Multi-UAV Coverage Path Planning Based on Hexagonal Grid Decomposition in Maritime Search and Rescue

Sung-Won Cho 1, Jin-Hyoung Park 1, Hyun-Ji Park 1 and Seongmin Kim 2

Abstract: In the event of a maritime accident, surveying the maximum area efficiently in the least amount of time is crucial for rescuing survivors. Increasingly, unmanned aerial vehicles (UAVs) are being used in search and rescue operations. This study proposes a method to generate a search path that covers all generated nodes in the shortest amount of time with multiple heterogeneous UAVs. The proposed model, which is a mixed-integer linear programming (MILP) model based on a hexagonal grid-based decomposition method, was verified through a simulation analysis based on the performance of an actual UAV. This study presents both the optimization technique’s calculation time as a function of the search area size and the various UAV routes derived as the search area grows. The results of this study can have wide-ranging applications for emergency search and rescue operations.

Keywords: mixed integer linear programming model; multi-UAV coverage path planning; hexagonal grid decomposition; maritime search and rescue

1. Introduction

In Korea, an average of 2631 marine accidents have occurred per year over the last six years (Figure 1). Out of 553 fatal accidents in 2020, 126 people were killed or are missing [1]. As the environment changes, including weather and ocean currents, marine accidents involving fishing boats or passenger ships are becoming more common [2]. Because marine accidents usually occur in environments where direct human access is difficult, the risk of death also increases, creating danger for the victim and rescuer alike [3]. To reduce the number of fatalities, a system that can respond to marine accidents under normal circumstances should be established, and first responders should be prepared to use it in an emergency [4,5].

Unlike land, the ocean is a difficult environment for humans to directly access in the event of an accident [6]. The time required for search and rescue (SAR) operations is determined by the amount and performance of available resources, and various tools and resources have been developed to address this issue [7–12]. Unmanned aerial vehicles (UAVs) are being introduced in several countries to improve maritime SAR operations [13]. Humans lose efficiency over time, whereas robots are able to maintain their efficiency at all times, thereby reducing the uncertainty that can arise in maritime SAR [14].

This study focuses on search operations that require rapid and efficient coverage of a target area in maritime SAR. A search operation, which can also be defined as a coverage path planning (CPP) problem, requires navigating a large area in an abbreviated period of time to rescue survivors. In general, robot path planning is classified into global path planning and local path planning [15,16]. Global path planning creates a complete path from the initial starting point to the target point based on deterministic or stochastic...
information before the mobile robot starts operating [17,18]. In contrast, local path planning generates an operation plan in real-time, responding to changes in operating environments and data collected by local sensors during navigation [19–22]. In this study, we propose a method for global CPP with deterministic information of UAVs to search for survivors at sea.

A CPP problem first divides the target area into subareas that need to be covered and then determines the order in which the divided area will be covered. As shown in Figure 2, the two methods for segmenting the region are cellular decomposition and grid-based decomposition [23–25]. In Figure 2a, the cellular decomposition method divides the area based on the characteristics of the search area and the UAV’s speed, weight, battery endurance, etc. [26–28]. Although the cellular decomposition method is simple to determine the order of cells to cover as it considers the characteristics of the region and the UAV when dividing, owing to the various characteristics, it is difficult to define the decomposition criterion. As shown in Figure 2b, the grid-based decomposition method divides an area into small pieces using a predetermined unit. The grid-based decomposition method divides the region without considering the region’s characteristics or the UAV, making it simple to divide the region but increasing the complexity of planning the coverage path.

For the CPP problem, the method of dividing a region has a direct impact on the method of coverage path planning [24,28,29]. For studies that use the cellular decomposition method [30–35], the coverage path is simply obtained by assuming that the order (i.e., path) in which the UAV moves to cover each cell is back-and-forth. The sweep direction is determined by the amount of energy the robot consumes moving in reverse when moving back and forth [26–28,30,33–40]. The cellular decomposition method has the advantage of quickly establishing a path at each cell using an algorithm with low computational
complexity. However, when multiple UAV types are used or an irregularly shaped region is searched, this method has difficulty finding an optimal path, and a local optimal solution is derived [41,42]. When the area is divided using the grid-based decomposition method, a large number of grid cells (i.e., the number of nodes) are created. Thus, it takes a relatively long time to generate a path. However, the grid-based decomposition method does generate an optimal path [43–47]. This study focuses on generating the optimal search path for multiple UAVs by segmenting the target area using the grid-based decomposition method.

In most grid-based decomposition studies [43–50], the shape of the grid cell is set to a square, which is the motivation for this study. The search path may derive differently as the location and number of adjacent nodes that the UAV can move to from each node vary depending on the shape of the grid cell. Additionally, the more sides the grid cell has, the more directions it allows for movement when planning the path. This allows paths to be explored efficiently but increases the computational complexity of the problem. Thus, this study proposes a hexagonal grid-based decomposition method and a mixed-integer linear programming (MILP) model to generate a search path that covers all generated nodes in the shortest amount of time with multiple heterogeneous UAVs.

The contributions of our study are twofold as follows. First, to the best of our knowledge, this is the first study to propose a MILP model for CPP based on hexagonal grid decomposition. The model proposed by [44] is a comprehensive model that does not consider the angle between adjacent nodes as a constraint, but it is time-consuming to derive the solution. Therefore, we develop a model that lowers the computational volume by utilizing the angles between adjacent grid cells. Second, we present a coverage path by considering multiple UAVs that operate simultaneously. Most studies on the multi-UAV CPP problem decomposed the target area into subregions and hierarchically interpreted the problem as a multiple CPP problem of a single UAV for each subregion. This method can generate an inefficient path because it cannot plan a path that entirely considers the performance of heterogeneous UAVs.

The remainder of this paper is organized as follows. Section 2 provides a detailed description of the problems and describes the proposed model. The computational experiments are explained and interpreted in Section 3. Finally, conclusions are drawn in Section 4.

2. Problem Description

The objective of this study is to enhance the efficiency and completeness of search operations in maritime SAR systems. Figure 3 shows the overall architecture of the SAR system, which consists of a monitoring system (Subsystem 1), a global planning system (Subsystem 2), and a local planning system (Subsystem 3).

Figure 3. Maritime search and rescue (SAR) system architecture.

Subsystem 1 identifies the accident and gathers information, including the accident area. Then, subsystem 2 is triggered to create a global SAR plan. Subsystem 2 generates a map for the given search boundary; on this map, a coverage path plan and mission plan are generated for the initial search operation and rescue operation, respectively. These
global plans provide a reference for subsystem 3 to create a local SAR plan (i.e., navigation). Subsystem 3 continuously modifies the existing global plans as sensors update environment information.

This study focuses on map generation and coverage path planning for the search operation of subsystem 2. At the level of subsystems 1 and 2, the details regarding the accident and environment are typically unknown or uncertain; thus, the coverage plans of the global strategies for initial search operations must ensure an exhaustive search within a boundary around the accident area. Because there is no room for delay, the completion time of the search is the most crucial consideration in a successful SAR. Therefore, this study aims to produce coverage path plans that minimize the search completion time, which is achieved by the proposed method for map generation.

Through a two-phase process, this study creates a path for multiple UAVs to search a given area. In phase 1, path planning begins by dividing the search area using the hexagonal grid decomposition method. Figure 4 shows the camera footprint in the UAV expressed as a hexagonal grid cell.

![Image 1](image1.png)  
**Figure 4.** Hexagonal grid cell.

Phase 2 transforms the generated hexagonal grid-based map into a graph to obtain input values for path generation. The graph is comprised of nodes centered at each hexagonal grid cell and edges connecting each grid and adjacent grids. The grid-decomposed area is converted to graph form, as shown in Figure 5, and all grids corresponding to the search area are covered. In this study, we set up a MILP model that creates a search path for multiple heterogeneous UAVs that covers all the nodes in the shortest time.

![Image 2](image2.png)  
**Figure 5.** Graph structure of the search area.
2.1. Phase 1: Hexagonal Grid-Based Decomposition

The search area in this study is divided into hexagonal grid cells, assuming each UAV moves through the grid’s center. Hexagonal grid-based decomposition is used accordingly from the point where the UAV departs. The size of the hexagon is the image size \( a \) as determined by the camera quality of the UAV (see Figure 6). Covered and uncovered grids are defined in the divided search area and used as input values to create the search path in phase 2.

![Figure 6. Input value of hexagonal grid-based decomposition.](image)

2.2. Phase 2: Multi-UAV Coverage Path Planning

To formulate a MILP model for CPP based on hexagonal grid decomposition, several assumptions are made. All the nodes in the search area that belong to set \( N^- \) must be covered. All nodes are covered only once by a UAV, and thus a collision-free path between UAVs is always guaranteed. Because the search operation is terminated once all the hexagonal nodes are covered, a dummy node \( i^{-1} \) is introduced as an ending node to indicate the completion of the search. \( N^+ \) is a set of searching nodes including the dummy ending node \( i^{-1} \). The starting node \( \theta(u) \) of the UAVs is outside the search area, and it is not in \( N^- \) or \( N^+ \). The UAVs must move from the current node to adjacent nodes, which indicate the physically neighboring nodes along the sides \( e = 1, 2, \ldots, 6 \) of the hexagon shown in Figure 7. The adjacent nodes and dummy ending node belong to \( N^{\text{ADJ}}(i) \) for node \( i \). The dummy ending node only has itself as an adjacent node, i.e. \( N^{\text{ADJ}}(i^{-1}) = \{i^{-1}\} \). To move to an adjacent node, cruising and rotational movements are taken into account. In detail, if a UAV passes the nodes \( i \to j \to k \) in sequence, the UAV straightly cruises for the paths \( i \to j \) and \( j \to k \), and rotates as the angle between the vectors \( \langle i, j \rangle \) and \( \langle j, k \rangle \) at node \( j \). Thus, the travel times of UAVs can be calculated as the sum of cruising time with speed \( v_u \) and rotational time with speed \( w_u \). Exceptionally, rotational time at the starting node is not considered.

A series consisting of the decision variable \( x_{uijes} \) can represent the path of the UAVs. Figure 7 shows an illustrative example to aid in understanding of the decision variable \( x_{uijes} \). As can be seen in Figure 7, the exit direction at node \( i \) is toward side \( e = 3 \) which connects to node \( j \) while the initial heading direction is toward side \( e = 2 \). In this situation, the decision variable \( x_{uijes} \) is 1 at \( e = 2 \), and the rotational time at node \( i \) is proportional to an angle between the vectors \( I(e = 2) \) and \( (i, j) \). Furthermore, for the travel between the starting node and first visited node, the decision variable \( x_{uijes} \) is 1 at \( e = 7 \). Additionally, for the virtual travel between the dummy ending nodes, the decision variable \( x_{uijes} \) is 1 at \( e = 8 \).
Figure 7. Illustrative example for notation $x_{uijes}$.

Mixed-Integer Linear Programming Model

In this section, we describe the MILP model for CPP based on hexagonal grid decomposition.

$$
\begin{align*}
\text{min} & \quad t_u^{\text{max}} \\
\text{s.t.} & \quad \sum_{i \in N^+} \sum_{j \in N^+} \sum_{e \in E} x_{uijes} = 1, \quad \text{for } u \in U, s \in S \\
& \quad \sum_{u \in U} \sum_{j \in N^+} \sum_{e \in E} x_{uijes} = 1, \quad \text{for } i \in N^- \\
& \quad x_{uijes} = 0, \quad \text{for } u \in U, i \in N^+, j \in N^+ \setminus \{N^{ADJ}(i)\}, e \in E, s \in S \\
& \quad \sum_{i \in N^+} \sum_{e \in E} x_{uijes} - \sum_{i \in N^+} \sum_{e \in E} x_{uijes(i-1)} = 0, \quad \text{for } u \in U, j \in N^+, s \in S \setminus \{1\} \\
& \quad \sum_{i \in N^{ADJ}(j)} y_{ui} = 1, \quad \text{for } u \in U \\
& \quad \sum_{j \in N^+} x_{uijes} - y_{ui} = 0, \quad \text{for } u \in U, i \in N^+, e = 7, s = 1 \\
& \quad x_{uijes} = 0, \quad \text{for } u \in U, i = i^{-1}, j = i^{-1}, e \in E \setminus \{8\}, s \in S \\
& \quad \sum_{k \in N^{ADJ}(i)} x_{uijes} = \sum_{e \in E} x_{uijes(i-1)} = 0, \quad \text{for } u \in U, i \in N^-, j \in N^{ADJ}(i) \setminus \{i^{-1}\}, s \in S \setminus \{1\} \\
& \quad t_u^0 = \sum_{i = \mathcal{P}(u)} \frac{d_{ij}}{v_u} \times y_{ui} + \sum_{i \in N^{ADJ}(\mathcal{P}(u))} \sum_{j \in N^+ \setminus \{i\}} \sum_{e = 7} \frac{r(\overrightarrow{(\mathcal{P}(u), i)}, \overrightarrow{ij})}{w_u} \times x_{uijes}, \quad \text{for } u \in U \\
& \quad t_u = t_u^0 + \sum_{i \in N^+} \sum_{j \in N^{ADJ}(i)} \sum_{s \in S} \left[ \sum_{e \in E} \frac{d_{ij}}{v_u} + \sum_{e \in E \setminus \{7, 8\}} \frac{r(l(e), \overrightarrow{ij})}{w_u} \right] \times x_{uijes}, \quad \text{for } u \in U
\end{align*}
$$
\[ t_u \leq p_u, \text{ for } u \in U \] (12)
\[ t_{u_{\text{max}}} \geq t_u, \text{ for } u \in U \] (13)
\[ t_u^0 \geq 0, t_u \geq 0, t_{u_{\text{max}}} \geq 0 \] (14)

\[ x_{uj}, y_{ui} \in \{0, 1\} \] (15)

The objective function (1) is the completion time that is the longest among the travel times of all the UAVs, and the function is minimized. Constraint (2) ensures that each movement step for each UAV has only one path and state. Constraint (3) forces each node in the defined search area to be covered only once, preventing duplicate paths from being reused by different UAVs. Constraint (4) limits movement to nodes that are adjacent to one another. Constraint (5) states that if the path of each UAV is \( i \to j, j \to k \) becomes the next path in a sequential manner. Constraint (6) defines a movement path from the starting node to one of the nearby nodes. Constraints (7)–(9) are relevant to the definition of the index \( e \) in the decision variable \( x_{uijes} \). Constraint (7) links the path-related decision variable \( y_{ui} \) from the starting node to the search area and the path-related decision variable \( x_{uijes} \) within the search area. If the path from the starting node is \( i^0 \to i \), the first movement step \((s = 1)\) in the search area sequentially leads to \( i \to j \), satisfying the definition of \((e = 7)\) as the first movement of index \( e \). Likewise, Constraint (8) satisfies the definition of \((e = 8)\) as the ending movement of index \( e \). Constraint (9) allows the index \( e \) to be the index \( e'(\langle i, j \rangle) \) of the side facing from the previous path, as defined by the decision variable \( x_{uijes} \). Constraint (10) specifies the time required for each UAV to travel from the starting node to the search area, while Constraint (11) defines the time required for each UAV to complete the search. The required time is the sum of the travel time between nodes in a linear distance and the rotation time of the UAV according to the angle formed by the continuous path. Constraint (12) ensures that each search completion time of a UAV is always less than or equal to the maximum duration time. Constraint (13) defines the time needed to complete a search after all nodes have been visited. Finally, Constraints (14)–(15) define the non-negative real and binary variable types of the decision variables.

3. Numerical Experiments
3.1. Experiment Design

We conducted numerical experiments to verify and analyze the methodology presented in this study. The hexagonal grid map generated in phase 1 is defined as an input value of the mathematical model presented in Section 2.2. Figure 8 is one of the target search areas used in the experiments, and the grid cells marked in green are used as the input value.
As mentioned in Section 1, the hexagonal grid decomposition method has a direct impact on the computational complexity of the mathematical model for path generation as well as the target value of the optimal solution. Hence, the time required to calculate the optimal solution in response to changes in the size of the search area is analyzed. In addition, the optimal UAV search path generated in response to changes in the shape of the search area is presented.

All UAVs maintain a constant altitude during a mission, and thus the camera footprint of all UAVs in this experiment is the same as a circle with a radius of 20 m. The performance of each UAV is shown in Table 1. The experiment was conducted using an Intel Core i7-7700HQ 2.80 GHz, 16.0 GB RAM, Windows 10 operating system environment, and a Gurobi 8.0.1 solver (https://www.gurobi.com (accessed on 17 September 2021)) was used to solve the MILP model. The Python programming language was used in all experiments.

| UAV               | $v_u$ (m/s) | $w_u$ (radian/s) | $p_u$ (s) |
|-------------------|-------------|------------------|-----------|
| UAV 1 (Quad-rotor)| 4           | 0.5              | 1800      |
| UAV 2 (Quad-rotor)| 4           | 1                | 1200      |
| UAV 3 (Quad-rotor)| 4           | 0.5              | 1500      |

3.2. Time Needed to Calculate the Optimal Solution in Response to Changes in the Size of the Search Area

The multi-UAV CPP model proposed in this study has high computational complexity. The complexity in finding an optimal path increases in direct proportion to the size of the search area. This section discusses the time needed for the mathematical model presented in Section 2.2 to calculate the optimal solution using the commercial solver (Gurobi 8.0.1). Twenty-seven instances with different search area sizes were used in this experiment while retaining similar shapes. Figure 9 shows the instances with the smallest and largest number of nodes, respectively.
Figure 9. Search areas of different test sizes. (a) 21 search nodes; (b) 45 search nodes.

Figure 10 shows the relationship between the time required to calculate the optimal solution using the commercial solver and the number of search nodes. When the number of search nodes was less than 27, an optimal solution can be found within a minute based on the instance used in this experiment. When the number of search nodes reaches 40, the commercial solver takes more than an hour to find the optimal solution. Figure 10 shows that as the size of the search area grows, the computational time grows exponentially, indicating an exponential growth in the computational complexity.

Figure 10. Time needed to calculate the optimal solution in response to changes in the size of the search area.

3.3. Search Path According to Changes in Search Area Shape

The optimal search path derived from the mathematical model presented in Section 2 is discussed in this section. As shown in Figure 11, we used various search areas to test the proposed model in this experiment. Figure 11a–c are instances used in Section 3.2, and the shapes of the search areas are similar but the sizes are different. Figure 11d–f are instances that have different irregularly shaped search areas.

Despite the similar shapes of the search areas, the search paths in Figure 12a–c appear in a completely different manner. As shown in Figure 12d–f, the mathematical model presented in this study can generate optimal routes for several types of search areas, even if they are not standardized.
3.4. Search Path According to the Change in the Shape of Grid Cells

In this section, the effectiveness achieved by the hexagonal grid-based area decomposition is validated. For the same instances in Figure 11, we tested the proposed MILP model on a map with square grid cells and on a map with hexagonal grid cells and compared the results. There is no structural difference in the MILP model except for considering four
adjacent sides instead of six adjacent sides in set $E$. Figure 13 shows the optimal search paths on the square grid map derived by the MILP model.

Compared to the results in Figure 13, on the hexagonal grid map in Figure 12, the search times of the served UAVs are reduced for all test instances. The search completion times on the hexagonal grid map are on average 10.59% lower than those on the square grid map. For instance (a), the worst case, the reduction reaches 23.39%. Furthermore, on the hexagonal grid map, the average difference between the search times of the served UAVs is 1.30 seconds, compared with an average difference of 4.02 seconds on the square grid map. Thus, the reduction is 67.7%. All these results are intuitively reasonable because the hexagonal grid decomposition allows more movable nodes for the served UAVs and thus creates relatively flexible paths.

![Figure 13. Optimal search path according to the change in the shape of grid cells. (a) 28 search nodes; (b) 43 search nodes; (c) 56 search nodes; (d) 43 search nodes (V-shaped area); (e) 28 search nodes (bullet shaped area); (f) 42 search nodes.](image)

3.5. Search Path According to the Change in the Number of Served UAVs

In this section, the performance of the proposed model with three UAVs is verified. Similar to the previous experiment, for the same instances in Figure 11, we tested the proposed MILP model with three UAVs. Figure 14 shows the optimal three-UAV search paths derived by the MILP model.

Compared to the results for two UAVs in Figure 12, the reduction in search completion times when using three UAVs is on average 28.4%. The difference between the maximum and minimum search times of the UAVs is 2.75 seconds on average. These results are encouraging in that the proposed method efficiently distributes UAV operations without any collisions despite the increasing number of served UAVs.
4. Conclusions

Marine accidents in Korea result in casualties and ongoing environmental and property damage. Because survivors can be exposed to increasingly dangerous conditions and uncertainty as time passes, it is critical for first responders to respond quickly. If an accident occurs, the search for survivors is fraught with difficulties resulting from various uncontrollable factors in the marine environment. Hence, public institutions such as the Ministry of Oceans and Fisheries, the Korea Coast Guard, and the Navy are making efforts to address this issue. In this study, multiple heterogeneous UAVs were utilized to search for casualties in a short period of time to protect lives from catastrophic marine accidents and to prevent the spread of damage. We proposed a hexagonal grid decomposition to represent the search area as a hexagonal grid map, and developed the MILP model that reduces the time required for all UAVs to complete a search using the graph for this grid map as an input. The proposed model in the numerical experiment was verified by a simulation analysis based on the performance of an actual UAV. This study presents both the optimization technique’s calculation time as a function of the search area size and the multiple UAV paths for each of the various search areas. The experimental results of this study provided the following two insights. First, the proposed model can be comprehensively operated even with changes such as the shape or size of the search area and the diversity or number of UAVs. Second, because the hexagonal grid cell has more adjacent grids than the square grid cell, the proposed model has a larger solution space and can derive a more efficient path.

This study confirmed the relationship between the number of search nodes and computational complexity based on a small sample size; subsequent experiments on more diverse environments could provide more comprehensive insight. In future works, related algorithms (dynamic programming, reinforcement learning, etc.) can be studied to solve larger-scale problems with reasonable computational time. In addition, this study which focuses on deterministic global planning can be extended in consideration of the factors that change over time such as weather conditions and ocean currents. Furthermore, CPP for search operations affects mission planning for rescue operations and local planning systems in maritime SAR systems, and future research can be conducted in an integrated manner. Lastly, not only can this study be used to aid in the decision-making of a quick
and accurate initial response to marine accidents, but we also expect the results from this study to lead to a wider application in land-based emergencies.

**Author Contributions:** Conceptualization, S.-W.C., H.-J.P. and J.-H.P.; methodology, S.-W.C. and H.-J.P.; software, H.-J.P. and S.K.; validation, S.-W.C. and H.-J.P.; formal analysis, S.-W.C.; writing—original draft preparation, S.-W.C. and H.-J.P.; writing—review and editing, S.-W.C. and H.-J.P.; supervision, S.-W.C. and J.-H.P.; funding, J.-H.P. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research is a part of the project titled “Development of Open Platform Technologies for Smart Maritime Safety and Industries” funded by the Korea Research Institute of Ships and Ocean Engineering (PES4070).

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The authors are grateful for the valuable comments from the editor and anonymous reviewers.

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

**Nomenclature**

| Set and parameters | Description |
|--------------------|-------------|
| $i$                | Index of node |
| $u$                | Index of UAV |
| $e$                | Index of the side that makes up the node’s hexagon or index of starting nodes and dummy node |
| $s$                | Index of UAV movement steps |
| $\hat{0}(u)$      | Index of UAV $u$’s starting node |
| $i^{-1}$           | Index of ending node (dummy node to indicate the end of the search) |
| $N^-$              | Set of nodes $i$ in the search area |
| $N^+$              | Set of nodes $i$ and ending node $i^{-1}$ in the search area, $N^+ = N^- \cup \{i^{-1}\}$ |
| $N^{ADJ}(i)$      | Set of nodes adjacent to node $i$ |
| $U$                | Set of UAV $u$ |
| $E$                | Set of indices with the following values: 1 (2, $\cdots$, 6) if a UAV is heading toward to the hexagonal side $e = 1 (2, \cdots, 6)$ of the node; 7 if it passed the starting node; 8 if it passed the ending node |
| $S$                | Set of UAV movement steps, including sufficiently large number |
| $d_{ij}$          | Distance between node $i$ and node $j$ (unit: m) |
| $e((i, j))$       | Hexagonal side $e$ of node $i$ that intersects the vector $(i, j)$ of nodes $i, j$ |
| $l(e)$            | Vector that is perpendicular to the hexagonal side $e$ with direction from the current node to heading side $e$ |
| $r(l_1, l_2)$    | Angle between vector $l_1$ and vector $l_2$ (unit: radian) |
| $v_u$             | Average cruising speed of UAV $u$ in straight line (unit: m/sec) |
| $w_u$             | Average rotational speed of UAV $u$ (unit: radian/sec) |
| $p_u$             | Maximum duration time for UAV $u$ (unit: sec) |

**Decision variables**

| $x_{u, i, e, s}$  | 1 if UAV $u$ is at node $i$ facing side $e$ at step $s$ and its subsequent path is equal to $i \rightarrow j$, 0 otherwise |
| $y_{ui}$         | 1 if UAV $u$ moves from the starting node to node $i$, 0 otherwise |
| $l_{i, u}^{0}$   | Travel time for UAV $u$ from its starting node to the initially reached node in the search area |
| $t_{u}$          | Total travel time for UAV $u$ to complete the search, $t_{\text{max}} = \max_{u \in U} \{t_{u}\}$ |
| $t_{\text{max}}$ | Resulting time to complete the search, $t_{\text{max}} = \max_{u \in U} \{t_{u}\}$ |
31. Choset, H.; Acar, E.; Rizzi, A.A.; Luntz, J. Exact cellular decompositions in terms of critical points of morse functions. In Proceedings of the 2000 ICRA, Millennium Conference, IEEE International Conference on Robotics and Automation, Symposia Proceedings (Cat. No. 00CH37065), San Francisco, CA, USA, 24–28 April 2000; Volume 3, pp. 2270–2277.

32. Acar, E.U.; Choset, H.; Rizzi, A.A.; Atkar, P.N.; Hull, D. Morse decompositions for coverage tasks. *Int. J. Robot. Res.* **2002**, *21*, 331–344. [CrossRef]

33. Rekleitis, I.; Lee-Shue, V.; New, A.P.; Choset, H. Limited communication, multi-robot team based coverage. In Proceedings of the IEEE International Conference on Robotics and Automation, ICRA’04, New Orleans, LA, USA, 26 April–1 May 2004; Volume 4, pp. 3462–3468.

34. Li, Y.; Chen, H.; Er, M.J.; Wang, X. Coverage path planning for UAVs based on enhanced exact cellular decomposition method. *Mechatronics* **2011**, *21*, 876–885. [CrossRef]

35. Balampanis, F.; Maza, I.; Ollero, A. Area partition for coastal regions with multiple UAS. *J. Intell. Robot. Syst.* **2017**, *88*, 751–766. [CrossRef]

36. Coombes, M.; Chen, W.H.; Liu, C. Boustrophedon coverage path planning for UAV aerial surveys in wind. In Proceedings of the 2017 International Conference on Unmanned Aircraft Systems (ICUAS), Miami, FL, USA, 13–16 June 2017; pp. 1563–1571.

37. Huang, W.H. Optimal line-sweep-based decompositions for coverage algorithms. In Proceedings of the 2001 ICRA, IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164), Seoul, Korea, 21–26 May 2001; Volume 1, pp. 27–32.

38. Maza, I.; Ollero, A. Multiple UAV cooperative searching operation using polygon area decomposition and efficient coverage algorithms. In *Distributed Autonomous Robotic Systems 6*; Springer: Tokyo, Japan, 2007; pp. 221–230.

39. Bochkarev, S.; Smith, S.L. On minimizing turns in robot coverage path planning. In Proceedings of the 2016 IEEE International Conference on Automation Science and Engineering (CASE), Fort Worth, TX, USA, 21–25 August 2016; pp. 1237–1242.

40. Torres, M.; Pelta, D.A.; Verdegay, J.L.; Torres, J.C. Coverage path planning with unmanned aerial vehicles for 3D terrain reconstruction. *Expert Syst. Appl.* **2016**, *55*, 441–451. [CrossRef]

41. Toth, P.; Vigo, D. The vehicle routing problem: Society for Industrial and Applied Mathematics. *Siam Monogr. Discret. Math. Appl.* **2001**. [CrossRef]

42. Barrientos, A.; Colorado, J.; Cerro, J.d.; Martinez, A.; Rossi, C.; Sanz, D.; Valente, J. Aerial remote sensing in agriculture: A practical approach to area coverage and path planning for fleets of mini aerial robots. *J. Field Robot.* **2011**, *28*, 667–689. [CrossRef]

43. Cabreira, T.M.; Ferreira, P.R.; Di Franco, C.; Buttazzo, G.C. Grid-based coverage path planning with minimum energy over irregular-shaped areas with UAVs. In Proceedings of the 2019 International Conference on Unmanned Aircraft Systems (ICUAS), Atlanta, GA, USA, 11–14 June 2019; pp. 758–767.

44. Cho, S.W.; Park, H.J.; Lee, H.; Shim, D.H.; Kim, S.Y. Coverage path planning for multiple unmanned aerial vehicles in maritime search and rescue operations. *Comput. Ind. Eng.* **2021**, *161*, 107612. [CrossRef]

45. Miao, X.; Lee, J.; Kang, B.Y. Scalable coverage path planning for cleaning robots using rectangular map decomposition on large environments. *IEEE Access* **2018**, *6*, 38200–38215. [CrossRef]

46. Lakshmanan, A.K.; Mohan, R.E.; Ramalingam, B.; Le, A.V.; Veerajagadshwar, P.; Tiwari, K.; Ilyas, M. Complete coverage path planning using reinforcement learning for tetromino based cleaning and maintenance robot. *Autom. Constr.* **2020**, *112*, 103078. [CrossRef]

47. Cheng, K.P.; Mohan, R.E.; Nhan, N.H.K.; Le, A.V. Graph theory-based approach to accomplish complete coverage path planning tasks for reconfigurable robots. *IEEE Access* **2019**, *7*, 94642–94657. [CrossRef]

48. Valente, J.; Sanz, D.; Del Cerro, J.; Barrientos, A.; de Frutos, M.Á. Near-optimal coverage trajectories for image mosaicing using a mini quad-rotor over irregular-shaped fields. *Precis. Agric.* **2013**, *14*, 115–132. [CrossRef]

49. Nam, L.; Huang, L.; Li, X.J.; Xu, J. An approach for coverage path planning for UAVs. In Proceedings of the 2016 IEEE 14th International Workshop on Advanced Motion Control (AMC), Auckland, New Zealand, 22–24 April 2016; pp. 411–416.

50. Choi, Y.; Choi, Y.; Briceno, S.; Mavris, D.N. Energy-constrained multi-UAV coverage path planning for an aerial imagery mission using column generation. *J. Intell. Robot. Syst.* **2020**, *97*, 125–139. [CrossRef]