Journal of
*Marine Science
and Engineering*

MDPI

Article

A Method for Multi-AUV Cooperative Area Search in Unknown Environment Based on Reinforcement Learning

Yueming Li *, Mingquan Ma, Jian Cao, Guobin Luo, Depeng Wang and Weiqiang Chen

National Key Laboratory of Autonomous Marine Vehicle Technology, Harbin Engineering University, Harbin 150001, China; mingquan@hrbeu.edu.cn (M.M.); caojian@hrbeu.edu.cn (J.C.); luoguobin@hrbeu.edu.cn (G.L.)
* Correspondence: liyueming@hrbeu.edu.cn

Abstract: As an emerging direction of multi-agent collaborative control technology, multiple autonomous underwater vehicle (multi-AUV) cooperative area search technology has played an important role in civilian fields such as marine resource exploration and development, marine rescue, and marine scientific expeditions, as well as in military fields such as mine countermeasures and military underwater reconnaissance. At present, as we continue to explore the ocean, the environment in which AUVs perform search tasks is mostly unknown, with many uncertainties such as obstacles, which places high demands on the autonomous decision-making capabilities of AUVs. Moreover, considering the limited detection capability of a single AUV in underwater environments, while the area searched by the AUV is constantly expanding, a single AUV cannot obtain global state information in real time and can only make behavioral decisions based on local observation information, which adversely affects the coordination between AUVs and the search efficiency of multi-AUV systems. Therefore, in order to face increasingly challenging search tasks, we adopt multi-agent reinforcement learning (MARL) to study the problem of multi-AUV cooperative area search from the perspective of improving autonomous decision-making capabilities and collaboration between AUVs. First, we modeled the search task as a decentralized partial observation Markov decision process (Dec-POMDP) and established a search information map. Each AUV updates the information map based on sonar detection information and information fusion between AUVs, and makes real-time decisions based on this to better address the problem of insufficient observation information caused by the weak perception ability of AUVs in underwater environments. Secondly, we established a multi-AUV cooperative area search system (MACASS), which employs a search strategy based on multi-agent reinforcement learning. The system combines various AUVs into a unified entity using a distributed control approach. During the execution of search tasks, each AUV can make action decisions based on sonar detection information and information exchange among AUVs in the system, utilizing the MARL-based search strategy. As a result, AUVs possess enhanced autonomy in decision-making, enabling them to better handle challenges such as limited detection capabilities and insufficient observational information.

Keywords: cooperative area search; multi-agent reinforcement learning; multi-AUVs

Citation: Li, Y.; Ma, M.; Cao, J.; Luo, G.; Wang, D.; Chen, W. A Method for Multi-AUV Cooperative Area Search in Unknown Environment Based on Reinforcement Learning. *J. Mar. Sci. Eng.* **2024**, *12*, 1194. https://doi.org/10.3390/jmse12071194

Academic Editor: Leszek Chybowski

Received: 2 April 2024
Revised: 12 July 2024
Accepted: 12 July 2024
Published: 16 July 2024

1. Introduction

In areas such as maritime search and rescue, intelligence reconnaissance, submarine resource exploration, and marine observation, AUVs [1] have demonstrated significant advantages with their autonomy and excellent endurance. However, as we continue to explore and develop the ocean, the area where AUVs perform search missions is also expanding, and the environment they face is becoming increasingly complex. A single AUV with limited endurance and detection capabilities is unable to cope. Therefore, when facing complex tasks, we generally consider multiple AUVs to collaboratively work [2]. The basic idea of multi-AUV cooperative control is to have multiple AUVs with relatively simple structures and functionalities work together as a team. This enhances the overall operational range and detection capabilities of the multi-AUV system. Within the multi-AUV system, each AUV can make behavioral decisions based on its own capabilities and sensor data, collaborating with other AUVs to accomplish the final mission. Additionally, AUVs operate in a marine environment with many unknowns and uncertainties. Moreover, underwater AUVs have limited sensing and communication capabilities, and these challenging conditions during operations place high demands on the AUVs' autonomous decision-making abilities. Therefore, researching methods for multi-AUV cooperative control to enhance the intelligence, autonomy [3], and coordination among AUVs is essential. This enables AUVs to have improved decision-making capabilities and work collaboratively with other AUVs to accomplish complex tasks. This technological direction is becoming an inevitable trend in the development of multi-AUV cooperative underwater search [4].

For regional search tasks, traditional methods often employ a comb-like scanning approach [5–7]. In this method, AUVs follow pre-planned trajectories for step-by-step scanning. However, under this approach, AUVs lack autonomous decision-making capabilities. In complex and unknown marine environments, AUVs relying solely on pre-planned trajectories may struggle to handle unexpected situations such as encountering obstacles, making it challenging to effectively complete search tasks. Furthermore, for military area search and reconnaissance missions, traditional methods involve AUV search trajectories that exhibit strong regularity. This lack of randomness can make it easier for enemy forces to detect and potentially target our AUVs, which is not ideal for ensuring the stealth and security of our operations. When dealing with large-scale regional search tasks, the traditional approach involves dividing the area into several subregions [8–10] and then assigning these subregions to individual AUVs for scanning using a comb-like pattern. Therefore, before initiating AUV searches, it is necessary to pre-plan the search area, which includes area partitioning, subregion allocation, and setting AUV comb-like scanning trajectories. For different search areas, planning often needs to be adjusted, and irregular search areas can pose challenges in terms of area division and AUV comb-like scanning. Additionally, due to the difficulties associated with underwater communication [11], it is challenging for AUVs to engage in frequent and high-bandwidth information exchange. Traditional approaches often require access to information such as the positions of each AUV and the completion status of subregion searches to facilitate real-time subregion allocation. This places high demands on communication between AUVs. If AUV communication is not smooth, it can have adverse effects on the collaborative regional search tasks of the AUV cluster [12,13]. Indeed, addressing challenges posed by weak communication environments and limited local observability is crucial for improving the performance of multi-AUV systems in cooperative regional search and exploring unknown marine environments. Research in the field of multi-AUV cooperative search technology is currently focused on several key areas:

1. Intelligent algorithms: Developing intelligent algorithms that enable AUVs to make autonomous decisions in complex and uncertain underwater environments. These algorithms should allow AUVs to adapt their search strategies dynamically based on their own sensor data and available information;
2. Adaptive search strategies: Designing cooperative search strategies that can adapt to different types of search areas. AUVs should be able to perform effective searches in irregular or unpredictable environments without the need for extensive pre-planning;
3. Robustness: Enhancing the robustness of multi-AUV systems to ensure they can continue operating effectively even in the presence of individual AUV failures or communication disruptions. Redundancy and fault-tolerant control mechanisms are important in this context;
4. Collaboration of multi-AUVs in the weak communication environments: Developing communication-efficient strategies that allow AUVs to collaborate effectively with minimal information exchange. Decentralized decision-making and coordination mechanisms can be beneficial in scenarios with limited bandwidth;

5. Exploration of unknown environments: Investigating methods for AUVs to explore and map unknown or uncharted marine environments. This involves combining exploration and mapping capabilities with cooperative search strategies;
6. Optimization and planning: Researching optimization techniques and path planning algorithms that help AUVs make informed decisions about their search trajectories, considering factors such as energy efficiency, time constraints, and mission objectives.

Overall, these research areas aim to advance the capabilities of multi-AUV systems and enable them to perform efficient and adaptive cooperative search missions in challenging underwater environments. We focus on researching methods for multi-AUV cooperative area search, starting from the perspectives of adaptive search strategies and collaboration under weak communication environments. Firstly, this paper establishes a first-order kinematic and a sonar detection model for AUVs. Using Dec-POMDP, we model the cooperative area search task. Secondly, we apply the multi-agent reinforcement learning QMIX method [14] to solve the Dec-POMDP mentioned above. Building upon prior work [15], we specifically design the QMIX method based on the maximum entropy mechanism and conduct comparative analysis of the training results. Finally, we perform simulation tests and conduct a preliminary exploration of reward function settings.

2. Background

Multi-AUV cooperative regional search enables efficient coverage and detection tasks in various areas, making it a hot topic of research in both academia and industry. Some multi-AUV cooperative search systems have already found practical applications in engineering projects. One such example is the research project called Generic Ocean Array Technology System (GOATS) conducted by the Massachusetts Institute of Technology (MIT). The GOATS project utilizes multiple AUVs equipped with underwater acoustic devices to form a mobile underwater detection network for searching for underwater mines in coastal waters [16–18]. Building on the GOATS project, a research team composed of the National Underwater Research Center (NURC) and MIT initiated a project in 2008 known as the Generic Littoral Interoperable Networked Technology (GLINT). In the GLINT project, a multi-AUV system is equipped with various sensors to autonomously detect, locate, and track specific targets [19]. In Europe, from 2012 to 2015, various research institutions from Italy, Estonia, the United Kingdom, Spain, and Turkey collaborated on a research project called Autonomous Robotic Systems for Oceanographic and Water-Archaeological Surveys (ARROWS) [20]. The project aimed to enhance underwater scanning efficiency using multi-AUV systems and conducted research on task allocation strategies [21] and underwater communication [22].

In the academic field, there have been many successful research efforts related to multi-AUV collaborative area search. Li et al. [23] proposed a distributed dynamic predictive control (DDPC) algorithm based on predictive control principles. This algorithm predicts the states of a multi-AUV system to obtain information about the task environment and updates the AUV states as inputs for online task optimization decisions, allowing them to determine the next moment's search area. Wang et al. [24] presented research on a multi-agent target search method for AUVs. This algorithm is based on multi-agent deep deterministic policy gradient (MADDPG) and incorporates temporal and spatial information into the reinforcement learning process. It also introduces specific rewards tailored for maritime target search scenarios. However, it does not consider the impact of sonar detection effectiveness, leading to a lack of target guidance information during the search process and subsequently affecting the efficiency of target search by AUVs. Liu et al. [25] defined the target search problem as a well-known Traveling Salesman Problem (TSP) with defined start and end points. They considered two competitive and non-communicable optimization objectives for underwater vehicles: total navigation distance and turning angles. The study also introduced mobility constraints for AUVs and utilized an improved ant colony algorithm for solving the problem. However, this research treated the search area as a known region and did not account for scenarios involving unknown search areas.

Cai et al. [26] focused on the maritime search and rescue mission of multi-AUV systems and studied the multi-robot coverage path planning (MCPP) problem. They introduced a new MCPP approach, which involved transforming the MCPP problem into two sub-problems: area partition and single AUV coverage path planning, solving each individually. Although this approach considered the guidance role of prior target information, it simply divided the search area and planned coverage paths for individual AUVs. It did not account for the collaboration between AUVs. Hu et al. [27] addressed the obstacle avoidance issue encountered by underwater vehicle formations during collaborative search and target capture. They introduced an energy-optimal formation obstacle avoidance strategy and an improved self-organizing map (SOM) path planning algorithm. However, this approach relied on forming a fleet of AUVs for search, which required frequent communication between the AUVs and placed relatively high demands on underwater communication capabilities. Bai et al. [28] introduced a biologically-inspired two-layer self-organizing map algorithm for dynamic task planning involving multiple autonomous underwater vehicles. This algorithm is developed for searching multiple targets in a 3D underwater environment affected by random ocean currents and dynamic uncertain obstacles. It focuses on addressing the decoupling of task assignment and path planning in initial task planning while considering energy consumption constraints. Li et al. [29] proposed a combined approach for autonomous underwater vehicles and unmanned surface vehicles (USVs) in target search tasks in unknown marine environments with no prior information. This approach involves a local dynamic predictive control framework combined with the Lévy flight (LDPC-Lévy). The LDPC-Lévy method involves assigning subregions for AUVs to search and planning suitable positions for USVs, ensuring reasonable communication distances while exploring the environment and searching for targets.

3. Method Review

Currently, multi-AUV cooperative area search technology is still in the development stage. Many of the methods in this field draw inspiration from the more mature field of unmanned aerial vehicle (UAV) swarm search [30–32]. These methods can be categorized based on their search approach into formation-based search [5–7], partition allocation-based search [8–10], and swarm intelligence-based autonomous search [33–35].

Formation-based search involves coordinating multiple AUVs to form a configuration that maximizes the search area, and then conducting a step-by-step search in this formation [36–38]. This approach draws inspiration from the single AUV comb-like search method but considers multiple AUVs as a unified formation [39] to plan the search coverage path for the entire formation. Formation-based search is simple and practical, allowing for comprehensive coverage without blind spots. However, it can face challenges in complex marine environments with factors like weak communication. Maintaining formation in such conditions can be difficult. Additionally, due to the constraints of the formation, AUVs may struggle to handle unexpected situations such as obstacle avoidance. As a result, the overall autonomy of AUV formations may be limited, and the level of coordination between individual AUVs may not be very strong. Hu et al. [27] address the energy consumption challenges encountered by AUV formations during collaborative search processes. They propose an energy cost-optimal formation search strategy and an improved self-organizing map path planning algorithm. Furthermore, they introduce a Formation Comprehensive Cost (FCC) model, which considers convergence time, transformation distance, and sensor network power consumption, to facilitate obstacle avoidance within the formation. Healey et al. [7] designed a multi-underwater robots system to address shallow water mine clearance issues. In this system, a supervisor robot located outside the minefield centrally controls all swimmer robots, ensuring a high mine clearance rate. However, the most significant drawback of this system is that if the supervising robot is damaged or disabled, all other robots will be left in an uncontrolled state, which can be a critical limitation in terms of system reliability and robustness.

Partition allocation-based search methods involve dividing the search area into several subregions initially and then dynamically allocating these subregions to individual AUVs using allocation algorithms. Once an AUV receives information about its allocated subregion, it proceeds to that subregion to perform coverage and search tasks. This approach allows for a more organized and efficient distribution of search efforts among multiple AUVs. Partition allocation-based search methods offer several advantages. They divide a large, irregularly shaped search area into multiple smaller, relatively regular subregions, simplifying the complexity of the search task. Additionally, AUVs can dynamically allocate subregions through information exchange, which enhances both the autonomy of individual AUVs and the overall coordination of the multi-AUV system. However, it is important to note that this approach also increases the computational load on the AUV system and places higher demands on communication between AUVs due to the need for real-time dynamic allocation of subregions. Welling et al. [8] utilized a multi-AUV system for collaborative search and target clearance tasks. They discussed task allocation problems and compared two allocation strategies based on the nearest distance and fuzzy logic in terms of time efficiency. Hoai An Le et al. [10] researched a hierarchical search planning model that divides the search area into multiple subspaces and then performs a second-level search planning within these subspaces. This two-stage search planning approach enhances the overall efficiency and precision of the search process.

Swarm intelligence-based autonomous search methods draw inspiration from the foraging behavior of biological groups like ants [40], wolf packs [41], etc. In these methods, each AUV makes its current best decision based on observable spatial information. Through iterative steps, the AUVs work collectively to ultimately achieve complete area search coverage. This approach leverages the power of decentralized decision-making and cooperation, similar to how natural swarms of animals collaborate to achieve complex tasks. Indeed, in recent years, the rise of artificial intelligence (AI) algorithms based on machine learning has opened up new research directions for multi-agent cooperative regional search methods. These AI algorithms can enhance the decision-making, adaptability, and coordination of multi-agent systems, making them more effective in complex and dynamic environments. Researchers are increasingly exploring the integration of machine learning techniques into multi-agent systems to improve their performance and autonomy in tasks such as regional search. The multi-agent cooperative search strategies derived through swarm intelligence and machine learning methods offer several advantages. They not only enhance the autonomous decision-making capabilities of AUVs in complex marine environments but also improve the coordination between AUVs. These strategies exhibit good adaptability to different environments, making them versatile and effective tools for a wide range of scenarios. The combination of swarm intelligence and machine learning contributes to more efficient and adaptive multi-agent systems for tasks like cooperative search and exploration. Cao et al.'s work [34] focuses on target search and tracking in unknown underwater environments. They propose an integrated algorithm for a multi-autonomous underwater vehicle collaborative team. This algorithm combines Glasius bio-inspired neural network (GBNN) and bio-inspired cascaded tracking control methods. The aim is to enhance search efficiency and reduce tracking errors. GBNN plays a central role in controlling the multi-AUV team's search for each target in this context. Liu et al.'s work [35] involves the establishment of a distributed multi-AUV cooperative search system (DMACSS) and introduces the autonomous cooperative search learning algorithm (ACSLA) integrated into DMACSS. It incorporates information fusion mechanisms and timestamp mechanisms, enabling each AUV in the system to exchange and fuse information during tasks. ACSLA is a customized learning algorithm trained using reinforcement learning methods.

In summary, the advantages and disadvantages of various search forms are shown in Table 1.

Table 1. Advantages and disadvantages of different search forms.

Collaborative Search Form	Advantages	Disadvantages
Formation-based search	1. Low method complexity, simple, and feasible 2. High area coverage rate 3. Suitable for large-scale, no-blind-zone searches	1. During the search process, the autonomy is low 2. Collaboration among AUVs is not strong 3. Challenging to deal with complex environments
Partitioned allocation search	1. Simplifying the search difficulty in complex areas 2. Good autonomy and inter-group coordination of AUVs	1. Strong dependence on communication 2. There may be delays if the computational load is high
Swarm intelligent autonomous search	1. High autonomy in AUV decision-making 2. Strong adaptability to different environments 3. Weak dependence on communication	1. Prone to getting stuck in local optima

To address the challenges of limited detection capability and insufficient observational information of AUVs during area search tasks, we have developed a multi-AUV cooperative area search system (MACASS). From the perspective of enhancing AUVs' autonomous decision-making capabilities, we integrated well-trained search strategies based on MARL into the action decision modules of each AUV within the system. The main contributions of this article are as follows:

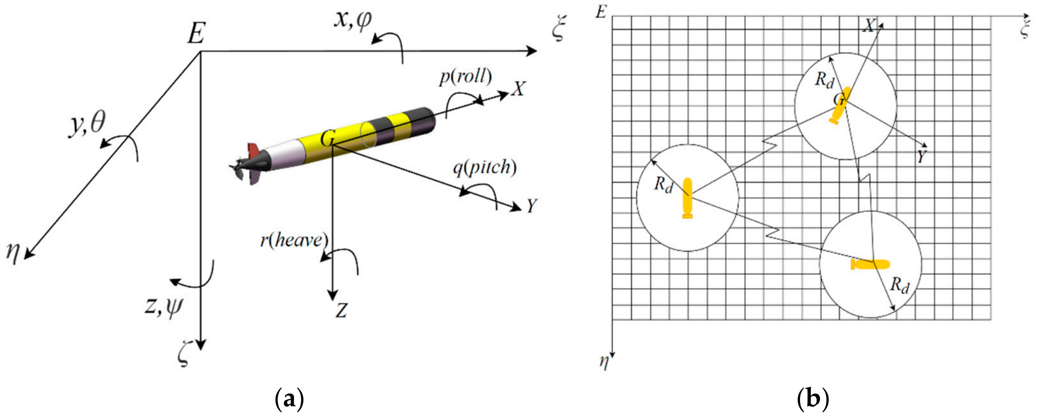
1. Dec-POMDP environmental modeling for search tasks: In this paper, the search task is modeled as a Dec-POMDP, and relevant global states, state transition functions, and reward functions are designed. Additionally, in underwater environments where AUVs have limited sensing capabilities and cannot access global environmental state information, the paper uses sonar-based detection information as the observations available to AUVs. Furthermore, considering the challenges of underwater communication, the multi-AUV system employs a distributed control approach to minimize communication requirements;
2. Cooperative area search algorithm based on MARL: We adopt the QMIX [14] algorithm to train the search strategy for the aforementioned Dec-POMDP and then based on the design approach using the maximum entropy mechanism for QMIX proposed by Guo et al. [15], we specifically design the QMIX algorithm for the conditions of multi-AUV cooperative area search tasks. Since the multi-agent reinforcement learning algorithm based on the maximum entropy mechanism draws inspiration from the soft actor-critic (SAC) algorithm [42], this approach is referred to as SAC-QMIX;
3. Multi-AUV cooperative area search system: After modeling the multi-AUV cooperative area search task as a Dec-POMDP and training search strategy, the well-trained search strategies are integrated into the decision-making models of each AUV. Subsequently, using a distributed control approach, these AUVs form the multi-AUV cooperative area search system. During the execution of the search task, each AUV within MACASS updates the search information map in real time based on sonar detection information and information exchange among AUVs in the system. They make action decisions by the search strategies integrated into the AUVs, ultimately cooperating with each other to complete the search task.

4. Dec-POMDP Modeling

This section begins by establishing the AUV kinematic and sonar detection model. Subsequently, it models the search task as a Dec-POMDP. It specifies AUV observations based on sonar detection information and the search information map. Additionally, it takes into account the AUV's kinematic characteristics to design corresponding actions and reward functions. This modeling approach helps create a framework for decision-making and coordination in multi-AUV cooperative search missions, considering both sensory data and the environmental context. The significance and explanations of the symbols and abbreviations used in modeling the cooperative area search task are summarized in the Appendix A.

4.1. AUV Kinematic Model

In order to provide a detailed description of the AUV's motion, this paper establishes an inertial coordinate system $E\text{-}\xi\eta\zeta$ and a body-fixed coordinate system $G\text{-}XYZ$, as shown in Figure 1a. The origin of the inertial coordinate system E is located at a certain point on the water surface. The $E\xi$ axis points north, the $E\eta$ axis points east, and the $E\zeta$ axis points downward. Due to the relatively low cruising speed of the AUVs performing the search tasks in this paper, the effects of the Earth's rotation can be ignored. Therefore, the coordinate system $E\text{-}\xi\eta\zeta$ is considered an inertial coordinate system. The body-fixed coordinate system, denoted as $G\text{-}XYZ$, is rigidly attached to the AUV itself. In this coordinate system, the origin G is located at the center of gravity of the AUV, the GX axis points along the AUV's longitudinal axis (forward), the GY axis is perpendicular to the GX axis and points starboard (to the right), and the GZ axis is perpendicular to both the GX and GY axes and points downward.



(**a**) (**b**)

Figure 1. (**a**) AUV coordination. (**b**) AUV projection on the horizontal plane.

When multiple AUVs are conducting a search in a region of size $L_x \times L_y \times L_z$, the search area is divided along the $E\zeta$ axis into multiple subregions $S_i, i \in \{1, 2, \cdots, I\}$ where $I = L_z/\delta_d$ and δ_d represents the maximum sonar vertical detection distance. Multiple AUVs search the various subregions sequentially. When each AUV completes the search of a subregion S_i, they immediately descend and move on to the next subregion S_{i+1} for further search. This process continues until the entire region has been completely searched. When multiple AUVs are searching subregion S_i, the variation in depth during AUV navigation is relatively small, and the AUVs can be considered to be operating at a fixed depth. Therefore, the problem of multiple AUVs searching subregion S_i can be transformed into a two-dimensional search problem on a plane s_i of size $L_x \times L_y$, as shown in Figure 1b. Additionally, it is assumed that the depth at which the target is located is the same as the depth of the AUVs performing the search task. In a two-dimensional plane, the motion of AUVs can be simplified to three degrees of freedom: heave, sway, and yaw. The kinematic equations for their motion are provided in the equation below, with further details available in reference [43].

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ r \end{bmatrix} \tag{1}$$

4.2. Sonar Detection Model

When AUVs are performing search tasks, their primary detection equipment is sonar. The detection probability of the sonar depends on the detection distance and the acoustic conditions of the operating environment. In this paper, only the detection distance between the sonar and the target is considered as the primary factor affecting sonar detection probability; the simplified sonar detection probability model [44] is represented by the following Equation (2):

$$P_d = \begin{cases} P_D \cdot \exp\left(-\frac{d}{D_s}\right), 0 < d \leq D_s \\ P_F, \quad d > D_s \end{cases} \tag{2}$$

In the equation, P_d represents the sonar detection probability, P_D stands for the sonar's detection precision, P_F represents the sonar's false alarm probability, d represents the detection distance, and D_s is the maximum detection distance of the sonar.

4.3. Search Information Map Model

From the previous information, it is evident that the problem of multiple AUVs searching subregion S_i can be transformed into a two-dimensional search problem in a plane s_i. Therefore, we discretize the two-dimensional plane s_i which has dimensions $L_x \times L_y$ into an $N_x \times N_y$ grid map. Each grid region is denoted as $grid(m,\ n)$, where $m \in \{1, 2, \cdots, N_x\}, n \in \{1, 2, \cdots, N_y\}$. When AUVs are performing search tasks, they update the gridded search information map using sonar detection information and information exchanged among AUVs. The search information map established in this paper consists of three main components:

1. Coverage map: This map indicates which areas have been covered or searched by the AUVs. Initially, at the start of the search task, all grid regions have a value of 0;
2. Uncertainty map: The uncertainty map represents the level of uncertainty associated with each grid region. Initial values are typically set to the max value;
3. Target presence probability map: This map estimates the probability of the presence of targets in each grid region. It provides information about how confident or uncertain the AUVs are about the presence of targets in each region. Initially, the probability of target existence in the grid region follows a uniform distribution, indicating no prior knowledge of target presence.

As the AUVs continue their search and gather more information through sonar detections and interactions with each other, these maps are updated in real time. The values in these maps evolve to reflect the changing understanding of the search environment, and AUVs use this information to make new decisions during the search process.

4.3.1. Coverage Map

At the beginning of the search task, the presence probability of targets within the sonar detection range of each AUV is generally low. Moreover, due to a lack of prior information about targets, the efficiency of AUV search efforts is often low. In order to encourage AUVs to explore unknown areas and accelerate the convergence speed of strategy learning, a coverage map is established. The coverage map is used to distinguish between grid regions that AUVs have explored and those that have never been explored. When an AUV proceeds to search a previously unexplored grid region $grid(m,\ n)$, the coverage value c_{mn} changes from 0 to 1, and it remains unchanged afterward. The update method for the coverage map is described by the following Equation (3):

$$c_{mn} = \begin{cases} 1, \text{if } \ grid(m,n) \in FOV \\ 0, \text{else} \end{cases} \tag{3}$$

4.3.2. Uncertainty Map

The uncertainty map is used to represent the uncertainty level within the search area. As the AUVs continue their search, the uncertainty about the environment gradually decreases. This map provides a measure of how much time AUVs have not detected those areas that have already been explored. Unlike the coverage map, the uncertainty map operates differently. When an AUV proceeds to search a grid region $grid(m, n)$, the uncertainty value k_{mn} for that grid region is set to its minimum value 0. As the AUV leaves that grid region, the uncertainty value k_{mn} slowly increases over time. The specific update formula is described as shown in Equation (4). The uncertainty map primarily serves the purpose of encouraging AUVs to re-search areas that have not been searched for a long time. It does so by gradually reducing the uncertainty values in regions that have been previously explored, thereby prioritizing unexplored or less recently explored areas for further investigation by the AUVs. This strategy helps ensure comprehensive coverage and exploration of the search area. The coverage map and the uncertainty map together assist AUVs in addressing the issue of low search efficiency during the early stages of the search when there is a lack of target information. These maps help AUVs identify areas that have already been explored (coverage map) and areas where uncertainty remains high (uncertainty map). This information guides AUVs to focus their efforts on unexplored or uncertain regions, improving their search efficiency and increasing the likelihood of finding targets.

$$k_{mn}(t+1) = \begin{cases} 0, & grid(m,n) \in FOV(t) \\ k_{mn}(t) + \frac{1}{T}, & grid(m,n) \notin FOV(t) \end{cases} \tag{4}$$

In the equation, k_{mn} represents the uncertainty value of the grid region $grid(m, n)$, and $FOV(t)$ represents the effective sonar detection range of each AUV at time t.

4.3.3. Target Presence Probability Map

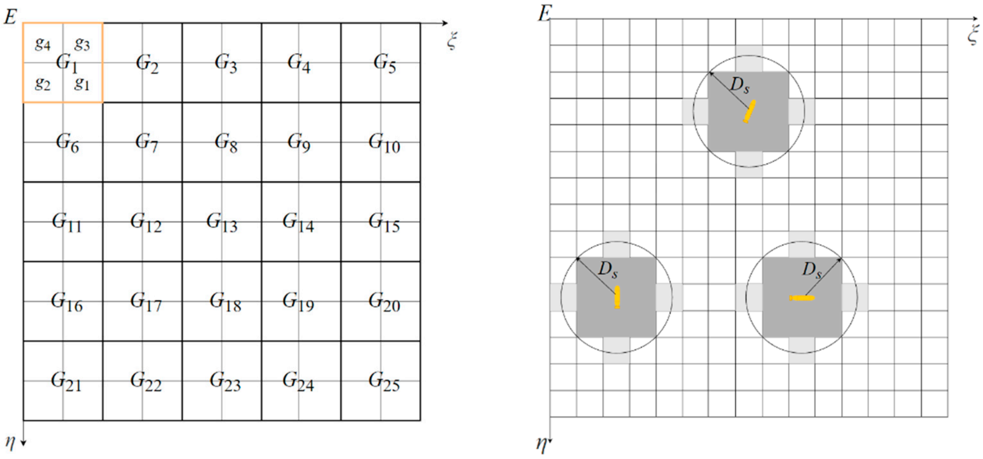
The target presence probability map is used to represent the probability of the presence of a target at different locations within the search area. It helps AUVs navigate toward potential targets more efficiently. Initially, when there is no prior information about the target's location, the probability of the target's presence is typically assumed to be equally distributed across the entire search area. As AUVs gather information and make detections, this map is updated to reflect changes in the likelihood of a target being present in various parts of the search area. This enables AUVs to focus their search efforts on regions with a higher probability of finding a target. The target presence probability is derived using the sonar detection model and the Bayesian formula. Each AUV initially calculates the target presence probability for grid regions within its sonar detection range based on the sonar detection results, as described by the formula. Subsequently, AUVs update their target presence probabilities through information exchange between them. Then, the target presence probabilities for grid regions across the entire search area are normalized to create the final target presence probability map. The target presence probability at the detection location is described by the following equation, with detailed derivations available in reference [45]:

$$P_j(t+1) = \begin{cases} \dfrac{(1-\overline{P}_{D_i}\cdot\overline{P}_{F_i}^{M_i-1})P_j(t)}{\sum_{k\in FOV_i}(1-\overline{P}_{D_i}\cdot\overline{P}_{F_i}^{M_i-1})P_k(t)+\sum_{k\notin FOV_i}(1-\overline{P}_{F_i}^{M_i})P_k(t)}, \text{if } j\in FOV_i, O(t)=1 \\ \dfrac{(1-\overline{P}_{F_i}^{M_i})P_j(t)}{\sum_{k\in FOV_i}(1-\overline{P}_{D_i}\cdot\overline{P}_{F_i}^{M_i-1})P_k(t)+\sum_{k\notin FOV_i}(1-\overline{P}_{F_i}^{M_i})P_k(t)}, \text{if } j\notin FOV_i, O(t)=1 \\ \dfrac{(\overline{P}_{D_i}\cdot\overline{P}_{F_i}^{M_i-1})P_j(t)}{\sum_{k\in FOV_i}(\overline{P}_{D_i}\cdot\overline{P}_{F_i}^{M_i-1})P_k(t)+\sum_{k\notin FOV_i}(\overline{P}_{F_i}^{M_i})P_k(t)}, \text{ if } j\in FOV_i, O(t)=0 \\ \dfrac{(\overline{P}_{F_i}^{M_i})P_j(t)}{\sum_{k\in FOV_i}(\overline{P}_{D_i}\cdot\overline{P}_{F_i}^{M_i-1})P_k(t)+\sum_{k\notin FOV_i}(\overline{P}_{F_i}^{M_i})P_k(t)}, \text{ if } j\notin FOV_i, O(t)=0 \end{cases} \tag{5}$$

In this formula, $P_j(t+1)$ represents the target presence probability in grid region j at time $t+1$, $P_j(t)$ and $P_k(t)$ represent the target presence probabilities in grid regions j and k at time t, P_{Di} represents the sonar detection probability for AUV$_i$, P_{Fi} represents the false alarm probability for AUV$_i$'s sonar, M_i represents the number of grid regions within AUV$_i$'s field of view, FOV_i represents the effective sonar detection range of AUV$_i$, $O(t)$ is a binary variable where $O(t) = 1$ represents AUV$_i$ discovering a target, and $O(t) = 0$ represents AUV$_i$ not discovering a target at time t.

4.4. Global State and Observation Space

In this section, based on the previously established search information map, we will define the global state ($\boldsymbol{S}$) and the observations ($\boldsymbol{O}$) for the AUV. The global state represents the complete description of the environment and the AUV's internal state. It includes information about the positions and states of all AUVs, the current state of the search information map (including coverage information, target presence information, and uncertainty information), the positions of any detected or suspected targets, and any other relevant variables that characterize the system at a given time. The global state $\boldsymbol{S}$ plays a significant role during the training phase of multi-agent reinforcement learning. It guides the AUVs to learn effective search strategies in situations where only local environmental information is observable. However, due to the large number of grid regions resulting from the discretization of search area S_i, the dimensionality of the global state $\boldsymbol{S}$ can become excessively high, leading to the "curse of dimensionality". To address this issue, the approach is to merge the grid regions of search area S_i into subregions, where multiple grid regions g are merged into a single subregion G. We refer to the grid region g as the secondary subregion and the region G as the primary subregion This merging process is illustrated in Figure 2a.



(**a**) (**b**)

Figure 2. (**a**) Grid region division. (**b**) Sonar detection range.

Based on the merged grid regions in the search information map, we design the global state $\boldsymbol{S}$ as a four-tuple:

$$\boldsymbol{S} = (\boldsymbol{C}, \boldsymbol{K}, \boldsymbol{P}, \boldsymbol{I}) \tag{6}$$

In this formula, $\boldsymbol{C} = \left(c_1, c_2, \ldots, c_{N_I}\right)$ represents the coverage information of each subregion G, $\boldsymbol{K} = \left(k_1, k_2, \ldots, k_{N_I}\right)$ represents the uncertainty information of each subregion G, $\boldsymbol{P} = \left(p_1, p_2, \ldots, p_{N_I}\right)$ represents the target existence probability information of each subregion G, N_I represents the total number of subregions G, $c_i, k_i, p_i \in \mathbb{R}$ represents the coverage, uncertainty, and target existence probability information of subregion G_i, which includes the average values of coverage, uncertainty, and target existence probability. $\boldsymbol{I} = (\boldsymbol{\eta}_1, \boldsymbol{\eta}_2, \ldots, \boldsymbol{\eta}_N)$ represents the position and orientation of AUVs, N represents the total number of AUVs. $\boldsymbol{\eta}_i = (x_i, y_i, \psi_i), x_i, y_i \in \mathbb{R}$ represents position coordinates of the i-th AUV along the ξ and η axes in the inertial coordinate system E-$\xi\eta\zeta$. $\psi_i \in [-\pi, \pi]$ represents the heading angle of the i-th AUV.

The AUV's sonar detection range is assumed to be the grid regions within a maximum sonar detection distance of D_s, and the number of grid regions within this range is assumed to be N_{FOV}. The AUV's sonar detection range is illustrated in Figure 2b. We combine the detection information obtained by AUV through sonar and the communication information among AUVs to form the AUV's observation ($\boldsymbol{O}$). The observation for the i-th AUV, denoted as $\boldsymbol{O}_i = (\boldsymbol{C}_i, \boldsymbol{K}_i, \boldsymbol{P}_i, \boldsymbol{\eta}_i, \boldsymbol{H}_i)$, includes the coverage value, uncertainty value, and target existence probability within the maximum sonar detection range of AUV$_i$. The variables relating to the relative positions and orientations of AUVs are also included in the observation, $\boldsymbol{H}_i = (\gamma_{i,1}, \gamma_{i,2}, \ldots, \gamma_{i,N-1})$ represents the relative spatial relationships between AUV$_i$ and the other AUVs, $\gamma_{i,j} = (\psi_{i,j}, d_{i,j})$ represents the relative spatial relationship between AUV$_i$ and AUV$_j$, $\psi_{i,j} \in [-\pi, \pi]$ represents the relative orientation angle between AUV$_i$ and AUV$_j$, and $d_{i,j} \in \mathbb{R}$ represents the distance between AUV$_i$ and AUV$_j$.

The calculation formula for $\psi_{i,j}$ is as follows:

$$\psi_{i,j} = \begin{cases} \psi_i - \pi - \arctan\left(\frac{y_j - y_i}{x_j - x_i}\right), \ x_j < x_i, y_j \geq y_i \\ \psi_i + \pi - \arctan\left(\frac{y_j - y_i}{x_j - x_i}\right), \ x_j < x_i, y_j < y_i \\ \psi_i - \arctan\left(\frac{y_j - y_i}{x_j - x_i}\right), \ x_j > x_i \end{cases} \tag{7}$$

The calculation formula for $d_{i,j}$ is as follows:

$$d_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{8}$$

4.5. Action Space Design

In the search process of multiple AUVs in the 2D plane s, we do not consider factors such as ocean currents. We assume that the AUVs travel at a constant speed, meaning that the AUV's longitudinal velocity u remains constant and is not affected by any lateral interference. In other words, the AUV's lateral velocity v is set to 0. Therefore, the action set a for AUVs executing the search task in this paper can be defined as $\{\Delta\psi_t\}$. It represents the change in the heading angle of AUVs at time t. Therefore, we define the action space of AUVs at time t as follows:

$$\boldsymbol{A} = \left(a_t^1, a_t^2, \ldots, a_t^N\right) \in \mathbb{R}^N \tag{9}$$

In Equation (9), a_t^i represents the action decided by the i-th AUV at time t, and N represents the number of AUVs. Taking into account the constraints imposed by the kinematic characteristics of AUVs, we impose the following restrictions on the action set a:

$$\begin{cases} -90^\circ \leq \Delta\psi_t \leq 90^\circ \\ a_t^i \in \{-90^\circ, -45^\circ, 0^\circ, 45^\circ, 90^\circ\} \end{cases} \tag{10}$$

In Equation (10), positive angle values indicate that the AUV is turning counterclockwise, negative angle values indicate that the AUV is turning clockwise, and an angle value of 0 indicates that the AUV is maintaining its current heading angle.

According to the AUV's two-dimensional kinematic equation in the inertial coordinate system E-$\xi\eta\zeta$, the AUV's position can be updated using the following equation:

$$\begin{bmatrix} x_{t+1} \\ y_{t+1} \end{bmatrix} = \begin{bmatrix} x_t \\ y_t \end{bmatrix} + \begin{bmatrix} \cos\psi_t & -\sin\psi_t \\ \sin\psi_t & \cos\psi_t \end{bmatrix} \begin{bmatrix} u_t \\ v_t \end{bmatrix} dt \tag{11}$$

In Equation (11), x_{t+1} and y_{t+1} represent the position coordinates of the AUV along the ξ and η axes in the inertial coordinate system at time t + 1. ψ_t represents the heading angle of the AUV at time t, u_t and v_t represent the longitudinal velocity and lateral velocity of the AUV at time t, and dt represents the time interval.

At time t + 1, the heading angle ψ_t can be updated using the following equation:

$$\psi_{t+1} = \psi_t + \Delta\psi_t \tag{12}$$

In Equation (12), ψ_t represents the heading angle of AUV at time t + 1, and $\Delta\psi_t$ represents the change in heading angle at time t.

4.6. Design of Reward Functions

The individual reward for an AUV in the multi-AUV cooperative area search problem can indeed be divided into multiple components to encourage specific behaviors. It is a common practice in reinforcement learning to create composite rewards that consider different aspects of the agent's behavior. Based on the search information map, we have identified four components for individual reward: coverage reward (r_c), uncertainty reward (r_k), target discovery reward (r_t), and collaboration reward (r_a). Therefore, the individual reward (r) for an AUV can be expressed as follows:

$$r = \alpha r_c + \beta r_k + \gamma r_t + \lambda r_a \tag{13}$$

In Equation (13), $\alpha, \beta, \gamma, \lambda \in \mathbb{R}$ represent the weight coefficients for each part of the reward.

4.6.1. Coverage Reward

When an AUV goes to a grid area $grid(m, n)$ that has never been searched before, i.e., the coverage value c_{mn} of grid area $grid(m, n)$ is 0, the AUV receives a coverage reward r_c, and the coverage value c_{mn} of that grid area $grid(m, n)$ changes from 0 to 1, indicating that it has been searched by the AUV, and its coverage value c_{mn} will not change thereafter. The specific update formula for coverage reward is as follows:

$$c_{mn} = \begin{cases} 1, \text{if } \ grid(m,n) \text{ is searched} \\ \qquad 0, \text{else} \end{cases} \tag{14}$$

If the coverage value of the grid area that the AUV goes to is 0, then the AUV receives a coverage reward. If the coverage value is already 1, the AUV does not receive any coverage reward.

4.6.2. Uncertainty Reward

For AUVs that have already searched a grid area $grid(m, n)$, even though its coverage value c_{mn} no longer changes, its uncertainty value k_{mn} continues to increase over time. Therefore, when an AUV goes to a grid area $grid(m, n)$ that has already been searched, it can still receive uncertainty rewards r_k related to its uncertainty value k_{mn}. The expression for uncertainty rewards is as follows:

$$r_k = \begin{cases} \frac{k_{mn}}{k_{\max}} - 1, \ k_{mn} < K \\ \frac{k_{mn}}{k_{\max}}, \ k_{mn} \geq K \end{cases} \tag{15}$$

In Equation (15), K is the tuning coefficient. When the uncertainty value k_{mn} of the grid area $grid(m, n)$ that the AUV is going to is greater than K, r_k is a positive reward. However, when k_{mn} is less than K, r_k is a negative reward.

4.6.3. Target Discovery Reward

The target discovery reward (r_t) is designed based on the probability of target existence in the grid cells within the AUV's sonar detection range. The probability of target existence represents the degree to which an AUV is inclined toward the presence of a target. If the grid cells within the AUV's sonar detection range have a higher probability of target existence, it implies that the AUV is closer to the target, and there is a greater likelihood of discovering the target in those areas. Essentially, the AUV is more likely to find the target in regions with a higher probability of target presence. Therefore, based on the probability of target presence in the grid cells within the AUV's sonar detection range, we have designed the target discovery reward (r_t), and its specific expression is as follows:

$$r_t = \frac{\sum_{grid(m,n)\in FOV} p_{mn}}{N_{FOV}} \tag{16}$$

In Equation (16), FOV represents the grid cells within the AUV's sonar detection range, and N_{FOV} represents the number of grid cells within the AUV's sonar detection range.

4.6.4. Collaborative Reward

In the process of multi-AUV searching within partitioned area S_i, the collaborative reward is designed to encourage individual AUVs to perform distributed area searches. This aims to maximize the coverage of the multi-AUV system's search area while avoiding the problem of AUVs clustering too closely and causing overlap in their sonar detection ranges. However, it is important to maintain a minimum distance between AUVs to ensure effective communication for sharing search information and updating the search information map. The specific formula for the collaborative reward may take into account factors related to AUV dispersion and the maintenance of communication within a certain range. The reward formula should encourage AUVs to strike a balance between spreading out to cover more area and staying close enough for efficient information sharing. Therefore, we design the collaboration reward (r_a), and its specific expression is as follows:

$$r_a = \begin{cases} -R,\ d_i < D_s \\ k\left(\frac{d_i - D_s}{D_c - D_s}\right), D_s < d_i < D_c \\ 0, D_c < d_i \end{cases} \tag{17}$$

In Equation (17), d_i represents the average distance between the i-th AUV and the other AUVs, D_s represents the maximum detection distance of the sonar, and D_c represents the communication distance between AUVs. When the average distance d_i between AUV$_i$ and the other AUVs is less than D_s, the AUVs are too close together, resulting in a significant overlap in their sonar detection ranges and a higher risk of collisions. This situation is unfavorable for the collaborative search of multiple AUVs, and AUV$_i$ receives a negative reward. When d_i is between D_s and D_c, the distance between AUV$_i$ and the other AUVs is appropriate, and AUV$_i$ receives a positive reward that is positively correlated with d_i. When d_i exceeds D_c, the average distance between AUV$_i$ and the other AUVs exceeds the communication distance, which hinders information sharing in the multi-AUV system. Therefore, AUV$_i$ does not receive any reward.

5. Algorithm Design

The QMIX algorithm [14] is a value-based algorithm in multi-agent reinforcement learning. Its core idea is to use a mixing neural network to combine the action-values $Q_i^\pi(\tau_i, a_i), i \in \{1, 2, \ldots, N\}$ of individual agents in a complex nonlinear manner to synthesize the joint action-value $Q_{tot}(\boldsymbol{s}, \boldsymbol{a})$. This is used to estimate the action-value of joint actions $\boldsymbol{a} = (a_1, a_2, \ldots, a_N) \in \mathbb{R}^N$ in a multi-agent system. Indeed, it is worth noting that in the action-values $Q_i^\pi(\tau_i, a_i)$ of individual agents, the history of observed actions τ_i is based on local observations o_i. This means that the action-values of each agent take into account the historical actions. Additionally, the joint action-value $Q_{tot}(\boldsymbol{s}, \boldsymbol{a})$ is monotonically increasing with respect to the action-values $Q_i^\pi(\tau_i, a_i)$ of individual agents, satisfying Equation (18). This constraint ensures consistency between the centralized strategy $Q_{tot}(\boldsymbol{s}, \boldsymbol{a})$ and the individual agent strategies $Q_i^\pi(\tau_i, a_i)$.

$$\frac{\partial Q_{tot}(\boldsymbol{s}, \boldsymbol{a})}{\partial Q_i^\pi(\tau_i, a_i)} \geq 0, \forall a_i \tag{18}$$

The QMIX algorithm based on the maximum entropy mechanism [15] builds upon the QMIX algorithm and incorporates ideas from the soft actor-critic algorithm, which is based on maximum entropy reinforcement learning. First, each agent adopts an actor-critic network framework, enhancing the existing network structure by introducing a policy network π_θ. Secondly, during the training process, a maximum entropy mechanism is introduced. Each agent not only aims to maximize the expected return but also seeks to maximize entropy. This mechanism makes the learning search strategies of agents more stochastic, encouraging agents to explore the environment more extensively and preventing the search strategy from converging prematurely.

5.1. Algorithm's Computation Process

After incorporating the maximum entropy mechanism, the expected return for each agent during the training process is no longer solely determined by the rewards provided by the environment. Instead, it is composed of both the rewards from the environment and the entropy associated with the actions taken by each agent. Therefore, the expected return in reinforcement learning is formulated as follows:

$$R(\boldsymbol{s}_t, \boldsymbol{a}_t) + \sum_{i=1}^{N} \alpha_i H_i(\pi(\cdot|\tau_{i,t})) \tag{19}$$

In Equation (19), $\boldsymbol{s}_t$ represents the global state of the environment at time t, $\boldsymbol{a}_t = (a_{1,t}, a_{2,t}, \ldots, a_{N,t})$ denotes the joint action taken by multiple agents at time t, $R(\boldsymbol{s}_t, \boldsymbol{a}_t)$ represents the joint reward received by multiple agents, and α_i is the entropy regularization coefficient. $H_i(\pi(\cdot|\tau_{i,t}))$ represents the entropy of the action $a_{i,t}$ taken by agent i at time t. The formula for calculating the entropy H_i for agent i at time t is as follows:

$$H(\pi(a_{i,t}|\tau_{i,t})) = -\textstyle\sum_{a \in A_i} \pi_i(a|\tau_{i,t}) \log(\pi_i(a|\tau_{i,t})) \tag{20}$$

The training objective is to learn the optimal policy π^* that maximizes the expected return. In other words, the optimal policy π^* satisfies the following equation:

$$\pi^* = \underset{\pi}{\operatorname{argmax}} \mathrm{E}_{\boldsymbol{s} \sim p, \boldsymbol{a} \sim \pi} \left[\sum_{t=0}^{\infty} \gamma^t \left(R(\boldsymbol{s}_t, \boldsymbol{a}_t) + \sum_{i=1}^{N} \alpha_i H_i(\pi(\cdot|\tau_{i,t})) \right) \right] \tag{21}$$

The joint action-value function $Q_{tot}(\boldsymbol{s}, \boldsymbol{a})$ for multiple agents can be calculated through a mixing neural network using the following expression:

$$Q_{tot}(\boldsymbol{s}, \boldsymbol{a}) = F_{mix}(q_1(\tau_1, a_1), q_2(\tau_2, a_2), \ldots, q_{\;N}(\tau_N, a_N)) \tag{22}$$

In Equation (22), $q_i(\tau_i, a_i)$ represents the action-value function for the i-th agent, and the F_{mix} function represents the mixing neural network.

The individual state-value function $v_i(\tau_i)$ for an agent can be obtained by taking the expectation of the action-value function $q_i(\tau_i, a_i)$ with respect to action a_i, and the calculation formula is as follows:

$$v_i(\tau_i) = \mathrm{E}_{a_i \sim \pi}[q_i(\tau_i, a_i)] \tag{23}$$

The overall state-value function $V_{tot}(\boldsymbol{s})$ for multiple agents can be obtained by taking the expected value of the joint action-value function $Q_{tot}(\boldsymbol{s}, \boldsymbol{a})$ with respect to the joint action $\boldsymbol{a}$. The calculation formula is expressed as follows:

$$\begin{aligned} V_{tot}(\boldsymbol{s}) &= E_{\boldsymbol{a} \sim \pi} Q_{tot}(\boldsymbol{s}, \boldsymbol{a}) \\ &= E_{\boldsymbol{a} \sim \pi}[F_{mix}(q_1(\tau_1, a_1), q_2(\tau_2, a_2), \ldots, q_N(\tau_N, a_N))] \end{aligned} \tag{24}$$

Due to the complexity of taking the expected value of the joint action-value function $Q_{tot}(\boldsymbol{s}, \boldsymbol{a})$ with respect to the joint action $\boldsymbol{a}$, to reduce computational complexity, Equation (24) can be approximated as follows:

$$\begin{aligned} V_{tot}(\boldsymbol{s}) &= E_{\boldsymbol{a}\sim\pi}[F_{mix}(q_1(\tau_1, a_1), q_2(\tau_2, a_2), \ldots, q_N(\tau_N, a_N))] \\ &\approx F_{mix}[E_{a_1\sim\pi}q_1(\tau_1, a_1), E_{a_2\sim\pi}q_2(\tau_2, a_2), \ldots, E_{a_N\sim\pi}q_N(\tau_N, a_N)] \\ &\approx F_{mix}[v_1(\tau_1), v_2(\tau_2), \ldots, v_N(\tau_N)] \end{aligned} \tag{25}$$

After incorporating the maximum entropy mechanism, the overall state-value $V(\boldsymbol{s})$ based on the maximum entropy mechanism can be expressed as a function of the individual state-values $v_i(\tau_i)$ of agents and the entropy $H_i(\pi(a_i|\tau_i))$. The specific formula is as follows:

$$\begin{aligned} V^{\pi}(\boldsymbol{s}) &= V_{tot}(\boldsymbol{s}) + \sum_{i=1}^{N} \alpha_i H_i(\pi(\cdot|\tau_{i,t})) \\ &= \mathrm{E}_{\boldsymbol{a}\sim\pi}\left[F_{mix}(q_1(\tau_1, a_1), q_2(\tau_2, a_2), \ldots, q_N(\tau_N, a_N)) + \sum_{i=1}^{N} \alpha_i H_i(\pi(\cdot|\tau_{i,t}))\right] \\ &\approx F_{mix}[\mathrm{E}_{a_1\sim\pi}q_1(\tau_1, a_1), \mathrm{E}_{a_2\sim\pi}q_2(\tau_2, a_2), \ldots, \mathrm{E}_{a_N\sim\pi}q_N(\tau_N, a_N)] + \sum_{i=1}^{N} \alpha_i H_i(\pi(\cdot|\tau_{i,t})) \\ &\approx F_{mix}[v_1(\tau_1), v_2(\tau_2), \ldots, v_N(\tau_N)] + \sum_{i=1}^{N} \alpha_i H_i(\pi(\cdot|\tau_{i,t})) \end{aligned} \tag{26}$$

The Bellman equation for the action-value function $Q^{\pi}(\boldsymbol{s}, \boldsymbol{a})$ regularized by entropy can be expressed as follows:

$$\begin{aligned} Q^{\pi}(\boldsymbol{s}, \boldsymbol{a}) &= \mathrm{E}_{\boldsymbol{a}\prime\sim\pi}\left[R(\boldsymbol{s}, \boldsymbol{a}) + \gamma\left(Q_{tot}(\boldsymbol{s}\prime, \boldsymbol{a}\prime) + \sum_{i=1}^{N} \alpha_i H_i(\pi(\cdot|\tau_i\prime))\right)\right] \\ &= R(\boldsymbol{s}, \boldsymbol{a}) + \gamma\left(V_{tot}(\boldsymbol{s}\prime) + \sum_{i=1}^{N} \alpha_i H_i(\pi(\cdot|\tau_i\prime))\right) \\ &= R(\boldsymbol{s}, \boldsymbol{a}) + \gamma V^{\pi}(\boldsymbol{s}\prime) \end{aligned} \tag{27}$$

Based on Equation (27), we can compute the target Q-values. Therefore, the loss function for the critic network can be represented as follows:

$$L_Q = \mathrm{E}_{\boldsymbol{s},\boldsymbol{a},\boldsymbol{s}\prime\sim D}\left[Q_{tot}(\boldsymbol{s}, \boldsymbol{a}) - \left(R(\boldsymbol{s}, \boldsymbol{a}) + \gamma \overline{V}(\boldsymbol{s}\prime)\right)\right]^2 \tag{28}$$

In Equation (28), $Q_{tot}(\boldsymbol{s}, \boldsymbol{a})$ represents the joint action-value function for multiple agents. It can be computed from individual action-value functions $q_i(\tau_i, a_i)$ of each agent using the mixing neural network as defined in Equation (22). $R(\boldsymbol{s}, \boldsymbol{a})$ denotes the joint reward obtained from the environment. $\overline{V}(s\prime)$ represents the target state-value function, which can be calculated based on the individual target state-value functions of each agent and the entropies of the agents, as defined in Equation (26).

The training objective of the actor network is to maximize the overall state-value $V^{\pi}(s)$. Therefore, the loss function of the actor network can be represented as follows:

$$L_{\pi} = -\mathrm{E}_{\boldsymbol{s}\sim D,\boldsymbol{a}\sim\pi} V^{\pi}(\boldsymbol{s}) \tag{29}$$

The above can be summarized, and the algorithm's computational flowchart is illustrated in the following Figure 3. The significance and explanations of symbols and abbreviations used in the design process of the algorithm are summarized in the Appendix A.

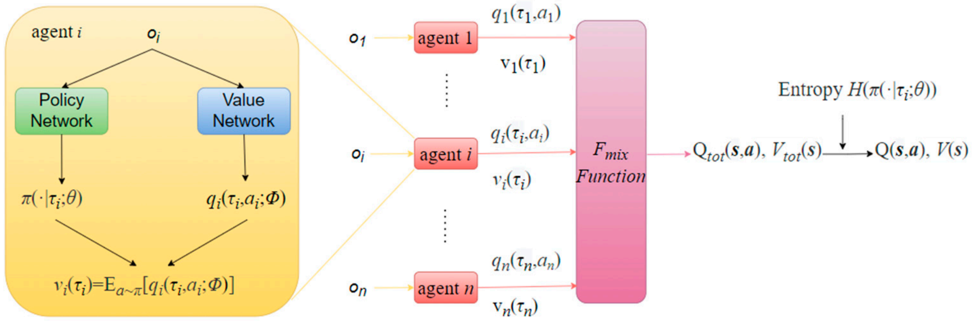


Figure 3. The computational flowchart.

5.2. Algorithm Framework Design

The SAC-QMIX algorithm consists of networks for multiple agents and a mixing neural network. The specific network architecture is designed as shown in the Figure 4. The design involves the following components:

1. Individual agent networks: Each agent has its own actor-critic neural network architecture. This network takes local observations as input and outputs action probabilities (actor) and action-value estimates (critic);
2. Mixing neural network (QMIX): The mixing neural network takes the individual agents' action-values and combines them in a complex, non-linear manner to estimate the joint action-values for the multi-agent system;
3. Maximum entropy mechanism: During training, a maximum entropy mechanism is incorporated into the agent networks, which encourages exploration and introduces randomness into the policies. This mechanism helps agents to learn more robust and diverse strategies;

According to the diagram in Figure 4, the agent's network includes both a policy network π_θ and a value network Q_ϕ. The mixing neural network consists of a hypernetwork and a feedforward neural network. The hypernetwork generates the weights w_1, w_2 and biases b_1, b_2 for the feedforward network based on the global state (s). The feedforward network combines the action-values Q_a of each agent to produce the joint action-value Q_{tot}. It is important to note that the weights w_1, w_2 of the feedforward network must remain positive to satisfy the condition in Equation (18).

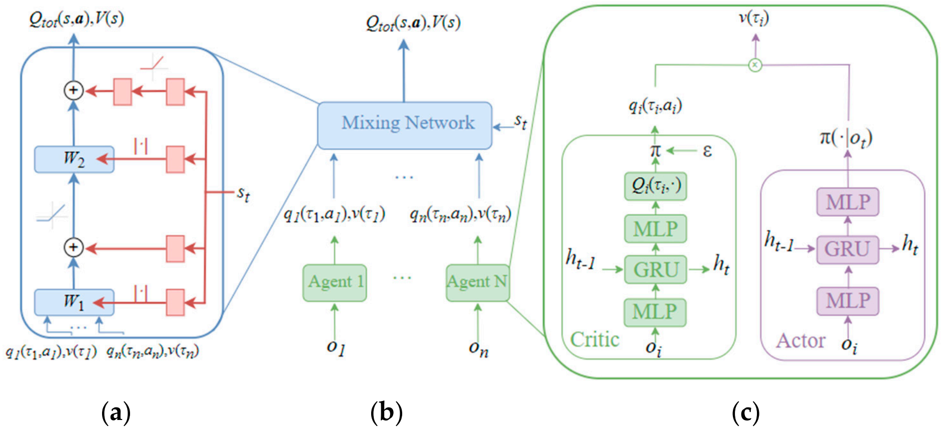


Figure 4. (**a**) The structure of the mixing neural network. (**b**) The overall structure SAC-QMIX consists of. (**c**) Network structure of the agents.

The training approach used in this paper for the algorithm is called centralized training with decentralized execution (CT-DE). This means that during the training process, the mixing neural network synthesizes the joint action-value function $Q_{tot}(\mathbf{s}, \mathrm{a})$ and the global state-value function $V(\mathbf{s})$ based on the individual agents' action-value functions

$q_i(\tau_i, a_i)$ and state-value functions $v_i(\tau_i)$ as well as the global environment state $\boldsymbol{s}$. However, after training is completed, each agent makes real-time action decisions based solely on observations o when executing tasks. The training framework of the algorithm is illustrated in Figure 5. This training method utilizes the global state $\boldsymbol{s}$ during the training process to compensate for the adverse effects caused by the insufficient observational information of the individual agent. The global state $\boldsymbol{s}$ contains information from all agents, which helps alleviate the instability issues that can arise during multi-agent reinforcement learning training.

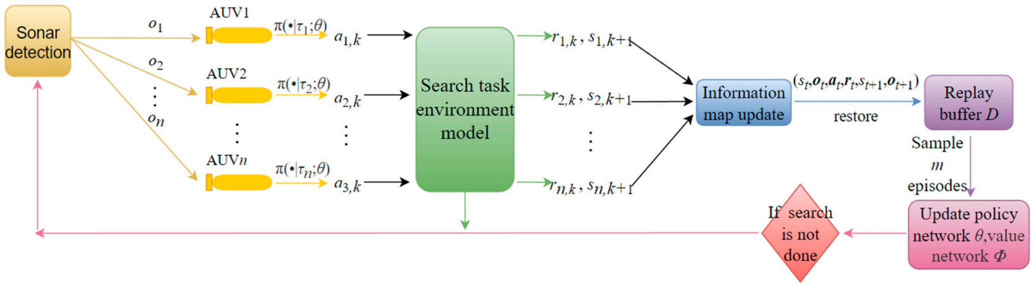


Figure 5. Training framework diagram.

Incorporating the computational process of the above algorithm with the algorithmic framework, the training process of the SAC-QMIX algorithm is as shown in Algorithm 1.

Algorithm 1. SAC-QMIX algorithm training process.

```
Input initial policy parameters $\theta$, Q − functions parameters , $\phi_1$, $\phi_2$, empty replay buffer $D$
Set target parameters equal to main parameters, $\phi_{\text{targ},1} \leftarrow \phi_1$ , $\phi_{\text{targ},2} \leftarrow \phi_2$
for $steps$ = 1:$M$ do:
    Observe the state $s$, obtain observations $o_i$, for $i$ = 1, 2, . . ., $n$
    Select action $a_i \sim \pi_\theta(\cdot|\tau_i)$, for $i$ = 1, 2, . . ., $n$
    Execute action $a_i$ in the environment, for $i$ = 1, 2, . . ., $n$
    Observe next state $\boldsymbol{s}\prime$, next observation $o\prime$, reward $r(\boldsymbol{s}, \boldsymbol{a})$, $done$ signal
    Store $(\boldsymbol{s}, \boldsymbol{o}, \boldsymbol{a}, r, d)$ in episode experience $exp$
    if s′ is terminal, $done$ is true then
        Supplement the episode experience $exp$ to the maximum episode experience
        Reset environment
    end if
    Store episode experience $exp$ in replay buffer $D$
    if it is time to update then
        Randomly sample a batch of episode experience $exp$ from replay buffer $D$
        Compute individual action − value $q_i$: $q_i^k = Q_{\phi,i}(\tau_k, a_k)$, $q^k_{\text{targ},i} = Q_{\phi_{\text{targ}},i}(\tau_k, a_k)$,
            for $i$ = 1, 2, for $k$ = 1, 2, . . ., $n$
        Compute individual state − value $v_i$:
            $v_i^k(\tau_i) = \mathrm{E}_{a_i \sim \pi}\left[q_i^k(\tau_k, a_k)\right]$ for $i$ = 1, 2 for $k$ = 1, 2, . . ., $n$
            $v^k_{\text{targ},i}(\tau_i) = \mathrm{E}_{a_i \sim \pi}\left[q^k_{\text{targ},i}(\tau_k, a_k)\right]$ for $i$ = 1, 2 for $k$ = 1, 2, . . ., $n$
        Compute united action-value function:
            $Q_{tot,i}(\boldsymbol{s}, \boldsymbol{a}) = F_{mix}\left(q_i^1(\tau_1, a_1), \ldots q_i^n(\tau_n, a_n)\right)$ for $i$ = 1, 2
        Compute the overall state − value function $V_{tot,i}(\mathbf{s}) = F_{mix}\left(v_i^1(\tau_1), \ldots, v_i^n(\tau_n)\right)$
            for $i$ = 1, 2
        Compute target Q-function $y$:
            $y(r, \boldsymbol{o}\prime, d) = r + \gamma\left(\min\limits_{i=1,2} F_{mix}\left(v^1_{\text{targ},i}(\tau_1'), \ldots, v^n_{\text{targ},i}(\tau_n')\right) + \sum\limits_{i=1}^{n} \alpha_i H_i(\pi(\cdot|\tau_i\prime))\right)$
        Update Q-network parameters using gradient descent:
            $\nabla_{\phi_i} \frac{1}{|B|} \sum\limits_{\exp \in B} \left(Q_{tot,i}(\boldsymbol{s}, \boldsymbol{a}) - y(r, \boldsymbol{o}\prime, d)\right)^2$ for $i$ = 1, 2
```

Algorithm 1. *Cont.*

```
        Update policy network parameters using gradient ascent:
            $\nabla_\theta \frac{1}{|B|} \sum_{\mathrm{exp}\in B} \left( \min_{i=1,2} V_{tot,i}(\mathbf{s}) + \sum_{i=1}^{n} \alpha_i H_i(\pi(\cdot|\tau_i)) \right)$
        Update the entropy regularization coefficient using gradient descent:
            $\nabla_\alpha \frac{1}{|B|} \sum_{\mathrm{exp}\in B} \left( \sum_{i=1}^{n} \left( H(\pi_\theta(a_i|\tau_i)) - \overline{H} \right) \right)$
        Soft update target network:
        $\phi_{\mathrm{targ},1} = \tau\phi_1 + (1-\tau)\phi_{\mathrm{targ},1}$
        $\phi_{\mathrm{targ},2} = \tau\phi_2 + (1-\tau)\phi_{\mathrm{targ},2}$
    end if
end for
```

6. Simulation Test

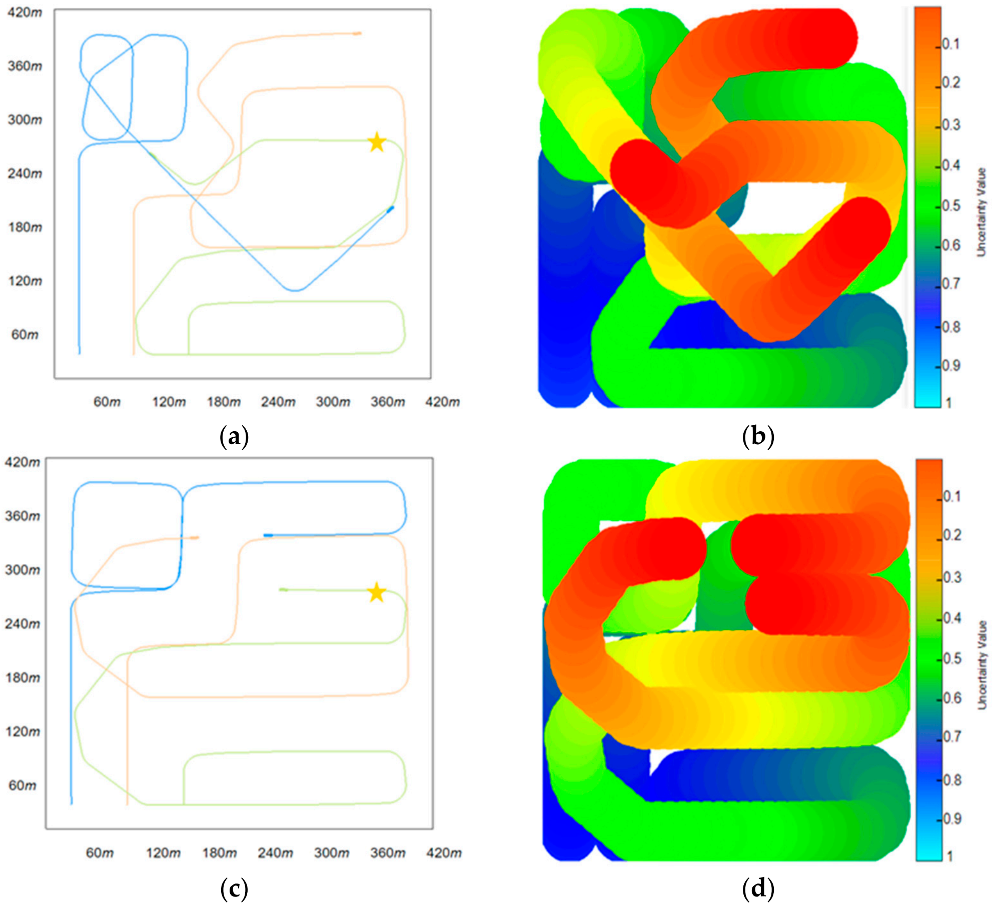
This paper sets the collaborative search area for multiple AUVs in two different scenarios. Scenario 1 is a square area with dimensions of 420 m × 420 m, and the Cartesian coordinate system's origin is located at the bottom left corner of the search area. The search area is divided into 7 × 7 primary subregions, with each primary subregion measuring 60 m × 60 m in size. The corresponding secondary subregions are 14 × 14. Scenario 2 is an irregular octagonal area with sides measuring 180 m. This scenario is primarily used to test the adaptability of the search strategy to irregular search areas. We used the QMIX algorithm as well as an improved version of the QMIX algorithm to train in both of these scenarios. Initially, all grid cells have a coverage mark of 0, an uncertainty value of the maximum value $k_{\max}$ equal to 1 and an uncertainty growth factor T equal to 50. In the corresponding uncertainty reward module, the tuning coefficient K is set to 0.5. The configuration includes three AUVs, each with a maximum sonar detection range (D_s) of 90 m, a sonar detection accuracy (P_D) of 0.8, a sonar false alarm probability (P_F) of 0.2, an effective communication distance between AUVs of 240 m, an AUV cruising speed (v) of 1 m per second, and an AUV action execution time (Δt) of 60 s. We established a neural network framework based on PyTorch in the Python platform and implemented the algorithm. The neural network parameters and learning training parameters are detailed in Table 2. Finally, we conducted simulation testing in a reinforcement learning gym environment.

Table 2. Training parameters table.

Parameter	Value
Actor learning rate lr_1	1×10^{-4}
Critic learning rate lr_2	1×10^{-4}
α learning rate lr_3	1×10^{-4}
Target entropy $\overline{H}$	0.2
Discount factor γ	0.99
Soft update coefficient τ	0.1
RNN dimension	64
Batch size	32
Max episodes M	30,000
Max steps T	1.5×10^6

Our simulation Scenario 1 is set up as follows: AUV 1 starts at (30 m, 30 m), AUV 2 starts at (90 m, 30 m), and AUV 3 starts at (150 m, 30 m). Each AUV has an initial heading of 90°. The target is located at (330 m, 270 m) and remains stationary.

Figure 6 shows the simulation diagram based on the reinforcement learning gym environment. Figure 6a represents the search trajectory map based on the QMIX algorithm, Figure 6c represents the search trajectory map based on the SAC-QMIX algorithm, and Figure 6b,d represent the corresponding uncertainty maps of the search areas. In Figure 6, the first AUV is represented in blue, the second AUV is represented in yellow, the third AUV is represented in green, and the golden pentagon star represents the target. The search areas of AUVs are assigned different colors based on the magnitude of uncertainty values, with uncertainty values increasing from red to light blue.



(**a**) (**b**) (**c**) (**d**)

Figure 6. (**a**) Search trajectory map based on QMIX. (**b**) Uncertainty map based on QMIX. (**c**) Search trajectory map based on SAC-QMIX. (**d**) Uncertainty map based on SAC-QMIX.

Our simulation Scenario 2 is set up as follows: the first AUV's initial position is at (30 m, 270 m), the second AUV's initial position is at (30 m, 210 m), and the third AUV's initial position is at (30 m, 150 m). The initial heading for each AUV is 0°. The target position is at (330 m, 270 m), and it remains constant. It is worth noting that, compared to the regular square search area in Scenario 1, the four corners of the search area in Scenario 2 are restricted zones. Therefore, the boundary conditions for the search area in Scenario 2 will be more complex. The final simulation results for Scenario 2 are shown in Figure 7. In Figure 7a, the search trajectory is based on the QMIX algorithm, and in Figure 7c, it is based on the SAC-QMIX algorithm. Figure 7b,d represent the corresponding uncertainty maps of the search area.

Based on the simulation results of the two scenarios, we can observe that both the search strategies using the QMIX algorithm and the SAC-QMIX algorithm achieve high area coverage rates and successfully locate the target. However, the trajectory pattern of the search based on the QMIX algorithm lacks regularity, indicating lower coordination among AUVs during the search process. In contrast, the search trajectory based on the SAC-QMIX algorithm appears more organized, suggesting better coordination among AUVs. The latest research progress of the multi AUVs cooperative area search system (MACASS) in unknown environments is shown in the Supplementary Materials.

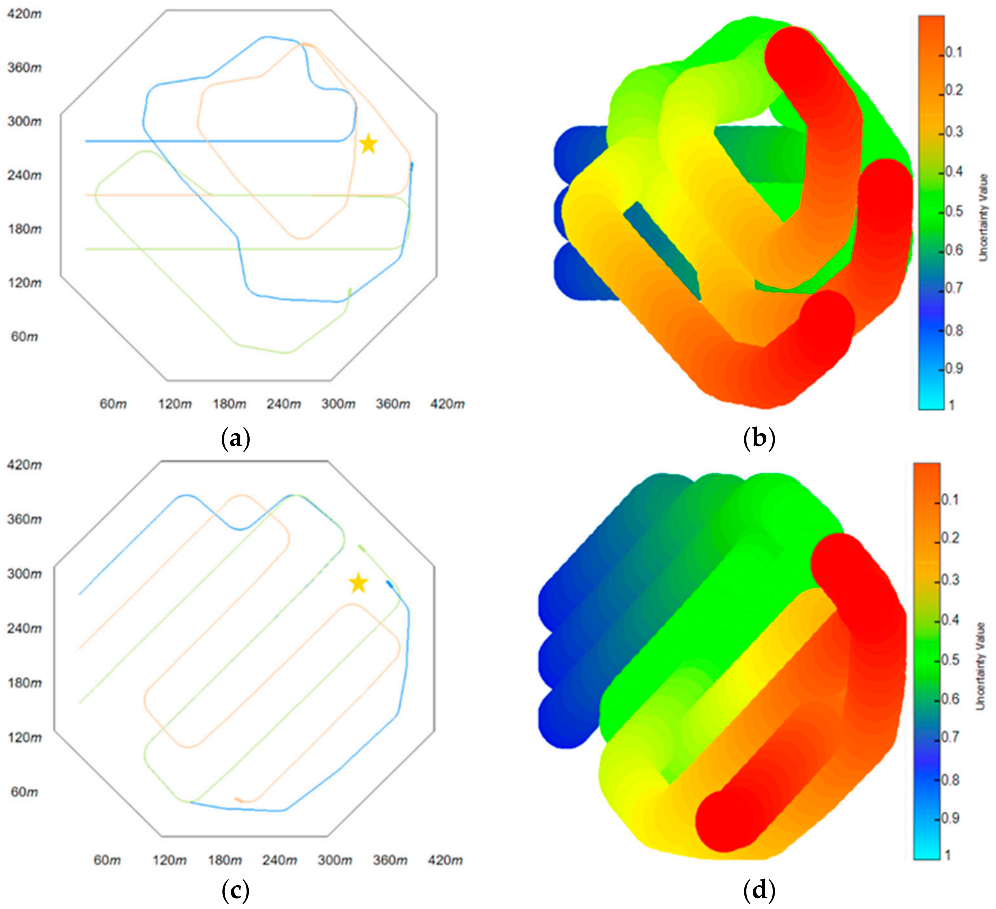


Figure 7. (**a**) Search trajectory map based on QMIX. (**b**) Uncertainty map based on QMIX. (**c**) Search trajectory map based on SAC-QMIX. (**d**) Uncertainty map based on SAC-QMIX.

We also conducted simulation tests on the fault tolerance performance of the MARL-based multi-AUV cooperative area search system. To simulate the potential single AUV breaks down in the multi-AUV cooperative area search system, we set a limited number of decision times for AUV 2. When AUV 2 reaches this limited number of decision times, it ceases to perform the search task, while the remaining AUVs continue with the search task. In the simulation test, we applied the search area from Scenario 1. The multi-AUV cooperative area search system utilized the search strategy based on the SAC-QMIX algorithm. We present the simulation results in Figure 8, where Figure 8a represents the initial stage of the search task, Figure 8b represents the stage when each AUV has made five decisions, Figure 8c represents the stage when AUV 2 breaks down, Figure 8d represents the stage when the remaining AUVs have made 15 decisions, and Figure 8e represents the stage when the remaining AUVs have made 25 decisions.

From the simulation results, it can be observed that even when one AUV experiences a failure and anchors, the remaining AUVs can still achieve a high area coverage rate in completing the area search task. This indicates that the MARL-based multi-AUV cooperative area search system possesses a certain degree of fault tolerance.

Furthermore, since the target information in the search area is initially unknown to each AUV, they can only obtain observational information through sonar detection and communication between AUVs during the search process and use this information to make action decisions based on the policy network. We illustrate the changes in the target probability map for AUVs during the search process in Figure 9. As shown in Figure 9, at the beginning of the search task, due to the lack of a priori information about the target, the target follows a uniform distribution in the search area. However, as the search task officially begins, each AUV calculates the target's probability of existence within the sonar detection range using the Bayesian formula based on sonar detection results. The target probability map is updated accordingly. Therefore, as the AUVs continue the search, the target's probability of existence in the grid areas without targets gradually approaches zero, and eventually, the target probability map converges to the target location.

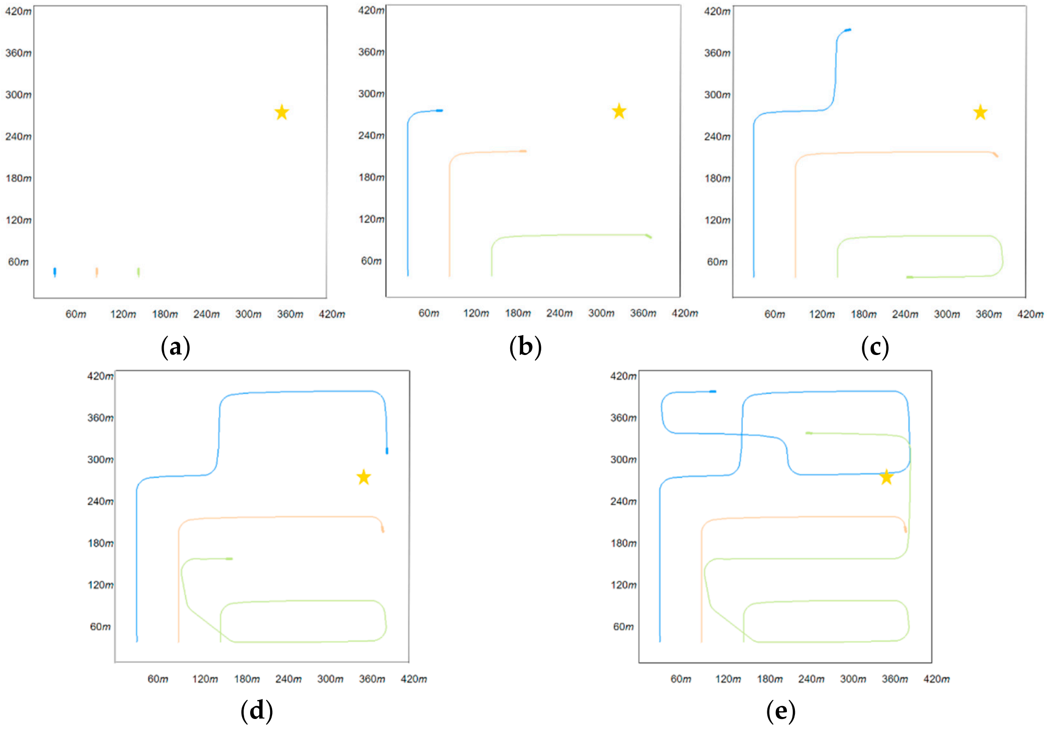


Figure 8. (**a**) Initial stage of the search task (**b**) Stage of decision-making with 5 decisions. (**c**) Stage when AUV 2 breaks down (**d**) Stage of decision-making with 15 decisions (**e**) Stage of decision-making with 25 decisions.

We compared the training results of the SAC-QMIX algorithm with the QMIX algorithm graph. The training results of both algorithms are shown in Figure 10, where Figure 10a represents the training reward graph and Figure 10b represents the search area coverage. Besides, we adopt the classic reinforcement learning method DQN (Deep Q-Network) to train the collaborative area search strategy network for multiple AUVs, and we use the training results as a contrast. The training method is also distributed training.

From the training results, we can observe that the SAC-QMIX algorithm converges faster than the QMIX algorithm. SAC-QMIX achieves a high level of search strategy in approximately 100,000 iterations. This is because the addition of the maximum entropy mechanism in the SAC-QMIX algorithm encourages agents to explore more during the training process, preventing them from getting stuck in local optima, thus speeding up the overall convergence process. The collaborative search strategy based on DQN did not converge to satisfactory results, and the reward and area detection rate remained consistently low.

Finally, we conducted ablation experiments on the coverage reward and uncertainty reward to explore the effects of different reward modules on the algorithm training results and the final test results. The algorithm used during the training process was the QMIX algorithm. The training results of different reward modules are shown in the Figure 11.

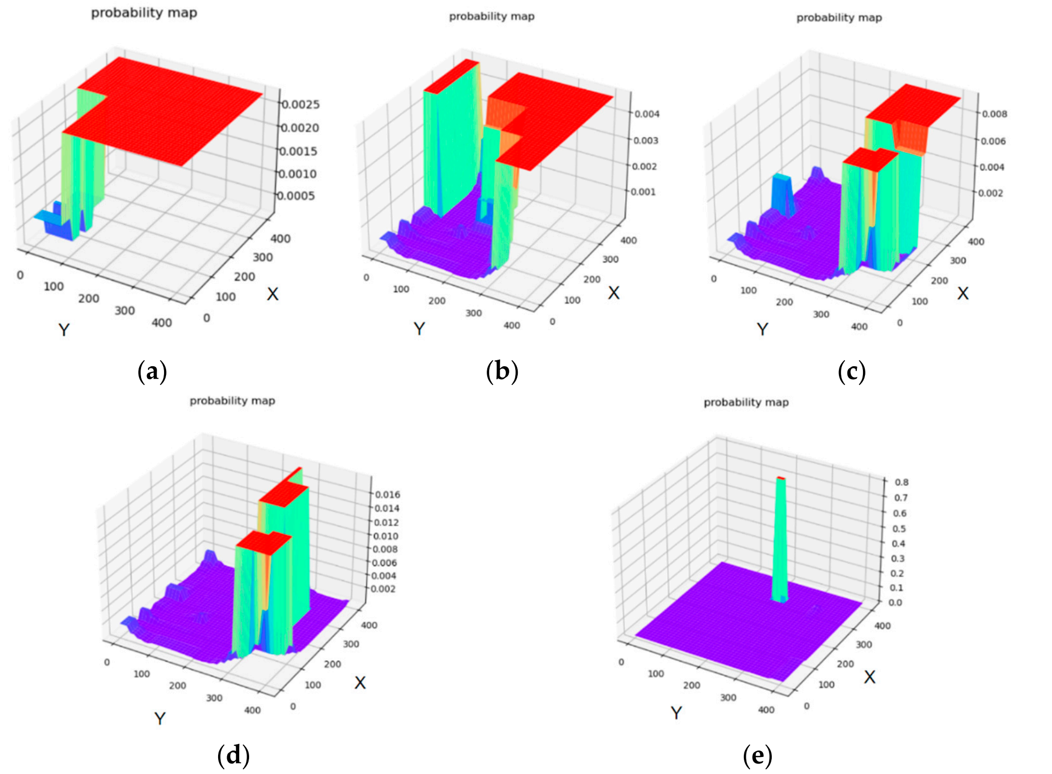


Figure 9. (**a**) Probability map for initial stages. (**b**) Probability map for 5 decisions. (**c**) Probability map for 8 decisions. (**d**) Probability map for 15 decisions (**e**) Probability map for 25 decisions.

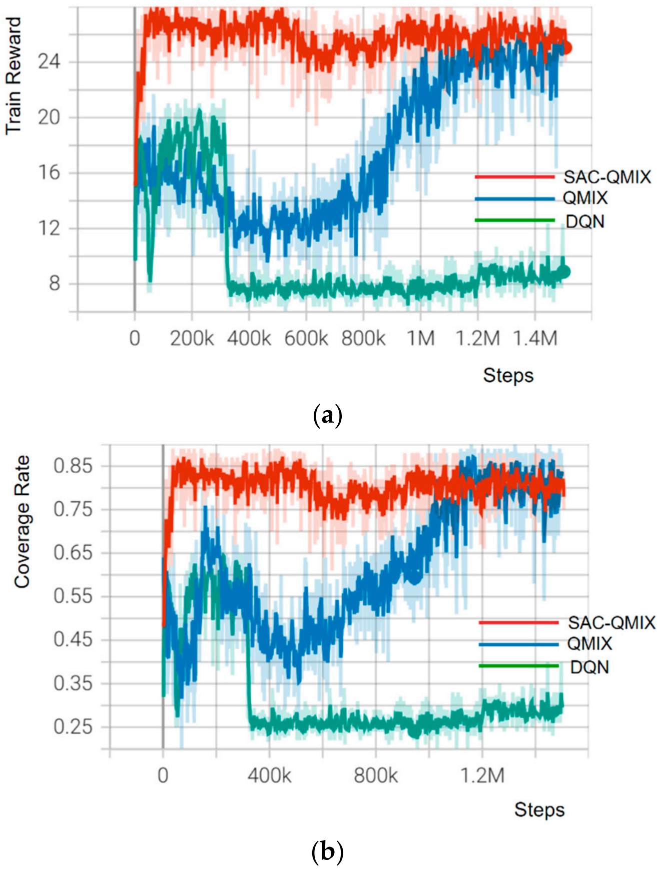


Figure 10. (**a**) Training curve of reward. (**b**) Training curve of coverage rate.

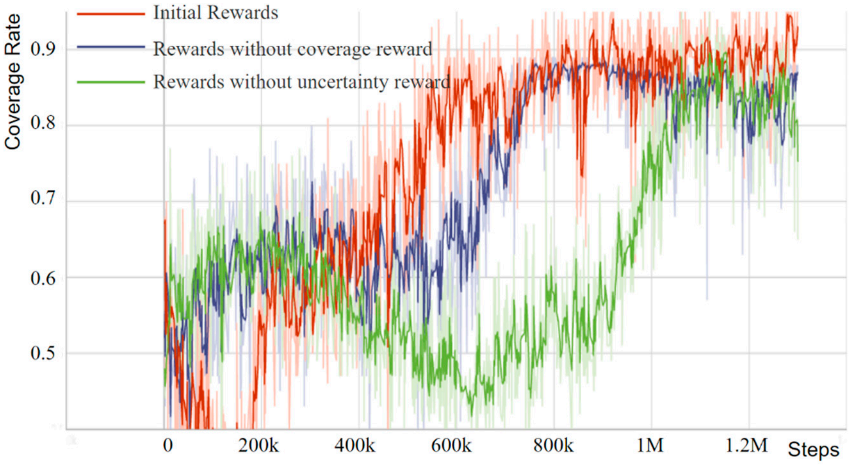


Figure 11. Training results under different reward settings.

From Figure 11, it can be observed that although the final training results under different reward settings are consistent, the AUVs can complete the area search task with a higher degree of exploration. However, under the initial reward setting, the algorithm converges faster, starting to converge around 600,000 steps. The algorithm without the coverage reward module converges slightly slower, starting to converge around 700,000 steps. On the other hand, the algorithm without the uncertainty reward module converges slowest, starting to converge around 1.1 million steps.

Corresponding to the ablation experiment mentioned above, this article conducts simulation tests based on the policy networks trained with different reward modules to validate the performance of different policy networks in searching for targets. The simulation area is a 600 m × 600 m square region, with the origin of the Cartesian coordinate system located at the bottom left corner of the search area. The initial positions of AUVs are as follows: AUV 1 starts at (30 m, 30 m), AUV 2 starts at (90 m, 30 m), and AUV 3 starts at (150 m, 30 m). The initial heading of each AUV is set to 90°, and the cruising speed of AUVs is 1 m/s. The sonar detection model used is the one established in Section 4.2. In each simulation test, each AUV can make a maximum of 40 decisions. The target's position (x_t, y_t) is randomly generated in each simulation test, and the simulation is conducted 300 times, where $x_t \in [0, 600]$, $y_t \in [0, 600]$; the results are shown in the Table 3.

Table 3. Table of test results with different reward modules.

Reward Module	Number of Tests	Successful Search Times	Success Search Rate
Initial reward	300	299	99.7%
Reward without coverage	300	278	92.7%
Reward without uncertainty	300	262	87.3%

From Table 3, it can be seen that policy networks trained with different reward modules can enable multiple AUVs to achieve a high success rate in completing the target search task. However, the success rate of searches without the coverage reward and without the uncertainty reward modules is significantly lower than that of searches under the initial reward module.

To compare and analyze the target search efficiency under different reward modules, this paper randomly selected 80 simulated tests from the successful target search simulations mentioned above and recorded the number of steps required to search for the target. The statistical results are shown in the Figure 12.

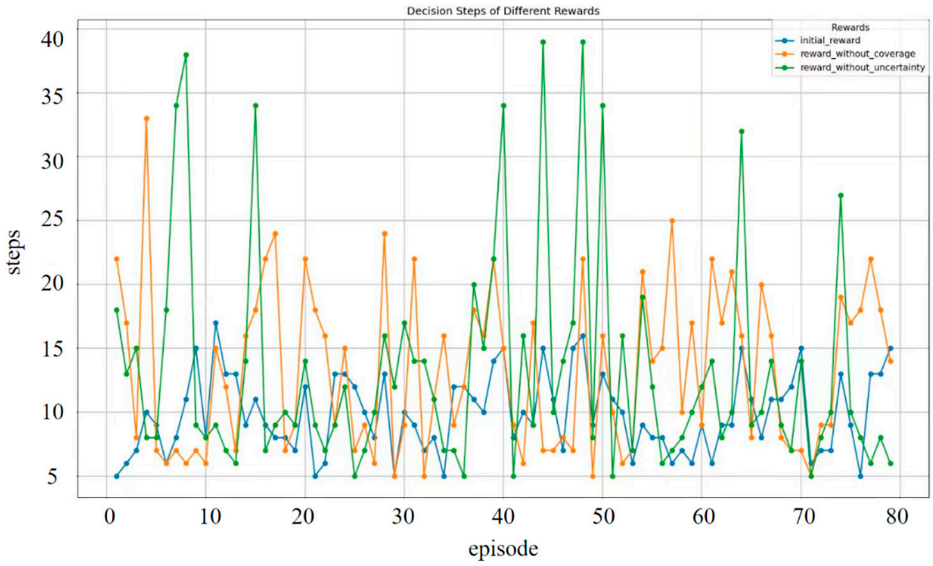


Figure 12. Number of target search decisions under different reward modules.

From Figure 12, we can see that the average number of decisions for the initial reward is 10, with a total variance of 9.49. For the network without coverage reward, the average number of decisions is 13, with a total variance of 40.23. Similarly, for the network without uncertainty reward, the average number of decisions is also 13, with a total variance of 74.32. The policy network based on the initial reward has the fewest average decision times and the smallest total variance, indicating the best performance. The policy network based on the reward without coverage and the reward without uncertainty have the same average decision times, but the total variance of the former is smaller, indicating better stability compared to the latter.

To analyze the training results under different reward modules as described above, we recorded the proportion of global rewards attributed to each module throughout the training process. The results are depicted in Figure 13. From the figure, it is evident that during the initial stages of training, the rewards for exploration and uncertainty accounted for a significant proportion. Specifically, coverage rewards constituted 30% of the global rewards, uncertainty rewards comprised 40%, while target discovery rewards represented only 3%. As training progressed, the proportions of coverage and uncertainty rewards decreased somewhat, whereas the proportion of target discovery rewards gradually increased and eventually stabilized around 30%. This indicates that each AUV began to focus its search on areas with a higher probability of target presence.

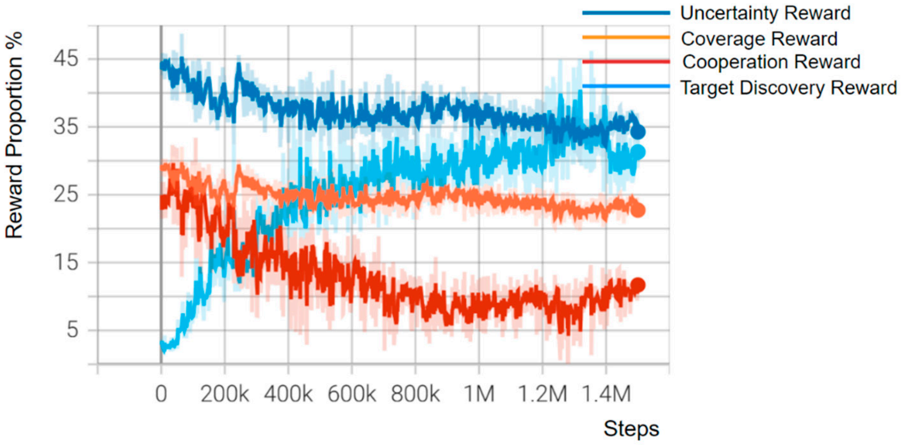


Figure 13. The proportion of global rewards attributed to different reward modules.

7. Conclusions

In this paper, we have transformed the problem of multi-AUV collaborative area search into a two-dimensional plane search problem and modeled the search process as a decentralized partially observable Markov decision process (POMDP). To address the challenge of limited perception capabilities of AUVs in underwater environments, we have constructed search information maps, including coverage maps, uncertainty maps, and target existence probability maps. Each AUV updates these maps through sonar sensing and communication with other AUVs to make action decisions during the search process. Furthermore, we have employed multi-agent reinforcement learning to solve the Markov decision process we established earlier. We first used the classical QMIX algorithm to solve the Markov decision process mentioned above, verifying the feasibility of multi-agent reinforcement learning in the problem of multi-AUV cooperative area search. Subsequently, we further analyzed and specifically designed the improved algorithm SAC-QMIX based on the QMIX algorithm and conducted a comparison between the two algorithms based on simulation test results and training outcomes. Finally, we validated the effectiveness of the established model and multi-agent reinforcement learning method through simulation tests. From the simulation test results, it can be observed that each AUV can complete the search task through action decisions generated by the policy network. Lastly, from the reward training curve, it is evident that the SAC-QMIX algorithm converges faster compared to the QMIX algorithm, but the final absolute performance is the same as the QMIX algorithm. This is a limitation of the SAC-QMIX algorithm and an area for improvement in the future. In the future, we can further research the multi-AUV cooperative area search system from the perspectives of target search time and searching moving targets. Additionally, we can compare the performance of more different algorithms and explore local path planning methods for the execution module in the MACASS.

Supplementary Materials: The following supporting information can be downloaded at https://www.mdpi.com/article/10.3390/jmse12071194/s1, Video S1: Presentation of Multi-AUV Cooperative Area Search Task in Unknown Environments. The latest research progress of the multi AUVs cooperative area search system (MACASS) that we have established has been provided. The video showcased the latest research progress of MACASS that we have established. From the video, it can be seen that each AUV has preliminarily achieved a high area detection rate in completing area search tasks in unknown obstacle environments while ensuring safety conditions.

Author Contributions: Conceptualization, Y.L. and M.M.; methodology, Y.L.; software, M.M.; validation, Y.L., M.M., and D.W.; formal analysis, Y.L.; investigation, M.M.; resources, G.L.; data curation, W.C.; writing—original draft preparation, M.M.; writing—review and editing, Y.L.; visualization, D.W.; supervision, J.C.; project administration, J.C.; funding acquisition, J.C. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financially supported by the National Natural Science Foundation of China (52071099, 52071104) and the National Key Laboratory of Autonomous Marine Vehicle Technology: 2023-SXJQR-SYSJJ01.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data are contained within the article and Supplementary Materials.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A

In Section 4, the significance and explanations of the symbols and abbreviations used in modeling the cooperative area search task are summarized in the following table:

Table A1. The significance and explanations of symbols and abbreviations used in Section 4.

Symbol and Abbreviation	Significance and Explanation
L_x	Length of the search area (m)
L_y	Width of the search area (m)
L_z	Height of the search area (m)
S_i	The *i*-th search subzone
δ_d	Height of the subzone (m)
s_i	Simplified subzone
P_d	Sonar detection probability
P_D	Sonar detection precision
P_F	Sonar’s false alarm probability
D_s	Maximum sonar detection range (m)
G	Primary subregion
$grid(m,n)/g$	The secondary subregion
c_{mn}	Coverage value of grid cell $grid(m,n)$
k_{mn}	Uncertainty value of grid cell $grid(m,n)$
p_{mn}	Target presence probability of grid cell $grid(m,n)$
$\boldsymbol{\eta}_i$	Pose information of the *i*-th AUV
FOV	The effective sonar detection range of each AUV
N_{FOV}	The number of grid cells within the AUV’s sonar detection range
$\boldsymbol{S}$	Global state
$\boldsymbol{O}_i$	Observation for the *i*-th AUV
$\boldsymbol{A}$	Joint actions of AUVs
$\psi_{i,j}$	Relative orientation angle between AUV$_i$ and AUV$_j$
$d_{i,j}$	Distance between AUV$_i$ and AUV$_j$
$\boldsymbol{H}_i$	Relative spatial relationships between AUV$_i$ and the other AUVs
r	Individual reward of AUV
r_c	Coverage reward
r_k	Uncertainty reward
r_t	Target discovery reward
r_a	Collaboration reward

In Section 5, this paper summarizes the significance and explanations of symbols and abbreviations used in the design process of the QMIX algorithm based on the maximum entropy mechanism as shown in the following table.

Table A2. The significance and explanations of symbols and abbreviations used in Section 5.

Symbol and Abbreviation	Significance and Explanation
$\boldsymbol{s}_t$	Global state of the environment at time *t*
$\boldsymbol{a}_t$	Joint action taken by multiple agents at time *t*
$R(\boldsymbol{s}_t, \boldsymbol{a}_t)$	Joint reward received by multiple agents
π_θ	Policy network
Q_ϕ	Value network

Table A2. *Cont.*

Symbol and Abbreviation	Significance and Explanation
$H_i(\pi(\cdot\|\tau_{i,t}))$	The entropy of the action $a_{i,t}$ taken by agent i at time t
$q_i(\tau_i, a_i)$	Action-value function for the i-th agent
F_{mix}	Mixing neural network
$Q_{tot}(\boldsymbol{s}, \boldsymbol{a})$	Joint action-value function
$v_i(\tau_i)$	Individual state-value function
$V_{tot}(\boldsymbol{s})$	Overall state-value function
$Q^{\pi}(\boldsymbol{s}, \boldsymbol{a})$	Action-value function regularized by entropy
$V(\boldsymbol{s})$	Entropy-regularized state-value function
$\overline{V}(s)$	Target state-value function

References

1. Sahoo, A.; Dwivedy, S.K.; Robi, P.S. Advancements in the field of autonomous underwater vehicle. *Ocean Eng.* **2019**, *181*, 145–160. [CrossRef]
2. Hadi, B.; Khosravi, A.; Sarhadi, P. A Review of the Path Planning and Formation Control for Multiple Autonomous Underwater Vehicles. *J. Intell. Robot. Syst.* **2021**, *101*, 67. [CrossRef]
3. Gafurov, S.A.; Klochkov, E.V. Autonomous unmanned underwater vehicles development tendencies. In Proceedings of the 2nd International Conference on Dynamics and Vibroacoustics of Machines (DVM), Samara, Russia, 15–20 September 2015; pp. 141–148.
4. Wang, L.; Zhu, D.; Pang, W.; Zhang, Y. A survey of underwater search for multi-target using Multi-AUV: Task allocation, path planning, and formation control. *Ocean. Eng.* **2023**, *278*, 114393. [CrossRef]
5. Zhang, J.Y.; Ning, X.; Ma, S.C. An improved particle swarm optimization based on age factor for multi-AUV cooperative planning. *Ocean. Eng.* **2023**, *287*, 115753. [CrossRef]
6. Yoon, S.; Qiao, C. Cooperative Search and Survey Using Autonomous Underwater Vehicles (AUVs). *IEEE Trans. Parallel Distrib. Syst.* **2011**, *22*, 364–379. [CrossRef]
7. Healey, A.J. Application of formation control for multi-vehicle robotic minesweeping. In Proceedings of the 40th IEEE Conference on Decision and Control (CDC), Orlando, FL, USA, 4–7 December 2001; pp. 1497–1502.
8. Welling, D.M.; Edwards, D.B. Multiple autonomous underwater crawler control for mine reacquisition. In Proceedings of the ASME International Mechanical Engineering Congress and Exposition, Orlando, FL, USA, 5–11 November 2005; pp. 257–262.
9. Miao, R.; Pang, S.; Jiang, D. Development of an Inexpensive Decentralized Autonomous Aquatic Craft Swarm System for Ocean Exploration. *J. Mar. Sci. Appl.* **2019**, *18*, 343–352. [CrossRef]
10. Hoai An Le, T.; Duc Manh, N.; Tao Pham, D. A DC programming approach for planning a multisensor multizone search for a target. *Comput. Oper. Res.* **2014**, *41*, 231–239. [CrossRef]
11. Yang, Y.; Xiao, Y.; Li, T.S. A Survey of Autonomous Underwater Vehicle Formation: Performance, Formation Control, and Communication Capability. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 815–841. [CrossRef]
12. Yan, Z.P.; Zhang, C.; Tian, W.D.; Zhang, M.Y. Formation trajectory tracking control of discrete-time multi-AUV in a weak communication environment. *Ocean. Eng.* **2022**, *245*, 110495. [CrossRef]
13. Chen, S.; Ho, D.W. Consensus control for multiple AUVs under imperfect information caused by communication faults. *Inf. Sci.* **2016**, *370*, 565–577. [CrossRef]
14. Rashid, T.; Samvelyan, M.; de Witt, C.S.; Farquhar, G.; Foerster, J.; Whiteson, S. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. In Proceedings of the 35th International Conference on Machine Learning (ICML), Stockholm, Sweden, 10–15 July 2018.
15. Guo, F.; Wu, Z. Learning maximum entropy policies with QMIX in cooperative MARL. In Proceedings of the 2nd IEEE International Conference on Electronic Technology, Communication and Information, ICETCI 2022, Changchun, China, 27–29 May 2022; pp. 357–361.
16. Edwards, J.R. Real-time classification of buried targets with teams of unmanned vehicles. In Proceedings of the Ocean's 2002 Conference and Exhibition, Biloxi, MS, USA, 29–31 October 2002; pp. 316–319.
17. Liu, T.-C.; Schmidt, H. AUV-based seabed target detection and tracking. In Proceedings of the Ocean's 2002 Conference and Exhibition, Biloxi, MS, USA, 29–31 October 2002; pp. 474–478.
18. Bovio, E.; Cecchi, D.; Baralli, F. Autonomous underwater vehicles for scientific and naval operations. *Annu. Rev. Control* **2006**, *30*, 117–130. [CrossRef]
19. Schneider, T.; Schmidt, H. Unified command and control for heterogeneous marine sensing networks. *J. Field Robot.* **2010**, *27*, 876–889. [CrossRef]

20. Allotta, B.; Costanzi, R.; Magrini, M.; Monni, N.; Moroni, D.; Pascali, M.A.; Reggiannini, M.; Ridolfi, A.; Salvetti, O.; Tampucci, M. Towards a robust system helping underwater archaeologists through the acquisition of geo-referenced optical and acoustic data. In Proceedings of the 10th International Conference on Computer Vision Systems, ICVS 2015, Copenhagen, Denmark, 6–9 July 2015; pp. 253–262.
21. Tsiogkas, N.; Frost, G.; Monni, N.; Lane, D. Facilitating multi-AUV collaboration for marine archaeology. In Proceedings of the MTS/IEEE OCEANS 2015, Genova, Italy, 18–21 May 2015.
22. Maurelli, F.; Saigol, Z.; Insaurralde, C.C.; Petillot, Y.R.; Lane, D.M. Marine world representation and acoustic communication: Challenges for multi-robot collaboration. In Proceedings of the 2012 IEEE/OES Autonomous Underwater Vehicles, AUV 2012, Southampton, UK, 24–27 September 2012.
23. Li, J.; Li, C.; Zhang, H. Distributed Dynamic Predictive Control for Multi-AUV Target Searching and Hunting in Unknown Environments. *Machines* **2022**, *10*, 366. [CrossRef]
24. Wang, G.; Wei, F.; Jiang, Y.; Zhao, M.; Wang, K.; Qi, H. A Multi-AUV Maritime Target Search Method for Moving and Invisible Objects Based on Multi-Agent Deep Reinforcement Learning. *Sensors* **2022**, *22*, 8562. [CrossRef]
25. Yan, Z.; Liu, W.; Xing, W.; Herrera-Viedma, E. A Multi-Objective Mission Planning Method for AUV Target Search. *J. Mar. Sci. Eng.* **2023**, *11*, 144. [CrossRef]
26. Cai, C.; Chen, J.; Yan, Q.; Liu, F. A Multi-Robot Coverage Path Planning Method for Maritime Search and Rescue Using Multiple AUVs. *Remote Sens.* **2023**, *15*, 93. [CrossRef]
27. Hu, X.; Shi, Y.; Bai, G.; Chen, Y. Collaborative Search and Target Capture of AUV Formations in Obstacle Environments. *Appl. Sci.* **2023**, *13*, 16. [CrossRef]
28. Bai, G.; Chen, Y.; Hu, X.; Shi, Y.; Jiang, W.; Zhang, X. Multi-AUV dynamic trajectory optimization and collaborative search combined with task urgency and energy consumption scheduling in 3-D underwater environment with random ocean currents and uncertain obstacles. *Ocean. Eng.* **2023**, *275*, 113841. [CrossRef]
29. Li, C.; Li, J.; Zhang, G.; Chen, T. IROA-based LDPC-Lévy method for target search of multi AUV-USV system in unknown 3D environment. *Ocean. Eng.* **2023**, *286*, 115648. [CrossRef]
30. Hou, K.; Yang, Y.; Yang, X.; Lai, J. Distributed Cooperative Search Algorithm With Task Assignment and Receding Horizon Predictive Control for Multiple Unmanned Aerial Vehicles. *IEEE Access* **2021**, *9*, 6122–6136. [CrossRef]
31. Fei, B.; Bao, W.; Zhu, X.; Liu, D.; Men, T.; Xiao, Z. Autonomous Cooperative Search Model for Multi-UAV With Limited Communication Network. *IEEE Internet Things J.* **2022**, *9*, 19346–19361. [CrossRef]
32. Xiao, J.; Wang, G.; Zhang, Y.; Cheng, L. A Distributed Multi-Agent Dynamic Area Coverage Algorithm Based on Reinforcement Learning. *IEEE Access* **2020**, *8*, 33511–33521. [CrossRef]
33. Rajnarayan, D.G.; Ghose, D. Multiple Agent Team Theoretic Decision-Making for Searching Unknown Environments. In Proceedings of the 42nd IEEE Conference on Decision and Control, Maui, HI, USA, 9–12 December 2003; pp. 2543–2548.
34. Cao, X.; Sun, H.B.; Jan, G.E. Multi-AUV cooperative target search and tracking in unknown underwater environment. *Ocean Eng.* **2018**, *150*, 1–11. [CrossRef]
35. Liu, Y.; Wang, M.; Su, Z.; Luo, J.; Xie, S.R.; Peng, Y.; Pu, H.Y.; Xie, J.J.; Zhou, R. Multi-AUVs Cooperative Target Search Based on Autonomous Cooperative Search Learning Algorithm. *J. Mar. Sci. Eng.* **2020**, *8*, 843. [CrossRef]
36. Huang, B.; Zhou, B.; Zhang, S.; Zhu, C. Adaptive prescribed performance tracking control for underactuated autonomous underwater vehicles with input quantization. *Ocean. Eng.* **2021**, *221*, 108549. [CrossRef]
37. Zhou, B.; Huang, B.; Su, Y.; Wang, W.; Zhang, E. Two-layer leader-follower optimal affine formation maneuver control for net-worked unmanned surface vessels with input saturations. *Int. J. Robust Nonlinear Control* **2023**, *34*, 3631–3655. [CrossRef]
38. Huang, B.; Zhang, S.; He, Y.; Wang, B.; Deng, Z. Finite-time anti-saturation control for Euler–Lagrange systems with actuator failures. *ISA Trans.* **2022**, *124*, 468–477. [CrossRef]
39. Huang, B.; Song, S.; Zhu, C.; Li, J.; Zhou, B. Finite-time distributed formation control for multiple unmanned surface vehicles with input saturation. *Ocean. Eng.* **2021**, *233*, 109158. [CrossRef]
40. Dorigo, M.; Birattari, M.; Stuetzle, T. Ant colony optimization–Artificial ants as a computational intelligence technique. *IEEE Comput. Intell. Mag.* **2006**, *1*, 28–39. [CrossRef]
41. Liu, C.; Yan, X.; Liu, C.; Wu, H. The Wolf Colony Algorithm and Its Application. *Chin. J. Electron.* **2011**, *20*, 212–216.
42. Haarnoja, T.; Zhou, A.; Abbeel, P.; Levine, S. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In Proceedings of the 35th International Conference on Machine Learning (ICML), Stockholm, Sweden, 10–15 July 2018.
43. Fossen, T.I. *Handbook of Marine Craft Hydrodynamics and Motion Control*; John Wiley and Sons: Hoboken, NJ, USA, 2011.
44. Jia, Q.; Xu, H.; Feng, X.; Gu, H.; Gao, L. Research on cooperative area search of multiple underwater robots based on the prediction of initial target information. *Ocean. Eng.* **2019**, *172*, 660–670. [CrossRef]
45. Shem, A.G.; Mazzuchi, T.A.; Sarkani, S. Addressing uncertainty in UAV navigation decision-making. *IEEE Trans. Aerosp. Electron. Syst.* **2008**, *44*, 295–313. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.