Exploring Rectangular Grid Environments

Mansoor Davoodi * Mehdi Khosravian Ghadikolaei† Mohammad Mehdi Malekizadeh‡

Abstract

In this paper we study the problem of grid exploring which is finding a shortest possible tour for a mobile robot moving in an unknown environment. We focus on a rectangular grid with m columns and n rows, denoted by $R(m,n)$, as the environment. We assume the robot can see only the four cells adjacent to its current cell. Under such conditions, we investigate different variants of exploration problem, and propose efficient bounds and algorithms for the shortest tour in which visits all the cells of $R(m,n)$. We show that for odd $R(m,n)$, $mn+1$ and for even $R(m,n)$, mn are the tight lower bounds for the length of the minimum tour, and propose efficient algorithms for finding such a tour. We show that the algorithms are optimal if the starting cell lies on the boundary of $R(m,n)$. Finally we propose a $(1+\frac{4}{mn})$-competitive algorithm when the robot starts at non-boundary cell.

1 Introduction

Exploration problems is one of the challenges of robot motion planning. Based on type of the environment and the characteristics of the robot, different kinds of exploring problems have been considered (e.g., blindness or visibility of the robot, type of environment and different services). Exploration refers to the task of finding a path, such that every point in the environment is seen from at least one point on the path [4]. In the exploration tour problem, path length minimization is studied in Manhattan and Euclidean metrics. The results have applications in lawn mower, searcher and cleaner robots.

Two models of environment are defined as follows: one without hole and barrier that is called *simple* environment, and one with hole and barrier [7]. When exploring is in a continuous environment, the visibility of the robot could be finite or infinite; but actually robots are usually blind and understand their surroundings using proximity sensors. Also, some tasks in the framework of exploring are considered for robots which necessitates their presence in the environment like lawn mowers that need to move all over the environment to cut the grass.

The grid exploration problem consists of finding the shortest possible tour, which visits every cell of a grid at least once [7]. We call two cells adjacent, if they share a common edge. At each step, the robot can move to an adjacent cell along the four main directions –north, south, east and west–. We assume that the cells have unit size, therefore, the length of the path is equal to the number of steps from one cell to the another and the robot has enough memory to store a map of known cells.

There are two variants of the grid exploration problem. In the *offline* variant, the robot gets a map as an input and computes a tour. In the *online* variant, the robot has limited visibility and can recognize only the four adjacent, without any prior knowledge [7]. A classical graph theory problem named Hamiltonian cycle is closely related to the exploration problem which it consists in determining whether or not a given graph contains a Hamiltonian cycle, i.e., a cycle which visits every vertex exactly once. This problem is well known to be $NP-complete$ [9]. Umans and Lenhart [12] presented an $O(n^4)$ time algorithm for finding Hamiltonian cycles in grids without hole, where n is the number of cells. Salman [11] introduced alphabet grid graphs and determined classes of alphabet grid graphs which contain Hamiltonian cycles, moreover, it is demonstrated that if an $R(m,n)$ (rectangle with m columns and n rows in which $m,n \geq 2$) is odd ($m \times n$ is odd), then the rectangle is non Hamiltonian and also is presented a pattern to find a Hamiltonian cycle for the rectangular grid graph for any even number m or n. Itai et al. [6] presented necessary and sufficient conditions for existence of Hamiltonian paths in rectangular grid graphs and proved that the problem for general grid graphs (can contains holes) is $NP-complete$, which implies that the grid exploration problem is $NP-hard$ as well [6]. This result is particularly interesting because it demonstrates that allowing holes in the input grids can make the problem much harder. Arkin et al. [1] presented an offline algorithm which achieves an approximation factor of $\frac{6}{5}=\frac{48}{40}$ in grids without holes and a factor of $\frac{53}{40}$ if it contains holes.

For online variant of grid exploration, Gabriely and Rimon [3] presented an upper bound $C+B$ for the

*Department of Computer Science and Information Technology, Institute for Advanced Studies in Basic Sciences (IASBS), Zanjan, Iran, mdmonfared@iasbs.ac.ir

†LAMSADE Paris Dauphine University, mehdi.khosravian-ghadikolaei@dauphine.fr

‡Department of Computer Science and Information Technology, Institute for Advanced Studies in Basic Sciences (IASBS), Zanjan, Iran, malekizadeh.mehdi90@gmail.com

length of exploration tour that C is the number of cells and B is the number of boundary cells. Icking et al. [5] presented an upper bound $C + \frac{1}{2}E - 3$ for length of exploration tour where E denotes the number of boundary edges and showed that the best possible competitive ratio is 2 for general grids. Miyazaki et al. [10] proved that an online version of depth-first search achieves this ratio, therefore, the focus moved to exploration of grids without holes. Icking et al. [5] described an online $\frac{4}{3}-competitive$ algorithm which assumes that the robot starts from the boundary of the grid. This ratio was improved to $\frac{5}{4} = \frac{20}{16}$ by Kolenderska et al. [8], who also showed that the best possible ratio is at least $\frac{20}{17}$. In this paper we show that the length of exploration tour in the grid environment which is always 2-colorable is: $S \geq C + \big||V^B| - |V^W|\big|$, where V^B is the set of black cells and V^W is the set of white ones. We also show that the length of optimal exploration tour in an $R(m, n)$ (rectangle with m columns and n rows in which $m, n \geq 2$) is equal to $C + 1$ if $R(m, n)$ is odd ($m \times n$ is odd) and C if R(m,n) is even ($m \times n$ is even), where $C = m \times n$. For the offline variant of exploring, we present an algorithm to compute the optimal exploration tour in a grid rectangle. This algorithm takes time linear in the length of the path, *O(mn)*.

We prove that, every strategy in the online variant for exploring of an $R(m, n)$ with C cells needs at least $C+2$ steps, therefore, the lower bound of competitive ratio is equal to $1 + \frac{2}{C}$. We also present an optimal algorithm to compute the exploration tour in the grid rectangle by an assumption that the starting cell lies on the boundary. Finally we present a $(1 + \frac{4}{C}) - competitive$ algorithm when the robot starts at a non-boundary cell.

2 A Lower Bound

The grid graph corresponding to a grid environment consists of one node for every cell in the grid environment. Two nodes are connected by an edge, if their corresponding cells are adjacent.

A graph is bipartite if its nodes can be partitioned into two sets, so that all edges have one endpoint in each set. Every bipartite graph is 2-colorable and has no cycle with odd length.

Proposition 1 *Grid graphs are bipartite.*

Every bipartite graph is 2-colorable. Moreover we can set white color to each odd node of the grid graph and black to the even nodes. This implies that each node in a grid graph has at most four adjacent with different color.

Theorem 2 *Every strategy for exploring a grid environment with C cells needs at least $C + \big||V^B| - |V^W|\big|$ steps, where $|V^B|$ and $|V^W|$ are the number of black and white cells, respectively.*

Proof. The grid environment is bipartite. Therefore, to explore all cells we have a cycle with consequently movement of black and white cells, because in each step of exploration we enter to a cell which has different color from the previous cell. Thus, if $\big||V^B| - |V^W|\big| = k \neq 0$, to explore all the cells in the larger set, we need to explore an equal number of cells in the other set which means k extra visited cells. □

3 Optimal Exploration Tour in $R(m, n)$

In the offline variant of the grid exploration problem, the entire grid is provided as input and the goal is to determine a shortest exploration tour. Even though we know that it is $NP - hard$ to solve grid exploration in general grids [6], the difficulty of the problem seems to vary greatly depending on whether or not the grids are allowed to have some holes. In the grid environment without hole, exploration problem is still open. however, we present an algorithm to explore an $R(m, n)$ in time linear in the length of the path, $O(mn)$.

Lemma 3 *(see [2]). $R(m, n)$ has a Hamiltonian cycle if and only if $m \times n$ is even.*

So the optimal exploration tour in $R(m, n)$ is Hamiltonian cycle too.

Lemma 4 *(see [11]). $R(m, n)$ contains no Hamiltonian cycle if $m \times n$ is odd.*

So, the length of exploration tour in $R(m, n)$ is $S \geq (m \times n) + 1$.

Lemma 5 *(see [6]). $(R(m, n), s, t)$ has a Hamiltonian path with started node s and final node t if:*

Necessary conditions:

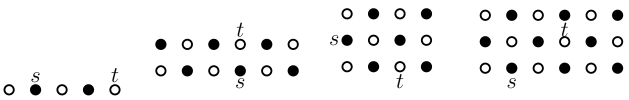
1. *R is even ($|V^B| = |V^W|$) and s and t have different color or*
2. *R is odd ($|V^B| = |V^W| + 1$) and s and t are colored by majority color.*

   And each the following conditions for the graph to have no s,t Hamiltonian path:
3. *$R(m, 1)$ is a 1-rectangle and either s or t is not a corner vertex (Figure 1(a)).*
4. *$R(m, 2)$ is a 2-rectangle and (s, t) is a non-boundary edge, that (s, t) is an edge and it is not on the outer face (Figure 1(b)).*
5. *$R(m, 3)$ is a isomorphic to a 3-rectangle $R'(m, 3)$ such that s and t are mapped to s' and t' and all of the following three conditions hold:*

(a) *m is even,*

(b) *s' is black, t' is white,*

(c) *${s'}_y = 2$ and ${s'}_y < {t'}_x$ (Figure 2(c)) or ${s'}_y \neq 2$ and ${s'}_y < {t'}_x - 1$(Figure 1(d)).*



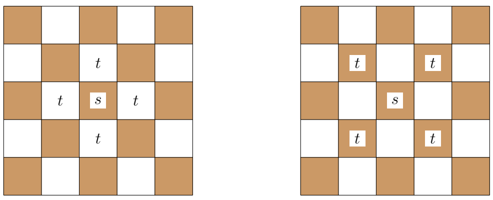
(a) (b) (c) (d)

Figure 1: Rectangular grid graphs in which there is no Hamiltonian path between s and t.

Then finding Hamiltonian path is done in liner time.

Corollary 6 *The length of the optimal exploration tour is $m \times n$ if $R(m, n)$ is even, and $(m \times n) + 1$ if $R(m, n)$ is odd.*

Proof. By considering Lemma 3, if $R(m, n)$ is even, we have a Hamiltonian path which starts from s and ends at one of four adjacent of s (Figure 2(a)). Thus, we can change our Hamiltonian path to exploration tour by moving toward s in one step. If $R(m, n)$ is odd, we have a Hamiltonian path which starts from s and ends at t, one of four diagonal neighbors of s (Figure 2(b)). Therefore, we can change our Hamiltonian path to exploration tour by moving toward s using traversing one common adjacent between t and s. $\square$



(a) Hamiltonian path in even rectangle (b) Hamiltonian path in odd rectangle

Figure 2: Conditions for Hamiltonian path in $R(m, n)$.

Corollary 7 *Finding an optimal exploration tour problem in an $R(m, n)$ without hole can be solved in time linear in the length of the path, $O(mn)$.*

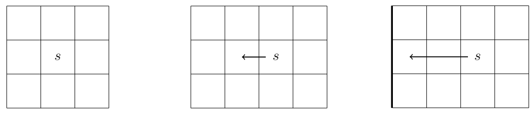
4 Competitive Complexity

In the online variant of the grid exploration problem the robot has a limited visibility and must explore the environment from a starting cell with no prior knowledge. Thus, the first question is whether the robot is still able to find the optimal solution or has to approximate the solution with a constant factor. There is a quick answer to this question.

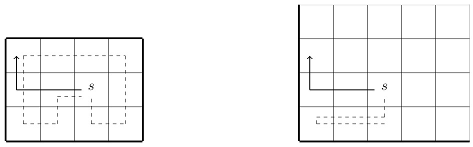
Theorem 8 *Every strategy in the online variant for the exploration of an $R(m, n)$ with C cells needs at least $C + 2$ steps.*

Proof. Since the robot knows nothing about the dimensions of $R(m, n)$ and its position in the environment, generally there are two different strategies for the robot's movements in two prior steps (Figure 3(a)):

1. **First Strategy**
This strategy decides to walk two steps to the west and by these movements robot meets the boundary of the environment (Figure 3(c)). The robot has two choices for the next step, move toward either the north adjacent or south adjacent. Without loss of generality, assume the robot moves to the north one (Figure 3(d)). In this state, the robot needs at least two additional steps for exploring the environment (Figure 3(d)). We can easily extend this pattern to build any rectangular environment of arbitrary size by extending height and width toward the north and east, respectively (Figure 3(e)). We can show easily that if the two first steps are toward another directions (north, south or east), the result is hold as well.



(a) (b) (c)



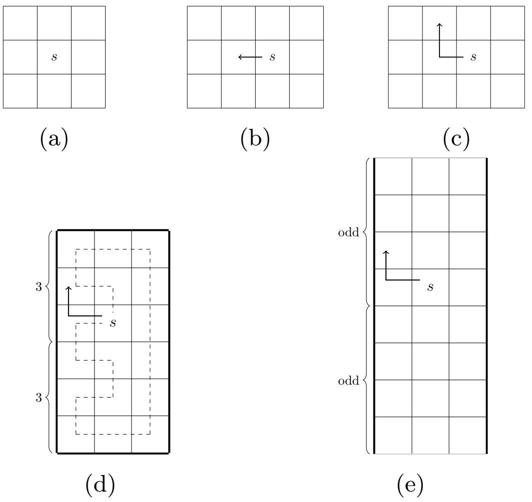
(d) (e)

Figure 3: Tight example for two additional steps in $R(m, n)$ with First Strategy. Dashed lines is the optimal exploration tour.

2. **Second Strategy**
This strategy decides to move two steps toward perpendicular direction (Figure 4(c)). We close our rectangle as shown in Figure 4(d). The robot must continue its exploration in two odd rectangles with width 3. Considering Corollary 1, we know the length of exploration tour is $C+1$ in each *odd* rectangle, where C is the number of cells. So, the strategy needs at least two additional steps for exploring the whole environment. We can easily extend this pattern to build rectangular environment of arbitrary size by extending the height toward the north and south as the height of each new rectangle is odd (Figure 3(e)). We can easily show that if the two first steps are toward the other perpendicular directions (east-south, west-north,...), our final result is hold.

By these two cases, we have shown that using any strategies in online variant, it is need at least $C + 2$ steps to explore $R(m, n)$, whereas the optimal strategy in offline variant needs C steps (Figure 3(d), Figure 4(d)). $\square$

Corollary 9 *Every strategy for the exploration of a rectangular grid environment with C cells is at least $1 + \frac{2}{C}$-competitive.*



(a) (b) (c)

(d) (e)

Figure 4: Tight example for two additional steps in $R(m,n)$ with Second Strategy. Dashed lines is optimal exploration tour.

Proof. The tight example is obtained by exploring the environment which is shown in Figure 3(d) and Figure 4(d). The length of the optimal tour is C, but the length of robot's tour which leads by an online strategy is at least $C+2$.

$$\frac{S_{online}}{S_{optimal}} = \frac{C+2}{C} = 1 + \frac{2}{C}$$

□

5 Algorithms of Exploration

5.1 Patterns of exploration in offline variant

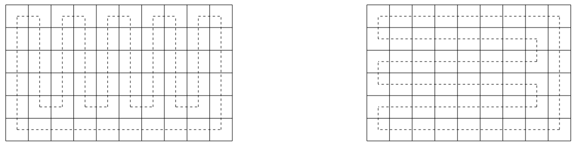
In Section 3, we proved that the exploration tour in the offline variant can be found in linear time in the length of the path, $O(mn)$. In this section, we present two patterns to explore each rectangular grid. Depending on even or odd rectangular grid these patterns can be algorithmically extended to $R(m,n)$, for any m and n.

1. *R(m,n)* is even:

Considering Lemma 1, if $R(m,n)$ is even, we have an exploration tour with the length C, where C is $m \times n$. Figure 5 gives an illustration of exploration tour with 2 examples $R(10,6)$ and $R(9,6)$. The patterns in this figure can be used for finding an optimal exploration tour of $R(m,n)$ for any even number m or n.

2. *R(m,n)* is odd:

Considering Corollary 1, if $R(m,n)$ is odd, we have an exploration tour of length $C+1$. An optimal exploration tour for $R(9,7)$ is shown in Figure 6. The pattern in this figure can be used for finding an optimal exploration tour of $R(m,n)$ for any odd number m and n, where $m,n \geq 3$.



(a) Exploration tour in $R(10,6)$.

(b) Exploration tour in $R(9,6)$.

Figure 5: Patterns of exploration tour.

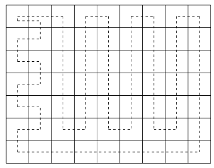
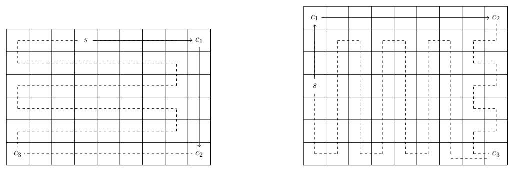


Figure 6: Exploration tour in $R(9,7)$.

5.2 Algorithms of exploration in online variant

In this section we present two algorithms for exploring $R(m,n)$ in online variant. If the robot starts at a boundary cell, we present an optimal exploration algorithm and when the starting cell is a non-boundary cell we present an $(1+\frac{4}{C}) - \textit{competitive}$ exploration algorithm.



(a) Exploration tour in $R(9,6)$.

(b) Exploration tour in $R(9,7)$.

Figure 7: Examples for algorithm's output in the online variant of exploring rectangular grid environment.

1. The starting cell lies on the boundary:

We present an algorithm to compute exploration tour of $R(m,n)$ when the starting cell lies on the boundary. The robot is able to recognize either the number of passing cells is even or odd. There are four possible directions (north, south, east and west) for the robot to move from one cell to an adjacent cell. Command CW denotes rotated clockwise in the environment. Every corner of the environment has only two adjacent and the robot can recognize them. The robot begins his exploration from starting cell in CW direction until reaches the first corner. Then he moves to the next corner and determines the number of cells between the corners is odd or even. If it is even, the robot must walk CW to reach the third corner and explores the remaining cells by the zigzag form between columns (rows) to reach the starting cell (Figure 7(a)). Otherwise, the robot explore between two rows (columns) by the zigzag form to reach the third corner and after that continue his exploration by the zigzag form between columns (rows) to reach the starting cell (Figure 7(b)).

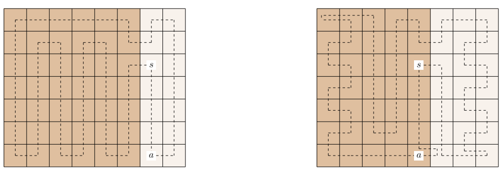
Algorithm 1 BOUNDRAY CONSTRAINT

```
Walk CW to reach the first corner–called c_1.
Walk CW to reach the second corner–called c_2.
if |c_1 c_2| is even then
    Walk CW to reach the third corner–called c_3.
    Explore all rows (columns) by the zigzag form parallel to
    c_2 c_3 to reach the starting cell.
else
    Walk between two last rows (columns) by the zigzag form
    parallel to c_1 c_2 until reach third corner–called c_3.
    Explore the rest rows (columns) by the zigzag form parallel
    to c_2 c_3 to reach the starting cell.
end if
```

We can easily prove that our algorithm finds optimal exploration tour. If the number of cells from c_1 to c_2 (denoted by $|c_1c_2|$) is even, then $R(m,n)$ is even, hence we can continue our exploration using defined pattern in the offline variant. In the other case, $|c_1c_2|$ is odd, hence $R(m,n)$ can be either odd or even. If $|c_2c_3|$ is odd, our first zigzag strategy ends at c_3. In this state, for continuation of exploration, we have to explore a visited cell again and after that it is easy explore the remaining rows (columns) by the zigzag form to reach the starting cell without visiting any extra cells. Therefore the algorithm finds the optimal exploration tour.

2. The robot starts at non-boundary cell:

We present a new algorithm for the case in which the robot starts at a non-boundary cell. The main idea in this algorithm is to subdivide the whole environment into two smaller rectangular environments based on the column of starting cell. In this algorithm the robot begins his movement toward the south until reaches the boundary cell, denoted by a. Then the robot continues his exploration by one step toward east. If this cell is a boundary cell, the robot continues his movement to explore the right rectangle and after that explores the left rectangles using Algorithm 1 (Figure 8(a)). Otherwise, the robot goes back to a and explores the left and right rectangles, respectively, using Algorithm 1 (Figure 8(b),(c)).



(a) Optimal exploration. (b) A tight example.

Figure 8: Examples for algorithm's output in the online variant of exploring rectangular grid environment.

Algorithm 2 GENERAL START POSITION

```
Walk to the south to reach a boundary cell–called a.
Walk to the east cell.
if is not a corner cell then
    Go back to a.
    Explore rectangle induced by s column and west cells by
    Algorithm 1 until reach the cell adjacent to the north s.
    Walk to east cell.
    Explore the remaining cells by Algorithm 1 until reach s.
else
    Walk to the north until reach boundary.
    Walk to the west cell.
    Walk to the south until reach the cell adjacent to the north
    s.
    Walk to the west cell.
    Explore the remaining cells by Algorithm 1 until reach s.
end if
```

Theorem 10 *The algorithm Genaral Start Position is* $(1 + \frac{4}{C}) - competitive$.

Proof. Suppose $R(m,n)$ is even in which m is even and n is odd. Suppose s lies on the m'-th column of the rectangle. In the worst case, if m' is odd, we have two odd rectangles $R(m',n)$ and $R(m-m',n)$ such that . However, both subdivided rectangles are odd and we can explore them with one additional step by using Algorithm 1. Also, as it is shown in Figure 8(c), the robot visits cell a and the eastern adjacent cell of a twice. So, we have four extra cell, in general.

$$\frac{S_{online}}{S_{optimal}} = \frac{C+4}{C} = 1 + \frac{4}{C}$$

□

6 Conclusion

Different variants of online exploring in a rectangular grid $R(m,n)$ for a single robot have been studied in this paper. Efficient bounds and algorithms have been proposed depending on the odd or even size of $R(m,n)$ and also locus of starting position. In all of these cases, we propose almost optimal online algorithms linear to the length of the output path. As a future work, investigating the problem for two or more robots is suggested. In fact in this paper, we assumed a very limited visibility for the robot, while it seems efficient collaborating robots under such assumption is challenging.

References

[1] E. M. Arkin, S. P. Fekete, and J. S. Mitchell. Approximation algorithms for lawn mowing and milling. *Computational Geometry*, 17(1):25–50, 2000.

[2] S. D. Chen, H. Shen, and R. Topor. An efficient algorithm for constructing hamiltonian paths in meshes. *Parallel Computing*, 28(9):1293–1305, 2002.

[3] Y. Gabriely and E. Rimon. Competitive on-line coverage of grid environments by a mobile robot. *Computational Geometry*, 24(3):197–224, 2003.

[4] C. Icking, T. Kamphans, R. Klein, and E. Langetepe. Exploring an unknown cellular environment. In *EuroCG*, pages 140–143, 2000.

[5] C. Icking, T. Kamphans, R. Klein, and E. Langetepe. Exploring simple grid polygons. In *Computing and Combinatorics*, pages 524–533. Springer, 2005.

[6] A. Itai, C. H. Papadimitriou, and J. L. Szwarcfiter. Hamilton paths in grid graphs. *SIAM Journal on Computing*, 11(4):676–686, 1982.

[7] T. Kamphans. *Models and algorithms for online exploration and search.* PhD thesis, University of Bonn, 2006.

[8] A. Kolenderska, A. Kosowski, M. Małafiejski, and P. Żyliński. An improved strategy for exploring a grid polygon. In *Structural Information and Communication Complexity*, pages 222–236. Springer, 2010.

[9] J. A. McHugh. *Algorithmic graph theory.* Prentice Hall Englewood Cliffs, 1990.

[10] S. Miyazaki, N. Morimoto, and Y. Okabe. The online graph exploration problem on restricted graphs. *IEICE transactions on information and systems*, 92(9):1620–1627, 2009.

[11] A. Salman, E. Baskoro, and H. Broersma. A note concerning hamilton cycles in some classes of grid graphs. *Journal of Mathematical and Fundamental Sciences*, 35(1):65–70, 2003.

[12] C. Umans and W. Lenhart. Hamiltonian cycles in solid grid graphs. In *Foundations of Computer Science, 1997. Proceedings., 38th Annual Symposium on*, pages 496–505. IEEE, 1997.