Journal of
Marine Science and Engineering

MDPI

Article

Hybrid Partition-Based Patrolling Scheme for Maritime Area Patrol with Multiple Cooperative Unmanned Surface Vehicles

Jiajia Xie [1], Rui Zhou [1], Jun Luo [1,2], Yan Peng [1,*], Yuan Liu [1], Shaorong Xie [1] and Huayan Pu [1]

[1] School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200444, China; bigjjia@shu.edu.cn (J.X.); zr901122@sina.com (R.Z.); luojun@shu.edu.cn (J.L.); liuyuanji@shu.edu.cn (Y.L.); srxie@shu.edu.cn (S.X.); phygood_2001@shu.edu.cn (H.P.)

[2] State Key Laboratory of Mechanical Transmission, Chongqing University, Chongqing 400044, China

[*] Correspondence: pengyan@shu.edu.cn

Received: 27 October 2020; Accepted: 17 November 2020; Published: 18 November 2020

Abstract: Multi-robot cooperative patrolling systems have been extensively employed in the civilian and military fields, including monitoring forest fires, marine search-and-rescue, and area patrol. Multi-robot area patrol problems refer to the activity that a team of robots works cooperatively and regularly to visit the key targets in the given area for security. Following consideration of the low cost and high safety of unmanned surface vehicles (USV), a team of USVs is organized to perform area patrol in a sophisticated maritime environment. In this paper, we establish a mathematical model considering the characteristics of the cooperative patrol task and the limited conditions of USVs. A hybrid partition-based patrolling scheme is proposed for a multi-USV system to visit targets with different importance levels in a maritime area. Firstly, a centralized area partition algorithm is utilized to partition the patrolling area according to the number of USVs. Secondly, a distributed path planning algorithm is applied to planning the patrolling path for each USV to visit the targets in a maritime environment to minimize the length of the patrolling path for the USV team. Finally, comparative experiments between the proposed scheme and other methods are carried out to validate the performance of the hybrid partition-based patrolling scheme. Simulation results and experimental analysis show the efficiency of the proposed hybrid partition-based patrolling scheme compared to several previous patrolling algorithms.

Keywords: unmanned surface vehicles; hybrid partition-based patrolling scheme; centralized area partition; distributed path planning; multi-USV system

1. Introduction

With significantly increasing dependence on marine resources and space, unmanned surface vehicle (USV) has been widely applied as a small, flexible, and high-speed maritime vehicle for diverse marine missions. Compared with a single USV, multiple USV system has higher task execution efficiency, higher flexibility, and maneuverability. With abundant sensors, such as differential global positioning system (GPS), inertial navigation, water quality monitor, camera, sonar, multiple coordinated USVs systems can perform hazardous marine missions and provide comprehensive environmental information to detect and monitor the particular area. Therefore, cooperation of multiple USVs is an essential research and development direction. In this paper, we aim to constitute manned vehicles with multiple cooperative USVs to accomplish repetitive, redundancy, and dangerous maritime area patrol tasks. The team of USVs needs to keep moving in the given area and visit the essential targets at regular intervals for area security. The challenge of multiple USVs patrol tasks for area patrol is how to coordinate them to visit all places with the shortest patrolling paths more frequently. In other words, how to assign targets to each USV and design a patrolling path for each USV is very critical [1].

Classic patrol tasks are divided into two categories. One is that all agents need to visit all vertices in the designated work region [2–5]. The other one is that the work region is partitioned into several sub-regions [6–9], and the agents are allocated to work only in one of the sub-regions without dynamic regional deployment. In the first type, it is straightforward to cause conflicts and collisions in path planning, resulting in unexpected collective behaviors and affecting overall performance. In the second type, a centralized unit performs partitioning before agents are assigned to one of the sub-regions to perform regular partitioned-based patrol tasks (i.e., patrolling path planning), which is easy to suffer low robustness when some emergencies occur. What is more serious is that a problem with the centralized unit can make the entire multi-agent patrolling system collapse.

In this paper, to overcome the problems concluded above, a hybrid partition-based patrolling scheme for multi-USV system to keep the protected area's safe is proposed. Multiple USVs patrolling planning is as follows. Firstly, an area partition algorithm is designed to partition the area according to the number of USVs and the importance levels of targets [10] in the area. Each USV is assigned to one of the partitions. Once one of the USVs suffers failure or breakdown, area re-partition will be executed to regenerate partitions and rearrange USVs in the new partitions. Secondly, a distributed path planning algorithm is proposed to plan an optimal patrolling path for each USV for global cooperative patrolling. Finally, various comparable experiments are designed to verify the performance of the proposed scheme. Simulation results demonstrate that the proposed scheme can accomplish the area patrol in the maritime environment effectively.

The main contributions of this paper are as follows:

- We present a centralized area partition algorithm based on the importance level, which can reduce conflict during patrolling. We divide the important targets with different importance levels into different partitions and assign USVs to work only in one of the partitions.
- We propose a distributed path planning algorithm for the multi-USV system to plan optimal paths to minimize the total length of the patrolling path in all partitions.
- We evaluate the proposed patrolling scheme with extensive simulations. The simulation results demonstrate the effectiveness of patrolling partition and patrolling path planning.

The remainder of this paper's organization is shown as follows: Section 2 reviews related representative works. Section 3 presents the system model. Section 4 provides an accurate analysis of our patrolling scheme. Section 5 shows experiment settings and discusses simulation results. Section 6 concludes this paper.

2. Related Works

In this section, we review related works about patrolling algorithms for multiple cooperative agents.

Various approaches were used to perform a multi-agent patrol task. Lin et al. [2] adopted a coordinated auction system to solve the patrolling problem. Each robot chooses the points they expect to visit by a coordinated auction system and improves the team's collective performance by continuous re-auction. Poulet et al. [3] proposed two different distributed, coordinated, and auction-based policies based on different social criteria. Othmani-Guibourg et al. [4] evaluated patrolling policies in a dynamic environment with an edge-Markovian evolving graph. A central coordinator is used to assign new goal node to the robots after the designed path is unavailable. Yan et al. [5] designed a frequency-based patrol algorithm. They utilized shared information to calculate expected idleness to solve decision conflicts. Fitzpatrick et al. [11] addressed a perimeter patrol problem with heterogeneous agents involving unmanned ground stations and unmanned air vehicles and presented an optimization approach built on max-plus probability. Farinelli et al. [12] presented a greedy baseline approach and a market-basedapproach based on sequential auction algorithms to tackle on-line coordination in the multi-robot system.

Different from the above works, multi-agent patrolling algorithms based on partition has been extensively developed. Sea et al. [7] improved the traditional k-means clustering algorithm based on non-uniform visiting frequency and workload to achieve graph partition and subgraph patrolling. Sugiyama et al. [13] proposed a negotiation-based learning method to address the multi-robots patrol problem. Wiandt et al. [14] proposed a self-organized partition algorithm to achieve distributed partition, in which robots transmit information by broadcasting to other robots without a third-party coordinator. Portugal et al. [1] thought that the local patrolling path in graph partitions could be improved by minimizing the longest local path. Sea et al. [15,16] presented a decentralized partitioning method for cleaning tasks of multiple agents under the grid environment. Sugiyama et al. [17] designed different visit frequencies for all purposes in the patrolling area and applied divisional cooperation to achieve the patrol tasks. Yasuyuki et al. [18] applied reinforcement learning based on discrete patrolling areas to perform security patrol tasks.

Besides, many optimization criteria have been designed to evaluate the performance of patrolling policies, such as minimum total patrolling path length [19], minimum total voyage time [20], the revisit time interval between two consecutive visits to each target [21], minimum the commute time between all pairs of patrolling points [22], and defined reward function [23,24].

3. System Models

This section analyzes the patrolling environment model, the USV model, and the patrol task goal of maritime cooperative area patrol, as shown in Figure 1. The main parameters in this paper are shown in Table 1.

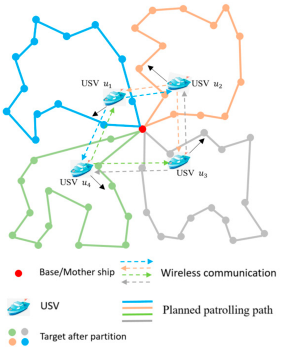


Figure 1. The system model of a multi-unmanned surface vehicle (USV) system considering no limit of maximum range. A team of USVs is applied to perform area patrol by dividing several important targets (colored solid circle) in the maritime area into different partitions and assigning a USV to visit the targets in the partition.

3.1. Patrolling Environment Model

We consider a maritime cooperative patrol task for a team of N USVs, $\mathrm{U} = \{u_1, u_2, \ldots, u_N\}$, as shown in Figure 1. A mother ship v_0 stops at the center of the given area and works as the USV base with the coordinate (x_0, y_0), i.e., departure point and home point of a patrol. A set of targets $V = \{v_1, v_2, \ldots, v_M\}$ are statically located in the given area $P \subset R^2$ in a maritime environment, where M is the number of targets. The coordinates of Target v_i is (x_i, y_i). All targets are distributed in the area with the mother ship as the center and R_0 as the radius. Multiple USVs are applied to perform area patrol by visiting these important targets in the area at the required frequency or regular intervals. Due to the characteristics of a realistic patrol mission, the targets are assigned importance levels that indicate its importance. The higher the target's importance, the higher the importance level, and the higher the frequency of visits. Each target v_m is assigned an importance level $w_m \in \{1, 2, \ldots, M_w\}$, i.e., the number of visits in patrol task, where M_w is the highest importance level. Inspired by the high degree of autonomy, independent processing of information and decision-making, and good fault tolerance, partition patrolling is introduced to a multi-USV cooperative patrolling system. According to the targets with different importance levels, we divide the area into N different disjoint partitions, $P = P_1 \cup P_2 \cup \ldots \cup P_N$ and $P_i \cap P_j = \varnothing, \forall 1 \leq i, j \leq N, i \neq j$ (see details in Section 4.2).$|P_i|$ is the number of targets in the i -th partition P_i. Then, allocate USV u_n to partition P_n.

Table 1. The summary of the main parameters.

Symbol	Description
P	The given area for patrolling
U	The set of USVs
N	The number of USVs
V	The set of targets in the given area P
M	The number of targets
(x_0, y_0)	The coordinate of the mother ship v_0
(x_m, y_m)	The coordinate of the m-th target v_m
w_m	The importance level of m-th target v_m
w_{P_n}	The total number of visits in the n-th partition P_n
M_w	The highest level of importance of targets
$d_{i,j}$	The straight sailing distance from v_i to v_j
$L_{nmax}(u_n)$	The maximum range of the n-th USV u_n
P'_n	The augmented partition of the n-th partition P_n
T_{P_n}	The expected task load of the n-th partition P_n
T_d	The difference in task load amongst all USVs
$L_{P_n}(l)$	The path of the l-th patrol in the n-th partition P_n
L_{P_n}	The total patrolling path of USV u_n in partition P_n
$X^l_{i,j}$	A binary coefficient
$F(P_n)$	The objective function

3.2. USV Model

A small team of USVs is deployed in the patrolling partition. The divisional team of USVs $\text{U} = \{u_1, u_2, \ldots, u_N\}$ is deployed to perform the patrol tasks, where N is the total number of USVs in the given patrolling area. The maximum range of the n-th USV u_n is represented as $L_{nmax}(u_n)$, since the diesel oil is finite. We assume that all targets are within the range of USV's maximum range, then we get $R_0 \leq \frac{L_{nmax}(u_n)}{2}$. Each USV is assigned to an individual partition to patrol all targets within the partition. The task of each USV u_n is to patrol the targets in the corresponding partition P_n under the limit of its maximum range. For this reason, starting from the mother ship v_0 to patrol the partition P_n once, the USV u_n may only be able to visit a part of the targets. Hence, each USV may patrol more than once to perform the required number of visits for the targets. A distributed patrolling path planning algorithm is presented to form the patrol path L_{P_n} for all the USVs, including several single patrolling paths $L_{P_n}(l)$.

Once one or some of the USVs suffers a breakdown, the corresponding partitions they are responsible for will be undertaken by other USVs which have completed the partition patrol task first, and the maritime patrolling area would be re-partitioned if necessary, and then the patrolling path of each USV is re-planned.

3.3. Patrol Task Goal

After partitioning patrolling targets to balance the workload, each USV u_n may start from the mother ship v_0 to patrol the partition more than once to accomplish each target's required number of visits because of the limited maximum range. The patrol task goal for each USV u_n is to minimize the sum of the length of patrolling path of several patrols, $len(L_{P_n})$.

4. Proposed Scheme

This section introduces the problem formulation for a cooperative patrol task. It is followed by a hybrid structure that is used in the designed patrolling algorithm for the multi-USV system, that is, centralized partition and distributed execution. Before the USV carries out the patrolling task, a centralized partition algorithm based on the importance level is conducted in the base station.

After that, the distributed patrolling path planning based on improved particle swarm optimization (PSO) is planned on each USV.

4.1. Problem Formulation

In real-world patrol missions, the more important target should be patrolled more frequently. We consider the maritime patrol task as patrolling targets with different importance levels that the target with higher importance levels should be visited more times. Hence, we try to balance the task load amongst all the USVs by allocating the targets reasonably and minimize the sum of the length of the patrolling path for each USV.

(1) Total weight, w_{P_n}: Total weight are the sum of the weight of targets patrolled by a USV in the partition P_n, which mean the sum of the required visits of all targets, defined as:

$$W_{P_n} = \sum_{m=1}^{|P_n|} w_m,\ \forall v_m \in P_n, \tag{1}$$

where $m \in \{1, 2, \ldots, |P_n|\}$, w_m is the importance level of the target v_m, which represents the different required number of visits required by the target v_m. w_m is used as the weight of the target v_m, and W_{P_n} is the sum of the weights of all targets in the individual partition, which is used as the constraint in Equation (7). We aim to optimize the patrol path while meeting the number of visits required by every target.

(2) The expected task load in a partition, T_{P_n}: Targets' importance levels and straight distance between each pair of targets are considered to evaluate the task load of the USV working in the partition, which is defined as the expected path length 6 that the USV has to travel to get the optimal patrolling path:

$$T_{P_n} = \sum_{v_m, v_j \in P_n} \frac{w_m \| v_m - v_j \|}{W_{P_n} - w_m}, \tag{2}$$

where $|P_n|$ is the number of targets in each partition P_n. $\|v_m - v_j\|$ denotes the Euclidean distance between the target v_m and target v_j.

The difference T_d 6 in task load amongst all USVs is introduced to balance the task load in different partitions with threshold T_h:

$$T_d = \frac{1}{K(K-1)} \sum_{i=1}^{K} \sum_{j=1}^{K} \left| T_{P_i} - T_{P_j} \right|, i \neq j, \tag{3}$$

$$T_d \leq T_h. \tag{4}$$

(3) Sailing distance, $d_{i,j}$: A USV moves from target v_i to target v_j, the optimal path is the straight path between this pair of targets $\left(v_i, v_j\right)$, which is defined as the Euclidean distance between these two targets.

$$d_{i,j} = \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2},\ \forall v_i, v_j \in P_n. \tag{5}$$

(4) Single patrolling path, $L_{P_n}(l)$: Limited by the maximum range of the USV u_n, it will take multiple times to patrol these targets in its partition. Every patrol starts from the mother ship, visits a part of the targets in the partition, and returns to the mother ship. The length of a single patrolling path is calculated by:

$$len(L_{P_n}(l)) = \sum_{i}^{|P_n|} \sum_{j}^{|P_n|} d_{i,j} X_{i,j}^{l},\ i \neq j, \forall l \in \{1, 2, \ldots, L\}, \tag{6}$$

$$1 \leq L \leq \left| W_{P_n} \right|, \tag{7}$$

where $X_{i,j}^{l}$ is a binary coefficient. $X_{i,j}^{l} = 1$ states clearly that a USV moves from v_i to v_j in the l-th patrol, whereas $X_{i,j}^{l} = 0$ is opposites. L indicates the number of patrols, which is considered that the maximum range of the unmanned ship is enough to cover the round-trip voyage from the base station to the farthest patrol target. $L_{P_n}(l)$ is a sequence set of targets visited in l-th patrol.

(5) Partition patrolling path, L_{P_n}: As mentioned above, the USV may patrol the partition L times to perform patrol tasks. When the USV returns to the mother ship for the last time, it indicates that all the targets have been visited with the required times. Therefore, the partition patrol task is finished, and the total length of the partition patrolling path is defined as:

$$len(L_{P_n}) = \sum\nolimits_{l=1}^{L} len(L_{P_n}(l)),\ \forall l \in \{1, 2, \ldots, L\}, \tag{8}$$

(6) Objective function, $F(P_n)$: In the maritime area patrol task, we aim to find a minimum patrolling path for all USVs to complete the partition-based patrol tasks within their respective partitions. Therefore, the objective function is denoted as:

$$F(P_n) = \min(len(L_{P_n})), \tag{9}$$

Subject to:

$$len(L_{P_n}(l)) \le L_{nmax}(u_n), \tag{10}$$

$$\sum\nolimits_{j=1}^{|P_n|} X_{0,j}^{l} = 1, \forall v_j \in P_n, \forall l \in \{1, 2, \ldots, L\}, \tag{11}$$

$$\sum\nolimits_{j=1}^{|P_n|} X_{j,0}^{l} = 1, \forall v_j \in P_n, \forall l \in \{1, 2, \ldots, L\}, \tag{12}$$

$$\sum\nolimits_{i=0}^{|P_n|} X_{i,j}^{l} - \sum\nolimits_{p=0}^{|P_n|} X_{j,p}^{l} = 0,\ \forall v_{j \in} L_{P_n}(l), \tag{13}$$

$$\sum\nolimits_{l=1}^{L} \sum\nolimits_{i=0}^{|P_n|} X_{i,m}^{l} = w_m,\ \forall l \in \{1, 2, \ldots, L\}, \tag{14}$$

where $L_{nmax}(u_n)$ indicates the maximum range of the n-th USV u_n. Thus, the length of every patrol path is limited by the USV's maximum range, as shown in Equation (10). They have to return to the mother ship and recharge before the next patrol. In every patrol, the constraint expressed by Equations (11) and (12) ensure that the USV starts from the mother ship before patrolling and return to the mother ship after completing a patrol. The constraint in Equation (13) guarantees that the number of arrivals and departures at a target are the same. Moreover, Equation (14) guarantees that all the targets are visited with the required number of visits.

4.2. Partition Algorithm Based on the Importance Level

For reducing task conflicts between USVs and allocating task to USVs reasonably and evenly, we divide the targets with different importance level into several disjoint partitions $P = P_1 \cup P_2 \cup \ldots \cup P_N$ according to the number of USVs, N. Then, we generate the shortest patrolling paths for each USV in its dispatched partitions. We introduce the main idea of the k-mean method to perform the target partition. Based on the multi-USV patrol task characteristics, we propose partition patrolling algorithms based on the importance level. The traditional k-means clustering algorithm divides the data into several categories based on the similarity of features. Our proposed algorithm improved the traditional k-mean method with a new calculation of centroids and new cost function due to the different importance levels of targets. The cost function of the partition algorithm is designed as:

$$J = \min \sum\nolimits_{n=1}^{N} \sum\nolimits_{v_m \in P_n} w_m \|v_m - p_n\|, \tag{15}$$

Subject to:

$$P_1 \cup P_2 \cup \ldots \cup P_N = P,$$

$$P_i \cap P_j = \varnothing,\ \forall 1 \le i, j \le N, i \ne j,$$

$$p_n = \frac{\sum_{v_i \in P_n} w_i v_i}{\sum_{v_i \in P_n} w_i}, \tag{16}$$

where n is the number of partitions, and p_n is the partition centroid.

The details of the proposed partition algorithm based on importance levels and the k-mean method are shown in Algorithm 1. The input of Algorithm 1 is the set of targets $V = \{v_1, v_2, \ldots, v_M\}$, the set of targets' importance levels $w = \{w_1, w_2, \ldots w_M\}$, and the number of partitions N. The output is the optimal partition $P^* = P_1 \cup P_2 \cup \ldots \cup P_N$. As is well-known, the initial centroids of the k-mean method have a significant influence on the final classification results. Therefore, we run the partition algorithm many times with different initial centroids (lines 1–2). To rationalize the classification results, the traditional centroids update is replaced by the weighted center of targets in the same partition to balance the task load amongst USVs, repeatedly until the method converges (lines 3–6). Subsequently, the expected task load and the task load difference are calculated for the partition (lines 8–9). The partitions that meet the conditions are saved in partition buffer (lines 10–11). Finally, the partition $P^* = P_1 \cup P_2 \cup \ldots \cup P_N$ with the lowest cost J^* is output as the optimal partition (line 14).

Algorithm 1: Partition algorithm based on the importance level

```
Input   Targets $V = \{v_1, v_2, \ldots, v_M\}$
Input   Targets' importance levels $w = \{w\}_1, w_2, \ldots w_M$
Input   Number of partitions $N$
Output  Partition $P^* = P_1 \cup P_2 \cup \ldots \cup P_N$
1: for iteration = 1 to $T$ do
2:     Randomly Choose $N$ targets as the initial centroids
3:     repeat
4:     For each target, select the nearest centroid as its partition
5:     Reset the centroids to the weighted center of all targets in each partition by (16)
6:     until convergence
7:     for $n$ = 1 to $N$ do
8:        Calculate the expected task load, $T_{P_n}$ by (2)
9:        Calculate the difference in task load amongst all USVs, $T_d$ by (3)
10:    if $T_d \le T_h$ then
11:           Save partitions and the corresponding cost $J$ by (15) in partition buffer
12:    end
13: end
14: Pick the partition $P^* = P_1 \cup P_2 \cup \ldots \cup P_N$ that has the lowest cost $J^*$
```

4.3. Patrolling Path Planning Algorithm Based on Improved PSO

When determining the results of partitions according to the USV team size and the balance of workload of all USVs, a patrolling path planning algorithm based on particle swarm optimization (PSO).

PSO designs a kind of massless particles to simulate the movement of birds in the flock. The particles have only two important terms: velocity and position. The velocity refers to the speed of moving, and the position refers to the direction of moving. Each particle searches for the optimal solution individually in the solution space and stores it as the current local optimal individual, and gain the best solutions by sharing the local optimal individual in the particle swarm and selecting the best individuals from all the particles. All particles in the particle swarm adjust their optimization direction based on individual optimal experience and collective optimal experience.

We design a PSO for the path problem in area patrol based on the partition patrolling algorithm proposed above to solve the area patrol task.

In the partition P_n, w_{P_n} is the total number of visits to all the targets. Each particle i has a position X_i, velocity V_i and previous local optimal position $pBest_i$. The particle swarm has a previous global optimal position $gBest$.

$$psize_n = w_{P_n} + 1, \tag{17}$$

$$X_i = \left[X_i^1, X_i^2, \ldots, X_i^{psize_n}\right], \tag{18}$$

$$V_i = \left[V_i^1, V_i^2, \ldots, V_i^{psize_n}\right], \tag{19}$$

$$pBest_i = \left[XpBest_i^1, pBest_i^2, \ldots, pBest_i^{psize_n}\right], \tag{20}$$

$$gBest = \left[gBest^1, gBest^2, \ldots, gBest^{psize_n}\right], \tag{21}$$

where X_i is a visited sequence of all targets in the partition P_n and the base station with a length of $psize_n$, including all the targets with the required number of visits. X_i^d is a column vector, containing information about the last visited target before visiting the target d.

In this paper, PSO is initialized as solutions that satisfy the requirement but maybe not the optimal. Then the optimal solution is found by iteration. In each iteration, particles update themselves by adopting the best experiences $pBest_i$ and $gBest$. Based on the local optimal solution and global optimal solution, the particle optimizes its velocity and position according to the following rules:

$$V_i' = wV_i + c_1 \times rand_1 \times (pBest_i - X_i) + c_2 \times rand_2 * (gBest - X_i), \tag{22}$$

$$w = \frac{(w_{ini} - w_{end})(G_{max} - G_i)}{G} + w_{end} \tag{23}$$

where c_1 and c_2 are the learning factors. $rand_1 \in [0,\ 1]$ and $rand_2 \in [0,\ 1]$ are random numbers used to enhance the randomness of the search. w is the inertia weight whose value is nonnegative, used to adjust the search range of the solution space. We introduce the linearly decreasing weight policy to dynamically adjust w, leading to a better solution by dynamically adjusting global and local search capabilities. w_{ini} and w_{end} are the initial inertia weight and the terminal inertia weight when the optimization process iterates to the maximum number of iterations G_{max}. G_i is the current iteration.

Then, we optimize the position of each particle. Suppose the current visiting target is d, the next target to be visited is n, which is generated by the following sets: feasible velocity F_V, feasible path F_P, and feasible target F_X.

$$A = \left\{n \middle| V_{(d,n)} > Pr_v \text{ and } (d,n) \epsilon V_i' \text{ and } (d,n)\ s.t.\Omega_{cs}\right\}, \tag{24}$$

$$F_V = randperm(A), \tag{25}$$

$$F_P = \{n | (d,n) \epsilon X_i \text{ and } (d,n)\ s.t.\Omega_{cs}\}, \tag{26}$$

$$F_X = \{n | (d,n) \epsilon P_n \text{ and } (d,n)\ s.t.\Omega_{cs}\}, \tag{27}$$

First, we randomly generate a feasible patrolling path. Then, we generate a new solution for the area patrol problem from the three feasible sets above. We generate a velocity set A by velocity cut for the updated velocity V_i' with a probability Pr_v and shuffle the elements in A to get a feasible velocity F_V, which meets the constraint set Ω_{cs} containing the constraints in Equations (10)–(14). The purpose of probabilistic constraints Pr_v d scrambling operations *randperm* is to increase the diversity of velocities. The feasible path F_P contains the targets available in the last position of the particle satisfying the constraints in Ω_{cs}, while the feasible target F_X contains all the targets. We select a feasible target from the set F_V, if not, from the set F_P, otherwise, from the set F_X. Due to the maximum range of the USV, when there is no suitable next target to visit, the USV returns to the base station, ends the current patrol, replenishes fuel, and starts the next patrol from the base station again. Therefore, we can obtain a new feasible solution.

Finally, we design a fitness function $fitness(X_i)$ to evaluate the performance of each particle i and deal with the patrol task goal of area patrol. The fitness function is designed as the total length of the patrolling path, calculated as follows:

$$fitness(X_i) = len(X_i). \tag{28}$$

Algorithm 2 shows the procedure of the patrolling path generated by PSO. First, a population of particles is initialized according to the targets in the partition, and the local optimal positions (i.e., the local optimal patrolling path)and the global optimal position (i.e., the global optimal patrolling path) are initialized (lines 2–6). Each particle has its initial position and velocity. Particle position indicates a possible feasible solution to the area patrol problem. In each iteration, the velocities of particles are updated according to their own optimal experience and neighboring optimal experience (line 9). The improved positions of particles are calculated based on the velocity update, which are full patrolling paths for targets in a partition. First, initialize the path length and set the starting point of the path as the base station (lines 10–11). Then, according to the constraints in the Equations (24)–(27), the next accessible patrol target is selected in turn until the predefined number of visits to all targets is completed as required. Finally, a new solution $X_i = \left[X_i^1, X_i^2, \ldots, X_i^{psize_n}\right]$ for the area patrol problem is generated for particle *i* (lines 12–15). PSO uses a fitness function to evaluate the quality of particles (line 16). The global optimal position of the particle swarm is updated iteratively, which is iterated until the maximum number of iterations (lines 17–18).

Algorithm 2: Patrolling path planning algorithm based on improved PSO

```
Input Partition $P_n$
Output the optimal position and $gcost$
1:for m = 1 to M
2:    for i = 1 to $psize_n$ do
3:      Initialize velocity $V_i$ and position $X_i$ for particle $i$
4:      Set $pBest_i = X_i$
5:    end
6:    $gBest = \min\{pBest_i\}$
7:    for iteration = 1 to N do
8:      for i = 1 to $psize_n$ do
9:         Update the velocity for particle $i$ by (22)-(23)
10:          $patrol_{num} = 0$
11:          $d = 0$
12:          while $(patrol_{num} < psize_n)$
13:             Generate the next target $k$ of target $d$ by (24)-(27)
14:             $X_i^k = [\langle d, k \rangle]$
15:          end
16:          Calculate the fitness of particle $i$
17:         $pBest_{i+1} = \min(pBest_i, X_i)$
18:          $gBest = \min(pBest_i, gBest)$
19:      end
20:    end
21:end
```

5. Experiment and Evaluation

In this section, we design experiments to evaluate the proposed scheme and analyze the simulation results.

5.1. Experimental Setup

In the experimental cooperative control scenario, we consider a 100 km × 100 km ocean area, where M targets are generated and located in the area randomly. Each target v_m is randomly assigned an importance level $w_m \in \{1, 2, 3\}$, which indicates visiting frequency to the target in the patrol task. A mother ship is in the middle of the area, as the recharge base station. N USVs are dispatched from the base to patrol the targets in the area. Considering that the fuel of every USV is limited, the limit of USV's maximum range $L_{nmax}(u_n)$ is set for each USV, that is, before the fuel is exhausted, the USV must return to the base station and replenish fuel, and then continue to visit the area to be visited. The size of targets to be patrolled is selected from the set {100, 200, 400}, while the USV team with different size is used to perform the patrol task, that is 2, 4, and 8, respectively.

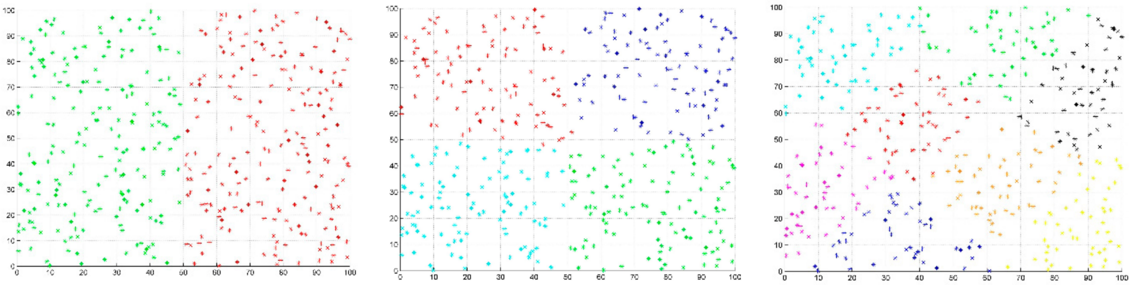
Besides, we also consider that it is essential that each target is not allowed to be visited twice in succession. The path length of each patrol of the USV is limited by the maximum range, that is, to guarantee that the USV returns to the base station before the fuel is exhausted.

To evaluate the performance of the proposed scheme, we conduct comparative experiments with Weighted Targets Sweep Coverage (WTSC), comprehensive learning particle swarm optimization (CLPSO), and random method (RM).

- WTSC: In this method, the patrolling path is generated by a centralized planning algorithm, that is, the WTSC proposed in [25].
- CLPSO: In this method, the patrolling path is generated based on the velocity and position update of CLPSO proposed in [26].
- The random method (RM): In this method, USV randomly selects an available target to visit.

5.2. Results Analysis

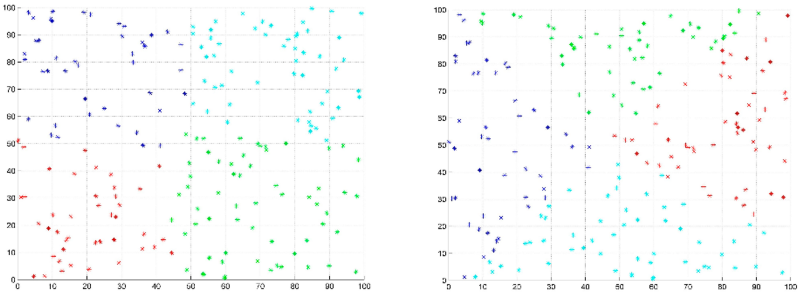
We evaluate the proposed partition algorithm based on the importance level with different sizes of targets and the different sizes of the USV team. Figure 2 shows the partition results of 400 patrolling targets with different USV team sizes and different importance level settings in the given ocean area. Figure 2a–c respectively represent the partition results of the 400 target evenly allocated to two, four, and eight USVs under the first importance level setting $w_m \in \{1,2,3\}$. The dots in Figure 2 represent the targets in the patrolling area. The targets marked with the same color belong to the same partition and are patrolled by the same USV.



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

Figure 2. The partition result of 400 targets allocated to the USV team of different sizes with different importance level settings: (**a**) 400 targets divided into two partitions for two USVs with; (**b**) 400 targets divided into four partitions for four USVs; (**c**) 400 targets divided into eight partitions for eight USVs.

Figure 3a,b show the partition results obtained after two kinds of importance level settings. Figure 3a shows the partition results of targets separately and randomly weighted by an importance level $w_m \in \{1,2,3\}$, while Figure 3b shows the results with an importance level $w'_m \in \{1,2\}$. It can be seen that when the importance levels of targets are different, the partition results generated by the partition algorithm are also different.



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

Figure 3. The partition result of 200 targets with different importance level settings allocated to four USVs: (**a**) 200 targets with an importance level $w_m \in \{1, 2, 3\}$ divided into four partitions for four USVs; (**b**) 200 targets with an importance level $w'_m \in \{1, 2\}$ divided into four partitions for four USVs.

Figure 4 shows the workload on each partition when the different sizes of targets are split into eight partitions according to the USV team's size. We can see that the workload on each partition is balanced, while the workload of each USV increases with the increase of the targets to be patrolled. Therefore, the proposed partition algorithm can divide the patrolling area into effectively-balanced partitions.

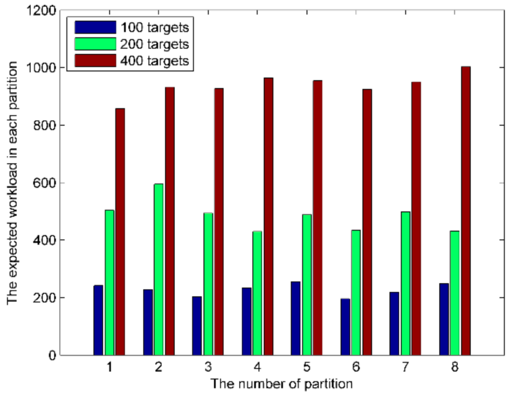
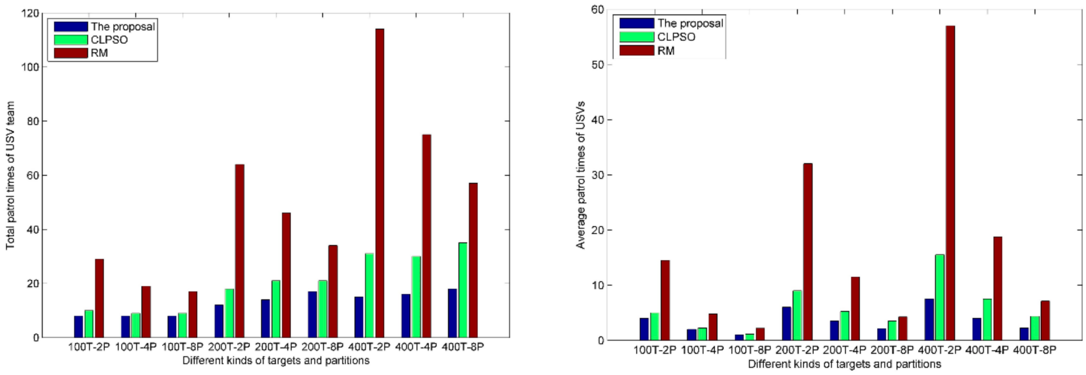


Figure 4. The workload on each partition when different numbers of targets are divided into eight partitions.

We compare the performance of the proposed partition-based patrolling algorithm with different methods. Figure 5 shows the total and average number of patrols with different size of USVs and the different size of targets in the ocean area, when the USVs are all limited by the maximum range $L_{nmax}(u_n) = 300$ km. In Figure 5, 100T-2P means 100 targets divided into two partitions allocated to two USVs. Figure 5 shows that the total number of patrols and the average number of patrols increase with the increase of targets to be patrolled. Besides, the limit of the maximum range also increases the number of times for USVs to return to the base station for refueling. The patrolling path generated by the proposed scheme is better than in other methods, which means the fewer number of total and average patrols for the USV team. The proposed scheme can achieve the best performance compared with other methods. As for the CLPSO [26], the velocity update of the particle only learns from its individual optimal experience with a certain probability. When the algorithm reaches the preset maximum number of iterations, it does not get a very good result, that is, its velocity update and position update methods are slightly poor. As for the RM, the points in the patrolling path are chosen randomly, which results in the high total and the average number of patrols for the USV team.



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

Figure 5. The total and the average number of patrols with different sizes of USVs and the different sizes of targets in the ocean area, limited by the maximum range: (**a**) Total number of patrols of the USV team; (**b**) Average number of patrols of the USV team.

Figure 6 shows the comparison of patrolling path length for USVs in each partition by the proposed scheme and other methods when 400 targets are divided into four partitions. We note that the proposed PSO-based patrolling path planning algorithm can better perform the patrol task in each partition with the predefined number of visits, which results in a shorter total length of the patrolling path than other methods [26].

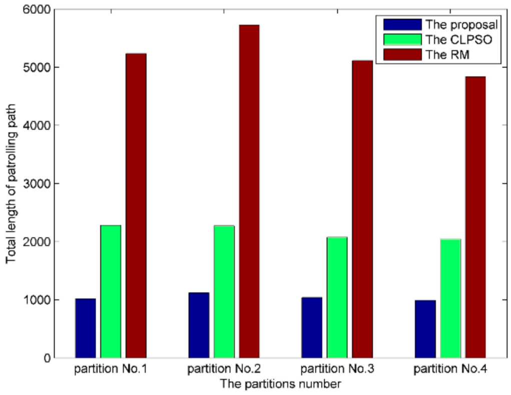


Figure 6. Comparison of patrolling path length for USVs in each partition by different patrolling planning algorithms when 400 targets are divided into four partitions.

Figure 7 shows the average patrolling length with different sizes of targets and fixed size of the USV team, not limited by maximum range, where the size of targets, M is 100, 200, 400, and the USV team's size, N is 8. The USV can dispatch from the base station, complete the patrol task in its partition with the predefined number of visits. It can be seen that the proposed scheme can achieve a shorter average patrolling length than WTSC [25] and CLPSO [26], which means the proposed scheme can obtain better patrolling paths with the minimum length compared with other schemes. The reason is that the proposed scheme can provide optimal patrolling path based on the local optimal experience and global optimal experience, whereas the WTSC and CLPSO are planning based on the local optimal operation.

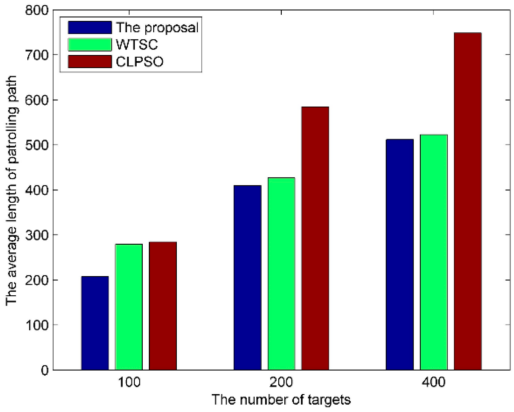


Figure 7. The average patrolling length with different sizes of targets and fixed size of the USV team, not limited by the maximum range.

On the contrary, Figure 8 shows the average patrolling length with different size of targets and eight USVs, limited by maximum range, $L_{nmax}(u_n) = 300$ km. The proposed scheme designs a new improved PSO to solve the combinational optimization problem modeled in Section 4.1. The goal of patrolling planning is to minimize the global patrolling path. We can see that the proposed scheme can obtain the best performance for partition patrolling under the limit of the maximum range of USVs. The average length of patrolling paths in the proposed scheme is shorter. The reason is that the proposed scheme can provide a better patrolling path through the centralized partition and distributed patrol planning. As for WTSC [25], a centralized planning method is adopted to assign the local optimal targets in a given area to each unmanned aerial vehicle in turn. The CLPSO [26] updates the velocity by learning from its own best experience without the global best experience of its neighbors. Besides, it can be observed that the average patrolling length with a maximum range limit is longer than that without a maximum range shown in Figure 7. The main reason is that when limited by a maximum range limit, to ensure that each USV can return smoothly after each patrol, each USV tries to visit more targets within the range and returns to the base station before the fuel is exhausted. Each USV patrols its partition repeatedly like this until all the targets are visited with predefined visits. In case of no limit of maximum range, USVs do not have to return to the base station for refueling many times.

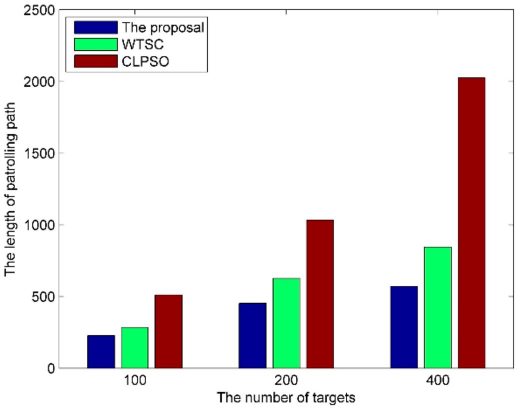


Figure 8. The change of average patrolling length with different sizes of targets and fixed size of the USV team, limited by the maximum range.

6. Conclusions

This paper has proposed a hybrid partition-based patrolling scheme of multiple USVs in an ocean scenario. Firstly, a partitioning algorithm based on importance level has been presented to divide the targets according to the USV team's size to balance the workload among multiple USVs. The importance level of each target shows its required times of being patrolled. Secondly, the partition's patrol task is mathematically expressed as a combinatorics optimization problem to achieve minimum patrolling path length. Thirdly, a distributed patrolling path planning algorithm based on the designed PSO has been proposed to plan the patrolling path for each USV in its partition. In path planning, the importance levels of the targets and the limit of the maximum range of the USVs are considered, which will increase the difficulty of patrolling planning. Finally, experiment results have demonstrated that the proposed partition-based patrolling scheme can improve the performance of cooperative patrol tasks compared with other schemes. As for the future work, the real-time environmental information is considered to realize online patrolling planning with dynamic change of the environment, such as the breakdown of the mother ship or the USV team.

Author Contributions: Conceptualization, J.X.; methodology, J.X.; software, R.Z.; validation, R.Z., J.L.; formal analysis, J.L. and S.X.; investigation, Y.L.; resources, Y.L.; data curation, Y.P.; writing—original draft preparation, J.X.; writing—review and editing, Y.P. and H.P.; funding acquisition, S.X. and Y.P. All authors have read and agreed to the published version of the manuscript.

Funding: Research supported by the National Natural Science Foundation of China (grant no 61625304, 91748116 and 61922053), Project of Shanghai Municipal Science and Technology Commission (grant no 17DZ1205000), and Shanghai Innovation Action Plan of Science and Technology, China (grant no 19DZ1207305).

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

References

1. Portugal, D.; Pippin, C.; Rocha, R.P.; Christensen, H. Finding optimal routes for multi-robot patrolling in generic graphs. In Proceedings of the 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, Chicago, IL, USA, 14–18 July 2014; IEEE: New York, NY, USA, 2014; pp. 363–369.
2. Hwang, K.S.; Lin, J.L.; Huang, H.L. Cooperative patrol planning of multi-robot systems by a competitive auction system. In Proceedings of the 2009 ICCAS-SICE, Fukuoka City, Japan, 18–21 August 2009; IEEE: New York, NY, USA, 2009; pp. 4359–4363.
3. Poulet, C.; Corruble, V.; Seghrouchni, A.E.F. Working as a team: Using social criteria in the timed patrolling problem. In Proceedings of the 2012 IEEE 24th International Conference on Tools with Artificial Intelligence, Athens, Greece, 7–9 November 2012; IEEE: New York, NY, USA, 2012; Volume 1, pp. 933–938.
4. Othmani-Guibourg, M.; El Fallah-Seghrouchni, A.; Farges, J.L.; Potop-Butucaru, M. Multi-agent patrolling in dynamic environments. In Proceedings of the 2017 IEEE International Conference on Agents (ICA), Beijing, China, 6–9 July 2017; IEEE: New York, NY, USA, 2017; pp. 72–77.
5. Yan, C.; Zhang, T. Multi-robot patrol: A distributed algorithm based on expected idleness. *Int. J. Adv. Robot. Syst.* **2016**, *13*, 1729881416663666. [CrossRef]
6. Portugal, D.; Rocha, R.P. On the performance and scalability of multi-robot patrolling algorithms. In Proceedings of the 2011 IEEE International Symposium on Safety, Security, and Rescue Robotics, Kyoto, Japan, 1–5 November 2011; IEEE: New York, NY, USA, 2011; pp. 50–55.
7. Sea, V.; Sugiyama, A.; Sugawara, T. Frequency-Based Multi-agent Patrolling Model and Its Area Partitioning Solution Method for Balanced Workload. In *International Conference on the Integration of Constraint Programming, Artificial Intelligence, and Operations Research*; Springer: Cham, Switzeland, 2018; pp. 530–545.
8. Lauri, F.; Koukam, A. A two-step evolutionary and ACO approach for solving the multi-agent patrolling problem. In Proceedings of the 2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence), Hong Kong, China, 1–6 June 2008; pp. 861–868.
9. Karypis, G.; Kumar, V. A fast and high quality multilevel scheme for partition irregular graphs. *SIAM J. Sci. Comput.* **1998**, *20*, 359–392. [CrossRef]
10. Chen, X. Police patrol optimization with security level functions. *IEEE Trans. Syst. Man Cybern. Syst.* **2013**, *43*, 1042–1051. [CrossRef]

11. Fitzpatrick, B.G.; Holmbeck-Cannell, K.; Laffin, M.; Wang, Y. Optimal perimeter patrol via max-plus probability. In Proceedings of the 2014 American Control Conference, Portland, OR, USA, 4–6 June 2014; pp. 5248–5253.
12. Farinelli, A.; Iocchi, L.; Nardi, D. Distributed on-line dynamic task assignment for multi-robot patrolling. *Auton. Robots* **2017**, *41*, 1321–1345. [CrossRef]
13. Sugiyama, A.; Sea, V.; Sugawara, T. Emergence of divisional cooperation with negotiation and re-learning and evaluation of flexibility in continuous cooperative patrol problem. *Knowl. Inf. Syst.* **2019**, *60*, 1587–1609. [CrossRef]
14. Wiandt, B.; Simon, V.; Kőkuti, A. Self-organized graph partitioning approach for multi-agent patrolling in generic graphs. In Proceedings of the IEEE EUROCON 2017–17th International Conference on Smart Technologies, Ohrid, Macedonia, 6–8 July 2017; pp. 605–610.
15. Sea, V.; Sugawara, T. Area partitioning method with learning of dirty areas and obstacles in environments for cooperative sweeping robots. In Proceedings of the 2015 IIAI 4th International Congress on Advanced Applied Informatics, Okayama, Japan, 12–16 July 2015; pp. 523–529.
16. Sea, V.; Kato, C.; Sugawara, T. Coordinated area partitioning method by autonomous agents for continuous cooperative tasks. *J. Inf. Proc.* **2017**, *25*, 75–87. [CrossRef]
17. Sugiyama, A.; Sea, V.; Sugawara, T. Effective task allocation by enhancing divisional cooperation in multi-agent continuous patrolling tasks. In Proceedings of the 2016 IEEE 28th international conference on tools with artificial intelligence (ICTAI), San Jose, CA, USA, 6–8 November 2016; pp. 33–40.
18. Yasuyuki, S.; Hirofumi, O.; Tadashi, M.; Maya, H. Cooperative capture by multi-agent using reinforcement learning application for security patrol systems. In Proceedings of the 2015 10th Asian Control Conference (ASCC), Kota Kinabalu, Malaysia, 31 May–3 June 2015; pp. 1–6.
19. Portugal, D.; Rocha, R. Msp algorithm: Multi-robot patrolling based on territory allocation using balanced graph partitioning. In Proceedings of the 2010 ACM Symposium on Applied Computing, Sierre, Switzerland, 22–26 March 2010; pp. 1271–1276.
20. Qiao, Y.; Yang, J.; Zhang, Q.; Xi, J.; Kong, L. Multi-UAV Cooperative Patrol Task Planning Novel Method Based on Improved PFIH Algorithm. *IEEE Access* **2019**, *7*, 167621–167628. [CrossRef]
21. Huang, L.; Zhou, M.C.; Hao, K.; Hou, E. A survey of multi-robot regular and adversarial patrolling. *IEEE/CAA J. Autom. Sin.* **2019**, *6*, 894–903. [CrossRef]
22. Alam, T.; Rahman, M.M.; Bobadilla, L.; Rapp, B. Multi-vehicle patrolling with limited visibility and communication constraints. In Proceedings of the MILCOM 2017–2017 IEEE Military Communications Conference (MILCOM), Baltimore, MD, USA, 23–25 October 2017; pp. 465–470.
23. Chen, X. Fast patrol route planning in dynamic environments. *IEEE Trans. Syst. Man Cybern. Part A: Syst. Humans* **2012**, *42*, 894–904. [CrossRef]
24. Chen, S.; Wu, F.; Shen, L.; Chen, J.; Ramchurn, S.D. Decentralized patrolling under constraints in dynamic environments. *IEEE Trans. Cybern.* **2015**, *46*, 3364–3376. [CrossRef] [PubMed]
25. Li, J.; Xiong, Y.; She, J.; Wu, M. A path planning method for sweep coverage with multiple UAVs. *IEEE Int. Things J.* **2020**, *7*, 8967–8978. [CrossRef]
26. Liang, J.J.; Qin, A.K.; Suganthan, P.N.; Baskar, S. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Trans. Evol. Comput.* **2006**, *10*, 281–295. [CrossRef]

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.