PID controllers are most commonly employed due to their reliability. These advantages can effectively compensate for the deficiencies arising from the black-box nature of DRL. Furthermore, DRL has the capability to enhance the intelligence of controllers, by simplifying the process of PID parameter tuning, significantly improving adaptability to different operational conditions, and enhancing control effectiveness. In recent years, there have been notable advancements in the control field concerning DRL-PID controllers (Guan and Yamamoto, 2021; Yu et al., 2022). In Wang et al. (2023), a supervised controller based on DRL-PID was employed to enhance the control precision of an indirect-contact heat exchanger. In Carlucho et al. (2020), the effectiveness of DRL-PID in control systems for mobile robots was evaluated using simulation platforms, and the results indicated its superiority over other adaptive PID methods. In Liu et al. (2023), a combination of DDPG and PID was applied to AUV motion control. However, it primarily focused on controlling the AUV's forward velocity and horizontal plane turning velocity, by simplifying the AUV model. To the best of our knowledge, in the aforementioned studies, there is a scarcity of research related to the three-dimensional path following of AUVs, and there is a pronounced lack of experimental validation of its feasibility.

Drawing inspiration from the previous studies, this paper proposes an adaptive PID controller based on DRL. The aim is to enable path following for underactuated AUVs suffering from actuator saturation and unknown external disturbances. Initially, a line-of-sight guidance methodology is employed to transform the AUV's path following mission into angular control problems. Subsequently, tailored state space and reward functions are meticulously designed for the AUV's path following mission to expedite algorithm convergence. Following this, rigorous comparative simulations were conducted between two reinforcement learning algorithms, Proximal Policy Optimization (PPO) (Schulman et al., 2017) and Soft Actor–Critic (SAC) (Haarnoja et al., 2018), both known for their reduced hyperparameter complexity and enhanced stability. Finally, the SAC-PID controller was rigorously validated through lake trials, to further affirm its feasibility and superiority.

The organization of the remaining sections of this paper is as follows: In Section 2, the underactuated AUV model and the path following mission are introduced. Section 3 illuminates the proposed SAC-PID controller in detail. In Section 4, comparative numerical simulations with different controllers are illustrated. In Section 5, experimental validation of the controller's performance are undertaken. Finally, Section 6 concisely conclude this research.

2. Problem formulation

This section describes the six-degree-of-freedom (DOF) mathematical model for the underactuated AUVs. Subsequently, the three-dimensional (3D) guidance law and the PID controller is introduced.

2.1. Underactuated AUV model

Considering an underactuated AUV depicted in Fig. 1, it is equipped with three actuating mechanisms: an aft propeller, a pair of vertical rudders, and a pair of horizontal rudders. Its motion encompasses six degrees of freedom, denoted as surge, sway, heave, roll, pitch, and yaw. Three coordinate systems are utilized, namely, the earth-fixed inertial coordinate system $\{O_e, x_e, y_e, z_e\}$, the body-fixed coordinate system $\{O_b, x_b, y_b, z_b\}$, and the path coordinate system $\{P_k, x_b, y_b, z_b\}$. The origin of the path coordinate system is established at the starting point P_k of the current path, with the x-axis pointing towards the next path point P_{k+1}, the y-axis oriented to the right, and the z-axis, perpendicular to the $P_k - x_p y_p$ plane, directed downwards.

The 6 DOF kinematic and kinetic equations of AUV are expressed in vector form as follows (Fossen, 2011):

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}(\boldsymbol{\eta})\boldsymbol{\nu}, \tag{1}$$

$$\boldsymbol{M}\dot{\boldsymbol{\nu}} + \boldsymbol{C}(\boldsymbol{\nu})\boldsymbol{\nu} + \boldsymbol{D}(\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{\tau}_d + \boldsymbol{\tau}, \tag{2}$$

where $\boldsymbol{\eta} = [x, y, z, \phi, \theta, \psi]^T$ represents the position and orientation of AUV, $\boldsymbol{\nu} = [u, v, w, p, q, r]^T$ denotes the vector encompassing linear and angular velocities, $\boldsymbol{\tau}_d \in \mathcal{R}^6$ signifies the vector of unknown environmental forces, and $\boldsymbol{\tau} = [X, Y, Z, 0, M, N]^T$ stands for the control input vector. Among them, although vector $\boldsymbol{\tau}$ contains five quantities, it only includes three independent control inputs. $\boldsymbol{J}(\boldsymbol{\eta}) \in \mathcal{R}^{6\times6}$, $\boldsymbol{M} \in \mathcal{R}^{6\times6}$, $\boldsymbol{C}(\boldsymbol{\nu}) \in \mathcal{R}^{6\times6}$, and $\boldsymbol{D}(\boldsymbol{\nu}) \in \mathcal{R}^{6\times6}$ respectively represent the Euler angle coordinate transformation matrix, inertial matrix, Coriolis and centripetal matrix, and damping matrix, whose specific forms can be found in this book (Fossen, 2011).

Subsequently, the modeling of control forces is addressed. X represents the thrust generated by the aft propeller, which is relatively straightforward to control. Many times, maintaining a constant propeller speed alone can result in a stable forward velocity. Hence, for the sake of simplicity, this paper omits velocity control. Y, Z, M, and N are associated with the aft rudders, and their modeling is as follows:

$$Y = Y_{uu\delta_v} u^2 \delta_v, \quad N = N_{uu\delta_v} u^2 \delta_v, \tag{3}$$

$$Z = Z_{uu\delta_h} u^2 \delta_h, \quad M = M_{uu\delta_h} u^2 \delta_h, \tag{4}$$

where $\delta_v \in [-\delta_{\max}, \delta_{\max}]$ and $\delta_h \in [-\delta_{\max}, \delta_{\max}]$ represent the angles of the vertical and horizontal rudders, respectively. $Y_{uu\delta_v}$, $N_{uu\delta_v}$, $Z_{uu\delta_h}$, and $M_{uu\delta_h}$ stand for various rudder force coefficients. $\delta_{\max}$ denotes the maximum rudder angle.

To enable precise simulation of AUV dynamics in subsequent sections, this paper sets the simulation time step to 0.05 s. In the lake trials, the AUV's actuating mechanisms, such as the rudder, should not switch angles too frequently due to their inherent performance limitations. Frequent angle changes can also lead to overheating and a series of related issues. Therefore, in the simulations presented later in this paper, the actual rudder angle update time step is designed to be 0.2 s.

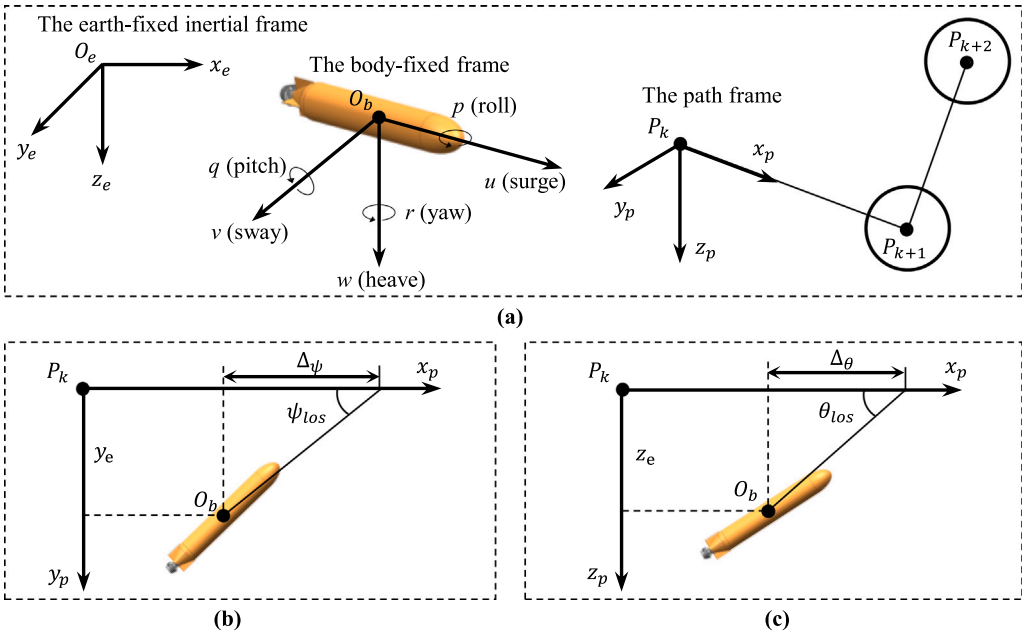


Fig. 1. Illustration of the path following mission of AUV.

2.2. Path following mission

The path following mission refers to the AUV's following of a time-independent path, which, in this paper, involves following a series of spatial waypoints. For this mission, Line-of-Sight (LOS) guidance is frequently used, as it can convert position errors into angular errors.

As shown in Fig. 1, $P_k = (x_k, y_k, z_k)$ represents the starting point of the current path, while $P_{k+1} = (x_{k+1}, y_{k+1}, z_{k+1})$ denotes the next waypoint. From this, the Euler angles ψ_p and θ_p of the path coordinate system relative to the earth-fixed inertial coordinate system can be computed as follows:

$$\psi_p = \arctan(\Delta y, \Delta x), \tag{5}$$

$$\theta_p = -\arctan(\Delta z, \sqrt{\Delta x^2 + \Delta y^2}), \tag{6}$$

where $\Delta x = x_{k+1} - x_k$, $\Delta y = y_{k+1} - y_k$, and $\Delta z = z_{k+1} - z_k$. To avoid singular values, the atan2 function can be used in programming calculations to return the angle. Assume that the current position coordinates of the AUV are denoted as $O_b = (x_0, y_0, z_0)$. The position errors y_e and z_e of the AUV relative to the path coordinate system can be computed using the following expressions:

$$y_e = -(x_0 - x_k) * \sin\psi_p + (y_0 - y_k) * \cos\psi_p, \tag{7}$$

$$z_e = (x_0 - x_k) * \cos\psi_p * \sin\theta_p + (y_0 - y_k) * \sin\psi_p * \sin\theta_p + (z_0 - z_k) * \sin\theta_p. \tag{8}$$

Subsequently, the desired yaw angle ψ_d and pitch angle θ_d for the AUV can be derived as:

$$\psi_d = \psi_p - \arctan(\frac{y_e}{\Delta_\psi}), \tag{9}$$

$$\theta_d = \theta_p - \arctan(\frac{z_e}{\Delta_\theta}), \tag{10}$$

where $\Delta_\psi > 0$ and $\Delta_\theta > 0$ represent lookahead distances.

Following that, the angular errors of the AUV in path following mission can be determined as below:

$$e_\psi = \psi_d - \psi, \tag{11}$$

$$e_\theta = \theta_d - \theta, \tag{12}$$

where ψ and θ represent the yaw and pitch angles of the AUV, respectively.

2.3. PID controller

Compared to position PID, incremental PID offers advantages such as error accumulation avoidance and faster response, making it more suitable for the underwater motion control of AUVs. Therefore, this paper adopts the incremental PID control, as described below:

$$\delta_v(t) = \delta_v(t-1) + K_{v,p}(e_\psi(k) - e_\psi(k-1)) + K_{v,i}e_\psi(k) + K_{v,d}(e_\psi(k) - 2e_\psi(k-1) + e_\psi(k-2)), \tag{13}$$

$$\delta_h(t) = \delta_h(t-1) + K_{h,p}(e_\theta(k) - e_\theta(k-1)) + K_{h,i}e_\theta(k) + K_{h,d}(e_\theta(k) - 2e_\theta(k-1) + e_\theta(k-2)), \tag{14}$$

where $K_{v,p}$, $K_{v,i}$, $K_{v,d}$ represent the proportional, integral, and derivative coefficients of the vertical rudder controller, and $K_{h,p}$, $K_{h,i}$, $K_{h,d}$ represent the coefficients of the horizontal rudder controller, respectively. Among them, the sampling time of the discrete system is incorporated in the integral and derivative coefficients.

The six-degree-of-freedom (DOF) mathematical model of AUV exhibits strong coupling, nonlinearity, and is influenced by complex water flow fluctuations. Therefore, it requires a series of PID parameters to match different situations, which ensures superior control performance for all scenarios, whilst it makes the tuning of PID parameters highly time-consuming and challenging.

3. The proposed adaptive controller

3.1. Soft actor critic

Markov decision process (MDP) is a framework for achieving objectives through the interaction of an agent with its environment, defined by a tuple $(\mathcal{S}, \mathcal{A}, p, r)$. $\mathcal{S}$ and $\mathcal{A}$ represent the state space and action space, respectively. The state transition probability $p : \mathcal{S} \times \mathcal{S} \times \mathcal{A} \to [0, \infty)$ represents the probability that the state will be $\boldsymbol{s}_{t+1} \in \mathcal{S}$ at the next moment given the current state $\boldsymbol{s}_t \in \mathcal{S}$ and action $\boldsymbol{a}_t \in \mathcal{A}$. $r : \mathcal{S} \times \mathcal{A} \to [r_{\min}, r_{\max}]$ denotes the reward provided by the environment.

The reinforcement learning problem can be regarded as policy search defined on a MDP, with the objective of learning policy $\pi(\boldsymbol{a}_t|\boldsymbol{s}_t)$ to maximize the cumulative reward $J(\pi)$. For SAC, an entropy term

$\mathcal{H}(\pi(\cdot|\boldsymbol{s}_t))$ is introduced into the cumulative reward $J(\pi)$, and its expression is as follows:

$$J(\pi) = \sum_{t=0}^{T} \mathbb{E}_{(\boldsymbol{s}_t,\boldsymbol{a}_t)\sim\rho_\pi}[r(\boldsymbol{s}_t,\boldsymbol{a}_t) + \alpha\mathcal{H}(\pi(\cdot|\boldsymbol{s}_t))], \tag{15}$$

where ρ_π stands for the state–action marginal of the trajectory distribution induced by a policy $\pi(\boldsymbol{a}_t|\boldsymbol{s}_t)$. α represents the temperature coefficient, which is used to adjust the importance of the entropy term and the reward term. Thus, while seeking to maximize the reward, the SAC agent also aims to explore different policies to avoid getting trapped in the local optima. This phenomenon will be confirmed in subsequent simulations.

Soft action-value function, also known as soft Q-function, whose Bellman equation can be derived as follows:

$$\begin{aligned} Q^\pi(\boldsymbol{s}_t,\boldsymbol{a}_t) = \sum_{t=0}^{T} \mathbb{E}_{(\boldsymbol{s}_t,\boldsymbol{a}_t)\sim\rho_\pi} \big[\, & r(\boldsymbol{s}_t,\boldsymbol{a}_t) \\ & + \gamma(Q^\pi(\boldsymbol{s}_{t+1},\boldsymbol{a}_{t+1}) + \alpha\mathcal{H}(\pi(\cdot|\boldsymbol{s}_{t+1}))) \big], \end{aligned} \tag{16}$$

where γ is the discount factor.

SAC employs double-Q trick similar to that of Twin Delayed Deep Deterministic Policy Gradient (TD3), utilizing two Q networks for estimation (parameters are represented by θ_1 and θ_2), alongside their corresponding target networks (parameters are represented by $\bar{\theta}_1$ and $\bar{\theta}_2$). Parameters θ_1 and θ_2 can be trained to minimize the following soft Bellman residual:

$$\begin{aligned} J_Q(\theta_i) = \mathbb{E}_{(\boldsymbol{s}_t,\boldsymbol{a}_t,r,\boldsymbol{s}_{t+1},d)\sim\mathcal{D}} \Bigg[\frac{1}{2} \Big(Q_{\theta_i}(\boldsymbol{s}_t,\boldsymbol{a}_t) - \big(r \\ + \gamma(1-d)(\min_{i=1,2} Q_{\bar{\theta}_j}(\boldsymbol{s}_{t+1},\tilde{\boldsymbol{a}}_{t+1}) + \alpha\mathcal{H}(\pi(\tilde{\boldsymbol{a}}_{t+1}|\boldsymbol{s}_{t+1}))) \big) \Big)^2 \Bigg], \end{aligned} \tag{17}$$

where $\mathcal{D}$ represents the replay buffer from which actions $\boldsymbol{a}_t$, states $\boldsymbol{s}_t$ and $\boldsymbol{s}_{t+1}$, rewards r, and completion flags d are sampled. The tilde symbol on $\tilde{\boldsymbol{a}}_{t+1}$ is used to differentiate it from the counterpart being sampled from the replay buffer; rather, it is sampled from the latest output of the policy network.

Next, the policy improvement is exponentially related to the soft Q-function. SAC employs the Kullback–Leibler divergence method as follows:

$$\pi_{\text{new}} = \arg\min_{\pi\in\Pi} \mathrm{D}_{\mathrm{KL}}\left(\pi(\cdot|\boldsymbol{s}_t) \middle\| \frac{\exp(\frac{1}{\alpha}Q^{\pi_{\text{old}}}(\boldsymbol{s}_t,\cdot))}{Z^{\pi_{\text{old}}}(\boldsymbol{s}_t)} \right), \tag{18}$$

where $Z^{\pi_{\text{old}}}(\boldsymbol{s}_t)$ represents partition function. The parameters of the policy network are denoted by $\boldsymbol{\phi}$. For the convenience of gradient computation in policy updates, SAC employs the reparameterization technique to obtain actions:

$$\tilde{\boldsymbol{a}}_t = f_{\boldsymbol{\phi}}(\epsilon_t;\boldsymbol{s}_t), \tag{19}$$

where ϵ_t is the noise vector, sampled from a Gaussian distribution. Parameters $\boldsymbol{\phi}$ can be trained to minimize the following objective function:

$$\begin{aligned} J_\pi(\boldsymbol{\phi}) = \mathbb{E}_{\boldsymbol{s}_t\sim\mathcal{D},\epsilon_t\sim\mathcal{N}} \big[& \alpha\log\pi_{\boldsymbol{\phi}}(f_{\boldsymbol{\phi}}(\epsilon_t;\boldsymbol{s}_t)|\boldsymbol{s}_t) \\ & - \min_{i=1,2} Q_{\theta_j}(\boldsymbol{s}_t, f_{\boldsymbol{\phi}}(\epsilon_t;\boldsymbol{s}_t)) \big]. \end{aligned} \tag{20}$$

The temperature coefficient α is used to adjust the weighting between the reward term and the entropy term. When the agent explores new areas, and the optimal policy is unclear, increasing α facilitates more extensive exploration. Conversely, when the agent has explored sufficiently, and the optimal policy can be roughly determined, α can be reduced to decrease the exploratory nature. Thus, we train α by minimizing the following objective function:

$$J(\alpha) = \mathbb{E}_{\boldsymbol{a}_t\sim\pi_t}[-\alpha\log\pi_t(\boldsymbol{a}_t|\boldsymbol{s}_t) - \alpha\bar{\mathcal{H}}], \tag{21}$$

where $\bar{\mathcal{H}}$ stands for the target entropy.

3.2. Basic settings

The design of the state $\boldsymbol{s}_t$ needs to comprehensively reflect the state of the AUV's motion system. First and foremost, it should include the AUV's pose and velocity, denoted as $\boldsymbol{\eta} = [x, y, z, \phi, \theta, \psi]^T$ and $\boldsymbol{\nu} = [u, v, w, p, q, r]^T$. Additionally, $\boldsymbol{s}_t$ should encompass the current AUV's rudder angles δ_v and δ_h.

The position information $[x, y, z]^T$ represents the AUV's absolute position, but replacing it with relative position information (i.e., PID input errors e_ψ and e_θ) can provide better guidance on steering. For attitude angles ϕ, θ, and ψ, they exhibit periodic characteristics, meaning that multiple angle values differing by 360° represent the same orientation. This characteristic is highly disadvantageous for the SAC agent's learning process. Therefore, we transform these angles into trigonometric forms to eliminate their periodicity, with $\Theta = [\sin\phi, \cos\phi, \sin\theta, \cos\theta, \sin\psi, \cos\psi]^T$. In this paper, the design of $\boldsymbol{s}_t$ is as follows:

$$\boldsymbol{s}_t = [e_\psi, e_\theta, \Theta^T, \boldsymbol{\nu}^T, \delta_v, \delta_h]^T. \tag{22}$$

The action vector $\boldsymbol{a}_t \in \mathcal{R}^6$ consists of the PID coefficients for two sets of PID controllers. Due to the critical importance of safety in AUV motion, we empirically set the range of action parameters as follows: $[0, 10]$, $[0, 1]$, $[0, 10]$ correspond to proportional coefficients, integral coefficients, and derivative coefficients, respectively.

The AUV's rudder angles should not be adjusted too frequently and rapidly, as it can potentially damage the actuating mechanisms. Additionally, to enhance the training convergence speed in the AUV motion control, it is advisable to make positive rewards more easily attainable. Otherwise, if the agent consistently receives negative rewards, it may have a tendency to prematurely terminate the mission. After testing various reward functions, the final chosen form is as follows:

$$r_t = \mathcal{T}e_\psi + \mathcal{T}e_\theta + \beta\frac{\Delta\delta_v}{\delta_{\max}} + \beta\frac{\Delta\delta_h}{\delta_{\max}}, \tag{23}$$

where β serves as the coefficient for adjusting the weights of different reward components, $\Delta\delta_v = \delta_v(t) - \delta_v(t-1)$ and $\Delta\delta_h = \delta_h(t) - \delta_h(t-1)$ represent the changes in rudder angle commands compared to the previous time step, and $\mathcal{T}$ is an operator defined as follows:

$$\mathcal{T}e(t) = \begin{cases} \exp(-|e(t)|), & |e(t)| \le e_{\min} \\ -\dfrac{|e(t)|}{\pi}, & \text{otherwise} \end{cases}, \tag{24}$$

where $e_{\min}$ represents the minimum error of the current episode from 0 up to the current time.

Additionally, similar to TD3, to enhance the stability of the training process, we introduce the policy network training delay and target Q network update delay during the training process.

In summary, this paper introduces an SAC-based adaptive PID controller for 3D path following of AUVs. To further illustrate the updating of various network parameters, we present a structural diagram as shown in Fig. 2. This diagram includes the updating processes of the actor network, critic networks (Q networks), target Q networks, and temperature coefficient α, as well as the deployment process of the actor network in the AUV. These processes are represented in different colors. Additionally, the detailed procedure is depicted in Algorithm 1.

4. Simulation

In this section and the next section, numerical simulation and real experiment will be conducted to validate the superiority of the proposed controller. The underactuated AUV platform used in both cases is a laboratory-developed vehicle equipped with a range of sensors, and it has an aft propeller, a pair of vertical rudders, and a pair of horizontal rudders.

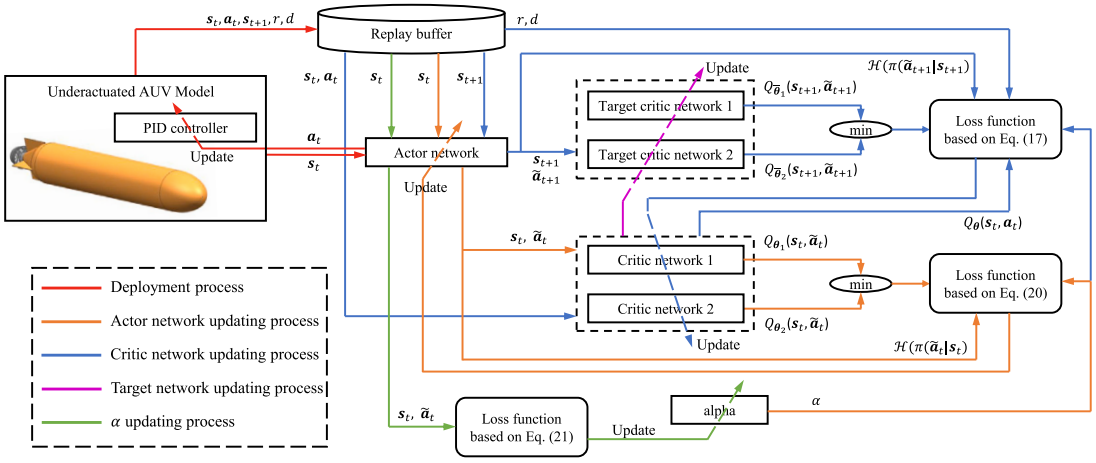


Fig. 2. Structural diagram of the proposed adaptive controller.

Algorithm 1: SAC-based adaptive PID controller

Initialize parameters $\boldsymbol{\theta}_1$, $\boldsymbol{\theta}_2$, $\boldsymbol{\phi}$;
Initialize target Q network parameters $\bar{\boldsymbol{\theta}}_1 \leftarrow \boldsymbol{\theta}_1$, $\bar{\boldsymbol{\theta}}_2 \leftarrow \boldsymbol{\theta}_2$;
Initialize an empty replay buffer $\mathcal{D}$;
for $j = 1$ **to** M **do**
  Reset the initial state $\boldsymbol{s}_0$;
  for $t = 0$ **to** T **do**
    Sample action $\boldsymbol{a}_t$ from policy $\pi_{\boldsymbol{\phi}}(\boldsymbol{a}_t|\boldsymbol{s}_t)$;
    Calculate the control inputs $\Delta\delta_v$ and $\Delta\delta_h$ based on the PID coefficients $\boldsymbol{a}_t$;
    Execute $\Delta\delta_v$ and $\Delta\delta_h$ to obtain $\boldsymbol{s}_{t+1}$, r_t, and d ;
    Store transition tuple $(\boldsymbol{s}_t, \boldsymbol{a}_t, \boldsymbol{s}_{t+1}, r_t, d)$ into the replay buffer $\mathcal{D}$;
    Sample a minibatch of the transition tuples ;
    Update the Q-function parameters by $\boldsymbol{\theta}_i \leftarrow \boldsymbol{\theta}_i - \lambda_Q \hat{\nabla}_{\boldsymbol{\theta}_i} J_Q(\boldsymbol{\theta}_i),\ i = 1, 2$;
    Update the policy network weights by $\boldsymbol{\phi} \leftarrow \boldsymbol{\phi} - \lambda_\pi \hat{\nabla}_{\boldsymbol{\phi}} J_\pi(\boldsymbol{\phi})$;
    Tune the temperature coefficient by $\alpha \leftarrow \alpha - \lambda_\alpha \hat{\nabla}_\alpha J(\alpha)$;
    Update the target Q networks by $\bar{\boldsymbol{\theta}}_i \leftarrow \tau\boldsymbol{\theta}_i + (1-\tau)\bar{\boldsymbol{\theta}}_i,\ i = 1, 2$;
  end
end

Table 1
Series of path points that the AUV needs to follow.

Points	1	2	3	4	5	6	7	8	9
x_d	0	0	0	30	100	170	200	200	200
y_d	0	60	120	190	220	190	120	60	−60
z_d	5	10	15	15	15	20	25	25	25

4.1. Simulation settings

The parameters of the underactuated AUV used are provided as follows:

(1) The mass and moments of inertia about the principal axes: $m = 600$ kg, $I_x = 20$ kg m^2, $I_y = I_z = 280$ kg m^2;
(2) The hydrodynamic added mass: $X_{\dot{u}} = -80$ kg, $Y_{\dot{v}} = Z_{\dot{w}} = -200$ kg, $K_{\dot{p}} = -10$ kg, $M_{\dot{q}} = N_{\dot{r}} = -130$ kg;
(3) The linear damping coefficients: $X_u = -80$ kg/s, $Y_v = Z_w = -530$ kg/s, $K_p = -30$ kg m^2/s, $M_q = N_r = -310$ kg m^2/s;
(4) The quadratic damping coefficients: $X_{u|u|} = -120$ kg/m, $Y_{v|v|} = Z_{w|w|} = -850$ kg/m, $K_{p|p|} = -100$ kg m^2, $M_{q|q|} = N_{r|r|} = -450$ kg m^2;

Table 2
Training parameters.

Parameter	Value
Policy network learning rate λ_π	0.0003
Q network learning rate λ_Q	0.001
Temperature coefficient learning rate λ_α	0.001
Target entropy $\bar{\mathcal{H}}$	−6
Reward discount factor γ	0.99
Target smoothing coefficient τ	0.005
Frequency of training policy	2
Frequency of updates for target Q network	4

(5) The rudder force coefficients of AUV: $Y_{uu\delta_v} = -Z_{uu\delta_h} = 40$ kg/m, $M_{uu\delta_h} = -N_{uu\delta_v} = 120$ kg.

Assuming that the time-varying environmental disturbances acting on the AUV are (Do and Pan, 2009):

$$\boldsymbol{\tau}_d = \begin{bmatrix} 0.15(m - X_{\dot{u}})d(t) \\ 0.15(m - Y_{\dot{v}})d(t) \\ 0.15(m - Z_{\dot{w}})d(t) \\ 0.1(I_x - K_{\dot{p}})d(t) \\ 0.1(I_y - M_{\dot{q}})d(t) \\ 0.1(I_z - N_{\dot{r}})d(t) \end{bmatrix}, \tag{25}$$

where $d(t) = 1 + 0.1(\sin(0.1t) - 1)\mathrm{N}(t)$, and $\mathrm{N}(t)$ denotes Gaussian random noise with mean 0 and variance 1.

The coordinates for the series of points $(P_k, P_{k+1} \ldots)$ in Fig. 1, which represent the path that the AUV follows in the simulation, are list in Table 1. The initial state of the AUV is set as: $\boldsymbol{\eta}(0) = [-30, -30, 4, 0, 0, \pi/4]^{\mathrm{T}}$ and $\boldsymbol{\nu}(0) = [4, 0, 0, 0, 0, 0]^{\mathrm{T}}$. The forward velocity of the AUV is set to 4 m/s.

Furthermore, due to the fact that the roll and pitch angles of the AUV during actual motion should not be too large, we have defined that an episode of the sailing mission will be terminated if ϕ exceeds 30° or θ exceeds 40°.

The simulation of AUV dynamics uses the RK45 method from the SciPy library with a time step of 0.05 s. The SAC training parameters are shown in Table 2.

4.2. Comparison of simulation results

To provide a clearer exposition of the excellence of the proposed method, we conducted in-depth detailed comparison analyses with the following two approaches:

(1) A PID controller with manually tuned parameters.

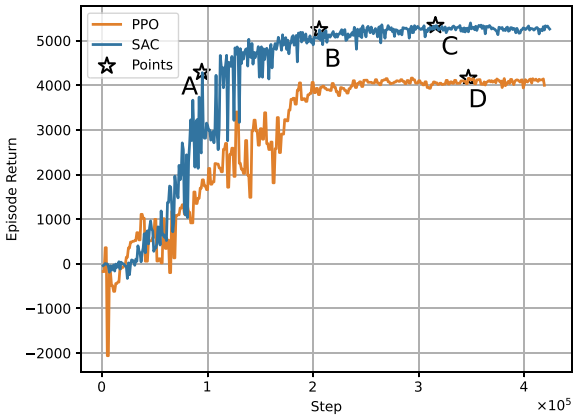


Fig. 3. Learning curves of the training process.

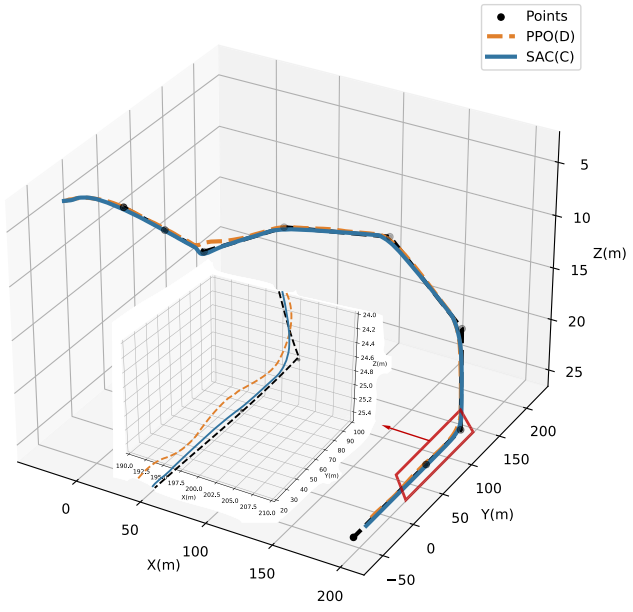


Fig. 4. Trajectory of the AUV based on SAC-PID and PPO-PID.

Table 3
Performance comparisons of three controllers.

		PID	PPO(D)	SAC(A)	SAC(B)	SAC(C)
IAE	e_ψ	120.41	118.67	111.15	67.26	**32.08**
	e_θ	115.55	41.59	59.52	31.42	**20.38**
σ	e_ψ	0.050	0.050	0.048	0.039	**0.030**
	e_θ	0.040	0.023	0.027	0.018	**0.011**

(2) An adaptive PID controller based on PPO, as described in Lai et al. (2023). To ensure a fair comparison, we maintained consistency in algorithm parameters and configurations.

PPO-PID and SAC-PID were trained for motion control of the AUV, and the training curves are shown in Fig. 3. Here, the horizontal axis represents the time steps during training, and the vertical axis represents the cumulative reward of the corresponding episode. In Fig. 3, the blue curve represents the training process based on SAC, while the yellow curve represents the training process based on PPO.

It is evident that the training process of the SAC-PID method is more stable compared to the PPO-PID method. This stability is particularly important in experiments because excessive fluctuation in control effects may make the AUV more prone to collisions, loss, and other catastrophic consequences. Moreover, the SAC-PID demonstrates faster convergence compared to the PPO-PID, significantly reducing training costs in experiments. Notably, the final return from the SAC-PID's training process stabilizes at around 5300, while the return from the PPO-PID stabilizes at around 4100. This indicates a 30% improvement for the former over the latter. It implies that the final control effect of the SAC-PID will be significantly superior to the PPO-PID. The above advantages can be attributed to the SAC algorithm, which considers policy entropy and offers a more comprehensive exploration of the action space, reducing susceptibility to suboptimal solutions.

To further analyze the controller's performance, we selected four typical episodes denoted as A, B, C, and D. Among them, episodes C and D were drawn from the stable phases of SAC and PPO training, while episodes A and B were sampled from the training process of SAC, as indicated in Fig. 3.

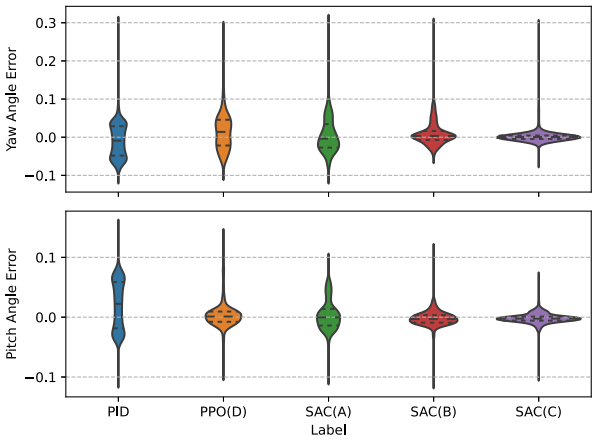


Fig. 5. Violin diagram of AUV's angular errors.

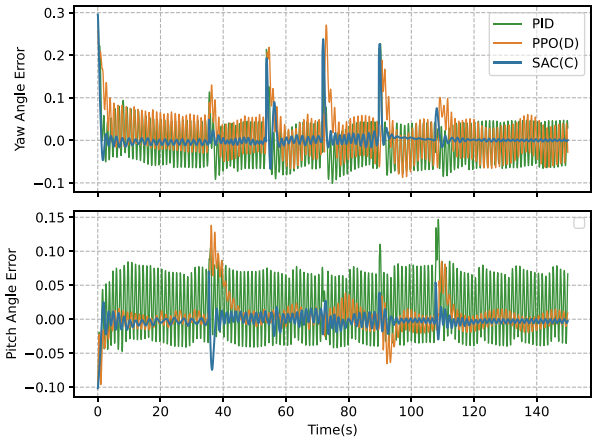


Fig. 6. Variation of AUV's angular errors over time.

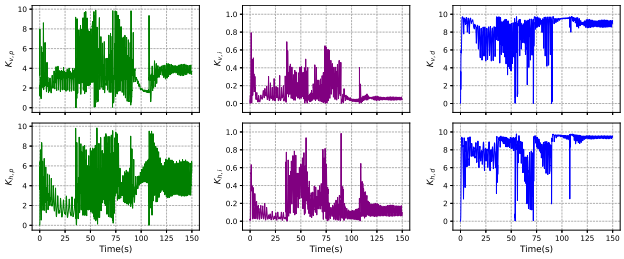
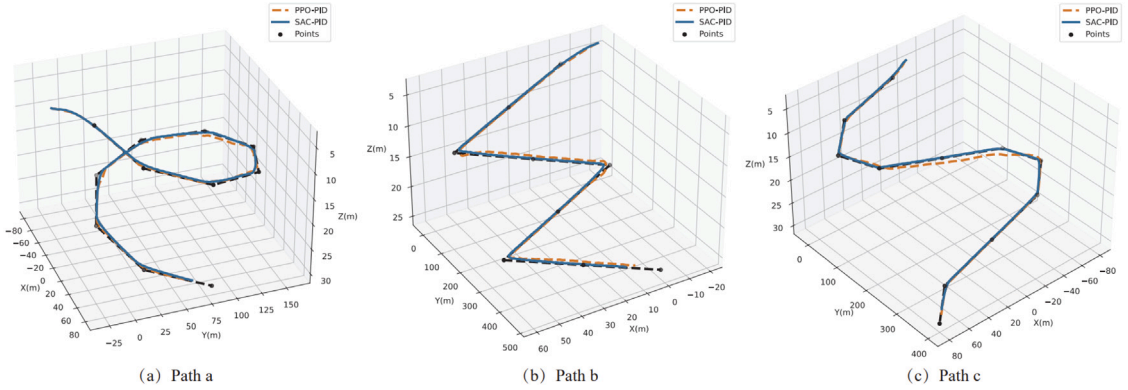


Fig. 7. Variation of PID coefficients based on SAC-PID.



(a) Path a (b) Path b (c) Path c

Fig. 8. Trajectory of the AUV under SAC-PID and PPO-PID on the alternative paths.

After the training of the SAC-PID controller reaches a stable state, taking point C as an example, the trajectory of AUV, controlled by SAC-PID, is depicted by the blue line in Fig. 4. It is evident that the AUV can stably follow the predefined path with minimal fluctuations. Similarly, following the training of the PPO-PID controller reaches a stable state, taking point D as an example, the trajectory of AUV, controlled by PPO, is shown by the yellow line in Fig. 4. It can be observed that the AUV can also follow the predefined path, but with larger fluctuations, indicating a poorer tracking performance compared to SAC.

As described in Section 2.2, the AUV's path following mission involves two angular errors, namely pitch angle error e_θ and yaw angle error e_ψ. We continue to analyze these tracking errors. For comparing the proposed SAC-PID controller with the PID and PPO-PID methods, the violin plot of errors is obtained in Fig. 5. From the violin plot, the distribution of errors can be clearly observed. Among them, the PID method exhibits the most dispersed error distribution, indicating the poorest control performance with a higher proportion of significant errors. After training with the PPO method, the error distribution becomes more concentrated around zero, signifying a notable improvement. With the gradual convergence of SAC, the error distribution becomes increasingly centered around zero, significantly outperforming the other two approaches. The curves depicting the tracking errors over time for the aforementioned methods are shown in Fig. 6. The results further support the conclusion that the SAC-PID controller exhibits the smallest tracking errors.

To precisely compare the control effectiveness of different methods, the errors e_θ and e_ψ are measured using two metrics: integral absolute error (IAE) and standard deviation (σ). The final results are presented in Table 3. It can be observed that after SAC training stabilizes, at point C, the IAEs of tracking errors are minimized. Compared to PPO, they are reduced by 73% and 51%, respectively, and compared to PID, they are reduced by 73% and 82%, respectively. Additionally, the standard deviations of tracking errors are also minimized. Compared to PPO, they are reduced by 40% and 52%, respectively, and compared to PID, they are reduced by 40% and 73%, respectively. This indicates that the proposed SAC-PID controller achieves the highest control accuracy and better robustness.

To further enrich the details of the proposed method, we plotted the variation curve of the PID parameters in the SAC-PID controller, as shown in Fig. 7. It can be observed that throughout the entire motion process of the AUV, the PID parameters are continuously adjusted by the actor network in SAC to achieve the better control performance.

4.3. Simulation results on alternative paths

To avoid the randomness of testing on a single path, simulations were conducted on three other typical paths. The comparison methods used here are consistent with Section 4.2. The final three-dimensional path following results are shown in Fig. 8, indicating that the SAC-PID controller achieves the best performance, enabling the AUV to follow the predefined paths more accurately.

Table 4
Performance comparisons of three controllers on the alternative paths.

		Path a			Path b			Path c		
		PID	PPO	SAC	PID	PPO	SAC	PID	PPO	SAC
IAE	e_ψ	147.96	114.54	**71.69**	148.15	95.42	**54.70**	156.28	106.26	**80.30**
	e_θ	113.86	55.56	**39.35**	98.44	40.89	**25.09**	114.91	54.35	**27.84**
σ	e_ψ	0.071	0.054	**0.050**	0.122	0.097	**0.070**	0.087	0.076	**0.071**
	e_θ	0.039	0.023	**0.017**	0.039	0.019	**0.011**	0.039	0.023	**0.012**

(a) AUV resting on water surface (b) AUV on a dive

Fig. 9. Lake trial of AUV.

To provide a detailed comparison of the control effectiveness, the errors e_θ and e_ψ are measured using two metrics: integral absolute error (IAE) and the standard deviation (σ). The final results are presented in Table 4, confirming that the SAC-PID controller achieves the smallest tracking errors across different paths.

In conclusion, through the aforementioned simulation verification and comparison, the results show that the proposed SAC-PID controller significantly outperforms the PID and PPO-PID controllers in multiple dimensions, such as control precision and convergence speed.

5. Experiment

5.1. Experiment settings

To further validate the effectiveness and reliability of the proposed controller, we conducted lake trials using our laboratory-developed AUV platform, as shown in Fig. 9. This underactuated AUV is equipped with an aft propeller, a pair of vertical rudders, and a pair of horizontal rudders.

The hardware framework of our laboratory-developed AUV platform is illustrated in Fig. 10, where the control system is designed for both the shipboard and AUV ends. Communication between them is facilitated via 433MHz wireless communication module, allowing for the exchange of position information via GPS. The AUV's hardware

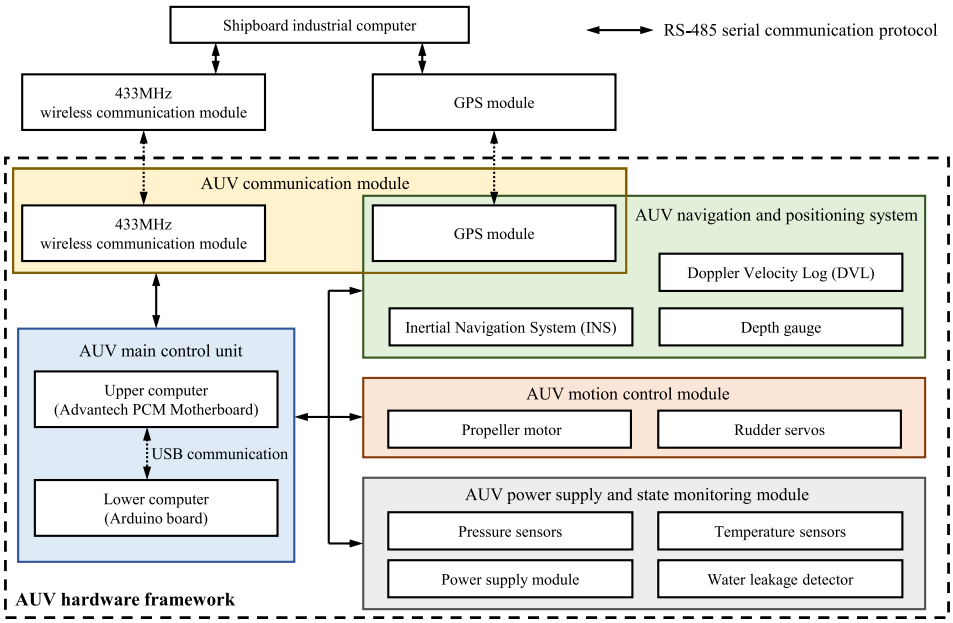


Fig. 10. Hardware framework of the laboratory-developed AUV platform.

framework comprises the main control unit, communication module, navigation and positioning module, motion control module, and power supply and state monitoring module. The main control unit consists of an Advantech PCM motherboard and an Arduino board. The communication module includes a wireless communication module and GPS module for communication with the shipboard control system. The navigation and positioning module incorporates Doppler Velocity Log (DVL), Inertial Navigation System (INS), depth gauge, and GPS, with GPS utilized solely for surface positioning. The motion control module consists of propeller motor and rudder servos. The power supply and state monitoring module is responsible for managing power supply and monitoring AUV's health, incorporating pressure sensors, temperature sensors, water leakage detectors, and the power system.

The lake trial process roughly involves the following steps: Initially, deploy the AUV at a suitable location on the water surface. Additionally, remotely control the AUV to move on the water surface for a short distance using the wireless communication module, until it reaches the vicinity of the designated dive location. Send a dive command to the AUV, after which it enters automatic mode. Initially, it increases motor speed to submerge, then it runs the control algorithms to follow the predefined path. After completing the following mission, it executes the surfacing procedure to resurface. Throughout the mission, the emergency subsystem continually monitors the operational status of the AUV. If any anomalies are detected, the subsystem will terminate the mission and activate the emergency surfacing procedure to prevent damage or loss of the AUV.

Given the importance of exploration in SAC and the potential risks and hazards in high-speed AUV trials, we implemented the following safety measures:

(1) When the AUV experiences significant sailing deviations, especially in depth, or when pitch and roll angles deviate significantly from the expected range, the current episode is terminated. The AUV is then commanded to surface actively.
(2) The AUV model used in the simulation is the same as the one used in the experiments in this section. To expedite convergence during experiments, the network parameters in the experiments were initialized with the pre-trained parameters from the simulation. It is important to note that this step was taken solely to reduce experimental costs, and the algorithm itself remained a model-free approach.

Table 5
Experimental training results.

Episode	Cumulative reward
10th	1125.3
20th	1181.3
30th	1230.9
40th	1384.7
50th	1554.6
60th	1620.5
70th	1643.1

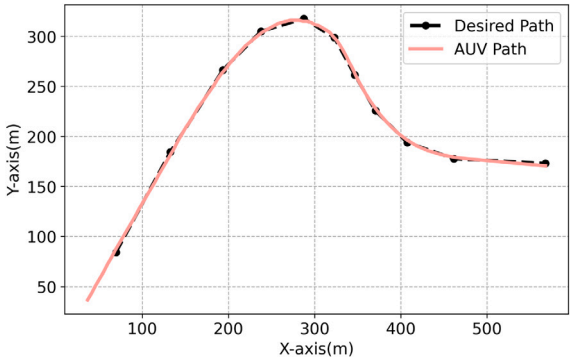


Fig. 11. Visualization of AUV's horizontal trajectory.

5.2. Experiment results

The AUV experiments consisted of over 70 episodes, and the results are presented in Table 5. It can be observed that as the training progresses, the SAC agent learns valuable knowledge through interaction with the environment, leading to a continuous increase in cumulative rewards. It is also notable that the network parameters pretrained through simulations provide a reasonable starting point for the experiments, and they do not represent the optimal parameters discovered during the experiments.

For visualization purposes, a specific set of episodes from the final convergence phase is selected for demonstration. Figs. 11 and 12 depict the AUV's horizontal trajectory and depth trajectory, respectively. These figures illustrate that the AUV can accurately follow the

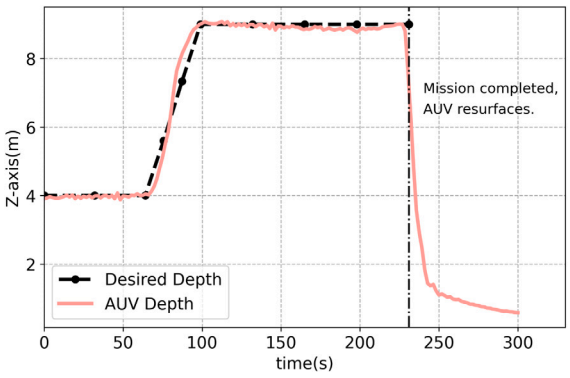


Fig. 12. Visualization of AUV's depth trajectory.

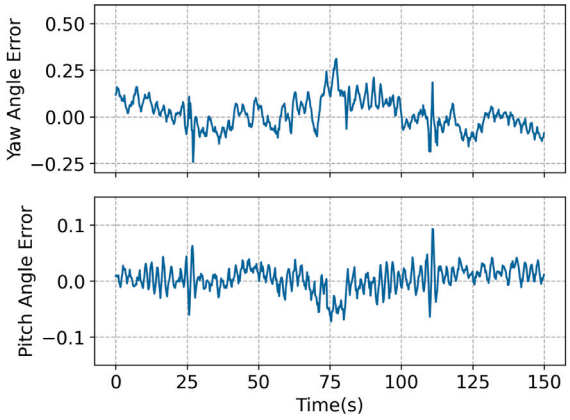


Fig. 13. Variation of AUV's angular errors over time. The IAEs of pitch angle error e_θ and yaw angle error e_ψ are 45.25 and 11.04, respectively. The standard deviations of pitch angle error e_θ and yaw angle error e_ψ are 0.081 and 0.022, respectively.

predefined path and maintain the desired depth during the trials. The curves depicting the tracking errors over time are shown in Fig. 13. The IAEs of pitch angle error e_θ and yaw angle error e_ψ are 45.25 and 11.04, respectively. The standard deviations of pitch angle error e_θ and yaw angle error e_ψ are 0.081 and 0.022, respectively. The error results indicate that the proposed SAC-PID controller still maintains high control accuracy in the AUV's lake trials.

In summary, the experimental results above demonstrate that the proposed adaptive SAC-PID controller can optimize control parameters and enhance control performance online during the AUV's actual underwater motion. The algorithm converges rapidly and exhibits high control accuracy, indicating strong engineering applicability.

6. Conclusion

In this paper, an adaptive PID controller based on the SAC for the control of AUV path following is introduced. This controller combines the interpretability of PID with the intelligence of reinforcement learning. It addresses issues such as difficult parameter tuning, challenges in handling various operating conditions and time-varying external disturbances, and the lack of clarity in control principles. Subsequently, a simulation platform is developed, and comparative analyses with other typical control methods demonstrate the superiority of the proposed controller. Finally, we validate the advancement and feasibility of SAC-PID as a controller through lake trials, addressing the current lack of experimental verification in this field. The results show that the SAC-PID controller significantly outperforms the PID and PPO-PID controllers in multiple dimensions, such as control precision and convergence speed. It underscores the successful integration of reinforcement learning algorithms with traditional control methods in practical systems. This integration ensures safety and enhances both adaptability and intelligence. This method is not only applicable to AUVs' control but also holds significance for the control of other mobile robots and complex industrial processes.

In our future work, we will focus on the following aspects. Firstly, we will consider control in scenarios involving sensor and actuator failures, which is likely to occur in the complex underwater environment. Secondly, we will utilize methods such as computational fluid dynamics to model parameters such as environmental disturbances more accurately. This will bring the simulation closer to real experimental scenarios. Finally, while the proposed controller is only capable of path following missions, our attention will focus on trajectory tracking control with temporal constraints on the path.

CRediT authorship contribution statement

Yuxuan Wang: Writing – original draft, Software, Methodology, Investigation, Conceptualization. **Yaochun Hou:** Methodology, Investigation, Data curation. **Zhounian Lai:** Software, Methodology. **Linlin Cao:** Supervision, Investigation. **Weirong Hong:** Project administration, Conceptualization. **Dazhuan Wu:** Writing – review & editing, Project administration, Funding acquisition.

Declaration of competing interest

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

Data availability

No data was used for the research described in the article.

References

An, S., Wang, L., He, Y., Yuan, J., 2022. Adaptive backstepping sliding mode tracking control for autonomous underwater vehicles with input quantization. Adv. Theory Simul. 5 (4), 2100445.

Anderlini, E., Parker, G.G., Thomas, G., 2019. Docking control of an autonomous underwater vehicle using reinforcement learning. Appl. Sci. 9 (17), 3456.

Bhat, S., Panteli, C., Stenius, I., Dimarogonas, D.V., 2022. Nonlinear model predictive control for hydrobatics: Experiments with an underactuated AUV. J. Field Robotics.

Bingul, Z., Gul, K., 2023. Intelligent-PID with PD feedforward trajectory tracking control of an autonomous underwater vehicle. Machines 11 (2), 300.

Carlucho, I., De Paula, M., Acosta, G.G., 2020. An adaptive deep reinforcement learning approach for MIMO PID control of mobile robots. ISA Trans. 102, 280–294.

Chen, H., Tang, G., Wang, S., Guo, W., Huang, H., 2023. Adaptive fixed-time backstepping control for three-dimensional trajectory tracking of underactuated autonomous underwater vehicles. Ocean Eng. 275, 114109.

Do, K.D., Pan, J., 2009. Control of Ships and Underwater Vehicles: Design for Underactuated and Nonlinear Marine Systems, vol. 1, Springer.

Dong, B., Lu, Y., Xie, W., Huang, L., Chen, W., Yang, Y., Zhang, W., 2022. Robust performance-prescribed attitude control of foldable wave-energy powered auv using optimized backstepping technique. IEEE Trans. Intell. Veh. 8 (2), 1230–1240.

Elmokadem, T., Zribi, M., Youcef-Toumi, K., 2016. Trajectory tracking sliding mode control of underactuated AUVs. Nonlinear Dynam. 84, 1079–1091.

Fang, Y., Huang, Z., Pu, J., Zhang, J., 2022. AUV position tracking and trajectory control based on fast-deployed deep reinforcement learning method. Ocean Eng. 245, 110452.

Fossen, T.I., 2011. Handbook of Marine Craft Hydrodynamics and Motion Control. John Wiley & Sons.

Guan, Z., Yamamoto, T., 2021. Design of a reinforcement learning PID controller. IEEJ Trans. Electr. Electron. Eng. 16 (10), 1354–1360.

Haarnoja, T., Zhou, A., Hartikainen, K., Tucker, G., Ha, S., Tan, J., Kumar, V., Zhu, H., Gupta, A., Abbeel, P., et al., 2018. Soft actor-critic algorithms and applications. arXiv preprint arXiv:1812.05905.

He, S., Kou, L., Li, Y., Xiang, J., 2020. Robust orientation-sensitive trajectory tracking of underactuated autonomous underwater vehicles. IEEE Trans. Ind. Electron. 68 (9), 8464–8473.

Huang, F., Xu, J., Wu, D., Cui, Y., Yan, Z., Xing, W., Zhang, X., 2023. A general motion controller based on deep reinforcement learning for an autonomous underwater vehicle with unknown disturbances. Eng. Appl. Artif. Intell. 117, 105589.

Jiang, P., Song, S., Huang, G., 2021. Attention-based meta-reinforcement learning for tracking control of AUV with time-varying dynamics. IEEE Trans. Neural Netw. Learn. Syst. 33 (11), 6388–6401.

Kong, S., Sun, J., Wang, J., Zhou, Z., Shao, J., Yu, J., 2022. Piecewise compensation model predictive governor combined with conditional disturbance negation for underactuated AUV tracking control. IEEE Trans. Ind. Electron. 70 (6), 6191–6200.

Lai, P., Liu, Y., Zhang, W., Xu, H., 2023. Intelligent controller for unmanned surface vehicles by deep reinforcement learning. Phys. Fluids 35 (3).

Liu, R., Cui, Z., Lian, Y., Li, K., Liao, C., Su, X., 2023. AUV adaptive PID control method based on deep reinforcement learning. In: 2023 China Automation Congress. CAC, IEEE, pp. 2098–2103.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., et al., 2015. Human-level control through deep reinforcement learning. Nature 518 (7540), 529–533.

Park, J.-Y., Jun, B.-h., Lee, P.-m., Oh, J., 2009. Experiments on vision guided docking of an autonomous underwater vehicle using one camera. Ocean Eng. 36 (1), 48–61.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O., 2017. Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347.

Sedghi, F., Arefi, M.M., Abooee, A., 2023. Command filtered-based neuro-adaptive robust finite-time trajectory tracking control of autonomous underwater vehicles under stochastic perturbations. Neurocomputing 519, 158–172.

Shen, C., Shi, Y., 2020. Distributed implementation of nonlinear model predictive control for AUV trajectory tracking. Automatica 115, 108863.

Shen, C., Shi, Y., Buckham, B., 2017. Trajectory tracking control of an autonomous underwater vehicle using Lyapunov-based model predictive control. IEEE Trans. Ind. Electron. 65 (7), 5796–5805.

Su, B., Wang, H., Li, N., 2021. Event-triggered integral sliding mode fixed time control for trajectory tracking of autonomous underwater vehicle. Trans. Inst. Meas. Control 43 (15), 3483–3496.

Wang, X., Cai, J., Wang, R., Shu, G., Tian, H., Wang, M., Yan, B., 2023. Deep reinforcement learning-PID based supervisor control method for indirect-contact heat transfer processes in energy systems. Eng. Appl. Artif. Intell. 117, 105551.

Wang, N., Su, S.-F., Yin, J., Zheng, Z., Er, M.J., 2017. Global asymptotic model-free trajectory-independent tracking control of an uncertain marine vehicle: An adaptive universe-based fuzzy control approach. IEEE Trans. Fuzzy Syst. 26 (3), 1613–1625.

Wei, H., Shen, C., Shi, Y., 2019. Distributed Lyapunov-based model predictive formation tracking control for autonomous underwater vehicles subject to disturbances. IEEE Trans. Syst. Man Cybern.: Syst. 51 (8), 5198–5208.

Wu, H., Song, S., You, K., Wu, C., 2018. Depth control of model-free AUVs via reinforcement learning. IEEE Trans. Syst. Man Cybern.: Syst. 49 (12), 2499–2510.

Yan, Z., Wang, M., Xu, J., 2019. Robust adaptive sliding mode control of underactuated autonomous underwater vehicles with uncertain dynamics. Ocean Eng. 173, 802–809.

Yang, N., Chang, D., Johnson-Roberson, M., Sun, J., 2022. Energy-optimal control for autonomous underwater vehicles using economic model predictive control. IEEE Trans. Control Syst. Technol. 30 (6), 2377–2390.

Yu, X., Fan, Y., Xu, S., Ou, L., 2022. A self-adaptive SAC-PID control approach based on reinforcement learning for mobile robots. Int. J. Robust Nonlinear Control 32 (18), 9625–9643.

Yu, C., Xiang, X., Wilson, P.A., Zhang, Q., 2019. Guidance-error-based robust fuzzy adaptive control for bottom following of a flight-style AUV with saturated actuator dynamics. IEEE Trans. Cybern. 50 (5), 1887–1899.

Zhang, C., Cheng, P., Du, B., Dong, B., Zhang, W., 2022. AUV path tracking with real-time obstacle avoidance via reinforcement learning under adaptive constraints. Ocean Eng. 256, 111453.

Zhang, W., Wu, W., Li, Z., Du, X., Yan, Z., 2023. Three-dimensional trajectory tracking of AUV based on nonsingular terminal sliding mode and active disturbance rejection decoupling control. J. Mar. Sci. Eng. 11 (5), 959.

Zhang, J., Xiang, X., Lapierre, L., Zhang, Q., Li, W., 2021. Approach-angle-based three-dimensional indirect adaptive fuzzy path following of under-actuated AUV with input saturation. Appl. Ocean Res. 107, 102486.