A Search and Coverage Algorithm for Mobile Robot

Seuk-Woo Ryu [1], Young-ho Lee[2] Tae-Yong Kuc [3] Sang-Hun Ji[4,] and Yong-Seon Moon[5]
[1,2,3] Department of Electrical Engineering, the Sungkyunkwan University, Suwon, Korea.
(Tel : +82-31-290-7202; E-mail: tykuc@yurim.skku.ac.kr)
[4,] Robot Convergence Research Center of the Korea Institute of Industrial Technology, Ansan, Korea.
(Tel : +81-31-8040-6363; e-mail: robot91@kitech.re.kr)
[5] Department of Electrical Engineering, Sunchon National University, Sunchon, Korea.
(E-mail: moon@sunchon.ac.kr)

Abstract - This paper presents a new work space modeling and search algorithm for complete coverage of robotic environment. The proposed algorithm named spatial cell diffusion (SCD) encodes the target area as groups of Gray codes for grid cells with size of instantaneous robot coverage and extends its sweep area by diffusing occupied cells outwards through continuous spiral movement. Based on the direct and rotational motions of robot, directions of search are categorized into eight ways by combining four cardinal points {N, S, W, E} with two senses of rotation {CW, CCW}. According to the direction of search selected, robot navigates and sweeps the target region cell by cell in consistent manner. While executing a coverage operation, robot changes its direction of motion and reverses the sense of rotation at a boundary cell of work space model. The spatial cell diffusion search algorithm proposed can be applied to on-line coverage of unknown environment as well as off-line coverage of known work space. The feasibility and effectiveness of the algorithm are demonstrated by simulation results for coverage work of target area.

Keywords - robot coverage, work space, Gray coded cell, space cell diffusion search

1. Introduction

A coverage algorithm has to guide a robot over all points of target workspace in a priori known or unknown environments. Once an effective coverage algorithm is constructed, it can be applied to an area coverage task as well as a sensor coverage problem. Area inspection, map building, surveillance, reconnaissance, floor cleaning, paint spraying, lawn mowing, field demining, plowing, and harvesting are applications which require a coverage algorithm for robotic automation.

In an effort to put it to practical use, the coverage problem has been studied by many researchers using various strategies [14], [17], [19], [16], [9], [2], [11], [6]. Two most popular approaches to work space modeling for robot coverage are cellular decomposition and grid-based representation [4], [5], [19], [11]. Based on the modeling approaches, the sensor coverage and area coverage problems have been attacked in the related literature. Lumelsky *et al.*[14] proposed the Seed Spreader algorithm which dealt with sensor coverage problem of robot vision. Assuming that vision sensor can identify all the intersection points, their algorithm guarantees complete coverage by providing the robot with parallel lines to follow. Choset and Pignon [4], Hert *et al.* [12], and Acar and Choset [1] further developed cellular decomposition algorithms which guarantee the complete coverage of regional cells by contiguous line sweep. Another decomposition approach to convex world model was proposed by Huang [10] in which a line sweep sequence is chosen so that it minimizes the amount of rotational motion of robot. In the work of Mannadiar and Rekleitis [15], the Boustrophedon cellular decomposition was combined with graph theory to generate an Euler tour which guarantees complete coverage of the known work space while minimizing the traveled path by the robot.

On the other hand, the grid-based modeling and coverage algorithms were developed by defining fine cells of grids for work area decomposition [19], [11], [20]. Similar to cellular decomposition methods, the grid-based coverage algorithms focus on the path planning of robot to guarantee complete coverage of work area by visiting every cell without missing. A grid-based coverage algorithm was proposed by Zelinsky *et al.*[19] which uses a kind of potential field and generates a path of steepest descent from the starting point to the goal. Gabriely and Rimon [11] presented a grid-based approach for complete coverage of continuous area. In their scheme, they decomposes the work area into square grids of size 2D, two times of tool size D, and their spanning tree algorithm generates a continuous path which visits contiguous grids without missing. Other grid-based algorithms developed by Zheng *et al.*[20] guarantee an optimal performance for known environment, where a grid is overlapped with the one with robot size.

In this paper, a new grid-based coverage algorithm is proposed for complete coverage of work area. Our coverage algorithm is aiming at handling a general coverage task including:

1) the case where a grid map of the environment can be acquired for off-line planning and

2) the case when a grid map of the environment is not

978-1-4577-0723-0 / 11 / $26.00

obtainable in advance.

In the grid map, the grid size is assumed to be less than or equal to the footprint of robot for area coverage task and the instantaneous unit coverage of a sensor for sensor coverage task, respectively. It is also assumed that the robot is equipped with not only a position and orientation sensor but an obstacle detection sensor to perform an on-line coverage work.

With the assumptions, a coverage algorithm is presented below. In the proposed coverage algorithm, the grid cells of target area are encoded as Gray codes to reduce memory usage of robot while performing area search and coverage work. The proposed coverage algorithm called spatial cell diffusion (SCD) scans Gray coded cells and extends its sweep area by diffusing occupied cells outwards through continuous spiral movement. Therefore, only the Gray code of former cell visited is required to memorize along with the current cell group and direction of search. This makes the memory requirement for coverage work with Gray coded cells reduces to O(1). On the other hand, based on the direct and rotational motions of robot, directions of search are categorized into eight ways by combining four cardinal points {N, S, W, E} with two senses of rotation {CW, CCW}. According to the direction of search selected, robot navigates and sweeps the target region cell by cell in consistent manner. While executing a coverage operation, robot changes its direction of motion and reverses the sense of rotation at a boundary cell of work space or obstacle. In our SCD algorithm, when the robot starts at the center of square area, its coverage time T is calculated as $T = \alpha n$ for free space, where α is the unit coverage time of single grid cell and n the number of grids in the work area. Otherwise, it amounts to $T = \alpha m + \beta$, $\beta \in [0, m\alpha)$ where m is the number of grids not occupied by an obstacle in the work space. For more details on the coverage time, see Section 3.

The outline of this presentation is as follows. Section 2 describes the cellular decomposition for work space modeling and representation as groups of neighboring grid cells for coverage execution along with the definition of possible directions of search. Section 3 explains our approach to off-line and on-line coverage problems. Section 4 demonstrates and analyzes experiment results by using a simulation program. The final section summarizes the paper.

2. Work Space Modeling

The operation of proposed coverage algorithm consists of work space representation and execution of coverage algorithm. In this section, a spatial grid based model is introduced for decomposition of work space.

To begin with, the geometry of target area is first divided into coordinated grid cells of area $A = W \times W$ as shown in Fig.1. Hence, the area of footprint of robot is assumed to be larger than or equal to A for area coverage problem and $W = \frac{R}{\sqrt{2}}$ for unit range R of instantaneous sensor measurement for sensor coverage task. In the figure, the origin of coordinate frame corresponds to the starting position of robot on the grid cell called center grid for the coverage work. The center coordinates of grid cell C_{ij} at location (iW, jW) is denoted by $G(i, j)$ for $i, j = 0, \pm1, \pm2, \pm3, \cdots$. and called sub-goal G_{ij}. Hence, $G(0,0)$ represents the center of grid cell C_{00} at the starting position.

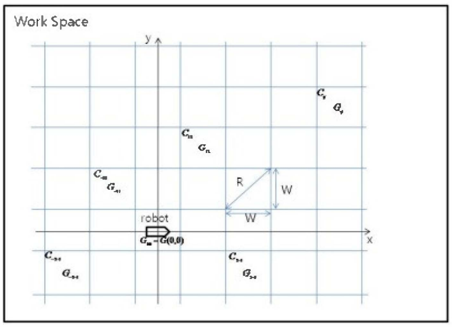

Fig.1. Coordinated Cellular Decomposition of Work Area

2.1 Coordinated Cell Group

With the cellular decomposition of work space, the grids are grouped into the set of neighboring cells as follows.

$S_k^1 = \{C_{ij} \mid G_{ij} = G(i, j)\}$ for $k = 1, 2, \ldots, n$,   (1)

where two coordinate values (i, j) are selected from the permutation $\Pi(A_k, 2)$ with repetition. That is,

$\Pi(A_k, 2) = \{i, j\}$,   (2)

where

$A_0 = \{\phi\}$,

$A_k = A_{k-1} \cup \left\{-\frac{k-1}{2}\right\}$ for odd k,

$A_k = A_{k-1} \cup \left\{\frac{k}{2}\right\}$ for even k.

For example, the first three groups of coordinated cells are determined as follows.

$S_1^1 = \{C_{00} \mid G_{00} = G(0,0)\}$,

$S_2^1 = \{C_{00}, C_{01}, C_{10}, C_{11}\}$,

$S_3^1 = \{C_{00}, C_{01}, C_{10}, C_{11}, C_{-10}, C_{0-1}, C_{-1-1}, C_{1-1}, C_{-11}\}$.

In the above definition of S_k^1, the superscript 1 denotes the first quadrant for the initial direction of robot movement from the starting position G_{00} to North CW(clockwise) or East CCW(counter clockwise). This will be clarified in the sequel.

Now, define the incremental set of cells as

$\Delta S_k(l) = S_k \setminus S_{k-1}$,   (3)

where $l = e_1$ for even k and $l = o_3$ for odd k from Table1.

Further, for completeness of notation, define for even k

i) $\Delta S_k(e_2)$ as the symmetrical displacement of $\Delta S_k(e_1)$ with respective to y axis.

ii) $\Delta S_k(e_3)$ as the symmetrical displacement of $\Delta S_k(e_1)$ with respective to $y = -x$.

iii) $\Delta S_k(e_4)$ as the symmetrical displacement of $\Delta S_k(e_1)$ with respective to x axis.

For odd k, define

i) $\Delta S_k(o_1)$ as the symmetrical displacement of $\Delta S_k(o_3)$ with respective to $y = -x$.

ii) $\Delta S_k(o_2)$ as the symmetrical displacement of $\Delta S_k(o_3)$ with respective to x axis.

iii) $\Delta S_k(o_4)$ as the symmetrical displacement of $\Delta S_k(o_3)$ with respective to y axis.

Then, it follows that

$$S_k^i = S_{k-1}^i \cup \Delta S_k(l^i), \quad (4)$$

where the superscript i represents the i-th quadrant for i=12,3,4 and each l^i takes one of the following four cases from Table 1.

Table 1 Standard space cell decomposition model

k	l^1	l^2	l^3	l^4
even	e_1	e_2	e_3	e_4
odd	o_3	o_4	o_1	o_2

In terms of movement direction for robot steering, the superscript i differentiates the following four cases and 8 possible directions of search in that

$$i = \begin{cases} 1 & \text{(North CW or East CCW)} \\ 2 & \text{(North CCW or West CW)} \\ 3 & \text{(South CW or West CCW)} \\ 4 & \text{(South CCW or East CW)} \end{cases} \quad (5)$$

The above definitions of four quadrants are depicted in Fig. 2.

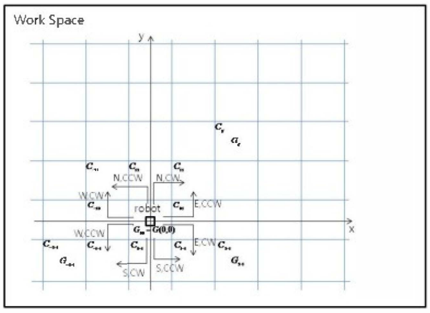

Fig.2. Definitions of Four Quadrants with 8 Directions of Robot Movement

As an illustrative example for the model of spatial cell decomposition, the first three cell groups of fourth quadrant, S_k^4, are expressed as

$S_1^4 = \{C_{00}\}$,

$S_2^4 = S_1^4 \cup \Delta S_2(e_4) = \{C_{00}, C_{10}, C_{1-1}, C_{0-1}\}$,

$S_3^4 = S_2^4 \cup \Delta S_3(o_2)$

$= \{C_{00}, C_{10}, C_{1-1}, C_{0-1}, C_{-1-1}, C_{-10}, C_{-11}, C_{01}, C_{11}\}$

With the definitions introduced so far, the next section presents an effective search and sweep algorithm derived for complete coverage of target area.

Alternatively, the movement direction of robot from the initial position to South CW or West CCW can be selected to model the set of coordinated grid cells as follows.

$$S_k^3 = \{C_{ij} |\, G_{ij} = G(i,j)\} \text{ for } k = 1,2,\dots,n, \quad (6)$$

where two coordinate values (i,j) are selected from the permutation $\Pi(A_k, 2)$ with repetition. Here,

$$\Pi(A_k, 2) = \{i,j\}, \quad (7)$$

where

$A_0 = \{\phi\}$,

$A_k = A_{k-1} \cup \left\{\frac{k-1}{2}\right\}$ for odd k,

$A_k = A_{k-1} \cup \left\{-\frac{k}{2}\right\}$ for even k.

Then, the incremental set of grid cells are defined as

$$\Delta S_k(l) = S_k \setminus S_{k-1}, \quad (8)$$

where $l = e_3$ for even k and $l = o_1$ for odd k from Table 1.

The cell groups S_k^i for other quadrants $i = 1,2,4$ are defined as in (4).

2.2 Types of Cells

There are 4 types of grid cells in the presented cellular decomposition of work space including obstacles.

T_0 : the cell belongs to an obstacle free region.

T_{10} : the cell partly overlapped with or touching obstacle or work space boundary and centered at reachable sub-goal G_{ij}.

T_{11} : the cell partly overlapped with obstacle or work space boundary and centered at unreachable sub-goal.

T_2: the cell completely occupied by an obstacle.

In this classification, the sub-goal positions G_{ij}'s of coordinates for cells $\{T_0, T_{10}\}$ are reachable by mobile robot, while cells $\{T_{11}, T_2\}$ are not. Further, T_0 has the four full directions of approach, $\{T_{10}, T_{11}\}$ at least one possible direction of access but less than four, and T_2 no direction of approach.

Based on the developed work space model, spatial cell diffusion (SCD) coverage algorithm is proposed for search and complete sweep of target area in the following section.

.

3. Spatial Cell Diffusion Coverage Algorithm

Above all, consider the coverage problem of an obstacle free region. In this case, there exists two kinds of cells in the region: free region cell T_0 and boundary cell $\{T_{10}$ or $T_{11}\}$. The boundary (touching) cells are further classified into Bridge Cell, Junction Cell, and Brook Cell.

1) **Bridge Cell** is defined as a boundary cell at which robot changes its movement direction from CW to CCW or vice versa.
2) **Junction Cell** is a boundary cell which splits the grid cells of forthcoming sweep region into two or more subgroups.
3) **Brook Cell** is the boundary cell whose sweep direction is aligned with the border line of work space or obstacle boundary.

Refer to Fig. 3 and 4 for more details.

3.1 SCD Coverage Algorithm

Input: Current position and orientation of robot, Gray coded grid map for off-line coverage, Search and encoding rule of Gray code for on-line coverage.

Procedure:

1) **(Optional)** When necessary, divide the target work area into two or more subareas and perform the steps below for each region independently.
2) Set the center grid for starting position of robot coverage.
   A. Set a global x-y coordinate system at the center grid.
   B. Reset a global x-counter and y-counter.
3) Choose the quadrant number i ($i = 1,2,3,4$).
4) For the quadrant number i selected, determine one direction of search from the pairs in (5).
5) **If (Off-line Search), then**
   {Referring to the coordinated cell group S_k^i, navigate Gray coded cells in accordance with the selected direction of search for grid cells surrounding the robot. Gray codes corresponds to a search tree, since two neighboring cells differ only a unit digit of subscript (or coordinate) each other.
   A. Increase or decrease the global and local x-y counters according to the movement of robot.
   B. Count the number of cells visited.
   }
   Else (On-line Search)
   {In accordance with the selected direction of search, navigate grid cells surrounding the robot generating the coordinated cell group S_k^i of Gray codes.
   A. Increase or decrease the global and local x-y counters according to the movement of robot.
   B. Count the number of cells visited.
   (Optional) In the presence of an obstacle, when necessary, circumnavigate the obstacle to identify the occupied cells and boundary cells.
   }
6) When robot reaches a boundary (touching) cell, do one of the following:
   A. For a Bridge Cell, change the direction of search, i.e., CW to CCW or CCW to CW.
   B. For Brook Cells, cover the cells.
   C. For a Junction Cell which separates the forthcoming grid cells into two or more small groups, record the Junction Cell and the number of separate cell groups. Cover the area first including cells consistent in the direction of cell diffusion. After covering one region, return to the Junction Cell recorded. If an inconsistent cell appears before completing the covering of an area, set the cell as a new starting cell with a new local coordinate system and local x-y counters. Cover the remaining region and return to the Junction Cell recorded.

   Then, check if the direction of search for new expansion of Gray coded cells is still consistent with the previous diffusion rule.
   A. If it is, continue the covering.
   B. Otherwise, set a new starting cell next to the boundary cell as the origin of local x-y coordinate system and set local x-y counters. Begin to generate and sweep Gray coded cells by choosing a new direction of search for the new starting cell.
7) When it completes the sweep of all the coordinated grid cells in terms of Gray codes, end.

In case of off-line search in 5), robot can circumnavigate an obstacle to identify its dimension and plan its coverage path around the obstacle. For example, see Fig. 5. The starting position, the origin of global coordinate system and the positions of Junction Cells are recorded. In the above coverage operation, counters are used for the robot to return to the previous Junction Cell for coverage of separate regions.

Fig. 3 illustrates the spatial cell diffusion approach for free space coverage, where the initial direction of movement has been chosen as North CW. In the figure, C_{op} is Bridge Cell at which robot changes its movement direction from CW into CCW while expanding its region of coverage. Junction Cell is also found in the figure and marked as C_{mn} and Brook Cells are denoted along the boundary of work area. The ordered coordinated cell groups are represented by Gray codes for search as follows.

$S_1^1 = \{C_{00}\}$,
$S_2^1 = S_1^1 \cup \Delta S_2(e_1) = \{C_{00}, C_{01}, C_{11}, C_{10}\}$,
$S_3^1 = S_2^1 \cup \Delta S_3(o_3) =$
$\{C_{00}, C_{01}, C_{11}, C_{10}, C_{1-1}, C_{0-1}, C_{-1-1}, C_{-10}, C_{-11}\}$.
$S_4^1 = S_3^1 \cup \Delta S_4(e_1) =$
$\{C_{00}, C_{01}, C_{11}, C_{10}, C_{1-1}, C_{0-1}, C_{-1-1}, C_{-10}, C_{-11},\}$
$\cup\{C_{-12}, C_{02}, C_{12}, C_{22}, C_{21}, C_{20}, C_{2-1}\}$.
$S_5^1 = S_4^1 \cup \Delta S_5(o_3) = \{C_{00}, C_{01}, C_{11}, C_{10}, C_{1-1}, C_{0-1},$
$C_{-1-1}, C_{-10}, C_{-11}, C_{-12}, C_{02}, C_{12}, C_{22}, C_{21}, C_{20}, C_{2-1}\}$
$\cup\{C_{2-2}, C_{1-2}, C_{0-2}, C_{-1-2}, C_{-2-2}, C_{-2-1}, C_{-20}, C_{-21}, C_{-22}\}$
$S_6^1 = S_5^1 \cup \Delta S_6(e_1)$,
$S_7^1 = S_6^1 \cup \delta S_8(e_1)$, $S_8^1 = S_7^1 \cup \delta S_7(o_3)$, $S_9^1 = S_8^1 \cup \delta S_{10}(e_1)$,
where $\delta S_8(e_1) \subset \Delta S_8(e_1)$, $\delta S_7(o_3) \subset \Delta S_7(o_3)$, and
$\delta S_{10}(e_1) \subset \Delta S_{10}(e_1)$.

Note in this coverage example that the coordinated cell groups S_7^1, S_8^1 and S_9^1 are different from the standard space cell decomposition model of Table 1. This is due to the change of movement direction at Bridge Cells from CW to CCW and vice versa. Nevertheless, the cell diffusion around boundary (touching) cells is consistent in that the direction of movement for visit of cell groups goes in order. Otherwise, a new starting cell after Junction Cell should have been generated. For comparison see Fig.4.

Remark 1: In general, the coverage time of SCD algorithm amounts to $T = \alpha n + \beta, \quad \beta \in [0, n\alpha)$, where α is the unit coverage time of single grid and n the number of reachable grids in the work space. In Fig. 3, it becomes $\beta = 7\alpha$ counting the additional time for grid cells visited twice after the Junction Cell.

Remark 2: The length of traveled path corresponding to the coverage time T of SCD algorithm is calculated as $l = W(n-1+m)$, where W is the unit length of grid cell, *n* the number of reachable grids in the work space, and *m* the number of grids with multiple visits.

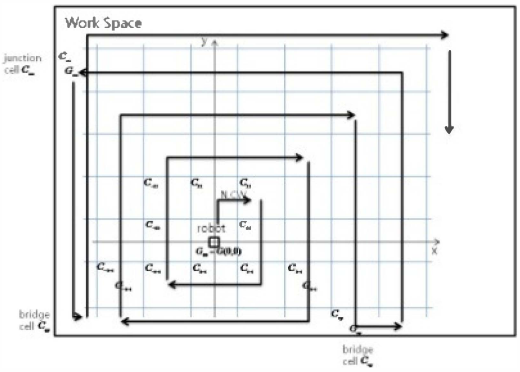

Fig.3. Spatial Cell Diffusion in Free Space

Remark 3: In the proposed coverage algorithm, since the Gray code is used instead of search tree representation, the memory requirement reduces to O(1) for the critical cells including the starting and Junction Cells.

When there are obstacles within the work space, the work space model includes four kinds of cells, $\{T_0, T_{10}, T_{11}, T_2\}$. In view of accessibility, type T_0 cells has 4 open directions of approach, while type T_{10} or T_{11} has 1, 2, 3, or 4 open directions and type T_2 no open direction of approach. An example of spatial cell diffusion approach for non-free space coverage is given in Fig. 4. In the figure, robot avoids the obstacles and steers to the next cell consistent in the direction of search. Fig. 5 shows the SCD coverage algorithm applied to a work area containing a matrix array of obstacles smaller than the size of grid cell. There are Bridge Cells along the bottom and left borders and Brook Cells on the top of boundary. To sweep each cell T_{10} or T_{11}, robot goes round each obstacle in the same direction as the SCD search. In this case, the grid cells occupied by small obstacles are subdivided into four smaller cells to compute coverage time. The total coverage time reads

$$\alpha n < T \le \alpha(n+3m) + \beta, \quad \beta \in [0, n\alpha),$$

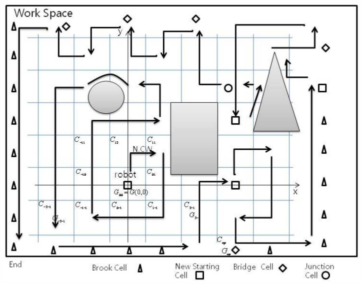

Fig.4. Spatial Cell Diffusion in Non-Free Space

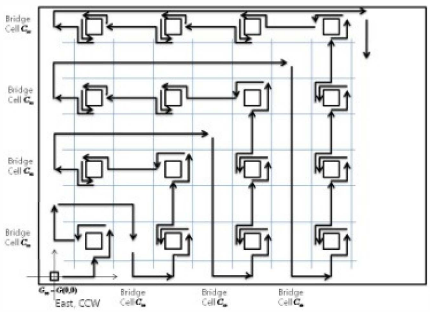

Fig.5. Spatial Cell Diffusion in Non-Free Space with Small Obstacle Array

where α is the unit coverage time of single grid, *n* the number of grids covering free space, and *m* the number of grid cells occupied by obstacles. Here, β denotes additional time for grid cells while returning to Junction Cells.

4. Simulation Examples

To test the feasibility and effectiveness of developed coverage algorithm, numerous simulations have been performed to search and cover various environments. In the sequel, only a couple of simulation results are demonstrated due to page limitation. Fig.6 shows two free space examples similar to Fig.3. In the first example, the initial direction of search is taken as E-CW and the SCD procedure is executed with only Bridge Cell operation without any Junction Cell, whereas there is a Junction Cell in the second example. Coverage examples with stationary obstacles in the work area are shown in Fig.7. In the figures, as a consequence of SCD search algorithm, two new start cells are generated after boundary cell inspection following the SCD procedure 6). Similarly, Fig.8

demonstrates complete coverage of an office environment with multiple rooms by applying the proposed coverage algorithm

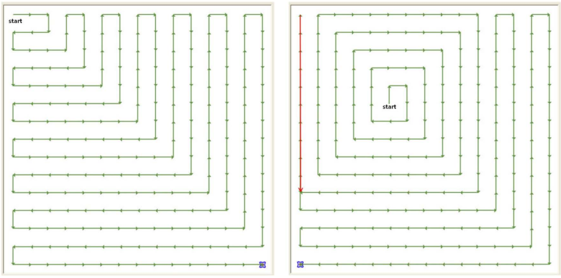

Fig.6. SCD Coverage in Free Space

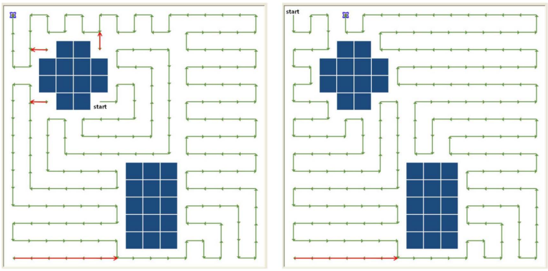

Fig.7. SCD Coverage in Non-Free Space

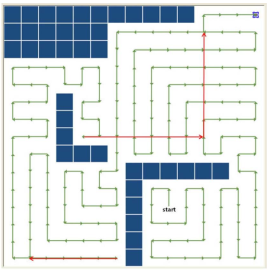

Fig.7. SCD Coverage in Non-Free Space

Table2 ~ Table4 compare the performance of SCD algorithm with random search and coverage method for three different work areas of Fig6 ~ Fig8, respectively.

From the comparison tables, it is obvious that the proposed SCD algorithm outperforms random coverage in terms of overall traveling distance, coverage time, rate of coverage, rate of multiple visit cells, and time limited performance of coverage rate.

Table2. Comparison of Free Space Coverage

	Traveling Distance (# of Cells)	# of Cells with Multiple Visit	Coverage Time			Limit Time (SCD Average time)			Rate of Coverage	Rate of Multiple Visit	Initial Direction of Search
			Hours	Minutes	Seconds	Hours	Minutes	Seconds			
SCD	235	10	0	2	53	-	-	-	100.00%	4.26%	N, CW
	225	0	0	2	44	-	-	-	100.00%	0.00%	E, CCW
	225	0	0	2	42	-	-	-	100.00%	0.00%	N, CCW
	225	0	0	2	42	-	-	-	100.00%	0.00%	W, CW
	235	10	0	2	48	-	-	-	100.00%	4.26%	W, CCW
	225	0	0	2	41	-	-	-	100.00%	0.00%	S, CW
	225	0	0	2	41	-	-	-	100.00%	0.00%	S, CCW
	225	0	0	2	41	-	-	-	100.00%	0.00%	E, CW
Random Search & Coverage	5153	4928	1	3	9	-	-	-	100.00%	95.63%	
	229	142	-	-	-	0	2	47	38.16%	62.01%	

Table3. Comparison of Non-Free Space Coverage

	Traveling Distance (# of Cells)	# of Cells with Multiple Visit	Coverage Time			Limit Time (SCD Average Time)			Rate of Coverage	Rate of Multiple Visit	Initial Direction of Search
			Hours	Minutes	Seconds	Hours	Minutes	Seconds			
SCD	199	15	0	2	42	-	-	-	100.00%	7.54%	N, CW
	208	10	0	2	29	-	-	-	100.00%	4.81%	E, CCW
	208	10	0	2	29	-	-	-	100.00%	4.81%	N, CCW
	203	19	0	2	24	-	-	-	100.00%	9.36%	W, CW
	203	19	0	2	24	-	-	-	100.00%	9.36%	W, CCW
	206	18	0	2	26	-	-	-	100.00%	8.74%	S, CW
	199	15	0	2	42	-	-	-	100.00%	7.54%	S, CCW
	203	19	0	2	24	-	-	-	100.00%	9.36%	E, CW
Random Search & Coverage	4951	4763	0	59	9	-	-	-	100.00%	96.20%	
	222	127	-	-	-	0	2	42	41.67%	57.21%	

Table4. Comparison of Coverage in Office Environment

	Traveling Distance (# of Cells)	# of Cells with Multiple Visit	Coverage Time			Limit Time (SCD Average Time)			Rate of Coverage	Rate of Multiple Visit	Initial Direction of Search
			Hours	Minutes	Seconds	Hours	Minutes	Seconds			
SCD	208	23	0	2	28	-	-	-	100.00%	11.06%	N, CW
	206	22	0	2	26	-	-	-	100.00%	10.68%	E, CCW
	309	24	0	2	43	-	-	-	100.00%	7.77%	N, CCW
	305	20	0	2	40	-	-	-	100.00%	6.56%	W, CW
	208	23	0	2	28	-	-	-	100.00%	11.06%	W, CCW
	206	22	0	2	26	-	-	-	100.00%	10.68%	S, CW
	213	28	0	2	31	-	-	-	100.00%	13.15%	S, CCW
	206	22	0	2	26	-	-	-	100.00%	10.68%	E, CW
Random Search & Coverage	4115	3930	0	51	1	-	-	-	100.00%	95.50%	
	207	149	-	-	-	0	2	31	25.44%	71.98%	

4. Conclusion

This paper presents a grid-based coverage algorithm named spatial cell diffusion (SCD) which encodes the target area as a group of Gray codes for grid cells with size of instantaneous robot coverage and extends its sweep area by diffusing occupied cells outwards through continuous movement. The coverage time and traveled distance of SCD algorithm are in proportion to the number of grid cells in the work space. The number of memory for on-line coverage using SCD algorithm reduces to O(1) for recording of critical cells and Junction Cells. The SCD algorithm can be applied to unknown environment as well as known work space. The feasibility and effectiveness of the algorithm are demonstrated by simulation results for coverage work of target area.

Acknowledgement

This work was supported in part by Knowledge and Economics Department of Korea Government through Original Industrial Technology Development Project for Collective Intelligence Robot (Grant 10037352).

References

[1] E.U. Acar and H. Choset, "Sensor-based Coverage of Unknown Environments: Incremental Construction of Morse Decompositions," *The International Journal of Robotics Research*, vol.21, no.4, pp.345-366, 2002

[2] E.U. Acar, H. Choset, Y. Zhang, and M. Schervish, "Path Planning for Robotic Demining: Robust Sensor-based Coverage of Unstructed Environments and Probabilistic Methods," *The International*

Journal of Robotics Research, vol.22, no.7-8, pp.441-466, 2003.

[3] K. Cheng and P. Dasgupta, "Dynamic Area Coverage using Faulty Multi-Agent Swarms," *IEEE/WIC/ACM Int. Conf. on Intelligent Agent Technology*, pp.17-23, 2007.

[4] H. Choset and P. Pignon, "Coverage path planning: The boustrophedon cellular decomposition," *Int. Conf. on Field and Service Robotics*, Canberra, Australia, 1997.

[5] H. Choset, "Coverage of known space: The boustrophedon cellular decomposition," *Autonomous Robots*, vol.9, pp.247-253, 2000.

[6] N. Correll and A. MArtinoli, "Collective Inspection of Regular Structures using a Swarm of Miniature Robots," M.H. Ang and O.Khatib (EDs.): *Experimental Robotics IX, STAR 21*, pp.375-385, Spring-Verlag Berlin Heidelberg, 2006.

[7] N. Correll and A. MArtinoli, "Robust Distributed Coverage using a Swarm of Miniature Robots," *IEEE Int. Conf. on Robotics and Automation*, pp.379-384, Rome, Italy, 2007.

[8] K. Easton and J. Burdick, "A coverage algorithm for multi-robot boundary inspection, " *IEEE Int. conf. on Robotics and Automation*, pp.727-734, 2005.

[9] H. Enders, W. Feiten, and G. Lawitzky, "Field test of autonomous cleaning in supermarkets," *IEEE Int. Conf. on Robotics and Automation*, pp.1779-1781, 1998.

[10] W.H. Huang, "Optimal Line-sweep-based Decompositions for Coverage Algorithms," *IEEE Int. Conf. on Robotics and Automation*, pp.27-32, Seoul, Korea, 2001.

[11] Y. Gabriely and E. Rimon, "Spanning-Tree Based Coverage of Continuous Areas by a Mobile Robot," *IEEE Int. Conf. on Robotics and Automation*, pp.1927-1933, Seoul, Korea, 2001.

[12] S. Hert, S. Tiwari, and V.J. Lumelsky, "A terrain covering algorithm for an AUV, " *Autonomous Robots*, vol.3, pp.91-119, 1996.

[13] J.W. Kang, S.J. Kim, M.J. Chung, J.H. Park, and S.W. Band, "Path planning for complete and efficient coverage operation of mobile robots," *IEEE Int. Conf. on Mechatronics and Automation*, pp.2126-2131, 2007.

[14] V.J. Lumelsky, S. Mukhopadhyar, and K. Sun, "Dynamic path planning in sensor-based terrain acquisition," *IEEE Trans. On Robotics and Automation,* vol.6, no.4, pp.462-472, 1990.

[15] R. Mannadiar and I. Rekleitis, "Optimal Coverage of a Known Arbitrary Environment," *IEEE Int. Conf. on Robotics and Automation*, pp.5525-5530, Alaska, USA, 2010.

[16] J.D. Nicoud and M.K. Habib, "The PEMEX autonomous demining robot, " *IEEE/RSJ Int. Conf. on Intelligent Robot Systems*, vol.1, pp.419-424, 1995.

[17] M. Schwager, J.J. Slotine, and D. Rus, "Unifying geometric, probabilistic, and potential field approaches to multi-robot coverage control," *Int. Symposium on Robotics Research*, Luserne, Switzerland, 2009.

[18] N.S.V. Rao ad S.S. Iyengar, "Autonoumous robot navigation in unknown terrains: visibility graph based methods," *IEEE transaction on Systems, Man and Cybernetics,* vol.20, no.6, pp.1443-1449, 1990.

[19] A. Zelinsky, R.A. Jarvis, J.C. Byrne, and S. Yuta, "Planning Paths of Complete Coverage of an Unstructured Environment by a Mobile Robot," *Int. Conf. on Advanced Robotics*, pp.533-538, Tokyo, Japan, 1993.

[20] X. Zheng, S, Jain, S. Koenig, and D. Kempe, "Multi-robot forest coverage," *IEEE Int. Conf. on Intelligent Robots and Systems*, pp.3852-3857, Alberta, Canada, 2005.