Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Complete Coverage Path Planning for Cleaning Task using Multiple Robots

Jeong H. Lee
ASRI, Department of Electrical Engineering
Seoul National University
Seoul, Korea
meeckee2@snu.ac.kr

Jeong S. Choi
ASRI, Department of Electrical Engineering
Seoul National University
Seoul, Korea
jsforce2@snu.ac.kr

Beom H. Lee
ASRI, Department of Electrical Engineering
Seoul National University
Seoul, Korea
bhlee@snu.ac.kr

Kong W. Lee
ASRI, Department of Electrical Engineering
Seoul National University
Seoul, Korea
parandl7@snu.ac.kr

***Abstract*— This paper proposes a novel path planning method for cleaning task using multiple robots in large environment. To do so, we suggested algorithm which partitions a given region into several smaller regions and plans the covering path which can completely cover these divided areas. The algorithm also allocates the planed areas to cleaning robots sequentially and generates the paths which robots can take, when moving from an area to another area, in the shortest time. For partitioning the given region into several small areas, the Virtual Door Algorithm is used, and for planning the paths, the template-based approach is used. Previously generated look-up table, which uses the Dijkstra algorithm, is used for determining the path from one are to another. The Task Area Allocation Algorithm, which allocates the divided areas to cleaning robots using look-up table, is used for planning paths that cover the area completely. Finally, we evaluated the performance of our algorithms which can completely cover the given region regardless of the number of cleaning robots and verify the effectiveness of our proposed algorithms through computer simulations.**

***Keywords*— path planning, partitioning, task allocation, cleaning robot, complete coverage**

I. INTRODUCTION

Over the past few decades, increasing number of researchers has focused their research on finding a solution to controlling robots for cleaning [1]-[4]. However, these studies are focused only on cleaning small environments (e.g. home) using a single or two cleaning robots. It is needless to say that it is effective to use multiple cleaning robots for cleaning task in large environment such as a train-station or an airport. Controlling each robot to clean each divided area after the given region is partitioned into several smaller areas is an effective solution in avoiding collisions between the cleaning robots. In this paper, we will suggest algorithm that partitions the given region into several small areas and allocates these divided areas to multiple robots. Also, we will present path planning algorithm for whole cleaning task.

Extensive studies related to partitioning a given region into several small areas have been conducted. Papadimitriou and Sideri have studied the optimum grid graph bisection problem [5]. This algorithm can divide the graph into two parts and recursively generate *n* partitions. However, its run time is unfeasible for most practical problems. Shermer has been studied a linear-time algorithm for bisecting a simple polygon in a pre-specified direction [6]. Hert and Lumelsky have studied the planar polygon partitioning algorithm to enable parallel under-sea coverage using submersible robots [7]. Agarwal, Hiot, Goo, and, Hghia have studied the rectilinear workspace partitioning problem [8]. They have considered the problem of maximally parallelizing the coverage of a contiguous rectilinear region. However, these algorithms are focused on area partitioning and not considered task area allocation. Therefore, we will study not only area partitioning but also task area allocation in this paper.

Numerous studies related to complete coverage of the given region have been conducted by robotics researchers. Oh, Choi, Park, and Zheng have studied a new map representation method as well as a complete coverage navigation method [9]. They have discussed a triangular cell map representation to make a shorter path and increase flexibility of coverage area than that of a rectangular cell map representation. They also proposed the complete coverage navigation and map construction methods that allow navigation of complete workspace without complete information about the environment. Pirzadeh and Snyder have proposed an indirect control strategy to deal with the coverage and search problem using an artificial potential field [10]. Choset has proposed a novel boustrophedon cellular decomposition approach by breaking down the workspace [11]. Hofner and Schmidt have proposed a template based on an approach that uses motion templates and motion mosaics [12]. Yang and Luo have proposed a neural network approach for complete coverage path planning with obstacle avoidance [13]. This approach is capable of autonomously planning collision-free paths for cleaning robots in a nonstationary environment. However, these algorithms are focused on complete coverage path planning using single robot or two robots. In this paper, we will study complete coverage path planning using multiple robots in large environment.

This paper is organized as follows: In section II, we give the problem formulation and some assumptions. In section III, we propose the Virtual Door Algorithm for dividing a given region into several small areas. In section IV, we propose the Task Area Allocation Algorithm for allocating divided tasks to cleaning robots. In section V, the feasibility and effectiveness are verified by computer simulation. Finally, conclusions are presented in section VII.

978-1-4244-2794-9/09/$25.00 ©2009 IEEE

II. PROBLEM FORMULATION

In this paper, we concern with the cleaning task in large environment. It goes without saying that cleaning task using multiple cleaning robots in more effective than that using a single cleaning robot in large environment such as a station or an airport (Fig. 1). Also, the centralized control method guarantees more completeness and effectiveness that the decentralized control method. Therefore, we use multiple cleaning robots for effectiveness and use the centralized control method to assign the cleaning task to multiple cleaning robots for completeness and effectiveness in this paper.

For assigning the task to multiple cleaning robots, the whole task needs to divide into several small tasks (sub-tasks). Thus, we partition the given region into several small areas and generate the complete coverage path of each divided areas. And, we regard the cleaning task through the generated path of each area as a sub-task. Then, we allocate the whole task to multiple cleaning robots by assigning the sub-tasks to each cleaning robot.

In this paper, some assumptions for solving this problem are follows:

- The given region consists of several rectangular areas.
- The number of the divided areas is larger than the number of the cleaning robots.
- One divided area is only allocated to one cleaning robot.

Each divided area is connected with at least one another divided area.

III. VIRTUAL DOOR ALGORITHM

In previous chapter, we formulate that the given region is divided into several small areas and the divided areas are assigned to each cleaning robot for allocating the whole cleaning task to multiple cleaning robots. In this chapter, we propose the algorithm, called the Virtual Door Algorithm in this paper, which divides the given region into several small areas whose shapes are rectangular.

Figure 1. Example of large environment

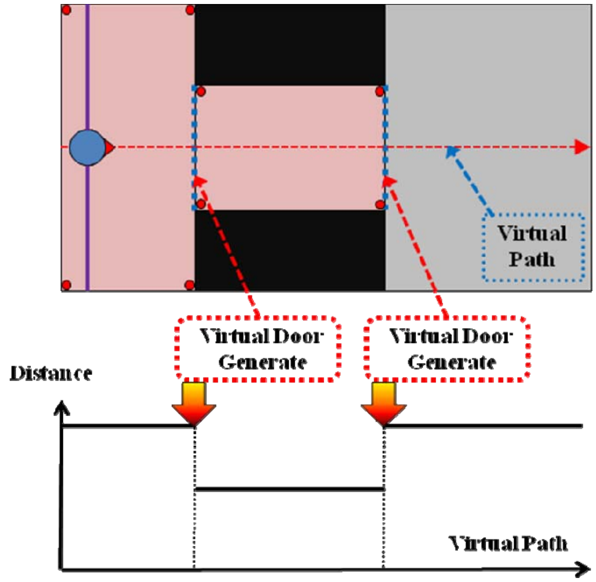

Figure 2. Concept of Virtual Door Algorithm

TABLE I. ALGORITHM FOR TASK AREA DIVITION

D = TASK-AREA-DIVIDE(*MapInfo*)
TASK-AREA-DIVIDE(*MapInfo*)
map_info ← GET_MAPINFO_MTX(MapInfo)
virtual_path_num ← horizon_size / path_width
h_cnt, v_cnt, area_cnt ← 1
WHILE h_cnt < virtual_path_num
WHILE v_cnt < vertical_size
IF map_info[h_cnt,v_cnt] is not included divided_area
dist[i].sum ← dist[i].up + dist[i].down
dist[i].diffsum ← dist[i].sum – dist[i-1].sum
IF dist[i].diffsum ≠ ZERO
SET_DOOR
sub_area[area_cnt] ← REGISTER
v_cnt ← v_cnt + 1
END
h_cnt ← h_cnt + 1
END

The basic concept of this algorithm is as follows. When the robot moves through the virtual path, the sum of the distance from the robot to the right-side obstacle and the distance from the robot to the left-side obstacle is uniform. But, the section on which the sum of distance suddenly changes is the boundary that the concept of the area changes as shown in Fig. 2. Therefore, we assume that the virtual door is set on the boundary that the concept of the area changes and use this virtual door to separate the region as the border. The algorithm is shown in Table I.

IV. Task Area Allocation Algorithm

In previous chapter, we partition a given region into several rectangular areas for allocating the whole cleaning task to multiple cleaning robots. In this chapter, the algorithm, called the Task Area Allocation Algorithm in this chapter, to allocate the cleaning task to each cleaning robot for effective cleaning task is proposed.

There are two points to be specially considered to use a number of the cleaning robots; the first is the number of cleaning robots and the second is the total cleaning time. In this paper, regardless of the number of cleaning robots, we propose the algorithm which all robots can clean the area evenly and be allocated to meet the above two elements.

Figure 3 shows the block diagram of out proposed task area allocation algorithm. The basic concept of this algorithm is as follows. First, starting areas of cleaning robots are selected using the number of given robots and we define these areas as *seed* areas. In this step, we select these seed areas which are same number as given cleaning robot's number in order by cleaning time of area. In these seed areas, the starting points of cleaning robots are located in the top left corner of these areas and the complete coverage path is generated in the shape of back and forth path. Next, the remained (non-allocated) areas are allocated to the cleaning robots. In this step, we use some merging method which each remained areas are merged into each cleaning robot. How to select the robot which will merge is that the robot whose cleaning time at this moment is smallest among cleaning robots is selected. Then, the closest point among corner points of remained areas from the robot's current position (the ending point of last merged area) is selected and the area which contains this selected corner point is allocated to the above selected robot. In order to fine the closest corner point, we make a look-up table which uses the Dijkstra algorithm from one corner point to other corner points. After the selected robot merges the closest area, the robot generates the path which can clean this closest area completely using a template-based approach and updates its cleaning time of merged areas. When all areas are not merged to cleaning robots, go to the step of selection robot, or when all areas are merged to cleaning robots, this process is terminated. Table II shows the task area allocation algorithm when the number of cleaning robots is given.

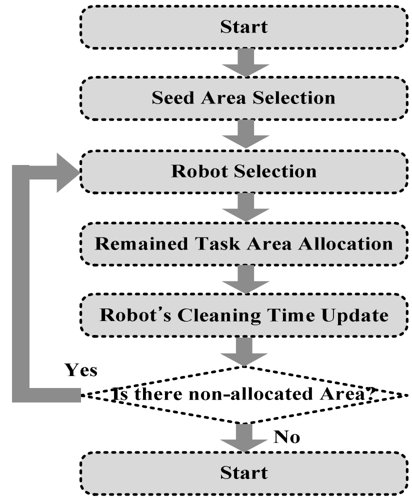

Figure 3. Block Diagram of Task Area Allocation

TABLE II. Algorithm for task area allocation

***M* = MERGE-SUBAREA(*G*, *seed*)**

MERGE-SUBAREA(*G, seed*)

```
rem_nodes_num ← G.tot_nodes_num
rem_nodes ← {true, … , true}
FOR i ← 1 to length(seed)
   M[i][1] ← seed[i]
   tot_area[i] ← G.node[seed[i]].opt_nec_area
   rem_nodes_num ← rem_nodes_num – 1
   rem_nodes[i] ← false
END
WHILE rem_nodes_num > 0
   merge_set ← FIND-MIN-ID(tot_area)
   {merge_node, trans_time}
   ← FIND-NEXT-NODE(G, merge_set, M, remain_nodes)
   cur_length_of_next_node ← length(M[merge_set])
   M[merge_set][cur_length_of_next + 1] ← merged_node
   tot_time[merge_set] ← tot_time[merge_set]
   + G.node[merged_node].opt_nec_time + trans_time
   rem_nodes_num ← rem_nodes_num + 1
   rem_nodes[merged_node + 1] ← false
END
```

V. SIMULATION RESULTS

To implement the suggested algorithms, we developed a simulator in this paper. In this simulator, the characteristics of cleaning robots, the information of given environment, and the number of cleaning robots were set to inputs and the cleaning times of allocated areas to each robot and the sum of the total cleaning times of the each cleaning robot were set to output. The part of the third floor of COEX in Korea was used as given environment.

The simulation was carried out using 1~10 cleaning robots in given environment. This environment is partitioned into 29 areas by the Virtual Door Algorithm as shown in Figure 4. Figure 5 shows the result of the task area allocation and complete coverage path planning when 4 cleaning robots are applied using the simulation. In this figure, A1 means the task areas allocated to the first cleaning robot. As the figure indicates, each robot is assigned the partitioned areas evenly and all robots completely cover whole environment.

Figure 6 shows the cleaning time results of the complete coverage path planning when 1~10 cleaning robots are applied. In this figure, Cal. means the ratio of whole cleaning time in given environment to cleaning time using 1~10 robots by dividing the whole cleaning time into the number of the robots and Simul. means the ratio of whole cleaning time to cleaning time using 1~10 robots using our proposed complete coverage path planning algorithm.

Table III shows the results of this simulation. Robot indicates the number of the cleaning robots applied to the simulation and the Time(cal.) indicates the values of dividing the whole cleaning time into the number of the robots. The Time(Simul.) indicates the maximum values among the cleaning times of the allocated areas of the each robot and the Ratio indicates the ratio of Time(Cal.) to Time(Simul.). We can see that the ratios using 2~5 cleaning robots are adjacent to 100%, but the ratios using more than 6 cleaning robots are rather not.

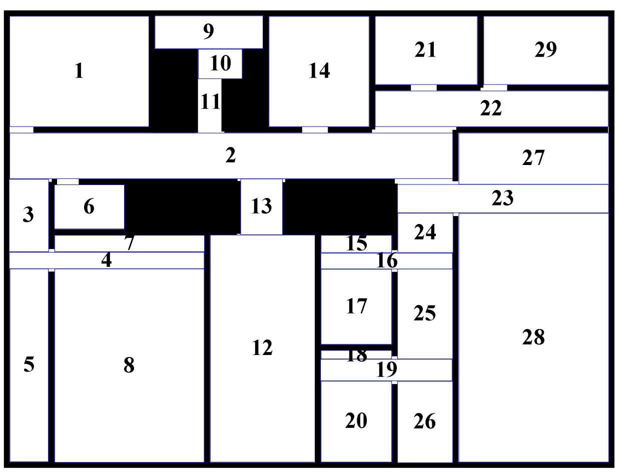

Figure 4. Result of the Virture Door Algorithm

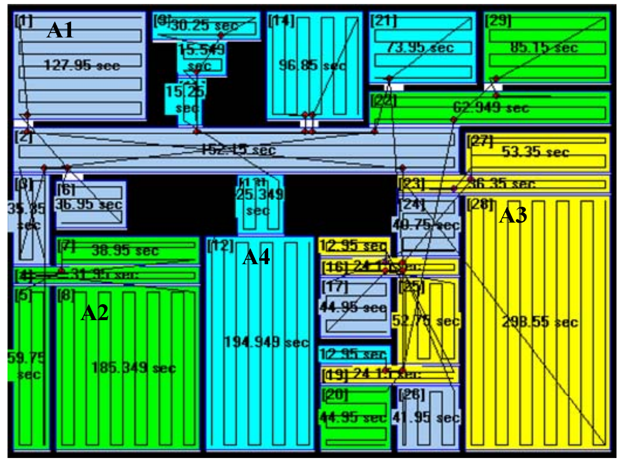

Figure 5. Result of the task area allocation and complete coverage path planning when 4 cleaning robots are applied in the COEX

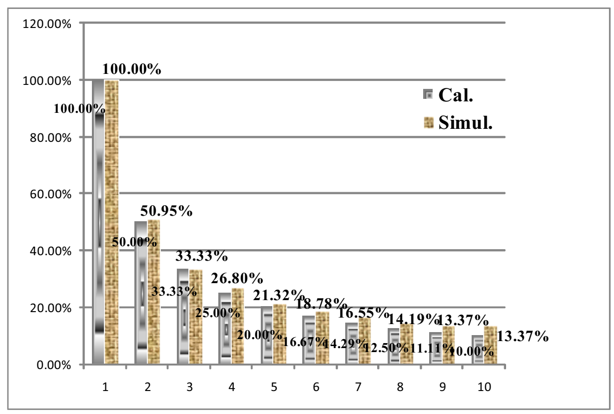

Figure 6. Result of the task area allocation when 4 cleaning robots are applied in the COEX

TABLE III. ALGORITHM FOR TASK AREA ALLOCATION

Robot	Time(Cal.) (sec)	Time(Simul.) (sec)	Ratio
1	2233.068	2233.068	100.00%
2	1116.534	1137.857	101.91%
3	744.356	744.182	99.98%
4	558.267	598.484	107.20%
5	446.614	476.113	106.61%
6	372.178	419.308	112.66%
7	319.010	369.640	115.87%
8	279.134	316.970	113.55%
9	248.119	298.550	120.33%
10	223.307	298.550	133.69%

VI. CONCLUSION

For cleaning task in large environment, it is more effective to use a number of cleaning robots than to use a single cleaning robot. In addition, for cleaning task using multiple cleaning robots, the whole cleaning task must be divided into several sub-tasks and be allocated to each robot effectively. In this paper, the whole cleaning task is allocated to each cleaning robot by dividing the given region into several small areas, by planning paths of each divided areas, and by allocating these planned paths to each cleaning robot. For dividing the given environment into several small areas, we suggest the Virtual Door Method. In addition, for effectively allocating the whole cleaning task to each robot, we plan the complete coverage paths of each divided area and allocate these planned paths to each cleaning robot using our suggested algorithm, Task Area Allocation Algorithm. We verify that the given large environment is effectively divided and the whole cleaning task is effectively allocated to multiple cleaning robots by our suggested algorithms using computer simulations. We assume that the suggested algorithms are applied not only to the cleaning task but also to the demining or the reconnaissance. In future, we will study the relation between the number of the cleaning robots and limited cleaning time and the relation between the number of the divided areas and total cleaning time.

ACKNOWLEDGMENT

This work was supported in part by a Korea Science and Engineering Foundation (KOSEF) NRL Program grant funded by a Korean government (MEST) (No. R0A-2008-000-20004-0), in part by the Growth Engine Technology Development Program funded by the Ministry of Knowledge Economy, and in part by the Brain Korea 21 Project.

REFERENCES

[1] Y. Oh and Y. Watanabe, "Development of small robot for home floor cleaning," Proceedings of the 41th SICE Annual Conference, vol. 5, pp. 3222–3223, August 2002.

[2] C. Chen and K. Song, "Complete coverage motion control of a cleaning robot using infrared sensors," IEEE International Conference of Mechatronics, pp. 543–548, July 2005.

[3] S. Kim, "Autonomous cleaning robot: Roboking system integration and overview," IEEE International Conference on Robotics and Automation, vol. 5, pp. 4437–4441, April 2004.

[4] J. Chi, G. Yang and J. Yang, "The on-line coverage algorithm and localization technique of the intelligent cleaning robot," IEEE International Conference on Sensing and Control, pp. 943–948, April 2008.

[5] C. H. Papadimitriou and M. Sideri, "The bisection width of grid graphs," Proceedings of the first annual ACM-SIAM Symposium on Discrete Algorithms, pp. 405–410, 1990.

[6] T. C. Shermer, "A linear algorithm for bisecting a polygon," Information Processing Letters, vol. 41, issue 3, pp. 105–120, March 1992.

[7] S. Hert and V. Lumelsky, "Polygon area decomposition for multiple-robot workspace division," International Journal of Computational Geometry and Applications, vol. 8, no. 4, pp. 437–466, 1998.

[8] A. Agarwal, L. Hiot, E. M. Joo and N. T. Nghia, "Rectilinear workspace partitioning for parallel coverage using multiple unmanned aerial vehicles," Advanced Robotics, vol. 21, no. 1-2, pp. 105–120, January 2007.

[9] J. S. Oh, Y. H. Choi, J. B. Park and Y. F. Zheng, "Complete coverage navigation of cleaning robots using triangular-cell-based map," IEEE Transactions on Industrual Electronics, vol. 51, no. 3, pp. 718–726, June 2004.

[10] A. Pirzadeh and W. Snyder, "A unified solution to coverage and search in explored and unexplored terrains using indirect control," IEEE International Conference Robotics Automation, vol. 3, pp. 2113–2119, May 1990.

[11] H. Choset, "Coverage of known spaces: the boustrophedon cellular decomposition," Autonomous Robots, vol. 9, no. 3, pp. 247–253, 2000.

[12] C. Hofner and G. Schmidt, "Path planning and guidance techniques for an autonomous mobile robot," Robotics and Autonomous Systems, vol. 14, no. 2-3, pp. 199–212, 1995.

[13] S. X. Yang and C. Luo, "A neural network approach to complete coverage path planning," IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics, vol. 34, no. 1, pp. 718–725, February 2004.