2015 15th International Conference on Control, Automation and Systems (ICCAS 2015)
Oct. 13-16, 2015 in BEXCO, Busan, Korea

Path Planning Algorithm to Minimize an Overlapped Path and Turning Number for an Underwater Mining Robot

Pandu Sandi Pratama[1], Jin-Wook Kim[1], Hak-Kyeong Kim[1], Suk-Min Yoon[2], Tae-Kyeong Yeu[2], Sup Hong[2], Sea-June Oh[3] and Sang-Bong Kim[1*]

[1] Department of Mechanical Design Engineering , Pukyong National University, Busan, 608-739, Korea (sbkim@pknu.ac.kr) * Corresponding author

[2] Technology Center for Offshore Plant Industries, KRISO, Daejeon, 305-343, Korea

[3] Division of Marine Engineering, Korea Maritime and Ocean University, Busan 606-791, Korea

Abstract: This paper considers the problem of minimizing an overlapped path and turning number to cover an uneven terrain ocean floor for an underwater mining robot. The purpose of this algorithm is to generate the shortest path to travel through the entire given bathymetric map with minimum overlapping path and minimum turning number based on binary map, cell decomposition, minimal sum of altitude method and depth-first search algorithm. To do this task, the following tasks are performed. Firstly, a binary map is obtained by applying threshold to the high slope regions of the ocean floor. Secondly, Morse cell decomposition method is applied to divide the whole area of work space into cells in the vertical and horizontal directions. To minimize the turning number, a minimal sum of altitude method is applied to the combination of vertical and horizontal cell decompositions. Finally, the coverage order is calculated by depth-first search algorithm. The simulation result shows that the proposed method generates the short coverage path with minimum overlapping, minimum turning number and fast coverage time compared to vertical and horizontal cell decomposition methods.

Keywords: mining robot, coverage path planning, depth-first search, minimum turning.

1. INTRODUCTION

Deep sea mining is a relatively new mineral retrieval process on the ocean floor. Since the discovery of manganese nodule that contains valuable minerals such as manganese, nickel, copper, and cobalt, the deep sea mining becomes a prospective solution to fulfill the demand for mineral resource in the future. The deep sea mining is relatively simpler than the common ground mining since the minerals are spread on the surface of ocean floor. However, the excavation cannot be done manually since it takes place on the ocean floor. Therefore, an underwater mining robot is required. The mining robot is a kind of remotely operated self-propelled underwater vehicle for collecting deep-seabed manganese nodules. The total concept of an automatic deep sea mining system as shown in Fig. 1 is composed of the collecting system including the mining robots, lifting system including flexible pipe, buffer, lifting pipe and lifting pump, surface vessel including mining vessel and ore carrier, and discharge system including discharge pipe and tiling discharge.

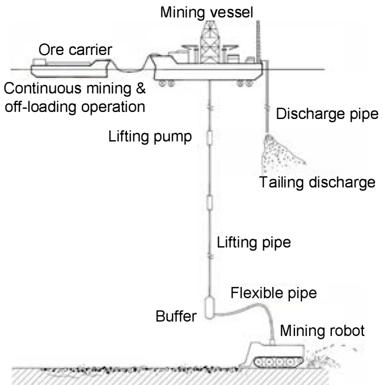


Fig. 1 Mining system.

To make the mining robot move efficiently to collect the manganese nodule in a given operation area, a path planning algorithm is needed. An optimal coverage path planning problem can be formulated as a generalization of the Traveling Salesman Problem (TSP) [1] for a continuous domain. Covering problem was tested to solve several applications such as lawn mowers [2], floor cleaning robots [3], unmanned aerial vehicles [4], and agricultural robots [5]. However, since the mining robot operates on the deep sea, the requirement is different from those applications.

There are considerable methods addressing the coverage path planning problem. The first method is cell decomposition based methods such as classical exact cell decomposition [7] and Morse cell decomposition [6]. In these methods, the cell covering sequence is calculated using optimal method such as genetic algorithm [8] to minimize the overlap. The second method is grid-based methods such as heuristic [4] and spanning tree [9] algorithms. In these algorithms, the turning number is not minimized. To solve this problem, a new path planning algorithm is needed.

This paper proposes an algorithm to generate the shortest path to travel through all the area with minimum overlapping path and minimum turning number based on the given bathymetric map. To do this task, the followings are done. Firstly, a binary map is obtained by applying threshold to the high slope regions of the ocean floor. Secondly, Morse cell decomposition method is applied to divide the whole area of work space into cells in the vertical and horizontal direction. To minimize the turning number, a minimal sum of altitude method is applied to the combination of vertical and horizontal cell decompositions. The simulation result shows that the proposed method generates the short coverage path with short overlapping path, has minimum turning number and fast coverage time compared with vertical and horizontal cell decomposition methods.

2. COVERAGE PATH PLANNING

This chapter proposes the new path planning algorithm. The input of the proposed algorithm is a bathymetric map. A bathymetric map is an elevation map of the mapped area.

The requirements for operating of a mining robot are as follows:

1. The mining robot must move through all the points in the target area to be covered completely.
2. The mining robot must fill the region with minimum overlapping paths.
3. Simple motion trajectories such as straight lines or circles should be used for simplicity in control.
4. The number of turning should be minimized for the mining robot to be connected to the mining vessel that couldn't turn easily.

A proposed coverage path planning method for this paper consists of three steps. The first step is to detect high slop regions defined as obstacles. The second step is to apply minimal sum altitude cell decomposition method for dividing the whole area of the work space of the ocean floor into smaller areas based on the obstacle position. The final step is to plan a shortest coverage path using a depth-first search algorithm method.

2.1 Binary map

Since the ocean surface is not flat, the high slope area should be avoided and assumed as a blocked area. To do this, a "slope map", $S(x,y)$, is calculated for the mapped area as the norm of the gradient of bathymetric map, $B(x,y)$, as in [11]:

$$S(x,y) = \|\nabla B\| = \left\| \frac{\partial B}{\partial x}\vec{i} + \frac{\partial B}{\partial y}\vec{j} \right\|, \tag{1}$$

where $\vec{i}, \vec{j}$ are the standard unit vectors in the X and Y axis, respectively. A binary map $T(x,y)$ is obtained by applying a user-defined slope threshold, δ_s, to the slope, $S(x,y)$ as follows:

$$T(x,y) = \begin{cases} 1 & \text{if } S(x,y) \geq \delta_s \\ 0 & \text{if } S(x,y) \leq \delta_s \end{cases}, \tag{2}$$

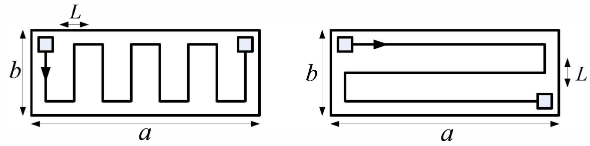
The example of binary map is shown in Fig. 2.

0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	0
0	0	0	0	0	0	0	0

Fig. 2 Binary map.

2.2 Minimal sum of altitude cell decomposition

The number of turns required to cover the region is a main factor for saving time cost. In the template-based path planner that the robot moved with the same sweep direction for every cell, it would generate too many unnecessary turns leading to low efficiency as in Fig. 3a [6]. However, if the sweep direction is similar with the shortest side of cell as shown in Fig. 3b, the minimum turning number can be obtained. Therefore, the turning number is related with the sweeping direction.



a. Inefficient turns b. Efficient turns

Fig. 3. Relation between turning number and sweep direction.

To minimize the turning number, a minimal sum of altitude (MSA) algorithm as in [10] was used. In MSA, the optimal decomposition was determined by the sweeping direction that minimized the sum of sub region altitude consisting of the diameter functions of hole and perimeter depending on the rotating angle θ of the polygonal as an obstacle. Altitude is defined as the minimum height of cell when the cell is rotated by θ degree. As shown in Fig. 3b, the turning number is proportional to the altitude of the cell. Therefore, to minimize the turning number, the sweep direction should be synchronized with the altitude of the cell.

For rectangular shape cell, the altitude $A(i)$ of the i^{th} cell can be calculated as:

$$A(i) = \min(a_i, b_i), \tag{3}$$

where a_i and b_i is the length and the width of the i^{th} cell and $i = (1,2,3,\cdots)$ is the cell number.

The sweep direction $Dir(i)$ of the i^{th} cell can be obtained as follows:

$$Dir(i) = \begin{cases} \text{Vertical} & \text{if } a_i > b_i \\ \text{Horizontal} & \text{if } a_i < b_i \end{cases}. \tag{4}$$

The sum of altitude $S(j)$ of the j^{th} way to split the graph is defined as:

$$S(j) = \sum_{1}^{n} A(i) \quad \text{for } i = 1,2,3,\cdots, \tag{5}$$

where n is the total number of cell and $j = (1,2,3,\cdots)$ is the number of way to split the graph $T(x,y)$ into sub graphs.

The main idea of the MSA is to find the minimum value of sum of altitude $S(j)$. The minimal sum of altitude (MSA) is defined as

$$MSA = \min(S(1), S(2), S(3), \cdots, S(k)), \tag{6}$$

where k is the total number of the possible way to split the graph $T(x,y)$ into sub graphs.

Finally, the total turning number to cover all area N is directly related to the MSA. Therefore, the MSA may reduce N. The value of N is depicted as follows:

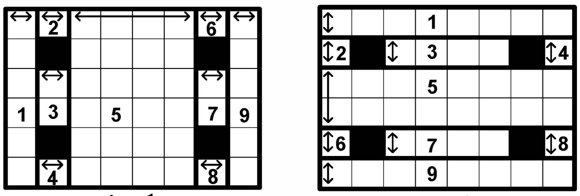
$$N = \frac{1}{L} MSA, \quad \text{for } i = 1,2,3,\cdots \tag{7}$$

where L denotes the robot width.

In [10], the cell decomposition can be obtained by dynamic programming formulation. This approach is difficult to apply in complex environment. To overcome this problem, a new approach is introduced in this paper. The optimal cell decomposition is obtained by combining the result of vertical cell decomposition and horizontal decomposition.

To do this, firstly, Morse cell decomposition for discrete map representation as in [6] is applied to the binary map in Fig. 2 to divide the whole area of the work space of the ocean floor into smaller areas called cells.

In Morse cell decomposition, it is assumed that the work environment has simple structure and is in non-polygonal space. The cell decomposition can be obtained by sweeping the line horizontally from the left to the right to find the critical points of the polygon obstacles and then drawing lines at only those vertices which can be extended vertically in both upward and downward directions. The result is shown in Fig. 4a. If the sweeping line is in vertical direction, the cell decomposition is shown in Fig. 4b.



$j=1$ $\quad$ $j=2$

a. Horizontal $\quad$ b. Vertical

Fig. 4. Horizontal and vertical cell decomposition.

The next step is to calculate the $S(j)$ of cell decomposition using Eq. (5). It can be seen that the sum of altitude for horizontal cell decomposition $S(1)$ is 12 seen from Fig. 4a and the sum of altitude for vertical cell decomposition $S(2)$ is 10 from Fig. 4b. Using Eq. (6), it can be calculated that the value of MSA is 10 obtained from vertical cell decomposition method.

To obtain the better MSA value, both vertical cell decomposition and horizontal are combined. The idea is by adding cells from cell decomposition with higher MSA to cell decomposition with lower MSA. In the case of Fig. 4, cells from 1 to 9 obtained from Fig. 4a are exchanged into Fig. 4b in turn as shown in Fig. 5. There are 9 ways to split the graph $T(x, y)$ into sub graphs as shown in Fig. 5.

Firstly, add one cell from cell decomposition with higher MSA to cell decomposition with lower MSA. Secondly, evaluate the current sum of altitude $S(j)$ using Eq. (5). If the sum of altitude of the current combination $S(j)$ is smaller or equal to that of the previous combination $S(j-1)$, the current combination is saved and will be used for next calculation. Otherwise, the current combination is discarded. Thirdly, repeat this process until all cells are added.

As shown in Fig. 5a, the sum of altitude $S(3)$ after adding the cell number 1 from Fig. 4a to the Fig. 4b is 9. Because $S(3)$ is less than the $S(2)$ that is 10, the current cell decomposition is saved. Next step is adding cell number 2 from Fig. 4a to Fig. 4b as shown in Fig. 5b. The current sum of altitude $S(4)$ becomes 10. Since the $S(4)$ is higher than $S(3)$, the current combination is discarded. This process is repeated until all cells from Fig. 4a are added to Fig. 4b. The results of this process are shown in Fig. 5. The number of cell and the sum of altitude of each step are shown in Table 1. Table 1 shows that the MSA is 8 obtained at $j=11$ as shown in Fig. 5i.

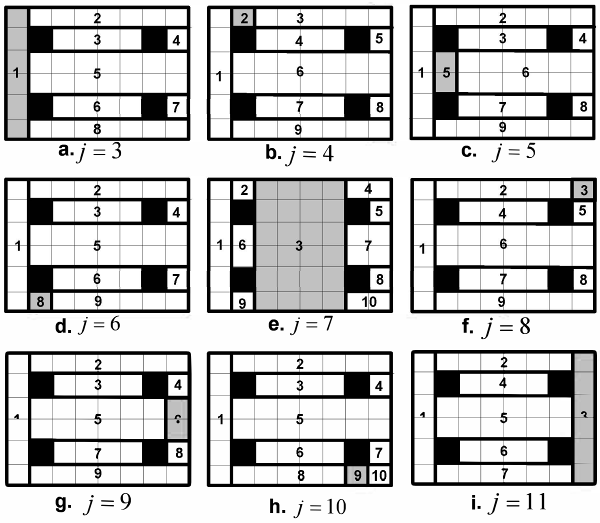


Fig. 5. Combination of vertical and horizontal cell decompositions.

Table 1 Number of cell and the sum of altitude of each step obtained from proposed method

Way to split the graph j	Total number of cell n	Sum of altitude $S(j)$
$j=1$	9	12
$j=2$	8	10
$j=3$	8	9
$j=4$	9	10
$j=5$	9	10
$j=6$	9	10
$j=7$	10	14
$j=8$	9	10
$j=9$	9	10
$j=10$	10	11
$j=11$	7	8

2.3. Generating coverage path

To generate the coverage path, firstly, the coverage sequence is calculated by depth-first search (DFS) algorithm [13]. The purpose of this algorithm is to obtain the coverage sequence with minimum overlap path. In DFS, each cell is assumed as a vertex. Given the cell decomposition result, the adjacent spanning tree connecting all vertices can be obtained. The spanning tree is defined by using the connected relationship, distance relationship and obstacles among cells. The DFS is a preorder traversal method. In this algorithm, each vertex is visited before its children. However, to prevent infinite loops, each vertex is only visited once.

For example, in Fig. 6, a vertex A is chosen as the start vertex. Since the vertex A has vertices B, C, and D as the children, the vertex B is chosen arbitrarily as next vertex. Instead of travelling to the children of the vertex A, travelling to the children of the vertex B is taken. The vertex E is chosen as the next vertex arbitrarily. However, since the vertex E doesn't have any children, the algorithm returns the vertex B, the parent of the vertex E and chooses the vertex F of the children of the previous vertex B. The next vertex is D. Since A has been visited, to prevent infinite loop, instead of visiting the vertex A, the next vertex chosen is the vertex C. The final result is A-B-E-F-D-C.

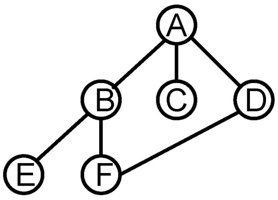


Fig. 6. DFS algorithm.

Finally, to travel between cells, a visibility graph-based path planning is proposed to minimize the overlap path. Visibility graphs may be used to find Euclidean shortest paths among a set of polygonal obstacles. This algorithm consists of two steps: constructing the visibility graph, and applying a shortest path algorithm such as Dijkstra's algorithm to the graph.

3. SIMULATION RESULT

To verify the effectiveness of our proposed method, several simulations are done. Fig. 7 shows the bathymetric map representing the ocean floor as input for simulation.

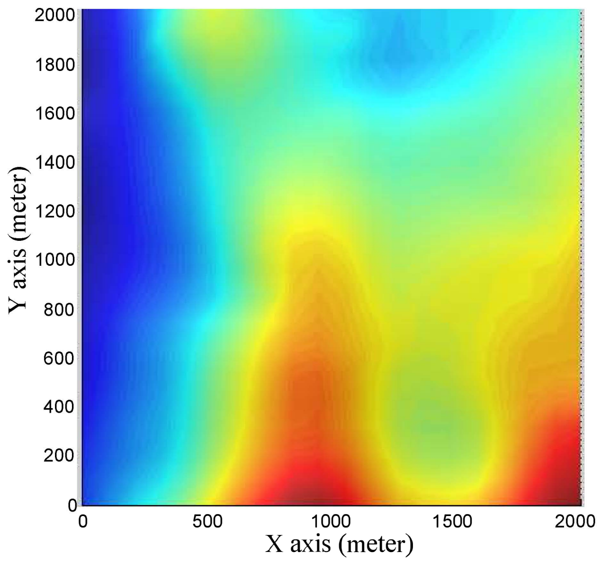


Fig. 7. Bathymetric map as input.

3.1 Binary map

Fig. 8 shows the binary map obtained from high-slope region detection by Eq. (2) of bathymetric map in Fig. 7. Filled black areas are the area that has slope larger than threshold value, and are considered as obstacles. Based on this map, the work environment is divided into several cells.

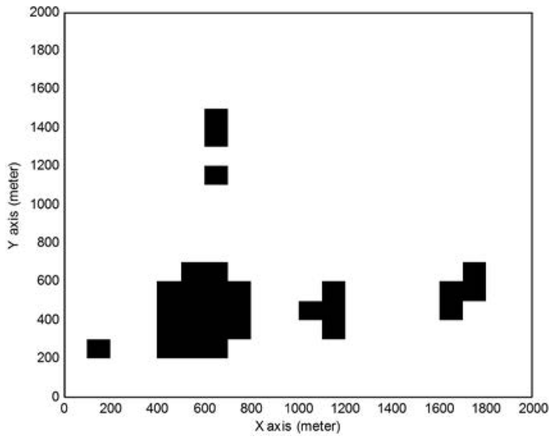


Fig. 8. Binary map.

3.2 Minimal sum of altitude cell decomposition

Figs. 9-11 show the comparisons among vertical cell decomposition, horizontal cell decomposition, and minimal sum of altitude cell decomposition.

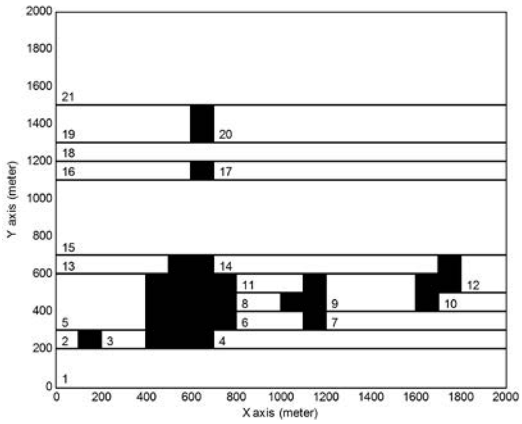


Fig. 9. Vertical cell decomposition.

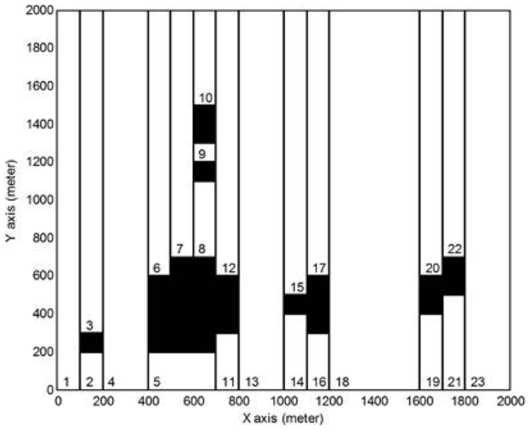


Fig. 10. Horizontal cell decomposition.

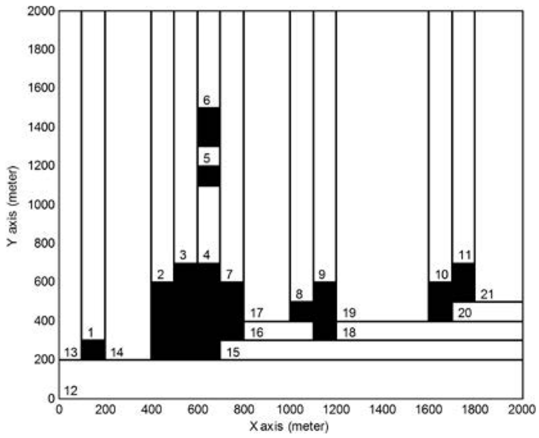


Fig. 11. Minimal sum of altitude cell decomposition.

Simulation result show that the sum of altitude of horizontal cell decomposition is 2425m, resulting 97 times turning. The sum of altitude for vertical cell decomposition is 2975m, resulting 119 times turn. The sum of altitude of proposed algorithm is 2275m, resulting 91 times turn. Compared with vertical and horizontal cell decompositions, the proposed algorithm reduces minimal sum of altitude and minimal turning number.

3.3 Total coverage

The coverage sequences obtained from DFS algorithm are obtained as follows:

Vertical: 21-19-18-16-15-12-10-7-4-1-2-5-3-13-6-8-11-14-9-17-20

Horizontal: 23-22-20-16-14-13-11-5-4-2-1-3-6-7-8-12-9-10-15-17-18-19-21

Proposed Algorithm: 21-11-10-19-9-8-17-7-4-3-2-14-1-13-12-15-16-18-20-5-6

Figs. 12 and 13 show the optimal coverage paths generated by the vertical cell decomposition and horizontal cell decomposition, respectively. Fig. 14 shows the optimal coverage path generated by proposed method.

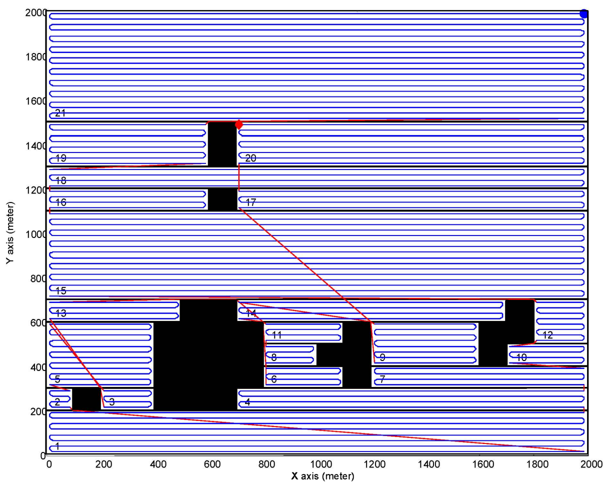


Fig. 12. Path generated by vertical cell decomposition.

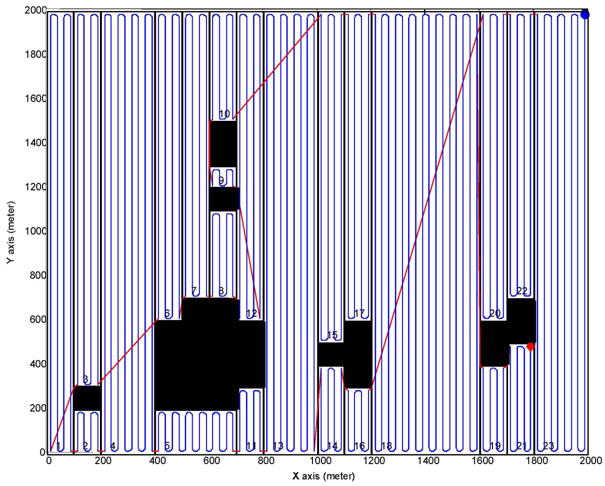


Fig. 13. Path generated by horizontal cell decomposition.

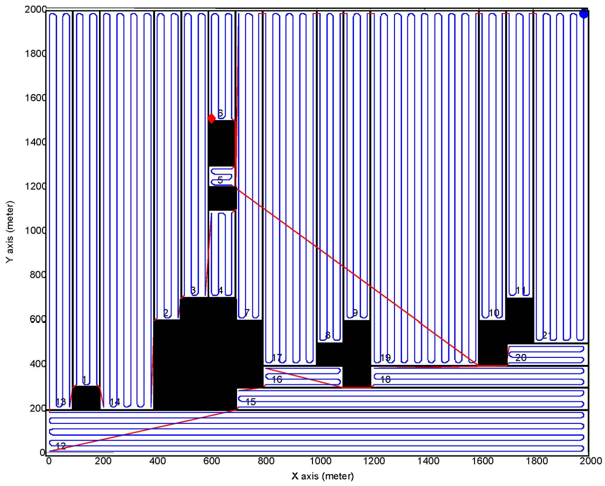


Fig. 14. Path generated by the proposed algorithm.

Table 2 shows the comparison of the path planning algorithms mentioned above. Table 2 shows that the proposed algorithm successfully generates the path with less overlapped path, with minimum turning number, and has fast coverage time compared to the vertical and horizontal cell decomposition as shown in Figs. 12-14.

Table 2 Comparison path planning algorithms.

Items	Vertical	Horizontal	Proposed
Overlapping path	9,881 m	6,5968m	5,610 m
Turning number	119 turn	101 turn	91 turn
Coverage time	12,096s	10,383s	9,647s

4. CONCLUSION

This paper proposed a path planning algorithm for a type of an underwater mining robot to coverage ocean floor to collect its minerals. By applying threshold to the high slope regions of the ocean floor, a binary map was created. Based on Morse cell decomposition the binary map was divided into cells in the vertical and horizontal direction. To minimize the turning number, a minimal sum of altitude method was proposed. The simulation result showed that the propose algorithm generated the shorter coverage path, less overlapped path, had less turning number and had faster coverage time compared to vertical and horizontal cell decomposition methods.

ACKNOWLEDGEMENT

This study is a part of the results of the R&D project, Development of a deep-Seabed Mining Technologies, supported by Korea Government, Ministry of Oceans and Fisheries. Authors are indebted to appreciate their full supports of this pioneering work.

REFERENCES

[1] E. M. Arkin, R. Hassin, "Approximation algorithms for the geometric covering salesman problem," *Discrete Applied Mathematics*, Vol. 55, pp. 197-218, 1994.

[2] P. M. Hsu, C. L. Lin, "Optimal planner for lawn mowers," *Proc. of the International Conference on Cybernetic Intelligent Systems (CIS)*, pp. 1-7, 2010.

[3] J. S. Oh, Y. H. Choi, J. B. Park, Y. F. Zeng, "Complete coverage navigation of cleaning robots using triangular-cell-based Map," *IEEE Transaction on Industrial Electronics*, Vol. 51, No. 3, pp. 718-726, 2004.

[4] J. Valante, J. D. Cerro, A. Barrientos, D. Sanz, "Aerial coverage optimization in precision agriculture management: A musical harmony inspired approach," *Computers and Electronics in Agriculture*, Vol. 99, pp. 153–159, 2013.

[5] G. Zuo, P. Zang, J. Qiao, "Path planning algorithm based on sub-region for agricultural robot," *Proc. of the International Asia Conference on Informatics in Control, Automation and Robotics (CAR)*, pp. 197-200, 2010.

[6] P. S. Pratama, J. W. Kim, J. H. Min, S. M. Yoon, T. K. Yeu, S. Hong, H. K. Kim, S. B. Kim, "Path Planning Method for Coverage of Uneven Terrain Environment Using Morse Cell Decomposition," *Proc. of the International Conference on Korean Society of Ocean Engineer*, pp. 151-154, 2014.

[7] D. H. Kim, G. Hoang, M. J. Bae, J. W. Kim, S. M. Yoon, T. K. Yeu, S. Hong, S. B. Kim, "Path tracking control coverage of a mining robot based on exhaustive path planning with exact cell decomposition," *Proc. of the International Conference on Control, Automation and Systems*, pp. 730-735, 2014.

[8] P. A. Jimenez, B. Shirinzadeh, A. Nicholson, G. Alici, "Optimal area covering using genetic algorithms," *Proc. of the International Conference on Advanced Intelligent Mechatronics*, pp. 1-5, 2007.

[9] Y. Gabriely, E. Rimon, "Spanning-tree based coverage of continuous areas by a mobile robot," *Annals of Mathematics and Artificial Intelligence*, Vol. 31, No. 10, pp. 77-98, 2001.

[10] W. H. Huang, "Optimal line-sweep-based decompositions for coverage algorithms," *Proc. of the International Conference on Robotics and Automation*, pp. 27-32, 2001.

[11] E. Galceran, M. Carreras, "Planning coverage paths on bathymetric maps for in-detail inspection of the ocean floor," *Proc. of the International Conference on Robotics and Automation*, pp. 4159-4164, 2013.

[12] R. C. Prim, "Shortest connection networks and some generalizations," *Bell Labs Technical Journal*, Vol. 36, pp. 1389-1401, 1957.

[13] E. Shimon, Graph Algorithms (2nd ed.), Cambridge University Press, pp. 46–48, 2011.