sciendo

POLISH MARITIME RESEARCH 4 (116) 2022 Vol. 29; pp. 36-44
10.2478/pomr-2022-0042

THREE-DIMENSIONAL PATH-FOLLOWING CONTROL OF AN AUTONOMOUS UNDERWATER VEHICLE BASED ON DEEP REINFORCEMENT LEARNING

Zhenyu Liang [1,2]
Xingru Qu [1*]
Zhao Zhang [2]
Cong Chen [2]

[1] School of Mechanical and Electronic Engineering, Dalian Minzu University, Dalian, China
[2] School of Naval Architecture and Ocean Engineering, Dalian Maritime University, Dalian, China

* Corresponding author: *quxingru@126.com (X. Qu)*

ABSTRACT

In this article, a deep reinforcement learning based three-dimensional path following control approach is proposed for an underactuated autonomous underwater vehicle (AUV). To be specific, kinematic control laws are employed by using the three-dimensional line-of-sight guidance and dynamic control laws are employed by using the twin delayed deep deterministic policy gradient algorithm (TD3), contributing to the surge velocity, pitch angle and heading angle control of an underactuated AUV. In order to solve the chattering of controllers, the action filter and the punishment function are built respectively, which can make control signals stable. Simulations are carried out to evaluate the performance of the proposed control approach. And results show that the AUV can complete the control mission successfully.

Keywords: Autonomous underwater vehicle (AUV), three-dimensional path following, deep reinforcement learning-based control, line-of-sight guidance, controller chattering

INTRODUCTION

As an efficient underwater operation equipment, the autonomous underwater vehicle (AUV) can complete many missions, such as subsea creature monitoring, marine hydrological environment detection, and seafloor mapping [1-3], and path-following control is one of the core technologies for AUVs to successfully complete those missions [4, 5].

Now, fruitful research approaches focused on model-based methods are being employed for the path-following control of an AUV, such as PID control, sliding mode control, fuzzy control and model predictive control [6-8]. Wan et al. propose a multi-strategy fusion control with delay method, avoiding the chattering caused by frequent switching [9]. Xia et al. combine the Lyapunov method with line-of-sight guidance to design dynamic control laws, and utilize fuzzy parameter optimization to solve the chattering of controllers [10]. Fang et al. propose a neural network-based gain observer to design dynamic and kinematic controllers [11]. Zhang et al. design an adaptive neural network controller, approximating the model uncertainties [12]. A sliding mode control-based procedure for the design of model predictive control is proposed in [13]. These control approaches are based on the constructed mathematical model, where the design of the controllers or the control laws depend on the model. However, an accurate model of an AUV is difficult to acquire directly because of the system complexity and underwater environment. Considering the practical constraints, the reinforcement learning (RL) based model-free control approach is receiving more attention and provides a promising alternative for motion control [14].

The RL vehicle learns the end-to-end connection between observations and actions by interacting with environment, where

the control design is implemented regardless of the system model [15]. The deep neural networks with arbitrary approximation capabilities can represent the complex relationship between inputs and outputs, and accurately learn from the dynamics of the model. With the aid of deep neural networks, the deep RL (DRL) achieves very impressive results. For example, by decomposing the path-following control into waypoint following control, Fang et al. design a Deep Deterministic Policy Gradient (DDPG)-based three-dimensional motion controller for an AUV [16]. Zielinski et al. propose vision-based navigation control laws for an AUV using the Advantage Actor-Critic (A2C) framework, and compare the control performance of different image processing methods [17]. Song et al. utilize the DDPG to design an end-to-end control method for target tracking of AUVs, avoiding the controller design through the complex dynamic model [18]. Meyer et al. simplify the sonar inputs and design an end-to-end controller by Proximal Policy Optimization (PPO), where collision avoidance is also considered [19]. Martinsen et al. utilize the DDPG to design a straight path-following controller of an underactuated vehicle, and the Gaussian reward is used to replace the traditional boundary reward, which significantly improves the training efficiency [20]. However, the above DRL-based control researches mainly focus on efficiency and accuracy, and have not considered the control chattering. The chronic chattering of controllers will impair the actuator performance. Moreover, there are lots of states and network nodes in the network design, slowing the network training and algorithm convergence.

Given the potential of model-free DRL in path-following control, this article researches the three-dimensional path-following control of an underactuated AUV based on the DRL. The path-following control is transformed into the heading control and pitch control. Kinematic control laws are employed using the three-dimensional line-of-sight (LOS) guidance, and dynamic control laws are employed using the twin delayed deep deterministic policy gradient (TD3). Besides, taking controller chattering into consideration, the smooth reward function and a first-order filter are introduced and thus make the control inputs smooth. Simulation results show that the proposed DRL-based method is capable of providing the required vehicle control along a three-dimensional path. Moreover, compared to the DDPG proposed in [16], the proposed control approach has remarkable effectiveness and superiority.

The rest of the article is organized as follows. The second section presents the problem formulation on the path-following of an AUV. The third section presents the DRL-based threedimensional path-following control design, including kinematic guidance design, dynamic control design and TD3 design. In the fourth section, simulation results and comparisons are provided to validate the proposed control approach. Finally, the last section concludes this article and indicates further work.

PROBLEM FORMULATION

As shown in Fig. 1, two coordinate frames in the three-dimensional space are introduced, including the earth-fixed frame $\xi\eta\zeta$ and the body-fixed frame xyz. P_i represents the desired waypoint, where $i = 1,..., n$. And the desired path can be obtained by connecting the neighbouring waypoints.

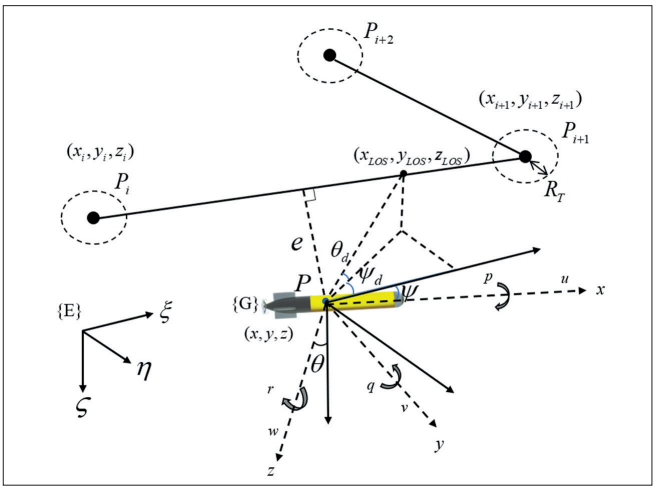


Fig. 1. Coordinate frames in three-dimensional space

According to [21], the roll motion of the AUV can be ignored and the motion model with five degrees-of-freedom can be written as

$$\begin{cases} \dot{\eta} = J(\eta)\nu \\ M\dot{\nu} + C(\nu)\nu + D(\nu)\nu + g(\eta) = \tau \end{cases} \tag{1}$$

where $\eta = [\xi, \eta, \zeta, \theta, \psi]^T$ with (ξ, η, ζ) being the positions of the AUV in the earth-fixed frame, and θ and ψ being the pitch angle and heading angle, respectively. $\nu = [u, v, w, q, r]^T$ represents the velocity of the AUV in the body-fixed frame, with u, v, w being the surge, sway, and heave velocities and q, r being the pitch and yaw velocities. M is the inertia matrix, $C(\nu)$ is the Coriolis-centripetal matrix. $D(\nu)$ is the fluid damping matrix. $g(\eta)$ is the restoring force vector. $\tau = [\tau_u, 0, 0, \tau_q, \tau_r]^T$ is the control inputs including the surge force τ_u, the pitch torque τ_q, and the yaw torque τ_r. $J(\eta)$ is the rotation matrix from the body-fixed frame to the earth-fixed frame, which is written as

$$J(\eta) = \begin{bmatrix} J_1(\eta) & 0_{3\times3} \\ 0_{2\times2} & J_2(\eta) \end{bmatrix} \tag{2}$$

with

$$J_1(\eta) = \begin{bmatrix} \cos\psi\cos\theta & -\sin\psi & \cos\psi\sin\theta \\ \sin\psi\cos\theta & \cos\psi & \sin\psi\sin\theta \\ -\sin\theta & 0 & \cos\theta \end{bmatrix}, \quad J_2(\eta) = \begin{bmatrix} 1 & 0 \\ 0 & \dfrac{1}{\cos\theta} \end{bmatrix} \tag{3}$$

Taking practical engineering into consideration, the pitch angle of an underactuated AUV is restricted as $-\pi/2<\theta<\pi/2$, and the heading angle is restricted as $-\pi/\psi<\pi$.

In this article, our objective is to design the DRL-based three-dimensional path-followingcontrol approach of an underactuated AUV such that the vehicle can follow the desired path. A geometrical illustration of the DRL-based path-following control is shown in Fig. 2. For the kinematic design,

the desired path is obtained by connecting the neighbouring waypoints. And the desired angles are calculated via the position error feedback. For the dynamic design, the control inputs of the underactuated AUV are calculated via the DRL network. Besides, in order to get smooth control actions and improve the training efficiency, additional rewards and an efficient network structure are designed.

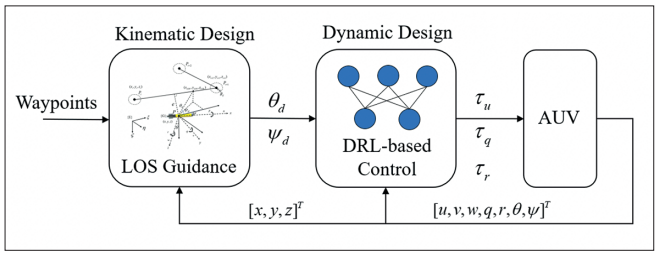


Fig. 2. Visualization of the DRL-based path-following control structure

DEEP RL-BASED PATH-FOLLOWING CONTROL DESIGN

As shown in Fig. 2, the DRL-based path-following control structure includes kinematic guidance design and dynamic control design. The given waypoints are transformed into the desired heading angle and pitch angle by using three-dimensional LOS guidance. And the DRL-based control design is used to calculate the control inputs, where states, actions, rewards and the TD3 network are designed.

KINEMATIC GUIDANCE DESIGN

Based on the three-dimensional LOS guidance [22], the desired kinematic control laws of the vehicle can be obtained. The three-dimensional LOS guidance is described in Fig. 1, where $P(x, y, z)$ represents the current positions of the AUV and P_iP_{i+1} represents the desired pathconnected by neighbouring waypoints. Considering the actual positions and desired path, thefollowing errors can be calculated, which are defined as e. Δ is the lookahead distance. ψ_d and θ_d are the desired heading angle and desired pitch angle, respectively.

Kinematic guidance can be achieved as follows. Firstly, the vehicle closes in on the desired path, which means that following errors converge to zero. Secondly, the vehicle moves along the desired path segment P_iP_{i+1}. The desired heading angle can be written as

$$\psi_d = \arctan 2(y_{LOS} - y, x_{LOS} - x) \tag{4}$$

Besides, the desired pitch angle can be written as

$$\theta_d = \arctan\left(\frac{z_{LOS} - z}{\sqrt{(x_{LOS} - x)^2 + (y_{LOS} - y)^2}} \right) \tag{5}$$

where $(y_{LOS}, x_{LOS}, z_{LOS})$ los los los x y z represents the LOS point shown in Fig. 1, which indicates the desired motion direction of the vehicle along P_iP_{i+1}.

The desired path is based on the neighbouring waypoints. With the switching of waypoints, the desired path is continually changed. Therefore, the switching scheme can be designed as

$$d = \sqrt{(x_{i+1} - x)^2 + (y_{i+1} - y)^2 + (z_{i+1} - z)^2} \tag{6}$$

with

$$i = \begin{cases} i, & d \geq R_T \\ i+1, & d < R_T \end{cases} \tag{7}$$

where $R_T > 0$ represents the switching radius, usually set to 1.5~5 times the length of the vehicle. d represents the distance between the actual position of the vehicle and the current waypoint. And i represents the ith waypoint.

DYNAMIC CONTROL DESIGN

By using the RL-based control [23], the desired dynamic control laws of the vehicle, regardless of the dynamic model, can be obtained, where the interaction between the vehicle and the environment is performed. This control process can be described as a Markov decision process, where the AUV performs actions under the current states and behaviour policy. And the vehicle obtains rewards with the states updated. By repeating this process, the optimal behaviour policy can be obtained and makes the AUV follow the desired path.

(1) States

The states consist of two components: motion states and error states, which can be written as

$$i = \begin{cases} S = [S_{motion}, S_{error}]^T \\ S_{motion} = [u, v, w, q, r, \theta, \psi]^T \\ S_{error} = [\theta_e, \psi_e, u_e]^T \end{cases} \tag{8}$$

where S_{motion} represents motion states and S_{error} represents error states. θ_e, ψ_e and u_e represent the pitch angle error, heading angle error, and surge velocity error. To be specific, there are

$$\begin{cases} \theta_e = \theta_d - \theta \\ \psi_e = \psi_d - \psi \\ u_e = u_d - u \end{cases} \tag{9}$$

where ψ_d and θ_d can be calculated by using the above three-dimensional LOS guidance.

However, there is a problem with ψ approaching π, $-\pi$, and ψ_d approaching $-\pi$, π; ψ_e will approach $\pm 2\pi$, and it will cause the AUV to approach the desired angle in the opposite direction. As shown in Fig. 3(a), the ψ_e will cause the AUV to rotate clockwise to approach the desiredheading angle, but a more efficient way is to rotate counterclockwise, as shown in Fig. 3(b).

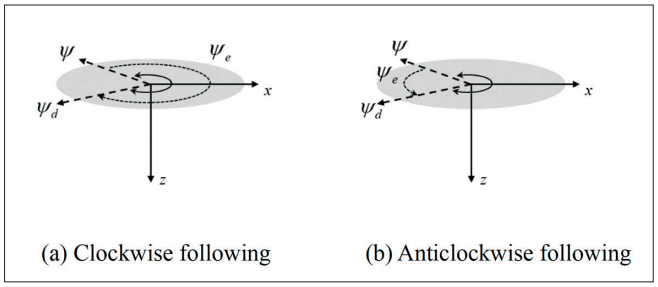


Fig. 3. Heading error calculation

Therefore, ψ_e is limited such that the vehicle can approach the desired signal in the direction of the small semicircle. The error can be written as

$$\psi_e = \begin{cases} \psi_e, \ |\psi_e| < \pi \\ \psi_e + 2\pi, \ \psi_e < -\pi \\ \psi_e - 2\pi, \ \psi_e > \pi \end{cases} \tag{10}$$

The state values have different units and scales. Excessive state values may lead to error gradients and influence the following performance. To avoid this problem, a normalization method is proposed to transform their measurements, which is beneficial for feature extraction and speeds up the training of the DRL. The method is as follows:

$$S_i = \frac{S_i}{S_{i,max}}, \quad i = 1,..., 10 \tag{11}$$

where S_i represents the ith state in (8) and $S_{i,max}$ represents the maximum velocity or angle. To be specific, $S_{i,max}$ represents the maximum velocity when $i \in \{1,..., 5\}$, and $S_{i,max}$ represents the maximum angle when $i \in \{8, 9, 10\}$.

(2) Actions

The underactuated AUV relies on the main propeller and rudders to control its attitude and position, so the actions include the surge force, pitch torque and yaw torque of the vehicle, which can be written as

$$a = [\tau_u, \tau_q, \tau_r]^T \tag{12}$$

Besides, the actions are limited to the vehicle's capability, such that control constraints should be considered, i.e., $a \in (a_{min}, a_{max})$, where a_{min} represents the minimum action and a_{max} represents the maximum action. Since backward motion is forbidden for the path-following of the AUV, the surge force is always positive.

In order to avoid the controller chattering, a first-order filter is first introduced, which can be written as

$$\begin{cases} \tau_{i,t} = (1 - a)\tau_{i,t-1} + au_{i,t} \\ a = \dfrac{\Delta t}{T_f + \Delta t} \end{cases} \tag{13}$$

where $i = q, r$. $T_f > 0$. Δt represents the step length and $u_{i,t}$ represents the action.

(3) Rewards

By setting proper and effective rewards, the vehicle is able to learn decision-making behaviour just like a human being. In the three-dimensional path-following control, following and maintaining the desired heading and pitch angles are essential. Besides, controller chattering should be avoided for as long as possible in the practical engineering. The rewards of an AUV are set as follows.

The first is the path-following reward. This reward prompts the vehicle to follow the desired path and is defined as

$$r_{pf} = -\Big(k_{pf}(|\theta_e|+|\psi_e|+u_e)+ k_1 \arctan(\theta_e^{\,2}) + \\ + k_2 \arctan(\psi_e^{\,2})+ k_3 \arctan(u_e^{\,2})\Big) \tag{14}$$

where k_{pf}, k_1, k_2 and k_3 are positive constants. u_e, θ_e and ψ_e make the reward more sensitive to the larger errors. And the arctangent function makes the reward gradient significant when the errors are close to zero, helping the AUV achieve path-following accurately.

The second is the smoothing reward. This punishment forces the vehicle to decrease the frequency and size of actions for as long as possible, which is defined as

$$r_s = - c_1\Big(|\tau_q|+|\dot{\tau}_q|\Big)- c_2\Big(|\tau_r|+|\dot{\tau}_r|\Big) \tag{15}$$

where c_1 and c_2 are positive constants. τ_q and τ_r are used to avoid large control inputs. $\dot{\tau}_q$ and $\dot{\tau}_r$ are used to smooth the control inputs, and avoid control chattering.

Therefore, the overall rewards for the path-following control of an AUV are written as

$$r = r_{pf} + r \tag{16}$$

In addition to states, actions and rewards, the ending condition is also defined. During the training process, the maximum training time of each episode is T_{end}. The training will be ended when T_{end} is arrived at and/or the desired path is followed. To be specific, the ending condition is written as

$$done \begin{cases} 1, & if\,(T = T_{end}) \text{ or } (d < R_T \,\&\, i = n \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

TD3 NETWORK DESIGN

Considering the Q value overestimation of DDPG, TD3 is proposed in the RL field, which includes clipped double Q-learning, delayed policy update and target policy smoothing [24]. The network of TD3 is shown in Fig. 4. A small batch (s, s', a, r) is sampled from the replay buffer and then the observation s' is input into an actor target, getting the next action a' The state-action (s', a') is input into two actor targets, where two Q values can be calculated. The final Q value and the mean square error are determined respectively, updating the critic model by backpropagation. And the actor model is

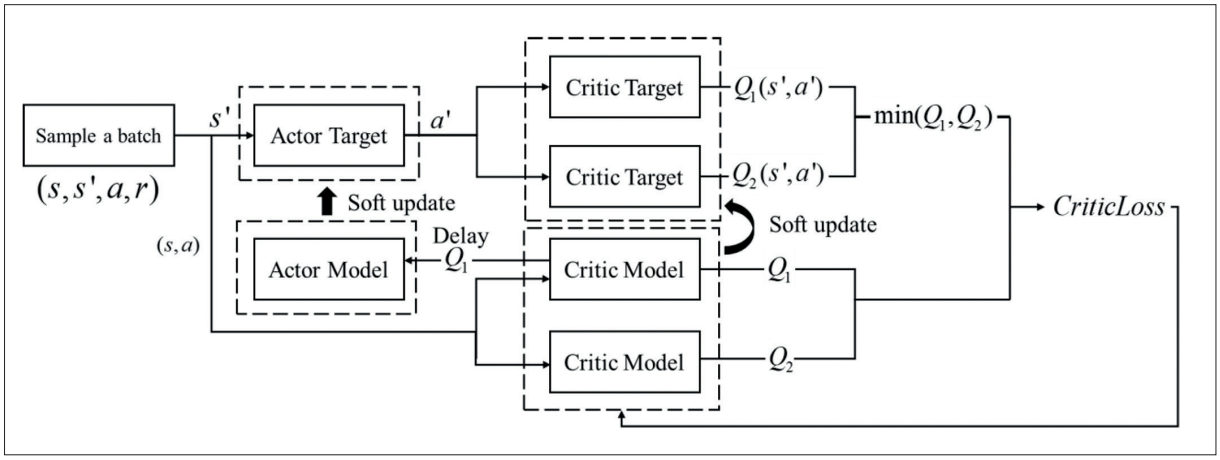


Fig. 4. Network structure of TD3

updated using the Q value calculated by the first critic model. The soft update is performed on all target networks.

Firstly, there exists a Q value overestimation by using the max operation under DQN and DDPG. The double Q-learning is introduced in TD3 where the critic network is updated. Besides, two networks of Q value are built and then the minimum value is used. There is

$$y(r, s') = r + \gamma \min_{i=1,2} Q_i(s', a') \tag{18}$$

where $y(r, s')$ represents the final Q value; $Q_1(\cdot)$ and $Q_2(\cdot)$ are the Q value generated by the two critic networks; r is the current reward; γ is the decay factor. With the final Q value determined, the mean square error is calculated by combining with the Q value and target value, and the loss function of the critic network can be obtained.

Secondly, the stability of the target network is the premise to realize the stable convergence of the policy network. The action with the largest expected reward is selected by using the policy network. In order to suppress the policy update in the wrong state, the error of the value estimation should be minimized. Therefore, by reducing the update frequency of the policy network, higherquality policy updates can be obtained. As shown in Fig. 4, the delay means that the policy network will be updated after the critic network is updated twice.

Thirdly, one problem of DDPG is that there may be overfitting to peaks in the value space, leading to local optima overfitting. Therefore, within the generation of the target policy, adding noise as regularization can smooth the calculation of the Q value and avoid overfitting. There is

$$y(r, s') = r + \gamma Q_{\theta'}\left(s', \pi_{\varphi'}(s') + \epsilon\right),$$
$$\epsilon \sim \text{clip}\left(N(0, \sigma), -c, c\right) \tag{19}$$

where ϵ represents the clipped noise, which follows a normal distribution. Adding disturbances to input actions of the target network makes the current net update within a certain range around the target network.

In the TD3 network design, the structure of the actor network and critic network is shown in Fig. 5, which has fewer nodes than the previous work [23]. The inputs of the actor network are states and the outputs are actions. The inputs of the critic network are state-action and the outputs are the Q value. To be specific, three fully-connected (FC) layers with 200, 100, and 3 nodes exist in the actor network, where the first two FC layers are the rectified linear unit (Relu) activation layer and the last one is the hyperbolic tangent (Tanh) activation layer. Four fully-connected (FC) layers with 100, 100, 200 and 1 nodes exist in the critic network, where the states and the actions can be concatenated together. All networks are updated by utilizing the Adam optimizer.

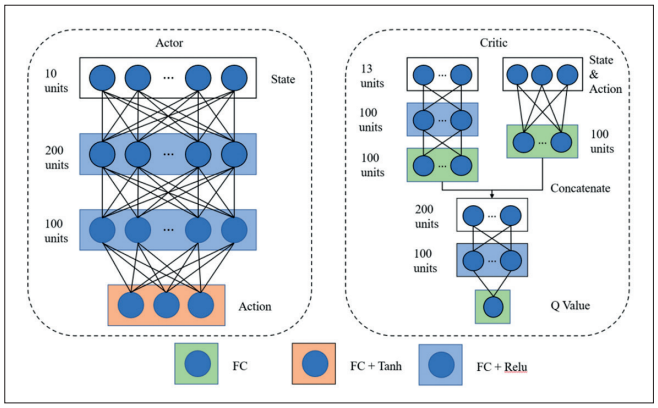


Fig. 5. Network structure of actor and critic

SIMULATION RESULTS

TRAINING RESULTS

The training consists of two parts. In Part I the desired surge and angle are introduced, and the desired signals can be tracked under the random initial attitude. In Part II two waypoints are randomly generated such that the vehicle can learn to track the desired path and switch path points. Note that Part II is based on Part I, and uses the desired angle generated by Part I. Part I consists of 300 episodes and Part II consists of 700 episodes, respectively. The training parameters and reset functions are as follows.

Tab. 1. Training parameters

Parameter	Value	Parameter	Value
Max episodes	1000	k_1	10
Max steps	3500	k_2	10
Actor learning rate	0.001	k_3	30
Critic learning rate	0.001	k_{pf}	3
Discount factor	0.99	c_1	0.1
Bath size	128	c_2	0.1
Reply buffer size	100000	T_{end}	350
Delay steps	2	R_T	5
Policy noise	0.2	Δ	4
Noise bound	[−0.5, 0.5]	T_F	0.2
Δt	0.1		

Tab. 2. Reset functions

Part I: Reset function
For every episode Initial posture = [Rand(-1,1)*pi, Rand(-1,1)*pi/12] Target posture = [Rand(-1,1)*pi, Rand(-1,1)*pi/12] Initial velocity = Rand(0,1) Target velocity = Rand(0,1)*0.5+1 End for
Part II: Reset function
For every episode Initial position = [0, Rand(0,1)*250, Rand(0,1)*50] Waypoint 1 = [250, Rand(0,1)*250, Rand(0,1)*50] Waypoint 2 = [500, Rand(0,1)*250, Rand(0,1)*50] Initial posture = [Rand(-1,1)*pi, Rand(-1,1)*pi/12] Initial velocity = Rand(0,1) Target velocity = Rand(0,1)*0.5+1 End for

Part 1: At the beginning of each episode, the initial heading angle of the AUV is randomly set within [-180°, 180°], the initial pitch angle is set within [-15°, 15°], and the desired heading angle and desired pitch angle are randomly set within [-180°, 180°] and [-15°, 15°], respectively. The initial surge velocity is set within [0,1], and the expected value is set within [1, 1.5].

Part 2: The initial positions of the AUV are set as $x = 0$, $y \in [0,250]$ and $z \in [0,50]$. The two target positions are respectively at $x = 250$, $y \in [0,250]$, $z \in [0,50]$ and $x = 50$, $y \in [0,250]$, $z \in [0,50]$. The initial heading angle and pitch angle are [-180°, 180°] and [-15°, 15°], and the initial surge velocity and desired value are [0, 1] and [1, 1.5]. During this training, the vehicle learns to follow the path connected by desired waypoints with desired velocities.

Episode rewards and average rewards are shown in Fig. 6. It can be seen that the rewards converge to a stable value. The AUV learns to control velocities in Part I and follow the desired path in Part II.

TESTING RESULTS

The testing is performed based on the trained model. The initial states of the AUV are shown in Table 3 and the waypoint positions are shown in Table 4. In order to evaluate the superiority of the proposed TD3-based control approach, comparisons with the DDPG proposed in [16] are shown. The testing results are shown in Figs. 7–10.

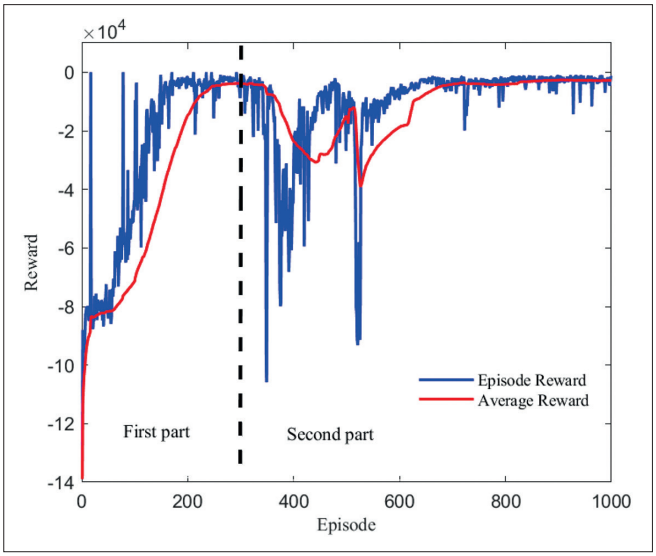


Fig. 6. Episode rewards and average rewards under two parts

Tab. 3. Initial states

State	Value	State	Value
Initial surge velocity	0.2	Initial position	[200,0,80]
Initial heading angle	0	Initial pitch angle	0
Desired surge velocity	1		

Tab. 4. Waypoint positions

Position	NO. 1	NO. 2	NO.3	NO. 4	NO. 5
ξ	100	135	205	240	160
η	70	210	200	120	80
ζ	45	70	85	90	100
Position	NO. 6	NO. 7	NO.8	NO. 9	NO. 10
ξ	60	60	180	320	320
η	160	300	320	240	40
ζ	120	120	120	70	40

Fig. 7 shows the three-dimensional path-following performance of an underactuated AUV by using the DDPG control and the TD3 control. It can be seen that the AUV can follow the desired path connected by the multiple waypoints. Besides, the TD3 control approach has higher following accuracy than the DDPG control approach. Fig. 8 shows the path-following errors, which can converge to the small neighbour of zero, and compared with the DDPG, the TD3 has smaller following errors. Note that the switching waypoints would inevitably influence the path-following performance. Fig. 9 shows the control inputs of the AUV, including the surge force, pitch torque, and yaw torque. By using the proposed control approach, the control inputs are stable and avoid chattering.

Fig. 10 shows the kinematic errors, including the surge velocity errors, yaw angle errors and pitch angle errors. It can be clearly observed that the control variables have bigger chattering by using the DDPG approach, and that the proposed TD3 can ensure higher following accuracy.

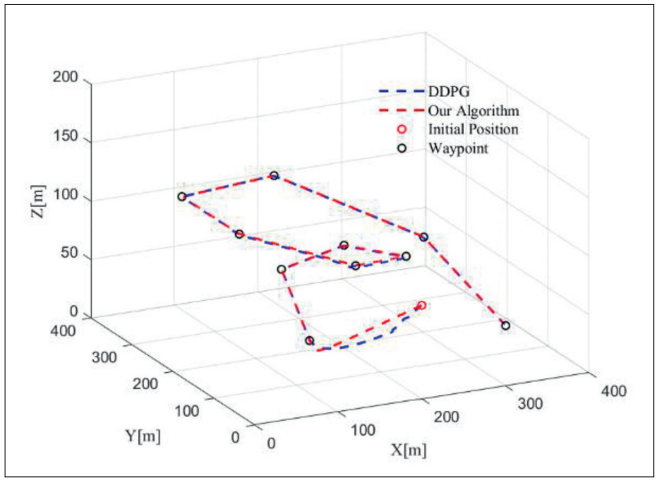


Fig. 7. Path-following performance of AUV

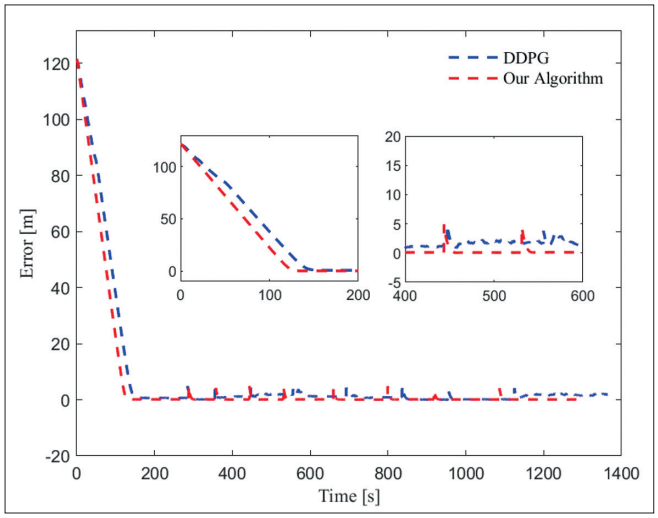


Fig. 8. Path-following errors of AUV

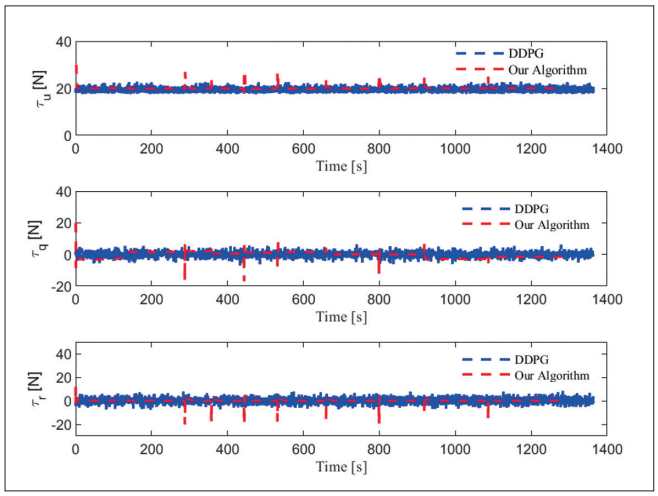
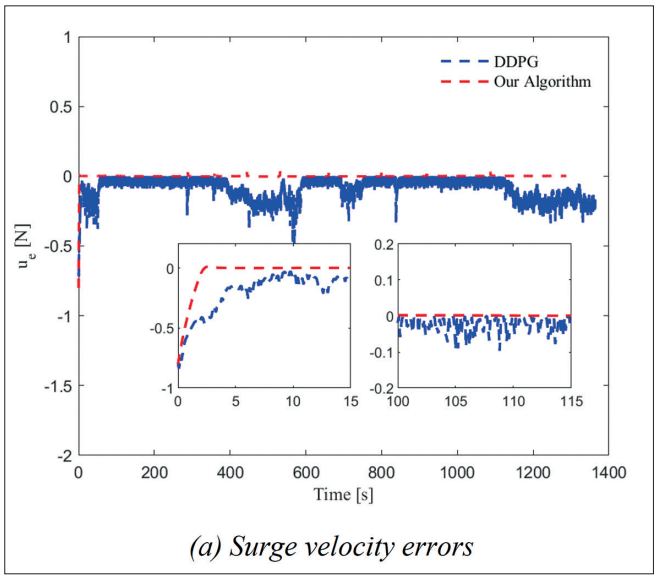
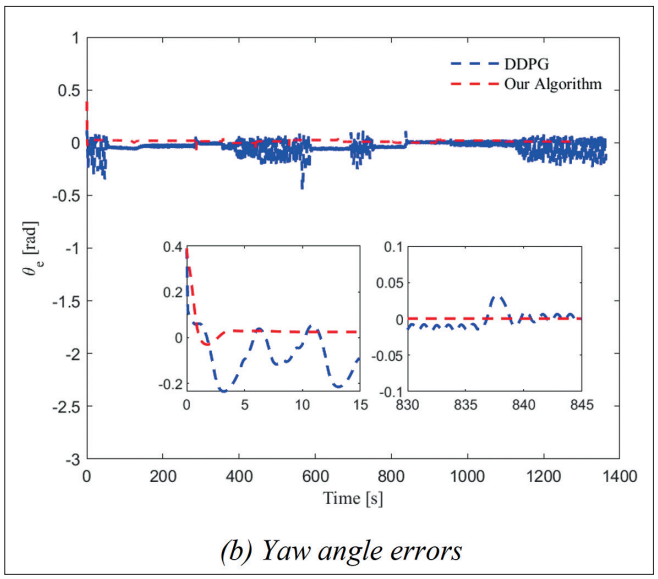


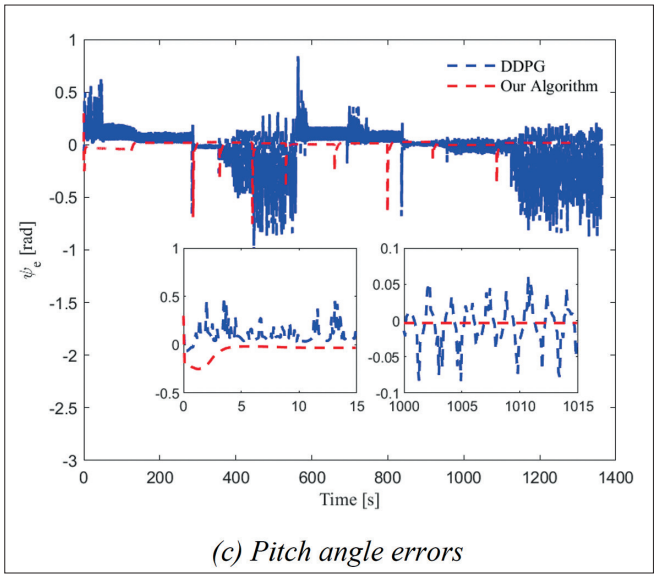
Fig. 9. Control inputs of AUV



(a) Surge velocity errors



(b) Yaw angle errors



(c) Pitch angle errors

Fig. 10. Kinematic errors of AUV

CONCLUSIONS

In this article, the DRL-based three-dimensional path-following control of an underactuated AUV is researched by using LOS guidance and TD3, where controller chattering is also considered. A reward function related to action punishment is designed, guaranteeing that the control inputs of the vehicle are smooth. Besides, an angle error calculation is proposed to solve the failure of path- following and guide the vehicle towards the targets. In order to accelerate the convergence of the algorithm, a two-part training method is adopted, and the observed values of state space are normalized before training. The simulation results demonstrate the effectiveness and superiority of the proposed three-dimensional path-following control approach. In future work, the external disturbances and obstacle avoidance of the underactuated AUV will be exclusively researched.

REFERENCES

1. I. Stenius et al., "A System for Autonomous Seaweed Farm Inspection with an Underwater Robot," Sensors, vol. 22, no. 13, Jul 2022, doi: 10.3390/s22135064.

2. L. Rowinski and M. Kaczmarczyk, "Evaluation of Effectiveness of Waterjet Propulsor for a Small Underwater Vehicle," Polish Marit. Res., vol. 28, no. 4, 2022, doi: 10.2478/pomr-2021-0047.

3. H. Choukri and L. Z. Qidan, "Path Following Control of Fully Actuated Autonomous Underwater Vehicle Based on LADRC," Polish Marit. Res., vol. 25, no. 4, 2018, doi: 10.2478/pomr-2018-0130.

4. L. Li, Z. Pei, J. Jin, and Y. Dai, "Control of Unmanned Surface Vehicle along the Desired Trajectory Using Improved Line of Sight and Estimated Sideslip Angle," Polish Marit. Res., vol. 28, no. 2, 2021, doi: 10.2478/pomr-2021-0017.

5. E. Vidal, N. Palomeras, M. Carreras, "Online 3D Underwater Exploration and Coverage," in IEEE/OES Autonomous Underwater Vehicle Workshop (AUV), Rectory Univ Porto, Porto, Portugal, 2018.

6. J. H. Wan et al., "Motion Control of Autonomous Underwater Vehicle Based on Fractional Calculus Active Disturbance Rejection," Journal of Marine Science and Engineering, vol. 9, no. 11, Nov 2021, doi: 10.3390/jmse9111306.

7. J. J. Zhou, X. Y. Zhao, T. Chen, Z. P. Yan, and Z. W. Yang, "Trajectory Tracking Control of an Underactuated AUV Based on Backstepping Sliding Mode With State Prediction," IEEE Access, vol. 7, 2019, doi: 10.1109/access.2019.2958360.

8. M. P. R. Prasad and A. Swarup, "Model predictive control of an AUV using de-coupled approach," International Journal of Maritime Engineering, vol. 160, Jan–Mar 2018, doi: 10.3940/rina.ijme.2018.a1.459.

9. 9. J. H. Wan et al., "Multi-strategy fusion based on sea state codes for AUV motion control," Ocean Engineering, vol. 248, Mar 2022, doi: 10.1016/j.oceaneng.2022.110600.

10. Y. K. Xia, K. Xu, Z. M. Huang, W. J. Wang, G. H. Xu, and Y. Li, "Adaptive energy-efficient tracking control of a X rudder AUV with actuator dynamics and rolling restriction," Applied Ocean Research, vol. 118, Jan 2022, doi: 10.1016/j.apor.2021.102994.

11. K. Fang, H. L. Fang, J. W. Zhang, J. Q. Yao, and J. W. Li, "Neural adaptive output feedback tracking control of underactuated AUVs," Ocean Engineering, vol. 234, Aug 2021, doi: 10.1016/j.oceaneng.2021.109211.

12. J. L. Zhang, X. B. Xiang, Q. Zhang, and W. J. Li, "Neural network-based adaptive trajectory tracking control of underactuated AUVs with unknown asymmetrical actuator saturation and unknown dynamics," Ocean Engineering, vol. 218, Dec 2020, doi: 10.1016/j.oceaneng.2020.108193.

13. H. N. Esfahani and R. Szlapczynski, "Model Predictive Super-Twisting Sliding Mode Control for an Autonomous Surface Vehicle," Polish Marit. Res., vol. 26, no. 3, 2019, doi: 10.2478/pomr-2019-0057.

14. C. X. Cheng, Q. X. Sha, B. He, and G. L. Li, "Path planning and obstacle avoidance for AUV: A review," Ocean Engineering, vol. 235, Sep 2021, doi: 10.1016/j.oceaneng.2021.109355.

15. S. Brandi, M. Fiorentini, and A. Capozzoli, "Comparison of online and offline deep reinforcement learning with model predictive control for thermal energy management," Automation in Construction, vol. 135, Mar 2022, doi: 10.1016/j.autcon.2022.104128.

16. Y. Fang, Z. W. Huang, J. Y. Pu, and J. S. Zhang, "AUV position tracking and trajectory control based on fast-deployed deep reinforcement learning method," Ocean Engineering, vol. 245, Feb 2022, doi: 10.1016/j.oceaneng.2021.110452.

17. P. Zielinski and U. Markowska-Kaczmar, "3D robotic navigation using a vision-based deep reinforcement learning model," Applied Soft Computing, vol. 110, Oct 2021, doi: 10.1016/j.asoc.2021.107602.

18. D. L. Song, W. H. Gan, P. Yao, W. C. Zang, Z. X. Zhang, and X. Q. Qu, "Guidance and control of autonomous surface underwater vehicles for target tracking in ocean environment by deep reinforcement learning," Ocean Engineering, vol. 250, Apr 2022, doi: 10.1016/j.oceaneng.2022.110947.

19. E. Meyer, H. Robinson, A. Rasheed, and O. San, "Taming an Autonomous Surface Vehicle for Path Following and Collision Avoidance Using Deep Reinforcement Learning," IEEE Access, vol. 8, 2020, doi: 10.1109/access.2020.2976586.

20. A. B. Martinsen and A. M. Lekkas, "Straight-Path Following for Underactuated Marine Vessels using Deep Reinforcement Learning," in 11th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles (CAMS), Opatija, Croatia, 2018, vol. 51.

21. T. I. Fossen, Handbook of Marine Craft Hydrodynamics and Motion Control, John Wiley, Chichester, UK, doi: 10.1002/9781119994138.

22. M. Breivik, T. I. Fossen, "Principles of guidance-based path following in 2D and 3D," in 44th IEEE Conference on Decision Control/European Control Conference (CCD-ECC), Seville, Spain, 2005.

23. T. Liu, Y. L. Hu, and H. Xu, "Deep Reinforcement Learning for Vectored Thruster Autonomous Underwater Vehicle Control," Complexity, vol. 2021, Apr 2021, doi: 10.1155/2021/6649625.

24. S. Fujimoto, H. V. Hoof, D. Meger, Addressing Function Approximation Error in Actor-Critic Methods, ICML, 2018.

CONTACT WITH THE AUTHORS

Zhenyu Liang

School of Mechanical and Electronic Engineering
Dalian Minzu University
Dalian, 116600

School of Naval Architecture and Ocean Engineering
Dalian Maritime University
Dalian, 116026
China

Xingru Qu
e-mail: quxingru@126.com

Zhao Zhang

Cong Chen

School of Mechanical and Electronic Engineering
Dalian Minzu University
Dalian, 116600
China