

Reducing AUV Energy Consumption Through Dynamic Sensor Directions Switching via Deep Reinforcement Learning

Jiawei Liu^{1,2*}, Yuanbo Xu^{1,2*}, Shanshan Song^{2†}, Lu Jiang³

¹MIC Lab, College of Computer Science and Technology, Jilin University, China

²College of Computer Science and Technology, Jilin University, China

³Information Science and Technology, Dalian Maritime University, China

jiawei23@mail.jlu.edu.cn, {yuanbox, songss}@jlu.edu.cn, jiangl761@dlmu.edu.cn

Abstract

Autonomous underwater vehicle (AUV) is crucial for marine applications such as ocean data collection, pollution monitoring, and navigation. However, their limited energy resources constrain their operational duration, posing a significant challenge for long-term operations. Due to the complex and unpredictable nature of the underwater environment, AUVs allocate energy to their sensing systems to sense the surrounding environment and avoid obstacles. Existing methods focus on reducing energy consumption on AUV computing and movement, neglecting sensing energy consumption and few attempts have been made to balance the AUV energy and sensing ability with a flexible sensing system. Along these lines, we consider both AUV energy consumption and flexible sensing abilities, and propose a deep reinforcement learning-based method to Reduce Energy Consumption by AUV Sensing system (RECS). Specifically, we build an AUV sensing system in a 2-dimension space, with controllable 8-direction sensing abilities to collect the environment information dynamically. Then we divide the underwater environment into several areas and assign weights on the edges of areas based on the AUV planned path. Additionally, we dynamically switch the sensors in different directions and radii to sense the edges of the area where the AUV is located. The Artificial Potential Field (APF) method is employed to re-plan the AUV path to avoid obstacles and reach the target point effectively. Experimental results demonstrate that compared to full sensors on, our method reduces energy consumption by 53.48% and is capable of generalizing to varying environments and varying sensing system radii.

Introduction

AUV, short for Autonomous Underwater Vehicle, is an unmanned robot capable of autonomously executing missions in underwater environments through pre-programmed program or self-learning algorithms (Yan et al.,2023a). AUVs perform various types of marine applications, such as ocean data collection (Wang et al.,2023b), pollution monitoring (Lin et al.,2022), and navigation (Vial et al.,2023). AUVs require environmental information provided by underwater

wireless sensor networks to plan paths for missions. However, pre-deploying underwater wireless sensor networks before each mission is not feasible due to the high deployment costs with underwater wireless sensor networks (Han et al.,2023). It implies that AUVs typically perform missions in unknown environments and utilize their sensing systems to sense the surrounding environment to obtain environmental information along the path from the start point to the target point.

Batteries are the sole energy source for AUVs, and due to the challenges of underwater recharging, AUV operational duration is limited by their energy consumption (Hou et al.,2022). Vision-based methods (Xue et al.,2022;Sun et al.,2023) or acoustic-based methods (Zhang et al.,2022) are utilized to construct AUV sensing systems and each method requires significant energy consumption. Despite the sensing systems consuming significant energy, it is imperative to consume energy to sense the surrounding environment to avoid obstacles due to the complexity of the underwater environment and the expensive AUVs (Chu et al.,2022). The range of sensing areas is a circular sensing area centered on the AUV with a radius r . AUVs moving from the start point to the target point generate numerous sensing areas, as shown in Figure 1. With full sensors on, the AUV successfully utilized its sensing system to sense the surrounding environment, reached the target point, and avoided obstacles, with 26% of its energy remaining. However, it is evident that the sensed areas covered by the AUV sensing system include some ineffective sensing areas. The ineffective sensing areas are defined as the sensing areas which do not sense the AUV planned path and consume additional energy compared to the part sensors on which the energy remaining is 50%. Thus, reducing ineffective sensing areas can effectively reduce the AUV energy consumption. To reduce ineffective sensing areas, we divided the sensing system into multiple switchable sensing directions to sense effective areas and proposed a method for dynamically controlling both the sensing direction and sensing radii.

We proposed a deep reinforcement learning-based method to reduce energy consumption by AUV sensing system (RECS). RECS reduces energy consumption utilizing the following novel designs:

1. We define the maximum radius that the sensing system covers as r and divide the unknown underwater envi-

*These authors contributed equally.

†Corresponding author.

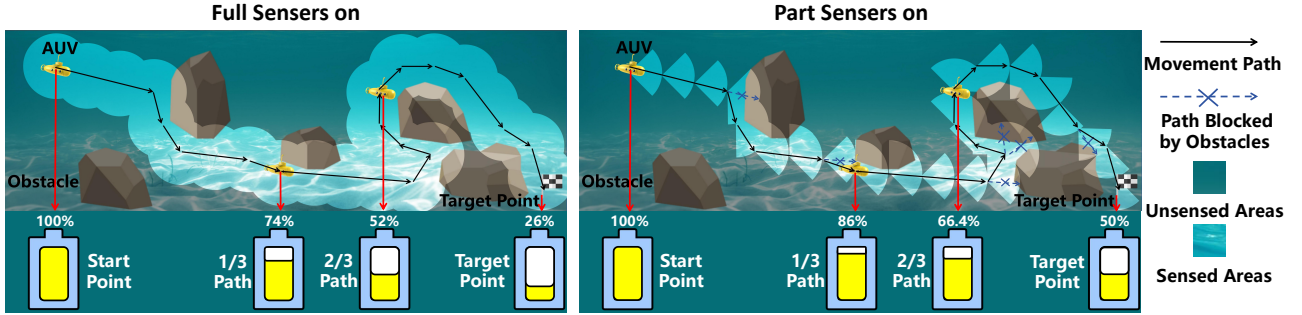


Figure 1: A comparison of the sensed areas, movement path, and remaining energy between fully activated AUV sensors and dynamically switched sensors. Dynamically switched sensors reduce the sensed areas and consume less energy without changing the planned path.

ronment into several square areas, with each square side length being $\frac{\sqrt{2}}{2}r$. This configuration guarantees that the AUV fully senses the entire square area regardless of its position within the square area.

2. We plan the AUV path based on the starting point and target point. Based on the areas traversed by this path, RECS assigns weights to the edges of all areas.
3. Before the AUV moves along the planned path, RECS selects the edges to be sensed based on the edge weights of the area where the AUV is located and assigns the sensing direction and sensing radius based on the relative position of the AUV to this edge. When obstacles are sensed, RECS re-plans the path and reassigns all edges of area weights.

Related Work

Trajectory optimization: (Mahmoodi and Uysal,2022) optimizes AUV energy usage by employing a hybrid propulsion system, real-time trajectory adjustments, energy-efficient operational modes, and maximizing solar energy harvesting based on environmental conditions. (Hou et al.,2022) proposes a framework for enhancing AUV energy efficiency by optimizing their trajectories, incorporating communication resource allocation, computation offloading, and data caching to balance energy consumption and operational requirements of IoUT devices. (Yu, Zheng, and Xu,2024) proposes a method that leverages ocean current data to plan energy-efficient paths, reducing energy consumption by allowing the AUVs to move with the current when possible. The approach enhances the overall efficiency of AUV operations by optimizing the balance between energy expenditure and mission success. (Choudhary et al.,2024) presents a hybrid optimization scheme designed to improve the energy efficiency of AUVs used for underwater data collection and leakage detection in underwater wireless sensor networks (UWSNs).

Communication and data transmission efficiency: (Chi et al.,2024) investigates the efficiency and optimization of energy consumption in underwater acoustic sensor networks by proposing a table forwarding-based election algo-

rithm and 3D path planning to balance energy usage and extend network lifespan, while considering sensor mobility and offshore currents. (Yan et al.,2023b) discusses a communication-aware motion planning strategy for AUVs in obstructed environments to improve data transmission efficiency and reduce energy consumption.

Collaborative and energy-efficient data collection: (Han et al.,2023) propose a selective line-of-sight technique used to smooth paths, reducing unnecessary turns and movements, which helps in conserving energy. (Wang et al.,2023b) proposes a collaborative data collection method for multiple AUVs, emphasizing energy-efficient strategies to extend the operational duration and enhance the coverage of ocean data collection missions. (Xi et al.,2022) presents an energy-aware data collection framework that optimizes the trajectories of AUVs to reduce energy consumption while maintaining effective data from underwater sensor networks.

(Xu et al.,2024b;Jiang et al.,2024a;Wang et al.,2023a) emphasize efficient prediction and decision-making in the absence of precise prior knowledge. (Xu et al.,2024a;Xu et al.,2022) provide novel approaches for task optimization in resource-constrained environments. (Yang et al.,2024;Jiang et al.,2024b;Xu et al.,2022b) offer insights into secure data processing and dynamic task optimization.

Background

Deep Reinforcement Learning

Deep reinforcement learning is typically formulated as a Markov Decision Process (MDP), represented by a tuple $M = \{S, A, P, r, \gamma\}$, where S denotes the set of states and A denotes the set of actions $A(s)$ associated with the state $s \in S$. The state transition probability is denoted by $P(s'|s, a)$, and r represents the reward function. Additionally, $\gamma \in [0, 1]$ is the discount factor, determining the weight of future rewards in the value of the discounted return. The smaller γ places more emphasis on immediate rewards, while values closer to 1 account for rewards further in the future. The agent follows a policy $\pi(a|s)$ to select actions a to be performed in state s . The policy $\pi(a|s)$ repre-

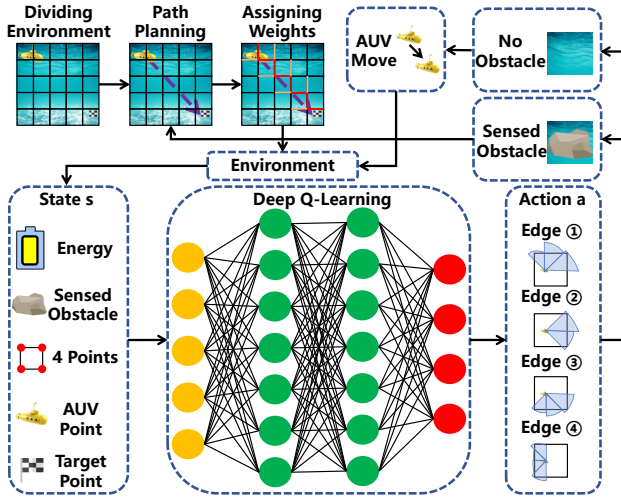


Figure 2: Schematic of RECS. At each step, RECS receives states from the environment and outputs the sensing edge for the area.

sents the probability of selecting action a given state s . The agent objective is to solve the Bellman optimality equation as shown in Equation 1.

$$v(s) = \max_{\pi} \sum_a \pi(a|s) \left[\sum_r p(r|s, a) r + \gamma \sum_{s'} p(s'|s, a) v_{\pi}(s') \right] \quad (1)$$

Q-learning, a classic reinforcement learning algorithm, is a transformation of the Bellman optimality equation (Watkins and Dayan, 1992). Deep Q-Network (DQN) integrates neural networks into reinforcement learning (Mnih et al., 2013). There is a main network and a target network in DQN. The main network is used to select actions and compute Q-values, with its parameters updated at every training step. The target network is introduced to ensure greater stability in the DQN training process, preventing sharp fluctuations and overestimation of target values, thereby improving convergence and overall training effectiveness. The experience replay mechanism is employed due to the bias caused by the non-uniform distribution of samples. By employing experience replay, DQN mitigates the correlation in the sampling process, thereby enhancing the stability and efficiency of the training process. RECS utilizes DQN as the core to dynamically control the AUV sensing system.

Artificial Potential Field

The concept of artificial potential fields creates a simulated potential field within the environment, where attractive forces draw the agent towards the target point, and repulsive forces push the agent away from obstacles (Khatib, 1986). By calculating the resultant force within the field, the agent gradually moves toward the target point while simultaneously avoiding obstacles (Pang, Zhu, and Sun, 2023). This potential field is usually formulated as a function of the distance to the target point and obstacles, and the agent navigates by moving in the direction of the steepest descent in

this potential field. Therefore, the AUV successfully navigates to the target point while dynamically avoiding obstacles, even in the absence of a detailed environmental map by the APF algorithm. RECS utilizes APF to help AUVs plan the path and avoid obstacles in unknown environments.

Method

RECS is shown in Figure 2. We divide the environment into several areas based on the sensing radius, and plan the path according to the AUV position and the target point position, and assign each edge weight according to the planned path when the AUV enters a new environment. Next, RECS inputs the state information and outputs the action which edges the AUV selects to sense and dynamically switches sensing directions and optimizes sensing radii to sense the edge. RECS returned to the path planning stage when the AUV sensed obstacles. The AUV moves along the planned path to an edge of its current area and continues sensing until it reaches the target point when no obstacle is sensed.

To switch the sensing directions, we divide the sensing system into eight 45° segments: upper-right, right-upper, right-lower, lower-right, lower-left, left-lower, left-upper, and upper-left, as shown in Figure 3a. We define the energy consumption with all sensors fully activated in the maximum sensing radius as E_f and the energy consumption for each exploration direction is $E_d = \frac{E_f}{8}$. The distance the AUV travels along the shortest path to reach the target point is denoted as d_s and the maximum sensing radius as r . In a no obstacle environment with all sensors fully activated, the total energy consumption required is $E_t = \frac{d_s E_f}{\max(r)}$. In RECS E_t refers to the energy consumed by sensing systems. The initial energy for the AUV is defined based on the distance to the target point in RECS, but it does not represent the total real available energy of the AUV. The total real available energy required for the AUV is distributed among various systems, including the propulsion system, communication system, and other subsystems (Chang et al., 2022). Therefore, if E_t exceeds the pre-defined energy, it will not result in mission failure. The AUV can reallocate energy from the allocations of other systems to the sensing system.

It is essential to understand the actual sensing system energy consumption and establish the relationship between sensing radii and energy consumption to simulate a real-world environment. We refer to the settings in (Danielis et al., 2022), and the energy consumption formula for the sensing system is as Equation 2:

$$E_f = \begin{cases} 80W, & 3.5km < r, \\ 35W, & 2.0km < r \leq 3.5km, \\ 8W, & 1.0km < r \leq 2.0km, \\ 2.5W, & r \leq 1.0km. \end{cases} \quad (2)$$

With the r increasing, energy consumption grows exponentially. Therefore, in RECS, the maximum range of r is set to be less than 1.0 km. Additionally, the setting is implemented so that when the r increases or decreases, E_f changes linearly with the r in RECS.

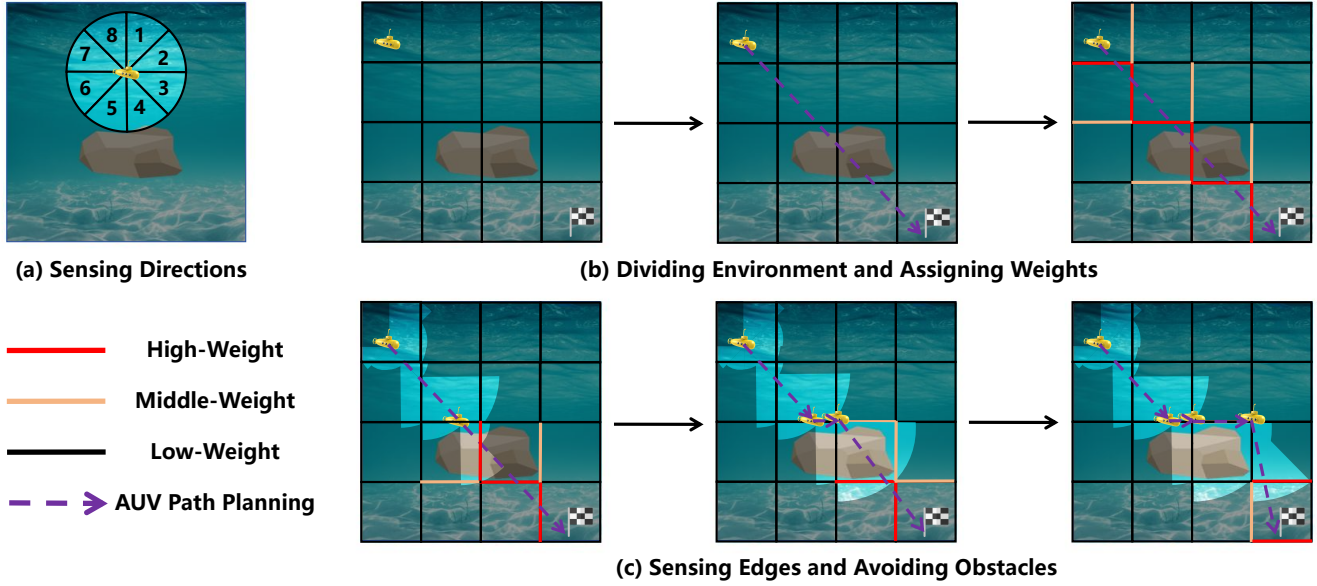


Figure 3: Sensing directions illustrates the division of the sensing system into eight directions. Dividing environment and assigning weights illustrate RECS divides the environment into square areas and assigns weights to the edges. Sensing edges and avoiding obstacles illustrate RECS sensed edges, avoids obstacles, and reassigned edge weights.

Dividing Environments

It is necessary to divide the underwater environment into several areas, due to the uncertainty of underwater environments and the necessity for the AUV to control the sensing directions and radii based on its current position. Therefore, a square modeling method is adopted to divide the environment into several square areas. The mission allocation uses square modeling, where the side length of each square is $\frac{\sqrt{2}}{2}r$ according to the AUV sensing radius. It ensures that when the AUV is at one vertex of the square, it can sense the entire square, as shown in Figure 3c. (Han et al.,2023) suggests that existing local path planning algorithms often generate paths that are too close to obstacles, thereby increasing the risk of collision, it is necessary to designate infeasible areas and mark the cells containing obstacles as off-limits for the AUV. However, this method may result in the AUV being unable to reach its destination. For example, if the target point is surrounded by obstacles marked as impassable areas for the AUV, the AUV will be unable to reach the target point. In the real environment, there might be sufficient space for the AUV to pass through. Therefore, RECS does not designate impassable areas but instead employs APF to maintain a safe distance from obstacles.

Assigning Weights

After dividing environments into several areas, RECS plans a path based on the positions of the starting point and the target point and assigns different weights to the edges of each area according to the planned path. Assigning weights helps RECS obtain rewards in DQN when sensing edges. Each time the AUV senses an edge, its current position and the sensed edge are recorded in a database to prevent re-

peat sensing. Therefore, the weight assignment represents only the weight for the AUV to sense the target edge from its current position and does not apply to sensing the same edge from other positions. The process from determining the AUV movement path to the partitioning of the environment and the assignment of weights to different areas are presented in Figure 3b. Colored edges indicate higher weights have been assigned to them. It can be observed that the edges with higher weights are all situated along the AUV planned path. Before introducing the specifics of the weight design, it is necessary to define the energy required to sense an edge as shown in Equation 3.

$$E_{\text{edge}} = \sum_{i=1}^n E_d \times \frac{r_i}{\max(r)}, \quad (3)$$

where E_{edge} represents the energy required to sense an edge; the r_i represents the sensing radius in the i -th direction.

We assign the weights into three weight categories: high-weight, medium-weight, and low-weight. The low-weight edges are marked in black in Figure 3b. The AUV will be penalized for sensing low-weight edges as Equation 4, due to its planned path does not pass through these edges. Sensing low-weight edges merely increases energy consumption without helping the AVU complete its missions.

$$W_{\text{low}} = -E_{\text{edge}} \times N_{\text{edges}}, \quad (4)$$

where N_{edges} represents the total number of edges sensed in this action, with a maximum of 4 and a minimum of 1.

The medium weights are marked in orange in Figure 3b as Equation 5. The medium-weight edges can be selectively sensed based on the current energy consumption. There are two scenarios for assigning medium-weight to edges: one is

the edge closest to a high-weight edge within the same area, and the other is all edges within an area sensing an obstacle.

$$W_{\text{medium}} = \begin{cases} W_{\text{low}}, & \text{if } E_{\text{edge}} \geq E_a, \\ -E_{\text{edge}}, & \text{otherwise,} \end{cases} \quad (5)$$

where E_a represents the average energy consumption for the AUV to sense edges in this action. When $E_{\text{edge}} \geq E_a$, it represents that the energy consumption for sensing a medium-weight edge is high, and the edge is regarded as a low-weight edge. Otherwise, it represents the energy required to sense the edge is low, only a minor penalty is applied.

The high weights are marked in red in Figure 3b as Equation 6. The edges traversed by the planned path are assigned high weight. RECS forbids the AUV movement until it senses a high-weight edge. When the AUV reaches the high-weight edge, it is regarded as a low-weight edge. In an environment without obstacles, the AUV reaches the target point by sensing only the high-value edges. Therefore, the AUV will receive a reward for sensing high-weight edges.

$$W_{\text{high}} = \left(\sum_{i=1}^n E_{d,i} - E_{\text{edge}} \right) \times (5 - N_{\text{edges}}), \quad (6)$$

where $\sum_{i=1}^n E_{d,i}$ represents the energy required to sense an edge when the sensing radius is at its maximum; $5 - N_{\text{edges}}$ represents that the fewer edges sensed during this action, RECS will get the greater reward.

Deep Reinforcement Learning

We utilize DQN as our core algorithm, which necessitates defining the state space, action space, and reward feedback. The state space is $S = \{s_e; s_o; s_p; s_c; s_t\}$ as shown as Figure 2 State S. State values within the state space will change after an action is executed. The components of the state space are defined as follows:

- s_e : The AUV sensing system energy consumption.
- s_o : Obstacles are sensed within the area.
- s_p : Four points in the area where the AUV is in.
- s_c : The AUV current position.
- s_t : The target position.

The action space is defined by the four edges of the area where the AUV is located: $A = \{a_u; a_l; a_d; a_r\}$. a_u represents the upper edge, a_l represents the left edge, a_d represents the lower edge, and a_r represents the right edge of the area. It does not mean that each action senses only one edge at a time. By combining all actions, there are a total of 14 possible actions. When an edge to be sensed is selected, the sensing direction is automatically configured based on the relative positions of the AUV and the edge. The sensing direction radius is set according to the position of the edge to be sensed and the edge is fully covered by the sensing area, as shown in Figure 2 Action A. After the edges are sensed, any straight-line path from the AUV position to the edge will be covered. It results in two possible scenarios: one where no obstacle is sensed and one where an obstacle is sensed.

When no obstacle is sensed, the AUV moves along the planned path until it reaches the sensed edge. The new position of the AUV and the sensing radius for each sensing

direction are input into the environment, which updates the state space. When an obstacle is sensed, the planned path may collide with the obstacle, and it is essential to re-plan the path to avoid obstacles. However, re-planning the path means RECS returns to the assigning weights stage due to the AUV new path altering the traversal edges.

We set rewards as positive and negative feedback to help train the RECS more effectively. Positive feedback includes the AUV reaching the target point, sensing the high-weight edge, sensing obstacles, and remaining energy when the AUV reaches the target point. Negative feedback includes the AUV sensing the low-weight edge and medium-weight edge, remaining energy less than zero, colliding with obstacles, and the energy consumed during each sensing.

Avoiding Obstacles

When an obstacle is sensed and it blocks the planned path, the AUV can not move along the planned path. In this case, RECS re-plans the path and reassigns edge weights, as shown in Figure 3c. Without designating impassable areas, the APF method is employed to avoid obstacles in the sensed obstacles area. We employ the attractive field and the repulsive field in APF to reach the target point and avoid obstacles, as shown in Equation 7 and Equation 8.

$$F_{\text{att}}(q) = -k_{\text{att}}(q - q_{\text{goal}}), \quad (7)$$

$$F_{\text{rep}}(q) = \begin{cases} k_{\text{rep}} \left(\frac{1}{\|\Delta q\|} - \frac{1}{d_0} \right) \frac{1}{\|\Delta q\|^2} \Delta q, & \text{if } \|\Delta q\| \leq d_0, \\ 0, & \text{if } \|\Delta q\| > d_0, \end{cases} \quad (8)$$

where $F_{\text{att}}(q)$ and $F_{\text{rep}}(q)$ represent the attractive and repulsive forces on position q ; $-k_{\text{att}}$ and k_{rep} represent the attractive and repulsive force constants; $\frac{1}{d_0}$ represents the distance threshold for the influence range of the repulsive force; Δq represents the gradient of the distance from the sensed obstacle to the AUV; However, APF may fall into the local minimum problem when the attractive and repulsive forces are equal in special cases. The local minimum problem is sometimes inevitable when an object moves in unknown environments because the object can not predict local minima before it detects obstacles forming the local minima (Park and Lee,2003). We resolved the local minimum problem by introducing random perturbations to both the attractive and repulsive forces in the APF.

Experiment

Experiment Settings

Simulations in a virtual environment inherently differ from the real-world environment. RECS primarily focuses on reducing energy consumption rather than path planning and AUV trajectory movement. Therefore, a series of constraints are introduced to address these limitations.

- All obstacles are regarded as static objects, such as reefs or other immovable objects.
- The AUV is treated as a point, with a collision only considered if it overlaps with an obstacle.

- The AUV movement speed is kept constant, ensuring that acceleration or deceleration does not cause the AUV to deviate from the planned path.
- The AUV sensing system energy can exceed the initially allocated energy.

The training environment features a map with dimensions of 100x100 units, with four different-sized obstacles randomly placed within the map. We randomly selected 64 different start and target point pairs on this map as the start and target locations for the AUV during training. A vectorized environment configuration was employed to improve the training process efficiency. The episode will terminate if any of the following conditions are met: (1) the AUV collides with obstacles; (2) the AUV reaches the target point. To facilitate calculations and visualization in the training environment, we set the maximum sensing radius r to 7.07, enabling RECS to divide the environment using a factor of 5. At this point, the side length of the square region is 5. The configuration can be proportionally scaled up or down in different environments or different AUV configuration without affecting the performance of RECS. We set the E_f at the maximum sensing radius r to 8. The energy consumption is a linear function of the sensing radius, with E_d ranging from 0 to 1. Since sensing is conducted edge-by-edge, this setting allows for precise determination of energy consumption for each action. The step size for each AUV movement is set to 0.5, facilitating precise calculation of the AUV travel distance and enhancing the accuracy of result visualization.

The training was conducted on an Ubuntu 22.04 machine with 128 GB of RAM and an Nvidia 4090Ti GPU.

The training process starts with the initialization of neural network parameters, specifically the weights, which serve as the foundation for learning during subsequent iterations. The batch size is set to 64 and the learning rate of 0.001. The epsilon value is initialized at 0.99, with an epsilon decay rate of 0.995, and a minimum epsilon value of 0.1. The Adam optimizer is utilized to update the weight parameters throughout the training process.

Prioritized experience replay is employed to disrupt the correlation in the training data, with a buffer size of 10,000 and a minimum size of 1,000. Prioritized experience replay samples experiences based on their importance (Schaul et al.,2016). The importance of an experience is typically determined by its Temporal Difference (TD) error, which quantifies the discrepancy between the predicted Q-value and the target Q-value. A larger TD error represents that the experience contributes to the network learning, and makes it more probable to be sampled during replay.

In each episode, RECS collects tuples (s,a,r,s',d) and stores them in its prioritized experience replay buffer. s and s' represent the AUV state information before and after executing an action, a is the action, r is the reward, and d represents the done or termination condition. To process subsequent data and ensure that RECS can be effectively applied to various scenarios and AUV configurations, the data in these tuples is normalized (Ioffe and Szegedy,2015). All data values are greater than -1 and less than 1 after normalization. During the training phase, soft updates are

Size	Distance		Energy		SD	SE	ER
	Full	RECS	Full	RECS			
100*100	49.6	50.1	58.8	25.2	2.3	1.2	57.1%
100*100	64.6	64.9	79.2	34.0	2.3	1.3	57.0%
100*100	78.5	79.0	93.2	45.8	2.4	1.4	50.8%
200*200	100.7	101.1	118.8	57.4	2.5	1.5	51.9%
200*200	129.8	130.1	150.8	73.7	2.5	1.4	51.1%
200*200	148.2	148.7	172.4	80.9	2.5	1.4	53.0%

Table 1: The comparative results show the differences in travel distance and energy consumption between RECS and full sensors on across various environments.

employed to gradually align the parameters of the target network with the main network. This method guarantees smoother changes in the target network, thereby avoiding abrupt fluctuations in the target values (Zhao et al.,2024).

Performance Comparison

To validate the performance of RECS and guarantee that RECS can generalize well in various environments, we utilized two different map sizes and conducted performance evaluations across 64 points using three different mission execution distances on each map. Specific performance evaluation metrics are employed to measure the performance of RECS. These metrics include the reduced energy consumption rate (ER) and the average number of sensed edges per step (SE). ER refers to the percentage of reducing energy reduction when utilizing RECS compared to full sensors on and directly measures RECS ability to reduce energy consumption relative to the full sensors on. SE represents the average number of sensed edges per step before the AUV reaches the target point and indirectly validates that the AUV can successfully reach the target point by sensing only the high-weight edges.

The comparison results of energy consumption between full sensors on and RECS are presented in Table 1. Full represents the full sensors on compared to RECS; SD represents the average number of sensing directions that are activated during each action; distance represents the movement distance of the AUV from the start point to the target point; energy represents the total energy consumption when the AUV reaches the target point. In Table 1, the energy consumption of RECS is lower than full sensors on in all environments. The minimum ER is 50.8%, with an average ER of 53.4%, indicating that RECS effectively reduces the energy consumption of the AUV sensing system. It helps extend AUV operational duration for missions. It is important to note that, since the start and target points are randomly selected, ER will exhibit slight variations due to the number of obstacles encountered by the AUV will vary. Consequently, differences in ER across different environments are expected and considered normal. In different map sizes and distances, the SE is consistently less than 2. It indicates that RECS successfully implements the assignment weights strategy, where the AUV primarily senses high-weight edges

Radius	Distance		Energy				ER
	Full	RECS	Full	RECS	SD	SE	
7.07	78.3	79.8	91.8	44.8	2.4	1.3	51.2%
14.1	78.7	79.5	95.2	44.5	2.5	1.3	53.2%

Table 2: Comparing the impact of different sensing radii on RECS.

and selectively senses medium-value edges without sensing low-value edges. Additionally, an SD of 2.4 confirms that each sensing cycle utilizes a minimal number of sensing directions, further contributing to the reduction of the AUV energy consumption. With the movement distance increasing, RECS achieves greater energy savings compared to full sensors on, indicating that RECS offers a significant advantage in long-distance missions. With the map size changing, ER does not exhibit significant variation, indicating that RECS possesses strong robustness and maintains its performance regardless of changes in map size. Although the distance to the target point increases slightly compared to the full sensors on, the increase is less than 1% compared to all distances. It indicates that the impact on the path is minimal and does not significantly increase the AUV travel cost. Overall, RECS successfully reduces the energy consumption of the sensing system without significantly increasing other system costs for the AUV.

Analysis on Transferability

To ensure that RECS performs effectively under different exploration radii, we tested 64 points with varying radii on a 200x200 map. The ER and SE values for these tests are presented in Table 2. The tested radii were 7.07 and 14.14, and the sensing system corresponding energy consumption of 8 and 16. It can be seen that both ER and SE show no significant changes compared to Table 1, indicating that RECS remains stable and effective under varying sensing radii. Although an increase in sensing radius leads to higher energy requirements for each sensing operation, the environment partitioning we use results in a corresponding increase in the size of the areas, leading to a relative reduction in the number of sensing operations required. The generalization capability of RECS across different sensing radii of the AUV sensing system is presented in Table 2.

Visualization

To compare the RECS sensing effectiveness with full sensors on, we placed four obstacles on a 100 × 100 map and visualized the path and sensing directions of the AUV with a sensing radius of 7.07, moving from the start point [53,21] to the target point [76,75] as Figure 4a and Figure 4b, and moving from the start point [26,54] to the target point [78,25] as Figure 4c and Figure 4d. The green point is the start point and the red point is the target point. The blue areas are the sensed areas. The red line on the obstacles represents the edges sensed by the sensing system. RECS performed selective edge sensing during its operation, with each action

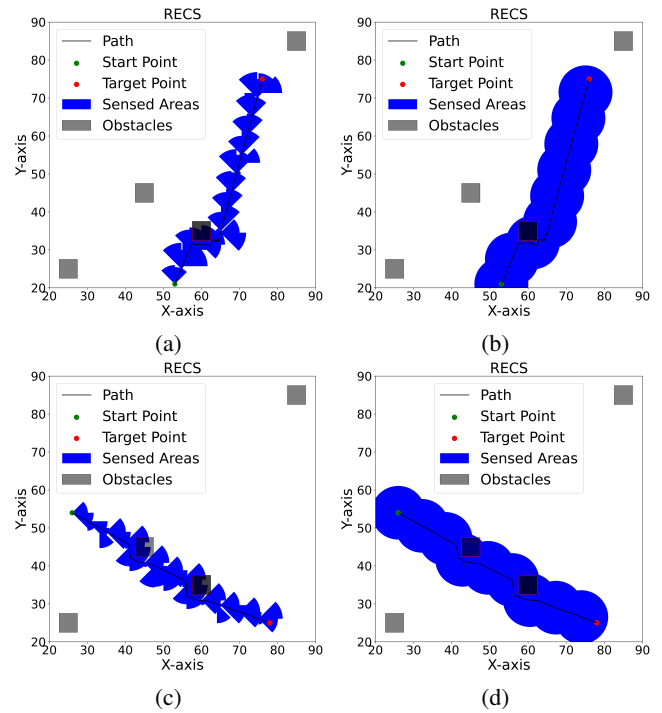


Figure 4: Visualization of movement paths and sensing areas in the simulated environment with RECS and full sensors on.

typically sensing only one or two edges. When sensed obstacles, the AUV successfully re-plans the path, avoids the obstacles, and successfully reaches the target point. Compared to full sensors on, RECS not only reduced energy consumption but also safely avoided obstacles RECS movement path is nearly identical to full sensors on.

The experimental results demonstrate that RECS can generalize effectively across various environments and various AUV configurations. RECS can adapt efficiently to various environments and mission distances presented in Table 1. RECS maintains similar performance across different AUV sensing radii presented in Table 2. These findings validate that RECS can effectively reduce the energy consumption of AUV sensing systems and can be successfully applied to various environments or AUV configurations.

Conclusion

This paper discusses a method-based deep reinforcement learning to address the challenge of reducing energy consumption in AUV sensing systems. It emphasizes how to divide the entire sensing system into different sensing directions and how the AUV selects the directions to sense. To address the challenge, we propose the RECS method, which divides the environment into different areas and assigns weights to each edge of areas. RECS selects the edges to be sensed based on their weights, and re-plans the path, and reassigns weights if obstacles are sensed. RECS uses DQN to select the edges to sense. In future work, we plan to set the simulated environment to a 3D space to ensure more realistic simulations and include movable obstacles.

Acknowledgments

This work is supported by the Natural Science Foundation of China No. 62472196 and No. 62101211, Jilin Science and Technology Research Project 20230101067JC and the National Key Research and Development Program of China No.2021YFC2803000.

References

- Chang, J.; Anderson, M.; Merrifield, S.; Nager, A.; Hess, R.; Young, R.; Kitchen, S.; and Terrill, E. 2022. Power Efficiency Autonomy for Long Duration AUV Operation. In *2022 IEEE/OES Autonomous Underwater Vehicles Symposium (AUV)*, 1–6. IEEE.
- Chi, D.; Tao, J.; Hu, Y.; Wang, H.; Wang, Z.; and Xu, Y. 2024. AUV-assisted information collection scheme with energy balance and low delay of underwater things. *Wireless Networks*, 30(1): 267–283.
- Choudhary, M.; Goyal, N.; Gupta, D.; Sharma, B.; and Sharma, N. 2024. An oceanographic data collection scheme using hybrid optimization for leakage detection during oil mining in mobility assisted UWSN. *Multimedia Tools and Applications*, 1–19.
- Chu, Z.; Wang, F.; Lei, T.; and Luo, C. 2022. Path planning based on deep reinforcement learning for autonomous underwater vehicles under ocean current disturbance. *IEEE Transactions on Intelligent Vehicles*, 8(1): 108–120.
- Danielis, P.; Parzyjegla, H.; Ali, M. A. M.; and Torres, F. S. 2022. Simulation model for energy consumption and acoustic underwater communication of autonomous underwater vehicles. *WMU Journal of Maritime Affairs*, 21(1): 89–107.
- Han, G.; Lai, W.; Wang, H.; and Zhu, S. 2023. Hybrid Algorithm-Based Full Coverage Search Approach With Multiple AUVs to Unknown Environments in Internet of Underwater Things. *IEEE Internet of Things Journal*.
- Hou, X.; Wang, J.; Bai, T.; Deng, Y.; Ren, Y.; and Hanzo, L. 2022. Environment-aware AUV trajectory design and resource management for multi-tier underwater computing. *IEEE Journal on Selected Areas in Communications*, 41(2): 474–490.
- Ioffe, S.; and Szegedy, C. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning*, 448–456. pmlr.
- Jiang, Y.; Xu, Y.; Yang, Y.; Yang, F.; Wang, P.; Li, C.; Zhuang, F.; and Xiong, H. 2024a. TriMLP: A Foundational MLP-Like Architecture for Sequential Recommendation. *ACM Trans. Inf. Syst.*, 42(6).
- Jiang, Y.; Yang, Y.; Xu, Y.; and Wang, E. 2024b. Spatial-Temporal Interval Aware Individual Future Trajectory Prediction. *IEEE Transactions on Knowledge and Data Engineering*, 36(10): 5374–5387.
- Khatib, O. 1986. Real-time obstacle avoidance for manipulators and mobile robots. *The international journal of robotics research*, 5(1): 90–98.
- Lin, C.; Han, G.; Zhang, T.; Shah, S. B. H.; and Peng, Y. 2022. Smart underwater pollution detection based on graph-based multi-agent reinforcement learning towards AUV-based network ITS. *IEEE Transactions on Intelligent Transportation Systems*, 24(7): 7494–7505.
- Mahmoodi, K. A.; and Uysal, M. 2022. Energy aware trajectory optimization of solar powered AUVs for optical underwater sensor networks. *IEEE Transactions on Communications*, 70(12): 8258–8269.
- Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; and Riedmiller, M. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*.
- Pang, W.; Zhu, D.; and Sun, C. 2023. Multi-AUV formation reconfiguration obstacle avoidance algorithm based on affine transformation and improved artificial potential field under ocean currents disturbance. *IEEE Transactions on Automation Science and Engineering*.
- Park, M. G.; and Lee, M. C. 2003. A new technique to escape local minimum in artificial potential field based path planning. *KSME international journal*, 17: 1876–1885.
- Schaul, T.; Quan, J.; Antonoglou, I.; and Silver, D. 2016. Prioritized Experience Replay. In Bengio, Y.; and LeCun, Y., eds., *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*.
- Sun, S.; Guo, H.; Wan, G.; Dong, C.; Zheng, C.; and Wang, Y. 2023. High-precision underwater acoustic localization of the black box utilizing an autonomous underwater vehicle based on the improved artificial potential field. *IEEE Transactions on Geoscience and Remote Sensing*, 61: 1–10.
- Vial, P.; Malagón, M.; Segura, R.; Palomeras, N.; and Carreras, M. 2023. GMM Registration: a Probabilistic scan matching approach for sonar-based AUV navigation. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 1033–1039. IEEE.
- Wang, E.; Xu, Y.; Yang, Y.; Jiang, Y.; Yang, F.; and Wu, J. 2023a. Zone-Enhanced Spatio-Temporal Representation Learning for Urban POI Recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 35(9): 9628–9641.
- Wang, J.; Liu, S.; Shi, W.; Han, G.; and Yan, S. 2023b. A multi-AUV collaborative ocean data collection method based on LG-DQN and data value. *IEEE Internet of Things Journal*.
- Watkins, C. J.; and Dayan, P. 1992. Q-learning. *Machine learning*, 8: 279–292.
- Xi, M.; Yang, J.; Wen, J.; Liu, H.; Li, Y.; and Song, H. H. 2022. Comprehensive ocean information-enabled AUV path planning via reinforcement learning. *IEEE Internet of Things Journal*, 9(18): 17440–17451.
- Xu, Y.; Wang, E.; Yang, Y.; and Chang, Y. 2022a. A Unified Collaborative Representation Learning for Neural-Network Based Recommender Systems. *IEEE Transactions on Knowledge and Data Engineering*, 34(11): 5126–5139.

Xu, Y.; Wang, E.; Yang, Y.; and Xiong, H. 2024a. GS-RS: A Generative Approach for Alleviating Cold Start and Filter Bubbles in Recommender Systems. *IEEE Transactions on Knowledge and Data Engineering*, 36(2): 668–681.

Xu, Y.; Yang, Y.; Wang, E.; Zhuang, F.; and Xiong, H. 2022b. Detect Professional Malicious User With Metric Learning in Recommender Systems. *IEEE Transactions on Knowledge and Data Engineering*, 34(9): 4133–4146.

Xu, Y.; Zhuang, F.; Wang, E.; Li, C.; and Wu, J. 2024b. Learning without Missing-At-Random Prior Propensity-A Generative Approach for Recommender Systems. *IEEE Transactions on Knowledge and Data Engineering*, 1–13.

Xue, K.; Liu, J.; Xiao, N.; Ji, X.; and Qian, H. 2022. A bio-inspired simultaneous surface and underwater risk assessment method based on stereo vision for USVs in nearshore clean waters. *IEEE Robotics and Automation Letters*, 8(1): 360–367.

Yan, J.; Peng, S.; Yang, X.; Luo, X.; and Guan, X. 2023a. Containment Control of Autonomous Underwater Vehicles With Stochastic Environment Disturbances. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 1–12.

Yan, J.; Zhang, L.; Yang, X.; Chen, C.; and Guan, X. 2023b. Communication-Aware Motion Planning of AUV in Obstacle-Dense Environment: A Binocular Vision-Based Deep Learning Method. *IEEE Transactions on Intelligent Transportation Systems*.

Yang, Y.; Zhang, C.; Song, X.; Dong, Z.; Zhu, H.; and Li, W. 2024. Contextualized Knowledge Graph Embedding for Explainable Talent Training Course Recommendation. *ACM Trans. Inf. Syst.*, 42(2): 33:1–33:27.

Yu, Y.; Zheng, H.; and Xu, W. 2024. Learning and Sampling-Based Informative Path Planning for AUVs in Ocean Current Fields. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.

Zhang, M.; Cai, W.; Xie, Q.; and Xu, S. 2022. Binocular-vision-based obstacle avoidance design and experiments verification for underwater quadrocopter vehicle. *Journal of Marine Science and Engineering*, 10(8): 1050.

Zhao, Z. Y.; Che, Y. L.; Luo, S.; Luo, G.; Wu, K.; and Leung, V. C. 2024. On Designing Multi-UAV aided Wireless Powered Dynamic Communication via Hierarchical Deep Reinforcement Learning. *IEEE Transactions on Mobile Computing*.