

Cooperative Motion Planning and Control of a Group of Autonomous Underwater Vehicles Using Twin-delayed Deep Deterministic Policy Gradient

Behnaz Hadi ^{1*}, Alireza Khosravi ¹, Pouria Sarhadi ²

¹ Department of Electrical and Computer Engineering, Babol Noshirvani University of Technology, Babol, Iran

² School of Physics, Engineering and Computer Science, University of Hertfordshire, Hatfield, UK

Abstract: The cooperative execution of complex tasks can lead to desirable outcomes and increase the likelihood of mission success. Nevertheless, coordinating the movements of multiple autonomous underwater vehicles (AUVs) in a collaborative manner is challenging due to nonlinear dynamics and environmental disturbances. The paper presents a decentralized deep reinforcement learning algorithm for AUVs that enables cooperative motion planning and obstacle avoidance. The goal is to formulate control policies for AUVs, empowering each vehicle to create its optimal collision-free path through adjustments in speed and heading. To ensure safe navigation of multiple AUVs, COLLision AVOIDance (COLAV) plays a crucial role. Therefore, the implementation of a multi-layer region control strategy enhances the AUVs' responsiveness to nearby obstacles, leading to improved COLAV. Furthermore, a reward function is formulated to consider four criteria: path planning, obstacle- and self-COLAV, as well as feasible control signals, with the aim of strengthening the proposed strategy. Notably, the devised scheme demonstrates robustness against disturbances. A comparative study is conducted with the well-established Artificial Potential Field (APF) planning method. The simulation results indicate that the proposed system effectively and safely guides the AUVs to their goals and exhibits desirable generalizability.

Keywords: Multi-AUVs; Motion planning; Obstacle avoidance; Ocean current; Deep reinforcement learning

1. Introduction

Multiple Autonomous Underwater Vehicles (AUVs) have become increasingly important for various applications such as deep-sea exploration, pipeline monitoring, and search and rescue operations [1, 2]. With the increasing complexity of AUV missions, there is a growing interest in employing AUV fleets. The utilization of multiple AUVs can potentially improve system performance, reduces costs, shortens mission duration, and increases the likelihood of mission success. Consequently, the navigation and motion control of such autonomous systems operating in the ocean environment are critical. Path planning is the process of determining a feasible, collision-free path that connects the starting and finishing points while taking into account the

*b.haadi@gmail.com

following factors: path length, arrival time, energy consumption, path smoothness, environmental uncertainties, and others [3, 4]. Although extensive research has been conducted on addressing the path planning problem in both single and multiple AUV applications [5], a smaller portion of solutions are based on machine learning (ML) techniques [6, 7]. Common methods that have been utilized in multiple AUVs' path planning include optimal control theory approaches [8], evolutionary algorithms [9], artificial potential fields (APF) [10], and the artificial neural network-based method [11, 12]. However, given the complexity and uncertainty of the marine environment, it can be challenging to generate a large number of possible scenarios, exposing these methods to reduced effectiveness in addressing external disturbances and fleet control issues in addressing external disturbances and fleet control issues [3]. To overcome these limitations, this study presents a novel approach to cooperative path planning of multiple AUVs. By leveraging the power of ML technologies, this approach paves the way for cooperative planning and control without relying on tedious if-then rule loops. This paper represents a step forward in the theory of this field, although the technology is still in its early stages [13].

Deep reinforcement learning (DRL) is a learning method that has been shown to be a beneficial and effective tool for tackling complicated problems and adapting to unpredictable situations in autonomous systems [14-16]. Recent studies in the field of marine system mission planning have shown an increasing tendency to employ DRL techniques [17]. In DRL-based path planning research, the single AUV motion planning problem [18-26] is comparatively more explored than the planning for multiple AUVs [7]. One example of the application of DRL in marine system mission planning is described by [18], where a hierarchical DQN with prioritized experience replay was presented to plan three-dimensional AUV paths. In order to reach this goal, the path planning problem was split into three layers, which made the state space smaller, and an artificial potential field was used to solve the sparse reward problem. Chu et al. proposed the twin-delayed deep deterministic policy gradient (TD3) algorithm for enhanced path planning in unmanned underwater vehicles (UUVs), emphasizing the use of multi-beam sonar data. The study includes a comparison analysis with other DRL algorithms [19]. Wu et al. employed DRL for a maritime search and rescue vessel coverage path planning framework for multiple persons in water to maximize the success rate in an effective time. Using DRL approaches, the framework attempted to dynamically alter the path planning strategy in response to real-time environmental conditions and operational restrictions [20]. In order to enhance localization accuracy and efficiency in underwater networks containing a large number of beacons, [21] implemented a DRL-based localization approach for an AUV. Through the strategic choice of beacons and the adjustment of their transmission power, the scheme enhances both localization precision and energy efficiency by taking into account variables like the number of selected beacons, AUV depth, and received signal strength. Bhopale et al. described how to prevent colliding with obstacles in AUVs using modified Q-learning. To reduce the risk of a crash, a danger area was constructed near the obstacles in this technique. Once the AUV entered this danger zone, exploration was stopped and exploitation was carried out until the danger zone was exited [22]. Hadi et al. have developed an adaptive motion planning and obstacle avoidance strategy for an AUV based on DRL. The presented method is robust to ocean currents [23]. In [24], Maximum a Posteriori Policy Optimization (MPO), a DRL technique, was utilized to address the motion control issue of an underactuated AUV. Extended state observers were used to estimate five-degrees-of-freedom

unknown disturbances. Xu et al. implemented DRL into USV path planning and dynamic COLAV using COLREGs. In this study, a deep deterministic policy gradient (DDPG) was employed to generate thrust and rudder inputs for vessel steering [27].

As previously mentioned, few studies have explored the use of DRL for multi-AUV path planning and obstacle avoidance. An ant colony optimization algorithm and a DQN were used to solve the problems of path planning and obstacle avoidance for multiple unmanned undersea vehicles (UUVs) [28]. The DRL approach was used to perform generalizable, distributed formation path planning for unmanned surface vehicles (USVs) [29]. Pan et. al. proposed flocking control of a swarm of underactuated unmanned surface vehicles (USVs) using DRL and the model predictive path integral (MPPI) method. A deep neural network was trained to learn each USV model. Formation control and collision avoidance tasks were achieved based on DRL and MPPI control designs. Zhao et al. used the PPO algorithm combined with navigation rules to design an obstacle avoidance model for multi-unmanned vessels [30].

In the current study, the mission involves navigating toward designated destinations for each AUV, ensuring avoidance of collisions with randomly distributed obstacles and other AUVs. Upon reaching the designated area, the AUV comes to a complete stop. If a new mission is assigned, it advances toward the subsequent destination. In another work by the authors, they proposed end-to-end formation motion planning and control for AUVs. The goal was to create optimal adaptive distributed controllers based on actor-critic frameworks. Two distinct obstacle-avoidance strategies were offered [31]. In the present study, DRL is extended to enable cooperative autonomous tasks in AUVs, where each AUV operates independently to achieve its objective while avoiding obstacles and other AUVs. It should be noted, another avenue in motion planning research is the target pursuit or encirclement problem. In target pursuit, the priority is to circumnavigate and enclose moving targets based on sensing information like range or information transmission through wireless networks along the path. For example, in [32] a distributed estimating and control approach is introduced to tackle the issue of distance-based target localization and pursuit for multiple AUVs. The localization and pursuit of an unknown underwater moving target are achieved by utilizing range measurements between the trackers and the target. In [33], a framework for multi-objective motion planning is introduced to achieve the localization and pursuit of multiple targets. A model predictive control method is employed to address the issue and ensure optimal control of pursuit vehicles while chasing moving target vehicles. A multi-agent cooperative pursuit optimization issue for maritime security protection using distributional soft actor-critic was solved in [34].

The majority of current research on AUV motion planning using DRL techniques is focused on solving single AUV path planning problems. Therefore, there is interest in a DRL-based decentralized adaptive controller with obstacle avoidance and self-COLAV for under-actuated AUVs. Using actor-critic architecture, optimal adaptive controllers identify the performance of the current control policy and update the controller to eliminate motion planning errors. The model for the Markov decision process (MDP) for end-to-end motion planning of the multiple AUVs has been presented. The system achieves end-to-end processing of information by directly mapping

the state information of the AUVs and the surroundings to control their heading and speed. Obstacle avoidance throughout the exploration process is considered a challenging task for AUVs. Since the proposed approach is a model-free method, reliance on the mathematical model has been eliminated, and an intelligent controller based on plant's behavior in an unknown environment has been developed. Obstacle avoidance is implemented using multi-layer control approaches. All AUVs are equipped with obstacle avoidance modules. When the obstacles are within the detection range of each AUV, they are entered into its MDP model, and the change of direction and speed required to avoid the obstacles is done when they enter the danger zone.

This research is a step forward in replacing burdensome, traditional rule-based methods that use models or geometry with novel learning-based algorithms. The paper's main contributions are as follows:

- A decentralized deep reinforcement learning algorithm for the path planning of multiple AUVs is proposed. In this framework, each AUV learns its own policy based on local observations and rewards without direct communication or coordination with other AUVs, which is desirable in underwater environments with limited communication capabilities.
- AUVs can make independent decisions based on local data, resulting in more scalable and resilient operations in real-world circumstances.
- At the sensing level, we developed decentralized collision avoidance strategies that map end-to-end sensing data directly to the desired collision-free speed and steering commands.
- For more realistic scenarios in underwater operations, the efficacy of the proposed algorithm under ocean disturbances is considered.

The present paper is organized as follows: In Section 2, the fundamentals of DRL are presented. AUVs' motion model is described in Section 3. The details of the DRL algorithm for distributed cooperative control and obstacle and self-COLAV, including a definition of the state space, action space, and reward schemes, are introduced in Section 4. Simulations of various scenarios are presented in Section 5. Section 6 is devoted to conclusions.

2. Problem statement

2.1. Cooperative motion planning mission

According to Figure 1, the primary mission of AUVs is to efficiently navigate towards predetermined goals while minimizing travel distance. Additionally, AUVs are designed to effectively circumvent encounters with stationary obstacles and prevent collisions with other AUVs. The initial positions of the AUVs are randomly determined within a designated geographical region. Obstacles within the movement space of AUVs are defined in a random manner. The selection of targets within a particular area is randomly conducted.

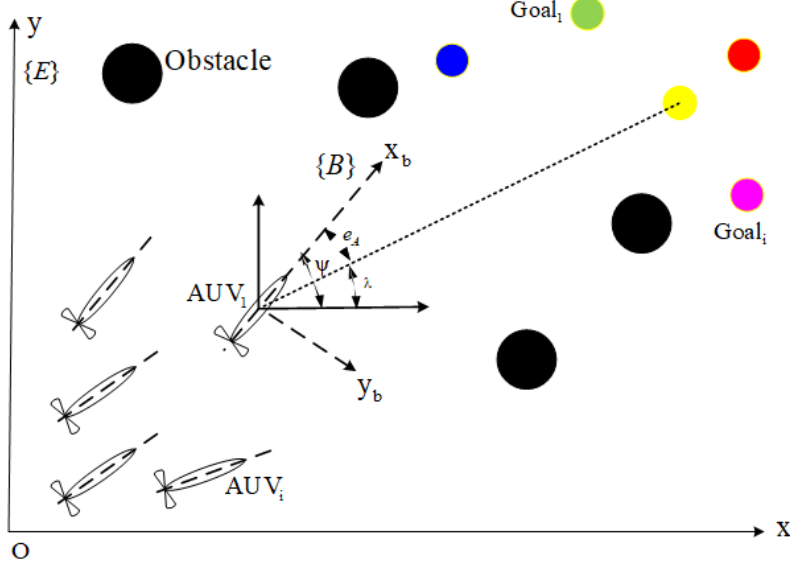


Fig. 1. Multi-AUVs cooperative motion planning

2.2. AUV motion model

To describe the motion of AUVs, two coordination systems are defined as displayed in Fig. 2: the body-fixed frame $\{B\}$, and the earth-fixed frame $\{E\}$. The motion of AUVs on a horizontal surface is suitable for applications where AUVs operate at a constant depth, such as seabed exploration. Excluding the roll, pitch and heave motions, the mathematical model of AUVs can be expressed by the following equations [35]:

$$\dot{\eta} = J(\eta)v \quad (1)$$

$$M\dot{v} + C(v)v + D(v)v + g(\eta) = \tau + \tau_d \quad (2)$$

where $\eta = [x, y, \psi]^T$ is the position vector of the AUV in the earth-fixed frame, which includes the (x, y) coordinate and the yaw angle $\psi \in [0, 2\pi)$. The velocity vector of the AUV in the body-fixed frame is $v = [u, v, r]^T$, where u is the surge velocity, v is sway velocity, and r is the yaw rate. $J(\psi) \in R^{3 \times 3}$ is the rotation matrix, M is the positive definite mass matrix, $C(v)$ is the coriolis terms and centripetal force matrix; and $D(v)$ is the damping matrix that are defined in [33]. $\tau = [\tau_u, 0, \tau_r]$ is the input control vector, which includes the surge force, the sway force (which is zero because the AUV is under-actuated), and the yaw moment. τ_d is disturbance. This model is solely utilized for data generation in training and simulations; the algorithm does not require an understanding of vehicle characteristics and is model-free data-driven.

2.3. Deep Reinforcement Learning (DRL) Algorithm

Reinforcement learning enables an agent to learn its intended behavior or policy through direct interaction with the environment (refer to Fig. 1). At every time step, the agent performs action a in the state s . Subsequently, it transitions to the new state of s' and receives the reward r from the

environment. Through extensive interaction with the environment, the agent discovers a strategy that maximizes the expected return on investment as a whole. The discounted sum of future rewards is defined as the expected return, as follows [36]:

$$R_t = r_{t+1} + \gamma r_{t+2} + \gamma^2 r_{t+3} + \dots = \sum_{k=0}^{\infty} \gamma^k r_{t+(k+1)} \quad (3)$$

where $\gamma \in [0, 1]$ is the discount factor that determines the current value of future rewards. The discounted expected state-value function is defined as follows, starting from the state s and following the π policy:

$$V^\pi(s) = E_\pi[R_t | s_t = s] = E_\pi\left[\sum_{k=0}^{\infty} \gamma^k r_{t+(k+1)} | s_t = s\right] \quad (4)$$

Similarly, the action-value function is defined as follows, which describes the value of the action a in the state of s under the π policy:

$$\begin{aligned} Q^\pi(s, a) &= E_\pi[R_t | s_t = s, a_t = a] \\ &= E_\pi\left[\sum_{k=0}^{\infty} \gamma^k r_{t+(k+1)} | s_t = s, a_t = a\right] \end{aligned} \quad (5)$$

The Bellman Optimality Equations for the state-value function and the action-value function are defined as follows:

$$V^*(s) = \max_a E[r_{t+1} + \gamma V^*(s_{t+1}) | s_t = s, a_t = a] \quad (6)$$

$$Q^*(s, a) = E[r_{t+1} + \gamma \max_{a'} Q^*(s_{t+1}, a') | s_t = s, a_t = a] \quad (7)$$

where $V^*(s) = \max_a Q^*(s, a)$ holds for all s . When Q^* is found through interactions, the equation $\pi^*(s) = \underset{a}{\operatorname{argmax}} Q^*(s, a)$ can be used to determine the best policy.

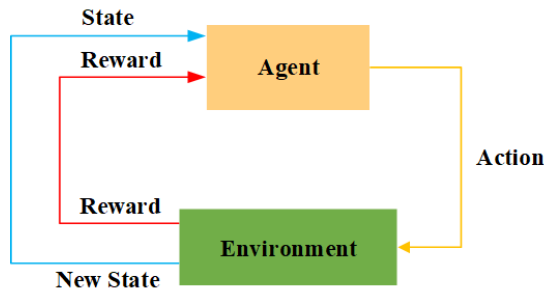


Fig. 2. Interaction of the deep agent with the environment in the Markov decision-making process

3. The structure of DRL for cooperative motion planning and obstacle avoidance of multi-AUVs

The design of AUV control is complicated by the highly nonlinear dynamics of underwater vehicles and the unknown disturbances caused by environmental conditions. Adaptation strategies that do not require dynamic models can be advantageous in this context. By employing actor-critic structures, RL algorithms converge to adaptive-optimal online solutions for systems that are entirely unknown. The (TD3) algorithm [37] is elaborated on to achieve an integrated cooperative motion planning and control algorithm, addressing a more comprehensive challenge than a standalone control problem that could be addressed using well-established classical techniques.

3.1. State and action representation

Each AUV aims to reach its destination safely by using the shortest possible path. Thus, the state space is based on the AUV's direction error from its target and distance from the identified obstacles.

$$S_{AUV_i} = \left[e_A, \frac{d_{Aj} - r_{det}}{r_{det}} \right], \quad i=1,2,3, \quad j=0,A \quad (8)$$

$$\frac{d_{Aj} - r_{det}}{r_{det}} = \begin{cases} \text{value} & \text{if obstacles or other AUVs are in AUV's sensor range detection} \\ 0 & \text{else} \end{cases}$$

where $e_A = \lambda - \psi$ is the direction error of the AUV toward the target, and λ indicates the direction of the target (Fig. 2). ψ is the heading angle of each AUV. d_{Aj} is the $N \times 1$ vector of the distance from identified obstacles or other AUVs, and N is the number of the objects. r_{det} is the maximum detection range of the AUV's sensor.

As stated previously, the AUVs intend to arrive at their destination safely and avoid obstacles and other AUVs during navigation, which requires speed and direction control. To accomplish this, the surge force τ_u and yaw moment τ_r are controlled.

$$A = [\tau_u, \tau_r] \quad (9)$$

3.2 Reward function

A detailed reward function is meticulously designed to shape the cooperative motion planning and obstacle avoidance mechanisms of the AUVs. This function strategically encourages desired behaviors while penalizing deviations from the intended course of action. It comprises three distinct sub-terms for the AUVs: efficiently guiding them towards targets, steering clear of potential hazards, and preventing collisions with other AUVs.

3.2.1 LOS tracking reward

This reward motivates the AUVs to follow the Line of Sight (LOS), thus enhancing the chances of reaching the target. This is expressed in the following equation:

$$r_1 = -|\lambda - \psi| \quad (10)$$

where λ is the LOS angle target with the AUV, and ψ is the AUV's heading angle.

3.2.2 Multi-layer obstacle and collision avoidance

This paper presents a two-layer region control concept for obstacle and self-collision avoidance, as shown in Figs. 3 and 4. Each AUV is surrounded by layers. Each layer has its own gain. The gains from the layers increase as they move from the outer to the inner layers. The rewards due to the inner layer with significant gain are only activated when an obstacle enters the inner layer region. This makes COLAV implementation more efficient. When an obstacle reaches the first layer, it is rewarded negatively. As it advances to the second layer, it receives increasingly severe negative rewards. Consequently, compared to receiving rewards with a constant gain, the AUV's sensitivity to its immediate surroundings increases. Each AUV learns a policy to avoid obstacles. According to Fig. 3, first, a circular shell is made for the AUV model. A collision is indicated by a shell overlapping an obstacle. The scaled distance of all obstacles inside sensors' detection range is entered into the corresponding AUV's MDP model, and thus, the reward function is defined as follows:

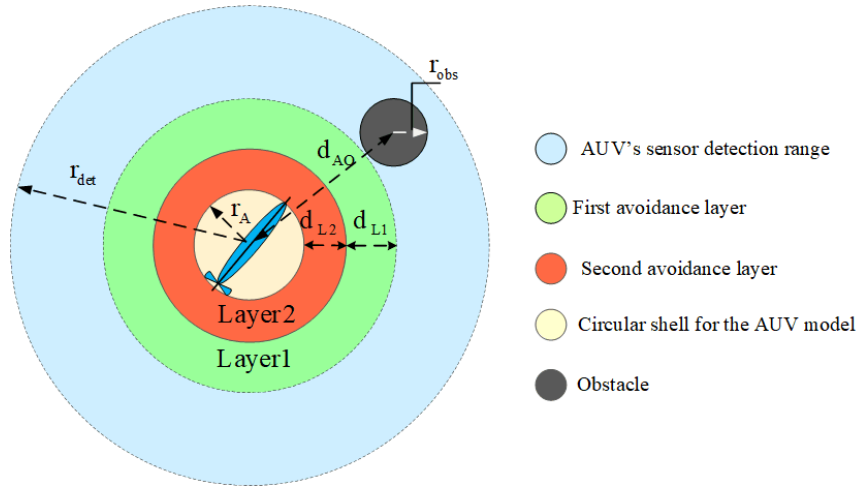


Fig. 3. Two-layer obstacle avoidance: each AUV senses the obstacles in the range of sensors detection

$$r_2 = \sum_{i=1}^{\ell} r_i, \quad r_i \begin{cases} 0 & \text{if } d_{AO} > d_{avoid} \\ -|d_{avoid} - d_{AO}| & \text{otherwise} \end{cases} \quad (11)$$

where ℓ denotes the number of obstacles sensed within the detection range of the AUV's sensors. d_{AO} is the distance between the AUV and the obstacle and $d_{avoid} = r_A + d_{L1} + d_{L2} + r_{obs}$, which r_A is the AUV's shell radius, d_{L1} and d_{L2} are the first and the second layer width distances respectively, and r_{obs} is the obstacle radius. If the distance of the AUV from the obstacles exceeds the defined danger zone with a radius of d_{avoid} , it receives a reward of zero; otherwise, it incurs penalized.

3.2.3 Self-COLAV reward

What?? This term of the reward reflects the AUVs' self-COLAV with respect to other AUVs, as formulated below:

$$r_3 = \begin{cases} 0 & \text{if } d_{AA} > d_{avoid} \\ -|d_{avoid} - d_{AA}| & \text{otherwise} \end{cases} \quad (12)$$

where, d_{AA} is the distance between two AUVs. This reward is designed to prevent AUV collisions. $d_{avoid} = 2(r_A + d_{L1} + d_{L2})$. Figure 4 demonstrates a two-layer region for self-collision evasion.

3.2.4 Total reward

The total reward function is defined as the weighted sum of all the mentioned reward functions:

$$r_L = w_1 r_1 + w_2 r_2 + w_3 r_3 + [w_4, w_5] r_E^T \quad (13)$$

where w_1, w_2, w_3, w_4 and w_5 are positive constants. Each of the aforementioned weightings represents the significance of its corresponding reward in the control policy. The selection of $r_E = [-|\tau_u|, -|\tau_r|]$ aims to reduce total control effort by incorporating the modulus of control signals to avoid the neutralization of negative and positive values.

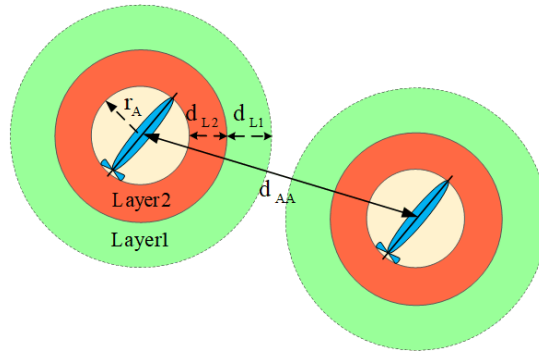


Fig. 4. Two-layer self-COLAV: indicating distance layers

The cooperative motion planning of multi-AUVs is presented in Algorithm 1 using the TD3 algorithm. The actor-critic reinforcement learning scheme AUV motion planning is shown in Fig. 5. An architecture with two hidden layers was used to estimate the value function (state-action) and policy for the DRL algorithm's actor and critic networks.

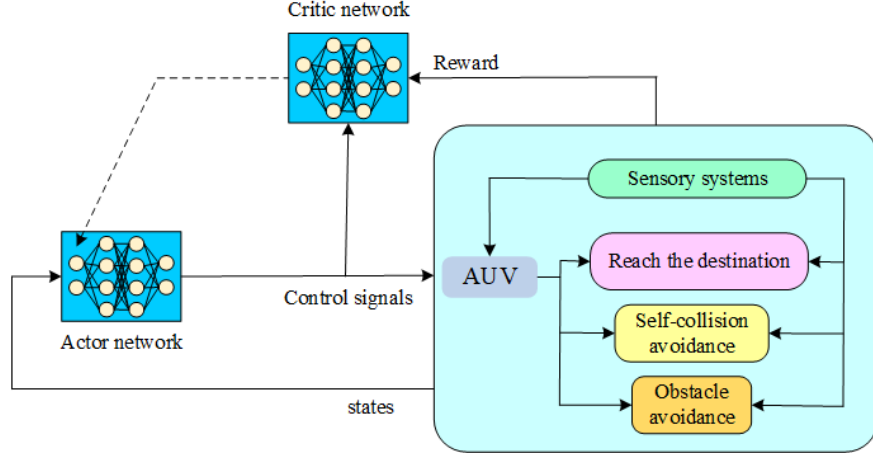


Fig. 5. Schematic diagram of AUV motion planning based on the proposed strategy

Algorithm1: Adaptive motion planning based on DRL

1. Configure initial poses, number of episodes, number of steps in each episode N , reply buffer size D , mini-batch size M , the learning rate for decision-making networks γ
2. Initialization online critic networks Q_{ϕ_1}, Q_{ϕ_2} and the online actor μ_{θ}
3. Initialization target critic networks $Q_{\phi'_1}, Q_{\phi'_2}$ and target actor network μ'_{θ}
4. For episode = 1, T do
 5. Initialize a random process (N_t) for action exploration
 6. Receive initial observation states (Eq. 8)
 7. For $t = 1, N$ do
 8. For each AUV, select the action $a_t = \mu(s|\theta) + N_t$ according to the current policy and the explored noise
 9. Executing the action a_t on AUV and observing the new state s_{t+1} (Eq.8) and obtaining the reward r_t (Eq.13)
 10. Storing the transfer (s_t, a_t, r_t, s_{t+1}) in the experience reply buffer D
 11. Sampling a random mini-batch $(M * (s_i, a_i, r_i, s_{i+1}))$ from the experience reply buffer D
 12. For $i=1, M$ do
 13. Calculation $a'_{i+1} = \mu'(s_{i+1}|\theta') + \text{clipped noise}$
 14. Calculating the value of target state-action using the equation $y_i = r_i + \gamma \min_{j=1,2} Q_{\phi'_j}(s_{i+1}, a'_{i+1}|\phi'_j)$
 15. End for
 16. Updating the critic networks by minimizing the cost function of Bellman equation errors
$$\phi_1 = r_i + \arg \min_{\phi_1} \frac{1}{M} \sum_{i=1}^M (y_i - Q_{\phi_1}(s_i, a_i|\phi_1)), \phi_2$$

$$= r_i + \arg \min_{\phi_2} \frac{1}{M} \sum_{i=1}^M (y_i - Q_{\phi_1}(s_i, a_i|\phi_2))$$
 17. If $t \bmod d$ then
 18. Updating the online actor policy (θ), using the sampled deterministic policy gradient $\nabla_{\theta} J(\theta) = \frac{1}{M} \sum_{s \in M} \nabla_{\theta} Q_{\phi_1}(s, \mu_{\theta}(s))$,
 19. soft updating the target networks: $\phi'_i = \tau \phi'_i + (1 - \tau) \phi'_i, \text{ for } i = 1, 2, \theta' = \tau \theta' + (1 - \tau) \theta'$
 20. End if
 21. End for
 22. End for

The proposed cooperative motion control and obstacle avoidance model seek an optimal policy for each agent using the TD3 algorithm. Before the start of the episodes, all the settings are done, including the initial position and orientation of multiple AUVs, the dimensions of the training area, the training rate, the exploration noise for the actor network, the smoothing noise for the target network, etc. In each episode, the variables of the velocity vector and the AUVs' position are reset. Multiple random target zones are selected, and the obstacles are distributed in the training zone randomly. In the algorithm, the critic has four networks, of which two have similar structures, i.e., online networks with ϕ_1 and ϕ_2 parameters, and target networks with ϕ'_1 and ϕ'_2 parameters. The actor has two online and target networks with similar structures, with θ and θ' parameters, respectively. In order to have proper exploration in state and action spaces while choosing the policy, the noise N_t is added (line 8). In each time step, by applying action to each AUV, new states s_{t+1} based on Eq. (9) are obtained, and the instant reward r_t based on the reward function designed for control purposes are received and stored in the experience buffer (D) along with the applied action and the system's current state. The buffer stores a predetermined number of state transitions and a mini-batch (M) is sampled from them (line 11). In the subsequent stage, an action is chosen for each state of the mini-batch. Applying the smoothing regularization strategy to the target policy and adding clipped noise to the deterministic output of the target actor network a' for the state s_{i+1} will prevent high variance when updating the critic network (line 13). The critic's target Q-values for all mini-batch samples are calculated. The online critic network's parameters ϕ_1 and ϕ_2 are updated by minimizing the mean square of the Bellman error (line 16). The training process of the online actor network is updated according to the random data sampled from the buffer, $M * (s_i, a_i, r_i, s_{i+1})$, based on a deterministic policy gradient with a lower frequency than the critic networks (line 18). By doing so, the critic network becomes more stable before being used for training the target network, and the errors are reduced. With a frequency equal to the update frequency of the actor's policy function, the parameters of the target actor network θ' and the target critic networks ϕ'_1 and ϕ'_2 are soft updated with the aim of enhancing the stability of the learning process (line 19). This procedure is repeated in every episode until all AUVs reach the destinations or the time of the episode's step is over.

4. Simulation results

The motion planning of under-actuated AUVs is evaluated in various scenarios to validate the efficiency of the proposed method. Cooperative motion planning is taken into account in the presence of obstacles. Furthermore, the algorithms' performance and robustness are assessed in the presence of ocean currents. This section discusses the acquired results. The training area measures $250 \times 250 \text{ m}^2$. The AUVs are placed in the one corner area of the training zone. The targets are circles with a radius of 3 meters at the boundary of the training zone. The dynamic model parameters used in AUV_i ($i = 1 - 5$) simulations are adapted from [38] and are as follows: $m_{11} = 200\text{kg}$, $m_{22} = 259\text{kg}$, $m_{33} = 80\text{kg}$, $d_{11} = (70 + 100|u|)\text{kg/s}$, $d_{22} = (100 + 200|v|)\text{kg/s}$, and $d_{33} = (50 + 100|r|)\text{kg/s}$. The authorized range of control signals is $\tau_u \in [150, 300] \text{ (N)}$ and $\tau_r \in [-20, 20] \text{ (N.M)}$. The obstacles are distributed randomly in an 8-meter

radius inside the training zone. An AMD Ryzen 7 3800XT 8-Core, 3.89 GHz CPU processor, and an NVIDIA GeForce RTX 2060 GPU were used to run algorithm 1 in MATLAB.

4.1 DRL parameter configuration and training

The actor-critic networks have a similar structure (Fig .6), which means that double-layer, fully connected networks of 400 and 300 neurons with a RELU activator function have been used. These values were chosen based on the dimensions of the state and action spaces, and they are adequate for approximating the state-action and policy functions in this study.

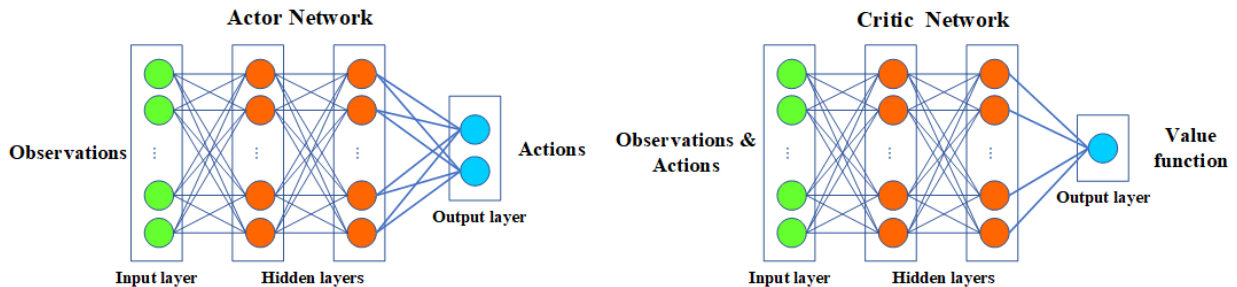


Fig. 6. Structure of the deep neural networks

The action is chosen using the Ornstein-Uhlenbeck process noise to fully leverage the state and action spaces. The following is a definition of the noise process:

$$N_{k+1} = N_k + \alpha_{OU}(\mu_{OU} - N_k) + \sigma_{OU}N_G(0,1) \quad (14)$$

where N_k and N_{k+1} are the Ornstein-Uhlenbeck process values at times k and $k + 1$, respectively. $N_G(0,1)$ is a Gaussian noise having a mean of zero and a standard deviation of one. The parameters of the Ornstein-Uhlenbeck process are α_{OU} , μ_{OU} , and σ_{OU} . The values of the TD3 parameters are shown in Table 1.

Table1: networks' parameters

Parameters	Value
Actor network learning rate	0.001
Critic network learning rate	0.0001
Discount factor	0.99
Memory size	1e6
Smooth update	0.005
Policy and target delay update	2
Target policy noise variance	0.1
Exploration variance	0.1
Sample time	0.5
α_{OU} (Constant)	1
μ_{OU} (The noise model's mean)	0
σ_{OU} (The noise model's variance)	0.1

In each episode, the AUV can travel 350 steps. The AUV's action is the surge force and the moment of yaw, so the heading and position of the AUVs are updated by equations (7) and (8). An episode ends when all AUVs reach the target areas or when the entire time step of each episode has passed. The training phase is conducted for three AUVs with initial positions of (20,10), (15,20), and (5,10), respectively. Obstacles are distributed randomly in the training area. The objective is to train AUVs so that they reach their destinations safely. Each AUV is equipped with a module for avoiding obstacles. Fig. 6 depicts the training diagrams of all three agents. As training episodes increase and the reward value stabilizes, the average reward steadily increases. Three AUVs are used in the training phase. However, five AUVs are used in testing situations to simulate the generalization of DRL capacity. In testing situations, five AUVs are used to simulate the generalization and scalability of proposed structure.

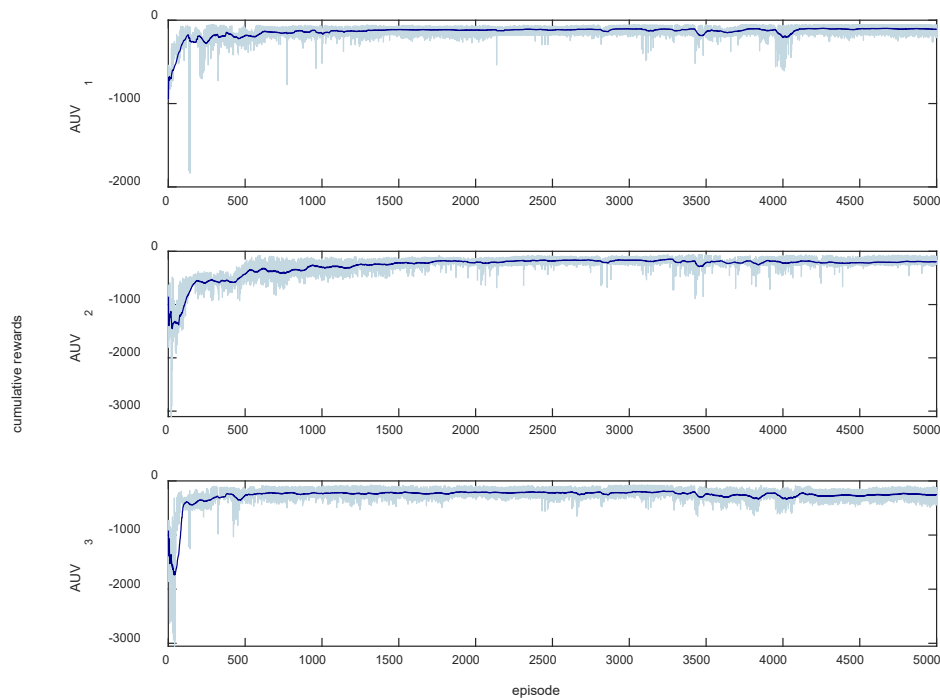


Fig. 7. Training result: the cumulative rewards per episode

4.2 Obstacle-free cooperative motion planning result

In the first section of the evaluation, obstacle-free path planning is considered. Five AUVs are used to assess the system's performance and demonstrate that the intelligent motion planning system has adequate generalization capability. Five AUVs move towards the designated targets at coordinates (10, 15), (20, 20), (20, 10), (25, 15), and (1, 8), respectively. The path of AUVs is depicted in Fig. 8. Each AUV successfully reaches the designated target.

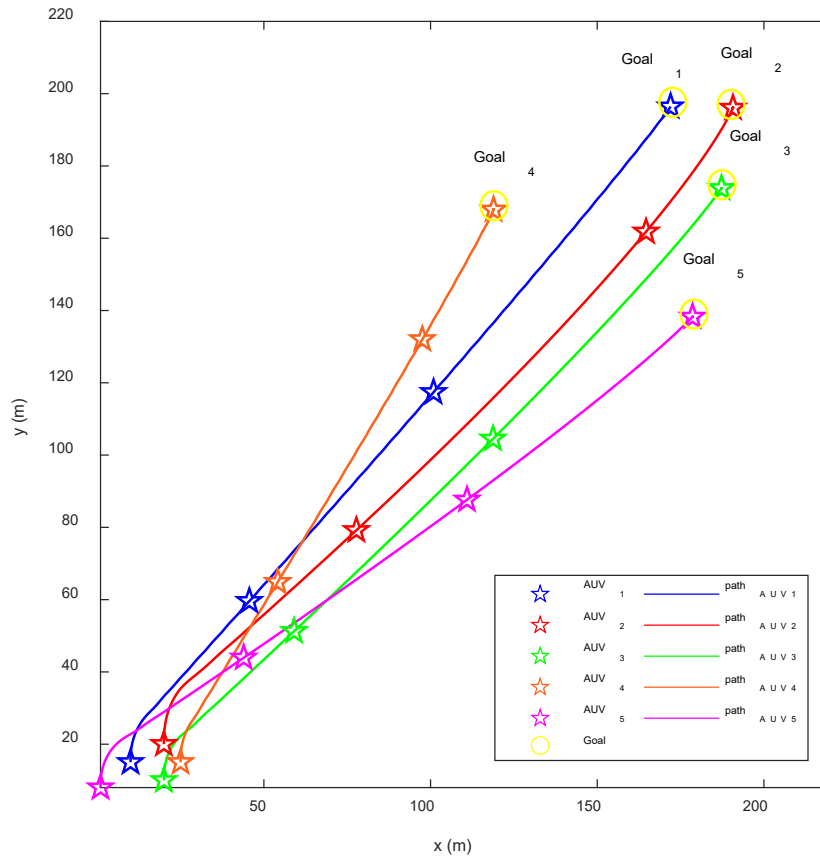


Fig. 8. Trajectories of five AUVs in an obstacle-free scenario

4.3 Cooperative motion planning with obstacles present

Static obstacles with a random distribution of 8 meters in radius are considered in the following scenario. It can be observed (Fig. 9) that AUVs choose their routes so that they avoid obstacles and do not collide with each other. Figures 10 and 11 depict the linear and angular velocities and the control signal. Figure 12 indicates the changing distance between AUVs and obstacles over time. It is evident that the AUVs do not collide with obstacles.

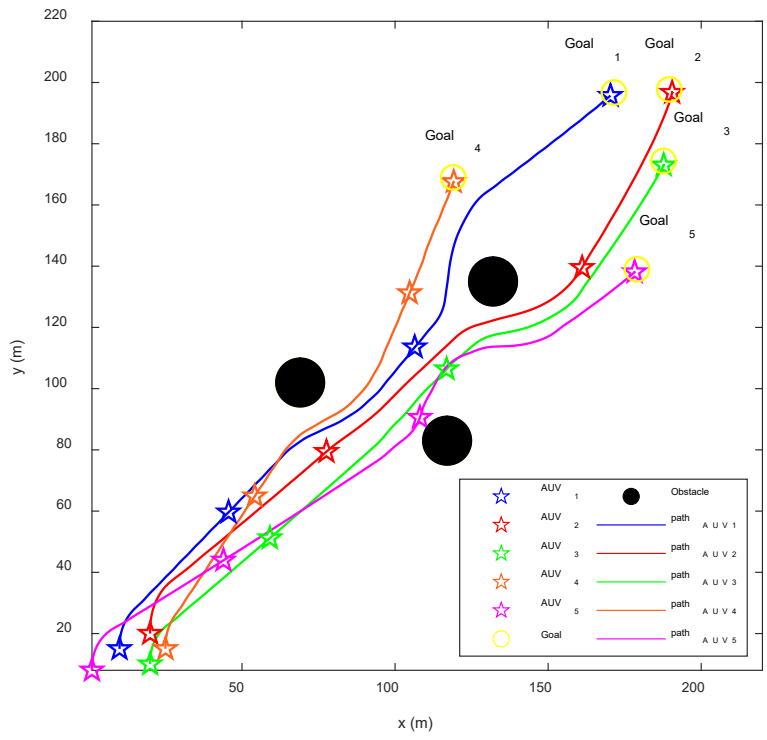


Fig. 9. Trajectories of five AUVs in the presence of obstacles

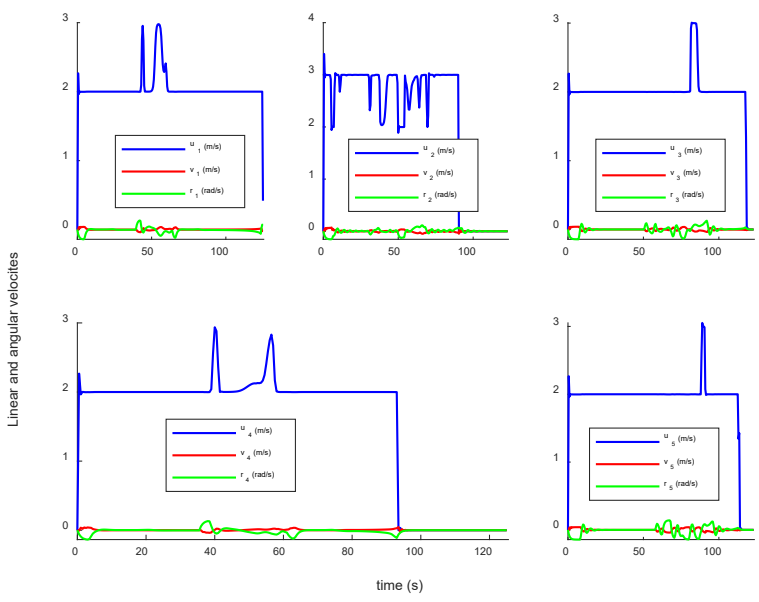


Fig. 10. State variables for five AUVs in the presence of the obstacles

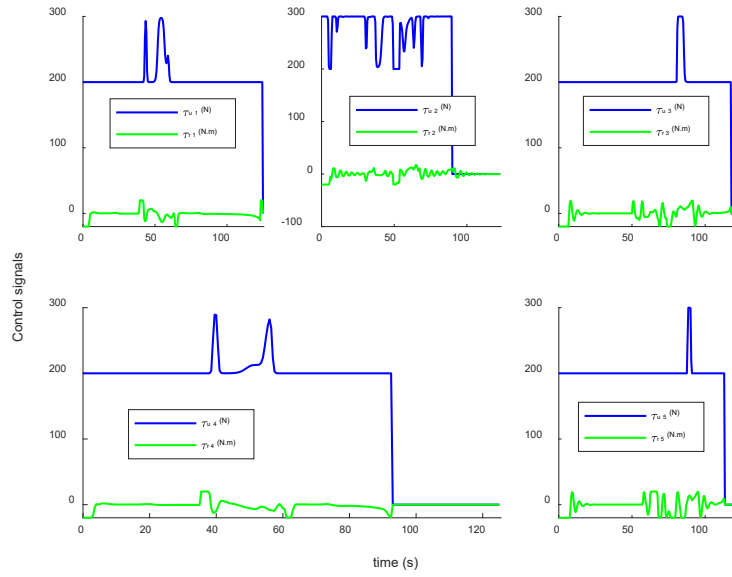


Fig. 11. Control signals for five AUVs in the presence of the obstacles

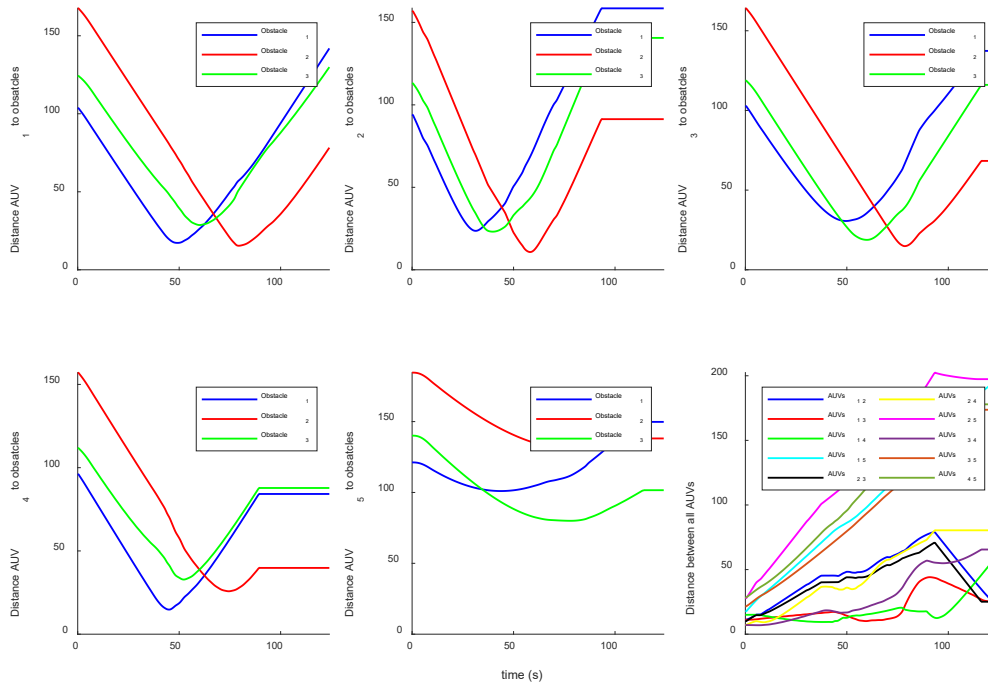


Fig. 12. Variation in the distance between AUVs and obstacles

In order to demonstrate the algorithm's capacity to do intricate tasks, a challenging mission was devised for some AUVs. This objective consisted of a series of successive targets. In the first scenario, AUVs initiated their movement from separate positions and effectively reached the pre-

established destinations. In the second scenario, four consecutive targets in different directions were considered for the second AUV (red color). In the third scenario, AUV3 (green color), four consecutive targets are considered, which include returning to the starting point and then going to the fourth target. In the fourth scenario, successive targets are designed for AUV1 (blue color). The simulation results in Fig. 13 show that the proposed approach has successfully completed complex missions and has generalized well to unseen scenarios.

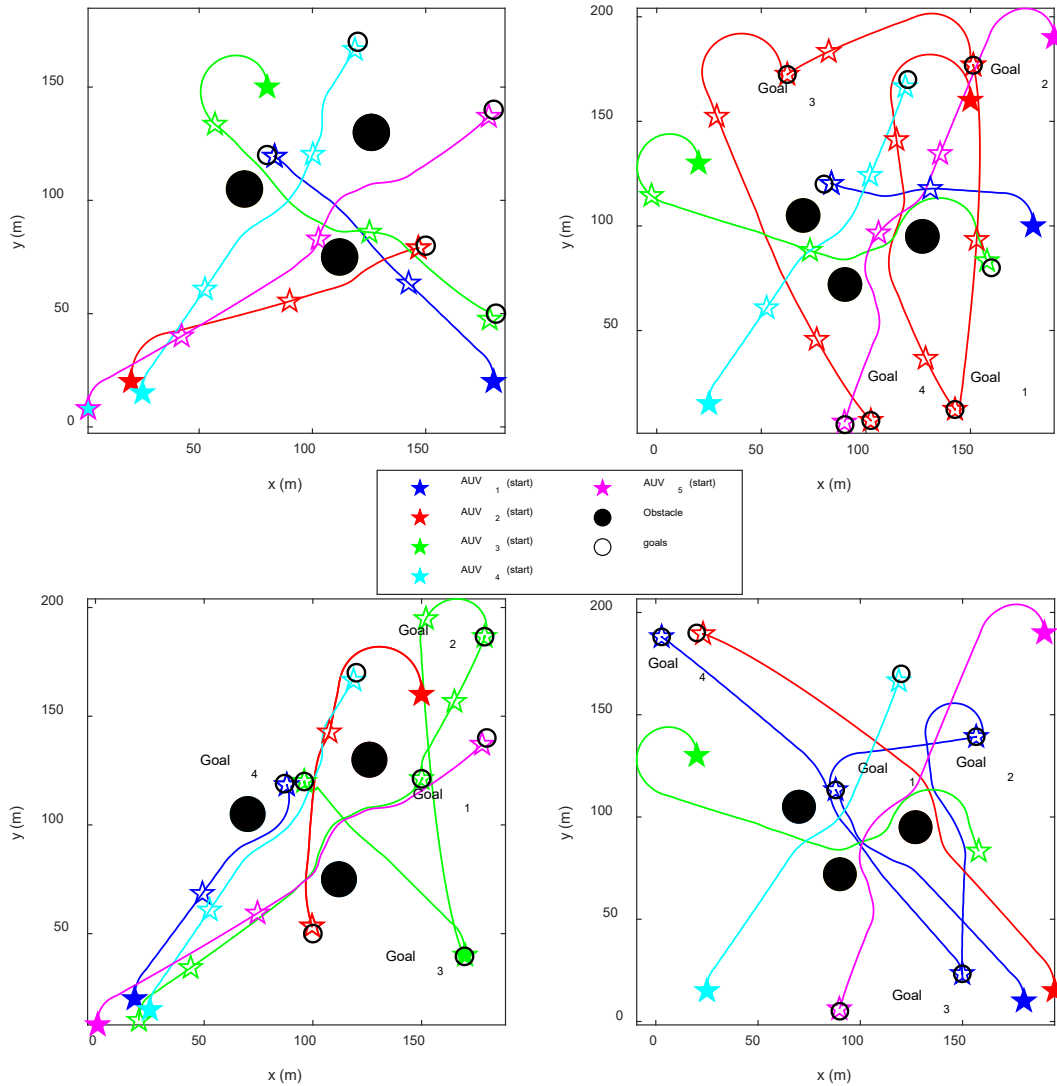


Fig. 13. The outcomes of simulations for single-goal and multi-goal scenarios

The AUVs undergo training to effectively navigate and prevent collisions with both stationary obstacles and other AUVs. To assess the algorithm's performance in the presence of a dynamic obstacle, one of the obstacles was simulated to be in motion in an east-to-west direction at a velocity of 0.3 meters per second.

Figure 14 depicts the AUVs' trajectory when they encounter the moving obstacle, as well as their distance from the obstacle border. According to the diagram, the fifth AUV passes the obstacle with the shortest distance.

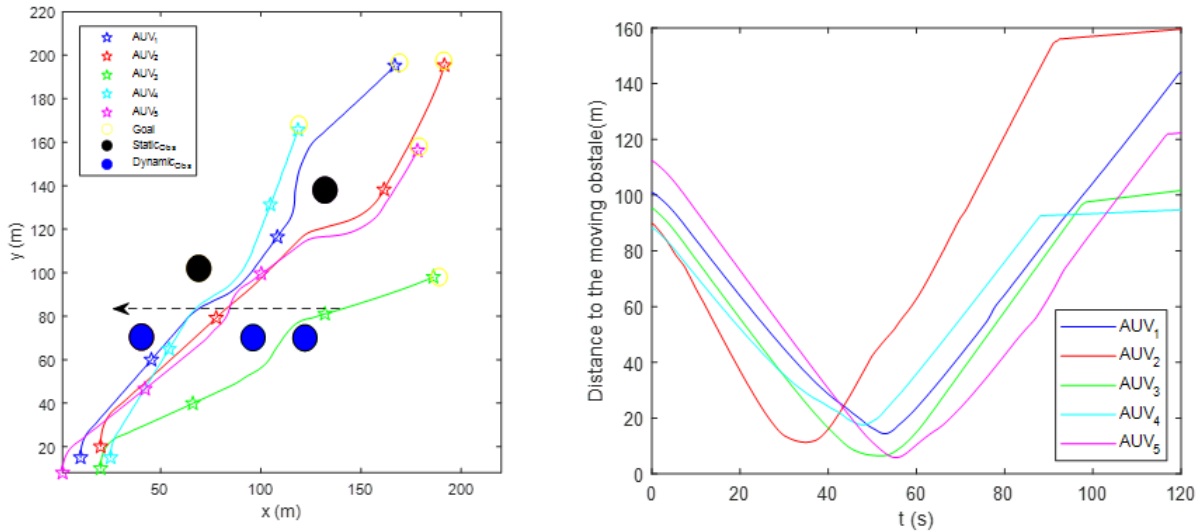


Fig. 14. AUV's trajectories in the presence of static and dynamic obstacles and the distance between the AUVs and the boundary of the dynamic obstacle along the movement path

4.4 Comparison between DRL and artificial potential field path planning

This section presents a comparative analysis between the method proposed in the article and the artificial potential field (APF) technique as one of the accepted approaches in motion planning. To achieve this, cooperative path planning consisting of five AUVs is performed using both the APF, and the results are compared under identical conditions. The task of path planning, which involves reaching targets, avoiding obstacles, and preventing collisions, is accomplished by the utilization of a modified artificial potential approach as outlined in reference [3]. For this purpose, the functions of attraction and repulsion are established. The force of attraction potential facilitates the achievement of the desired objective. The repulsive potential force diminishes the possibility of inter-particle collisions and the probability of encountering obstacles. Figure 13 depicts the APF path planning with a dashed line as the DRL outcomes (solid line). Based on results, the DRL paths are desirable. To comprehensively evaluate the performance of algorithms, quantitative criteria are employed. These criteria include distance traveled, minimum distance to obstacles (MDTO), average distance to obstacles (ADTO), integral of the absolute control effort ($IACE = \int_0^{t_f} |\tau_r| dt$), and integral of the absolute control effort rate ($IACER = \int_0^{t_f} \left| \frac{d\tau_r}{dt} \right| dt$). Based on the results presented in Table 2, the proposed DRL algorithm demonstrates comparable performance compared to APF. Selecting an optimal path by the DRL algorithm results in a lower average distance to obstacles compared to the APF method. Moreover, based on the IACE and IACER

criteria, the quality of control signals generated by the DRL method compared to the APF is more desirable for implementation.

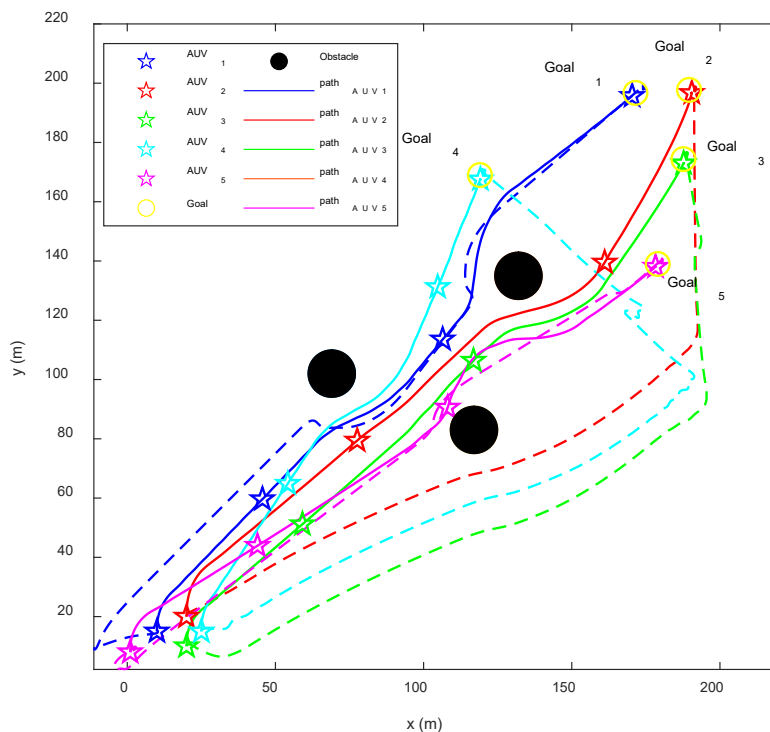


Fig. 15. Comparison of path planning results: The solid line is DRL path planning, and the corresponding dashed line is APF path planning results.

Table2: Performance evaluation of DRL and APF methods

	DRL					APF				
	Travelled Distance (m)	MDTO (m)	ADTO (m)	IACE	IACER	Travelled Distance (m)	MDTO (m)	ADTO (m)	IACE	IACER
AUV1	249.4	6.72	62.79	8.42	7.75	344.74	8.98	15.74	27.74	404.08
AUV2	254.79	4.29	75.58	10.33	9.22	338.04	6.60	13.25	29.24	448.17
AUV3	239.86	8.09	63.11	8.26	6.07	338.74	26.03	14.47	29.33	444.89
AUV4	183.35	6.60	57.11	10.21	4.45	345.19	10.75	13.60	29.72	395.13
AUV5	227.90	3.76	63.87	10.70	8.85	322	323	13.33	35.17	532.99

4.5 Ocean currents and cooperative motion planning

Underwater currents in the ocean are vertical and horizontal flow patterns caused by different factors, such as wind friction and gravity. To include ocean currents and how they affect AUV movement, the equations of motion can be written in terms of relative velocity [39]:

$$v_r = v - v_c \quad (15)$$

where $v_c = [u_c, v_c, 0]^T$ denotes the ocean current in a body-fixed frame, $v = [u, v, r]^T$ represents the linear and angular velocities of AUV in a body-fixed frame. For the 2-D current model, we have:

$$\begin{aligned} u_c &= V_c \cos(\beta_c - \psi) \\ v_c &= V_c \sin(\beta_c - \psi) \end{aligned} \quad (16)$$

where ψ is the AUV's heading. V_c and β_c are the speed and direction of the ocean current. As a result, the dynamic equations of the AUV are used based on the relative velocity[39].

Cooperative motion planning begins under the constant direction and amplitude of the ocean current to evaluate the robustness of the designed model. Figure 14 presents the AUVs' paths to the destinations with and without ocean current. The ocean current's speed is 0.15 m/s, and its direction relative to the x-axis is 70 degrees. Despite a little deviation in the AUV's path, the results show that the group can safely arrive at its destination. As a result, the DRL-based cooperative motion planning system is adaptable and robust enough.

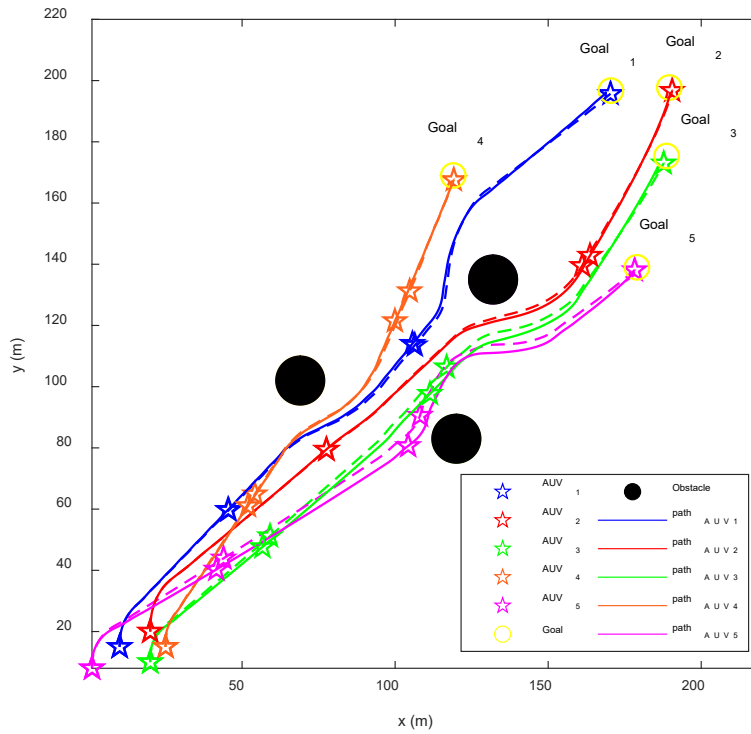


Fig. 16. The trajectories of AUVs in the ocean current, dashed line: AUVs' trajectories without ocean current, solid line: AUVs' trajectories with ocean current

Finding the maximum ocean current that the system can withstand depends on the testing scenario and becomes stochastic in general. However, we tested various conditions to determine the

maximum tolerability of the algorithm for this particular scenario. We applied four currents—North-South, South-North, West-East, and East-West—with various magnitudes to assess tolerances. Our findings indicate that the best case was North-South with a tolerance of 0.45 m/s, while West-East was the worst case with a 0.2 m/s tolerance. These results are acceptable, given the challenging scenario and the considerably lower speed of operation for these AUVs, approximately ~ 2 m/s. Therefore, the developed algorithm demonstrates acceptable robustness against external disturbances like ocean currents, which are common in practice.

Overall, the developed algorithms are evaluated in multiple testing scenarios. AUVs success is determined by their ability to reach random targets, avoid collisions with static obstacles arbitrarily placed in their path, and maintain a safe distance from one another. Additionally, AUVs achieve objectives in non-training directions and on consecutive missions in various directions without collisions. The proposed algorithm demonstrates proper robustness when confronted with the ocean current. Moreover, the developed algorithm demonstrated acceptable performance even when confronted with moving obstacles. The suggested method outperforms the APF method in terms of performance measures such as travel distance, average distance to obstacles, the minimum distance to obstacles, integral of the absolute control effort (IACE) and integral of the absolute control effort rate (IACER).

5. Conclusion

This paper proposes a novel cooperative motion control and obstacle avoidance approach based on machine learning for multiple AUVs that does not require complex control design, unlike most conventional analysis techniques. The TD3 algorithm, a policy gradient algorithm with a deterministic policy for continuous action, has been employed. Continuous control signals are appropriately calculated and applied to the AUVs to accomplish the mission. A multi-layer control concept is developed for obstacle- and self-collision avoidance so that AUVs' sensitivity to the distance from surrounding obstacles is increased. To evaluate the efficacy of the suggested end-to-end motion planning and control method, various simulation scenarios are conducted. Moreover, the performance in the presence of ocean currents demonstrates the robustness of the proposed scheme under realistic conditions. In summary, the key findings are as follows:

- The proposed AI-based algorithm successfully accomplishes the cooperative task, avoiding collisions with obstacles and other AUVs.
- The method demonstrates robustness against disturbances caused by oceanic currents.
- The generated control signals are feasible.
- Performance indicators, including travel distance, minimum distance to obstacles, IACE, IACER, and average distance to obstacles, reveal comparable results compared to the APF method.
- The generalizability of the method is demonstrated by the successful performance of a group of AUVs in challenging tasks.

Therefore, the proposed technique demonstrates merits for further considerations. Future work can include the experimental implementation of the suggested algorithm.

6. References

- [1] P. Sarhadi, A. R. Noei, and A. Khosravi, "Model reference adaptive PID control with anti-windup compensator for an autonomous underwater vehicle," *Robotics and Autonomous Systems*, vol. 83, pp. 87-93, 2016/09/01/ 2016.
- [2] Y. Yang, Y. Xiao, and T. Li, "A survey of autonomous underwater vehicle formation: Performance, formation control, and communication capability," *IEEE Communications Surveys Tutorials*, vol. 23, no. 2, pp. 815-841, 2021.
- [3] B. Hadi, A. Khosravi, and P. Sarhadi, "A review of the path planning and formation control for multiple autonomous underwater vehicles," *Journal of Intelligent and Robotic Systems*, vol. 101, no. 4, pp. 1-26, 2021.
- [4] Z. Zeng, K. Sammut, L. Lian, A. Lammass, F. He, and Y. Tang, "Rendezvous path planning for multiple autonomous marine vehicles," *IEEE Journal of Oceanic Engineering*, vol. 43, no. 3, pp. 640-664, 2017.
- [5] D. Li, P. Wang, and L. J. I. A. Du, "Path planning technologies for autonomous underwater vehicles-a review," *Ieee Access*, vol. 7, pp. 9745-9768, 2018.
- [6] M. Reda, A. Onsy, A. Ghanbari, A. Y. J. R. Haikal, and A. Systems, "Path planning algorithms in the autonomous driving system: A comprehensive review," *Robotics and Autonomous Systems*, p. 104630, 2024.
- [7] C. E. Okereke, M. M. Mohamad, N. H. A. Wahab, O. Elijah, and A. J. I. A. Al-Nahari, "An Overview of Machine Learning Techniques in Local Path Planning for Autonomous Underwater Vehicles," *IEEE Access*, 2023.
- [8] Y. Zhuang, H. Huang, S. Sharma, D. Xu, and Q. Zhang, "Cooperative path planning of multiple autonomous underwater vehicles operating in dynamic ocean environment," *ISA transactions*, vol. 94, pp. 174-186, 2019.
- [9] L. Zhi, Y. J. J. o. M. S. Zuo, and Engineering, "Collaborative Path Planning of Multiple AUVs Based on Adaptive Multi-Population PSO," *Journal of Marine Science Engineering*, vol. 12, no. 2, p. 223, 2024.
- [10] X. Li and D. Zhu, "An Adaptive SOM Neural Network Method for Distributed Formation Control of a Group of AUVs," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 10, pp. 8260-8270, 2018.
- [11] Z. Huang, D. Zhu, and B. Sun, "A multi-AUV cooperative hunting method in 3-D underwater environment with obstacle," *Engineering Applications of Artificial Intelligence*, vol. 50, pp. 192-200, 2016.
- [12] X. Cao, H. Sun, and G. E. Jan, "Multi-AUV cooperative target search and tracking in unknown underwater environment," *Ocean Engineering*, vol. 150, pp. 1-11, 2018.
- [13] W. Cai, Z. Liu, M. Zhang, C. J. R. Wang, and A. Systems, "Cooperative Artificial Intelligence for underwater robotic swarm," *Robotics and Autonomous Systems*, vol. 164, p. 104410, 2023.
- [14] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis, "Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers," *IEEE Control Systems Magazine*, vol. 32, no. 6, pp. 76-105, 2012.
- [15] L. Buşoniu, T. de Bruin, D. Tolić, J. Kober, and I. Palunko, "Reinforcement learning for control: Performance, stability, and deep approximators," *Annual Reviews in Control*, vol. 46, pp. 8-28, 2018.
- [16] V. B. Gjørnum, I. Strümke, J. Løver, T. Miller, and A. M. Lekkas, "Model tree methods for explaining deep reinforcement learning agents in real-time robotic applications," *Neurocomputing*, vol. 515, pp. 133-144, 2023.
- [17] P. Sarhadi, W. Naeem, and N. J. I.-P. Athanasopoulos, "A Survey of Recent Machine Learning Solutions for Ship Collision Avoidance and Mission Planning," *IFAC-PapersOnLine*, vol. 55, no. 31, pp. 257-268, 2022.
- [18] Y. Sun, X. Ran, G. Zhang, H. Xu, X. J. J. o. m. s. Wang, and engineering, "AUV 3D path planning based on the improved hierarchical deep Q network," *Journal of marine science engineering*, vol. 8, no. 2, p. 145, 2020.
- [19] Z. Chu, Y. Wang, D. J. I. T. o. S. Zhu, Man,, and C. Systems, "Local 2-D Path Planning of Unmanned Underwater Vehicles in Continuous Action Space Based on the Twin-Delayed Deep Deterministic Policy Gradient," *IEEE Transactions on Systems, Man, Cybernetics: Systems*, 2024.

- [20] J. Wu, L. Cheng, S. Chu, and Y. J. O. e. Song, "An autonomous coverage path planning algorithm for maritime search and rescue of persons-in-water based on deep reinforcement learning," *Ocean engineering*, vol. 291, p. 116403, 2024.
- [21] C. Liu *et al.*, "Efficient Beacon-Aided AUV Localization: A Reinforcement Learning Based Approach," *IEEE Transactions on Vehicular Technology*, 2024.
- [22] P. Bhopale, F. Kazi, N. J. J. o. M. S. Singh, and Application, "Reinforcement learning based obstacle avoidance for autonomous underwater vehicle," *Journal of Marine Science Application*, vol. 18, pp. 228-238, 2019.
- [23] B. Hadi, A. Khosravi, and P. J. A. O. R. Sarhadi, "Deep reinforcement learning for adaptive path planning and control of an autonomous underwater vehicle," *Applied Ocean Research*, vol. 129, p. 103326, 2022.
- [24] F. Huang *et al.*, "A general motion controller based on deep reinforcement learning for an autonomous underwater vehicle with unknown disturbances," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105589, 2023.
- [25] J. Yan, K. You, W. Cao, X. Yang, and X. J. I. T. o. I. V. Guan, "Binocular Vision-Based Motion Planning of An AUV: A Deep Reinforcement Learning Approach," *IEEE Transactions on Intelligent Vehicles*, 2023.
- [26] M. Xi *et al.*, "An Information-Assisted Deep Reinforcement Learning Path Planning Scheme for Dynamic and Unknown Underwater Environment," *IEEE Transactions on Neural Networks Learning Systems* 2023.
- [27] X. Xu, P. Cai, Z. Ahmed, V. S. Yellapu, and W. Zhang, "Path planning and dynamic collision avoidance algorithm under COLREGs via deep reinforcement learning," *Neurocomputing*, vol. 468, pp. 181-197, 2022.
- [28] L. YongZhou, L. GuangYu, and G. Xuan, "Multi-UUV Path Planning Study with Improved Ant Colony Algorithm and DDQN Algorithm," in *2021 IEEE 7th International Conference on Control Science and Systems Engineering (ICCSSE)*, 2021, pp. 143-148.
- [29] S. Wang, F. Ma, X. Yan, P. Wu, and Y. Liu, "Adaptive and extendable control of unmanned surface vehicle formations using distributed deep reinforcement learning," *Applied Ocean Research*, vol. 110, p. 102590, 2021.
- [30] L. Zhao and M.-I. Roh, "COLREGs-compliant multiship collision avoidance based on deep reinforcement learning," *Ocean Engineering*, vol. 191, p. 106436, 2019.
- [31] B. Hadi, A. Khosravi, and P. Sarhadi, "Adaptive formation motion planning and control of autonomous underwater vehicles based on deep reinforcement learning," *IEEE oceanic engineering*, 2023.
- [32] N. T. Hung, F. F. Rego, and A. M. Pascoal, "Cooperative distributed estimation and control of multiple autonomous vehicles for range-based underwater target localization and pursuit," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 4, pp. 1433-1447, 2021.
- [33] N. T. Hung, N. Crasta, D. Moreno-Salinas, A. M. Pascoal, and T. A. Johansen, "Range-based target localization and pursuit with autonomous vehicles: An approach using posterior CRLB and model predictive control," *Robotics and Autonomous Systems*, vol. 132, p. 103608, 2020.
- [34] Y. Hou, G. Han, F. Zhang, C. Lin, J. Peng, and L. Liu, "Distributional Soft Actor-Critic-Based Multi-AUV Cooperative Pursuit for Maritime Security Protection," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1-12, 2023.
- [35] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.
- [36] R. S. Sutton and A. G. Barto, "Introduction to reinforcement learning," 1998.
- [37] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *International conference on machine learning*, 2018, pp. 1587-1596: PMLR.
- [38] R. Cui, S. S. Ge, B. V. E. How, and Y. S. J. O. E. Choo, "Leader-follower formation control of underactuated autonomous underwater vehicles," *Ocean Engineering*, vol. 37, no. 17-18, pp. 1491-1502, 2010.
- [39] T. I. Fossen, "Guidance and control of ocean vehicles," University of Trondheim, Norway: Printed by John Wiley & Sons, Chichester, England, ISBN: 0 471 94113 1, Doctors Thesis, 1999.

Declarations

Funding

The authors declare that no funds, grants, or other support were received during the preparation of this manuscript.

Competing Interests

The authors have no relevant financial or non-financial interests to disclose.

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.

Author Contributions

All authors contributed to the study conception and design. Material preparation, data collection and analysis were performed by Behnaz Hadi. The first draft of the manuscript was written by Behnaz Hadi and all authors commented on previous versions of the manuscript. All authors read and approved the final manuscript.

Ethics approval Not applicable

Consent to participate Not applicable

Consent to publish Not applicable