Voronoi-Based Space Partitioning for Coordinated Multi-Robot Exploration

Ling Wu, Miguel Angel Garcia, Domenec Puig and Albert Sole

Abstract—Recent multi-robot exploration algorithms usually rely on occupancy grids as their core world representation. However, those grids are not appropriate for environments that are very large or whose boundaries are not well delimited from the beginning of the exploration. In contrast, polygonal representations do not have such limitations. Previously, the authors have proposed a new exploration algorithm based on partitioning unknown space into as many regions as available robots by applying K-Means clustering to an occupancy grid representation, and have shown that this approach leads to higher robot dispersion than other approaches, which is potentially beneficial for quick coverage of wide areas. In this paper, the original K-Means clustering applied over grid cells, which is the most expensive stage of the aforementioned exploration algorithm, is substituted for a Voronoi-based partitioning algorithm applied to polygons. The computational cost of the exploration algorithm is thus significantly reduced for large maps. An empirical evaluation and comparison of both partitioning approaches is presented.

Index Terms—Multi-robot, exploration, polygonal model, K-means.

I. INTRODUCTION

MULTI-ROBOT coordinated exploration of unknown environments is a challenging task that has been attracting considerable attention in the mobile robotics and artificial intelligence communities due to its potential applications to surveillance, search & rescue and space exploration problems.

The main aim of robot exploration is to unveil the structure and contents (basically free space and obstacles) of an unknown region of space. This goal must be solved under some constraints typically related to minimizing time and/or energy consumption. When the region to be explored is so large that those constraints cannot be satisfied with a single robot, a team of robots must be considered. In this case, the exploration algorithm must guarantee a coordinated deployment of the available robots such that, at least, the parallel exploration yields a reasonable speedup and efficiency, and eventually satisfies other application-dependent criteria, such as maximizing robot dispersion [11].

The majority of approaches to multi-robot exploration (e.g., [1], [4], [6], [11], [12], [14] are based on *occupancy grids* [3], [9] as the basic space representation model. Occupancy grids constitute a simple yet effective way of modelling the world as a regular discretization of space. Every cell in the grid keeps

Ling Wu, and Domenec Puig are with Rovira i Virgili University.

E-mail: ling.wu/domenec.puig@urv.cat
Miguel Angel Garcia is with Autonomous University of Madrid.

Email:miguelangel.garcia@uam.es

the probability of a portion of space being empty or occupied by an object/obstacle.

However, occupancy grids are not suitable enough for modelling and processing large environments. On the one hand, hundreds of thousands of cells must be kept and processed even for workspaces of moderate size. This has severe implications in computational time and memory consumption. On the other hand, the maps are rectangular and bounded by their initial size. If the real world goes beyond the limits of the current grid, there is no means to efficiently extend the map.

The aforementioned inefficiency and lack of flexibility of occupancy grids have motivated the use of alternative world representations based on polygonal models (e.g., [8], [10], [13]). In that case, the exploration algorithm is modified to deal with points and regions instead of cells, in some cases by modelling the free space as a graph of basic convex regions (e.g., trapezoids and triangles in [8], [10]), and in other cases by defining graphs of curves (roadmaps) contained in the empty space (e.g., [2], [13]).

The present paper goes into this direction as it describes and evaluates the application of a polygonal world model as the core spatial representation of a previously proposed multi-robot exploration algorithm [11] originally based on occupancy grids. This algorithm differs from previous approaches in that it explicitly forces the spread of the available robots across the unknown space by successively clustering the unknown cells through the well-known K-Means algorithm.

In this way, the algorithm prevents in a simple and effective way the greedy-like behaviour of many previous proposals, which usually drive the various robots according to local optimization criteria basically dependent on the current location of the robots and their nearby exploration targets, which are constituted by the frontiers of the already explored workspace. In taking these frontiers into account, they somewhat lose sight of the global goal of exploration, which is discovering the overall unknown area.

However, when large environments are considered, the repetitive execution of K-Means over hundreds of thousands of cells has a severe impact on the computational cost of the exploration algorithm. This is why a more efficient polygonal model proves to be beneficial.

This paper is organized as follows. Section II summarizes the exploration algorithm originally proposed in [11]. Section III describes the adaptation of that algorithm to the proposed polygonal space representation. Section IV shows an experimental comparison between both approaches. Finally, conclusions and further research lines are given in Section V.

Fig. 1.   Multi-robot coordinated exploration based on K-Means space partitioning on an empty working space. A partition of unknown space into eight regions ($K = 8$) is shown. Every region is assigned to a single robot. The white area represents the space that has already been explored, while the coloured regions are the regions produced by K-Means.

Fig. 2.   The same exploration method as in Fig. 1. applied to an unstructured working space populated with scattered obstacles. Each region is assigned to a single robot ($K = 8$). The black areas are already-found obstacles and the irregular spots and rectangles in the background are obstacles that have not been found yet.

II. GRID-BASED MULTI-ROBOT EXPLORATION WITH K-MEANS

The multi-robot exploration algorithm previously proposed in [11] has the following main features:

- The workspace is modelled as an occupancy grid (discretized rectangular map) whose cells can be in one of three states: $free$, $unknown$ and $occupied$ (obstacle).
- The size and shape of the map is predetermined.
- The whole map is initially unknown.
- All robots share the same global map and can communicate with a central decision agent without communication failure.
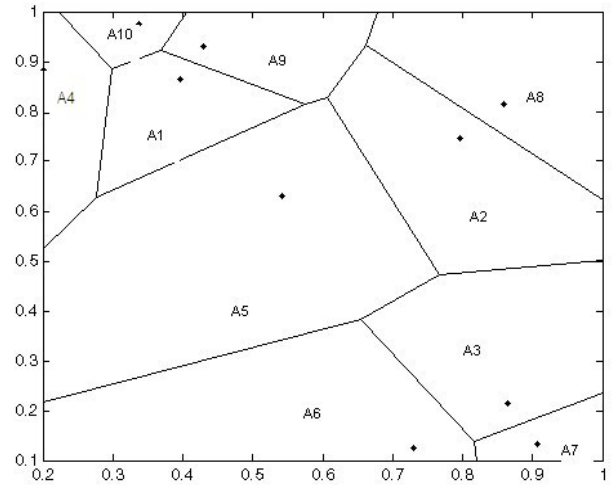

Fig. 3.   A Voronoi diagram of 10 sites (points). Space is partitioned into 10 *voronoi cells* A_i, each of which is constituted by all the points which are closer to one site than to any other site. The edge between two areas consists of a subset of equidistant points to two sites. Some cells, such as A10 and A4, extend to infinite.

- The robots are able to localize themselves within the map.

The exploration algorithm starts with all robots scanning the environment from their starting positions. The initial *free cells*, *occupied cells* and *frontier cells* (unknown cells adjacent to free cells) are thus determined.

The following steps are then repeated until all cells in the map are known (either empty or occupied):

1) The remaining unknown cells are clustered into as many disjoint regions as available robots by applying K-Means, with K being the number of robots. Each robot is then assigned to its closest region according to the Euclidean distance to the regions centroids.
2) Each robot is assigned the frontier cell with the lowest cost. The cost of a frontier cell with respect to a robot is calculated by summing up: (a) the length of the shortest path between the robot and the frontier cell, (b) the Euclidean distance from the frontier cell to the centroid of the region assigned to that robot, provided the cell does not already belong to that region, (c) a constant penalization in case the frontier cell is within the sensor range of another frontier cell assigned to a different robot (this guarantees the repulsion between robots).
3) All robots start moving to their assigned frontier cells until the first robot reaches its destination.

The first two steps are referred to as the *decision stage*. In turn, the third step is referred to as the *motion stage*.

Fig. 1 and Fig. 2 respectively show a snapshot of the map state after the decision stage during the exploration of both a blank and an unstructured environment with eight robots. The unexplored space is partitioned by applying the K-means clustering algorithm. Each of the eight robots is assigned to one of the regions. Once that assignment has been performed, each robot starts moving to its own region.
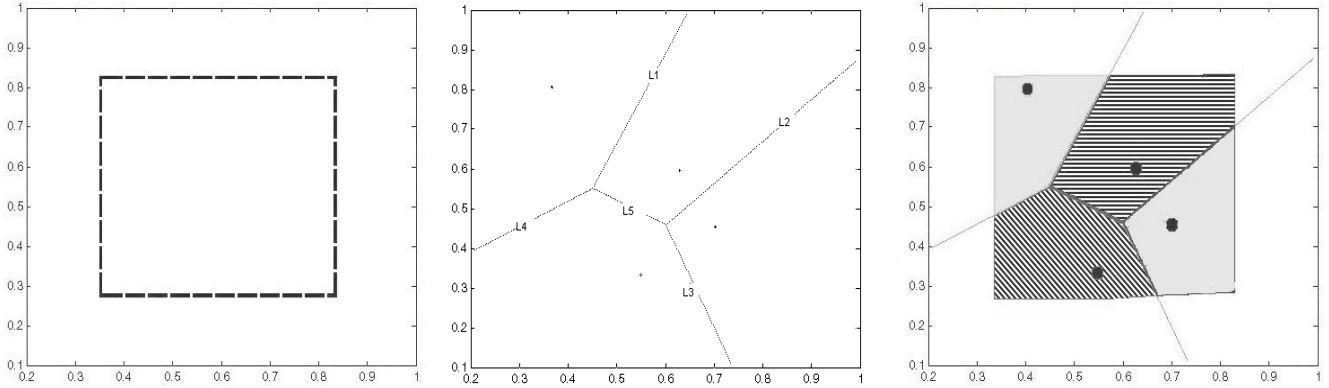
Fig. 4. (*a*) Polygon to be partitioned. (*b*) Original Voronoi diagram. (*c*) Constrained Voronoi diagram and cells.

A. Grid-based K-Means partitioning

During the decision stage, all unknown cells in the grid are clustered by applying the K-Means algorithm according to the following iterative process:

1) Randomly choose K unknown cells $\mathbf{C}_i$, $1 \le i \le K$, as prototypes of the K classes.
2) For every unknown cell, compute its Euclidean distance to the K prototypes and classify the cell into the class i of its closest prototype $\mathbf{C}_i$.
3) Determine the center of mass $\mathbf{M}_i$ of the unknown cells classified into every class i.
4) If $\mathbf{C}_i = \mathbf{M}_i$ for all i (convergence condition), go to step 5. Otherwise, substitute every $\mathbf{C}_i$ for its corresponding $\mathbf{M}_i$ and proceed from step 2.
5) All unknown cells are partitioned into K stable disjoint regions.

III. POLYGONAL-BASED MULTI-ROBOT EXPLORATION WITH VORONOI DIAGRAMS

The multi-robot exploration algorithm described in the previous section can be adapted to a polygonal world model representation in a straightforward manner by considering that:

- The workspace is modelled as a disjoint set of closed polygons. Every polygon has one of three states: *free*, *unknown* or *occupied* (obstacle). The map is constituted by the union of all polygons. Its size and shape are arbitrary.
- The whole map is initially constituted by a single, unknown polygon.
- Free and occupied polygons are included in the map after robot sensing, and subtracted from the unknown polygon/s to which they belong. An efficient library is utilized for Boolean operations between polygons [7].
- Edges of unknown polygons that are adjacent to free polygons are referred to as *frontier edges*.
- Robot path planning is performed in the interior of the available free polygons by applying any cellular decomposition algorithm.

However, the distinctive feature of the exploration algorithm described in [11] is the partitioning of unknown cells during the decision stage by applying K-Means, which is an algorithm specifically devised for clustering clouds of points (cells in this scope) and, hence, not suitable for partitioning polygonal maps.

Notwithstanding, the K-Means clustering algorithm can be mimicked when dealing with polygonal maps by means of *Voronoi diagrams*.

A. Voronoi diagrams

The Voronoi diagram [5] of a set of 2D points, also referred to as *sites*, $\mathbf{C}_i$, $1 \le i \le K$, is a partition of that space into K disjoint convex regions known as *Voronoi cells*. Every region V_i is defined by the points in space that are closer to $\mathbf{C}_i$ than to any other C_j, $j \ne i$.

Although the boundaries of regions are constituted by curves in general, the majority of efficient implementations of Voronoi diagrams currently generate polyline boundaries. Fig. 3 shows an example of a Voronoi diagram corresponding to a set of sites.

As mentioned above, Voronoi diagrams are theoretically defined as a partition of space (2D space in this case). Therefore, they have no exterior limits and, hence, extend to infinity. This means that some Voronoi cells may be unbounded and have infinite area. For instance, this is the case of the cells associated with points A4, A10, A8, A6 and A7 in the example of Fig. 3. However, the proposed multi-robot exploration algorithm only requires the partitioning of those polygons corresponding to unknown regions (unknown polygons). This means that the obtained Voronoi diagrams must be further constrained to the interior of those unknown polygons.

In order to constrain a Voronoi diagram to the interior of a given closed polygon that is to be partitioned, an AND logical operation [7] must be performed between that polygon and the cells of the original Voronoi diagram. The result is a set of closed Voronoi cells that are globally bounded by the polygon to be partitioned. This is illustrated in Fig. 4.

If the unknown area to be partitioned is made up of a set of disconnected regions, and, thus, defined by a collection of closed, separate polygons, the AND operation is carried out between the cells of the Voronoi diagram and each of those polygons in arbitrary order.
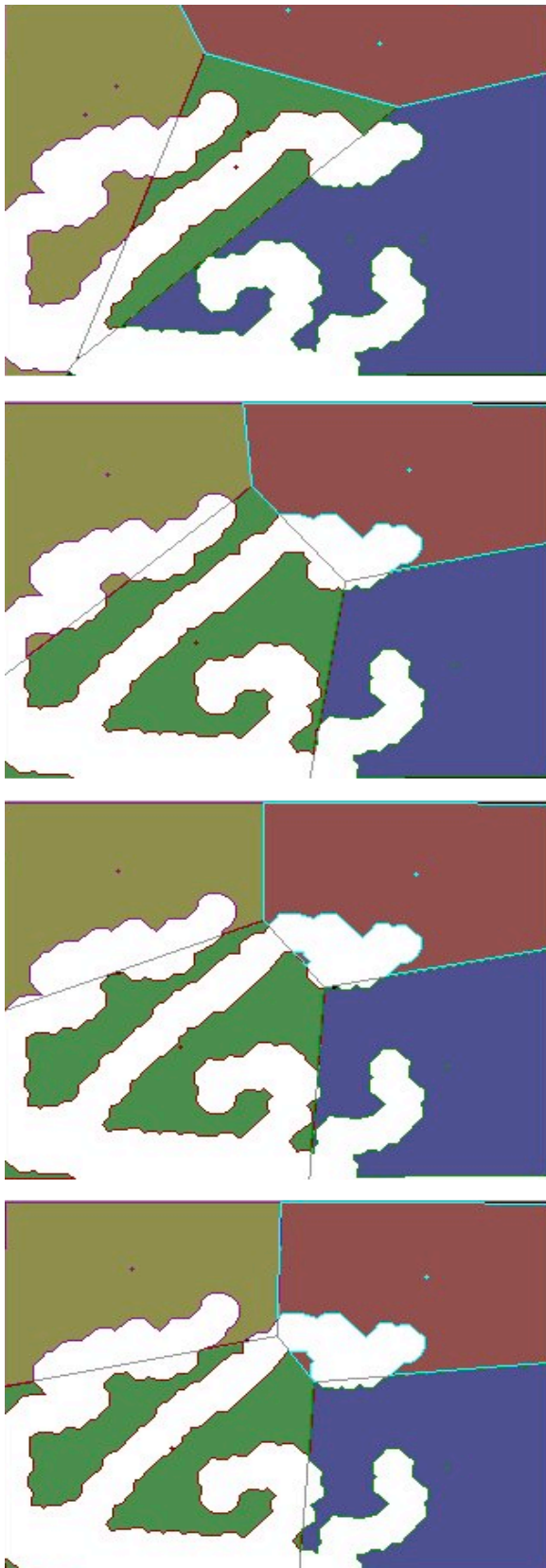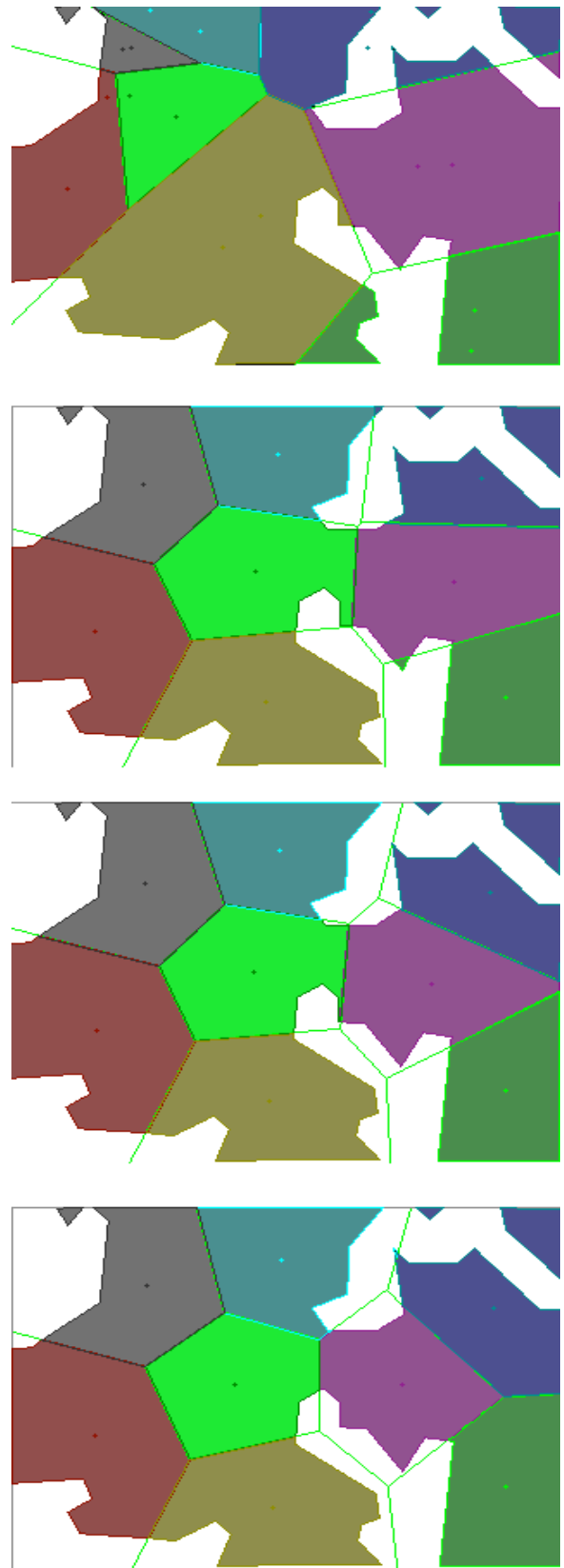
Fig. 5. An example of partitioning sequence of an unknown region with Voronoi diagrams according to the iterative procedure described in section III-B with four robots ($K = 4$). The bottom image shows the final partition after convergence of the constrained Voronoi cells.

Fig. 6. An example of partitioning sequence of an unknown region with Voronoi diagrams according to the iterative procedure described in section III-B with eight robots ($K = 8$). The bottom image shows the final partition after convergence of the constrained Voronoi cells.

B. Polygonal-based Voronoi partitioning

The K-Means partitioning algorithm described in section II-A is adapted to the proposed polygonal model by applying Voronoi diagrams according to the following iterative process.

1) Randomly choose K points $\mathbf{C}_i$, $1 \leq i \leq K$, contained in the polygons corresponding to the current unknown regions in the map.
2) Compute the Voronoi diagram associated with the current set of $\mathbf{C}_i$.
3) Constrain the cells of the Voronoi diagram to the current unknown polygons (see section III-A).
4) Determine the center of mass $\mathbf{M}_i$ of every constrained Voronoi cell.
5) If $\mathbf{C}_i$ - $\mathbf{M}_i < \varepsilon$ for all i, ε being a convergence parameter, go to step 6. Otherwise, substitute every $\mathbf{C}_i$ for its corresponding $\mathbf{M}_i$ and proceed from step 2.
6) The set of unknown polygons is partitioned into K stable disjoint regions.

Fig. 5 illustrates the aforementioned iterative process of Voronoi diagrams until they converge considering the exploration of a blank map with 4 robots. In the top image, each region has two centroids: the original one and the centroid calculated from the current Voronoi diagram. The partition becomes stable when those centroids converge and, thus, every region has a single centroid. The Voronoi diagram in the bottom image shows the stable partition corresponding to that example. Fig. 6 shows another example of the evolution of the Voronoi partitioning process for 8 robots and the same map used in Fig. 5. The white areas in both figures correspond to space that has already been explored.

IV. EXPERIMENTAL RESULTS

As indicated in section II, the most time consuming stage of the multi-robot exploration algorithm proposed in [11] is the successive partitioning of unknown cells by applying the K-Means clustering algorithm. However, this time can be significantly reduced by replacing the original world model representation based on occupancy grids for an equivalent representation based on polygons, and then applying constrained Voronoi diagrams as described in section III.

In order to compare both approaches in analogous conditions, the same maps have been processed with both the grid-based and the polygon-based multi-robot exploration algorithms. Since the number of iterations until convergence of both partitioning algorithms (see sections II-A and III-B) may slightly differ, the iterative partitioning process for both approaches has been forced to stop after the same number of iterations (currently 30 iterations).

For example, Fig. 7 plots the partitioning time for both the grid-based (K-Means) and polygon-based (Voronoi) approaches corresponding to every exploration step (moment at which one of the robots reaches its assigned frontier, and the decision stage, including repartitioning of unknown regions, is rerun; see section 2) considering 4 robots and a blank map whose associated occupancy grid has 320x240 cells. In these experiments, every cell accounts for an area of 0.16 m^2 (the cell sides are 40 cm long). Hence, this map

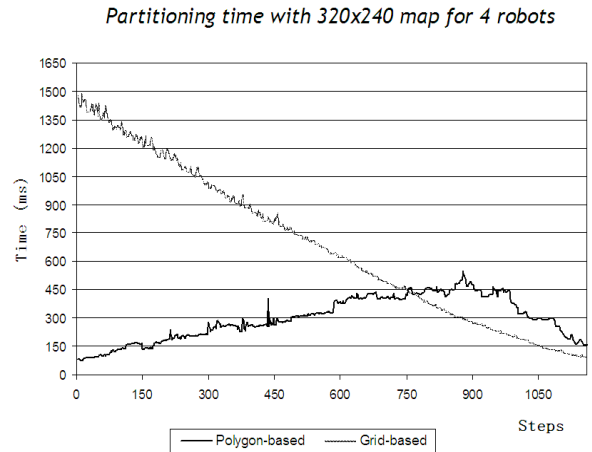Partitioning time with 320x240 map for 4 robots

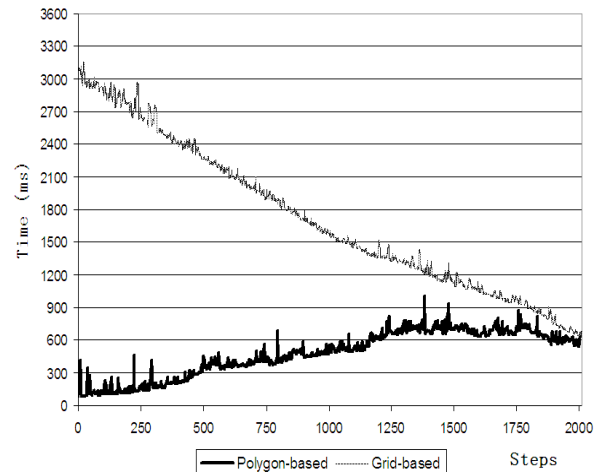Fig. 7. Partitioning times during the exploration of a medium-size blank map of 320x240 cells with 4 robots.

Fig. 8. Partitioning times during the exploration of a big-size blank map of 400x400 cells with 4 robots.

represents a workspace of 12,288 m^2. Similarly, Fig. 8 plots the partitioning times for both approaches corresponding to the exploration with 4 robots of a blank map constituted by 400x400 cells (25,600 m^2).

As the exploration proceeds, the partitioning time corresponding to the grid-based approach goes down steadily. The reason is that the number of unknown cells (the unknown area) to which the K-Means clustering algorithm must be applied is progressively reduced as the robots scan the environment.

In contrast, the computation time of the polygon-based partitioning algorithm starts growing as the number of edges that define the unknown polygons increases. The computational cost of the Voronoi diagrams algorithm only depends on the constant number K of robots. Constraining the Voronoi edges to the interior of the map is also related to the number of robots. However, the Boolean AND operations required to constrain those diagrams to the available unknown polygons depend on the number of edges of those polygons, and that keeps changing as the robots explore the environment —

TABLE I

COMPARISON OF AVERAGE COMPUTATION TIMES BETWEEN K-MEANS AND VORONOI PARTITIONING METHODS, CORRESPONDING TO A MEDIUM-SIZE BLANK MAP AND DIFFERENT NUMBERS OF EXPLORING ROBOTS.

	KMeans-based partitioning	Voronoi-based partitioning
2 Robots	326ms	151ms (46,39%)
4 Robots	592ms	298ms (50,23%)
6 Robots	961ms	604ms (62,90%)
8 Robots	1253ms	836ms (66,75%)

TABLE II

COMPARISON OF AVERAGE COMPUTATION TIMES BETWEEN K-MEANS AND VORONOI PARTITIONING METHODS, CORRESPONDING TO A BIG-SIZE BLANK MAP AND DIFFERENT NUMBERS OF EXPLORING ROBOTS.

	KMeans-based partitioning	Voronoi-based partitioning
2 Robots	682ms	301ms (44,10%)
4 Robots	1723ms	464ms (26,95%)
6 Robots	1983ms	1046ms (52,75%)
8 Robots	2699ms	1346ms (49,86%)

notice that although the total area of the unknown polygons keeps going down during the process, the number of edges of their boundaries may go up and down depending on the intricacy of their shapes (see Fig. 5).

However, the exploration reaches a point after which the unknown polygons become small enough as to start decreasing the number of edges of their boundaries. From that point on, the computation time of the polygon-based partitioning algorithm keeps declining. This behaviour can be clearly appreciated in the experiments depicted in Fig. 7 and 8.

It is important to remark that the benefits of the polygon-based partitioning algorithm are not relevant for small maps since, in that case, the number of cells to be partitioned is small. Hence, cell-based partitioning cannot be expensive. However, the proposed technique is clearly superior for big maps as expected.

A comparison between the partitioning times by considering different numbers of robots has also been carried out. As can be seen in Fig. 9, 10, 11, 12, 13 and 14, Voronoi-based partitioning performs better than KMeans-based partitioning for both middle size and big size maps with the different configurations of robots. It can also be noticed that the computation time of Voronoi-based partitioning rises with the number of robots. The reason is that the polygons that bound the already-explored regions (white areas in the figures) increase their complexity (number of edges) as the number of robots goes up (see Fig. 1). That complexity can be significantly reduced if those polygons are conveniently simplified by applying polygonal approximation algorithms, such as shown in Fig. 6.

Table I and Table II show the benefits of Voronoi-based partitioning over cell-based partitioning. The average computation times for both approaches corresponding to two blank maps and four different configurations of robots, which have
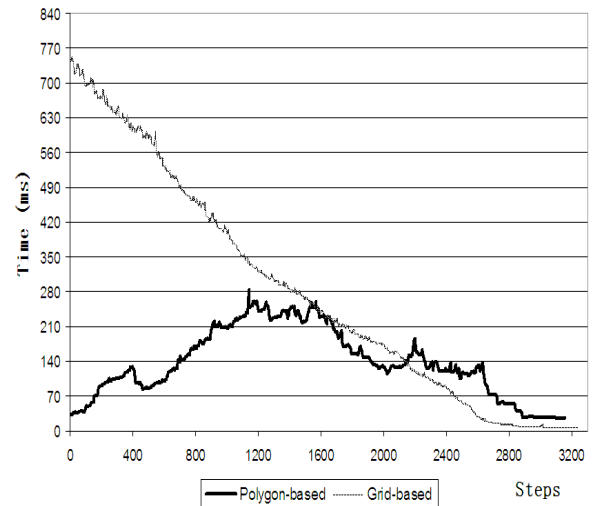

Fig. 9. Partitioning times during the exploration of a medium-size blank map of 320x240 cells with 2 robots.
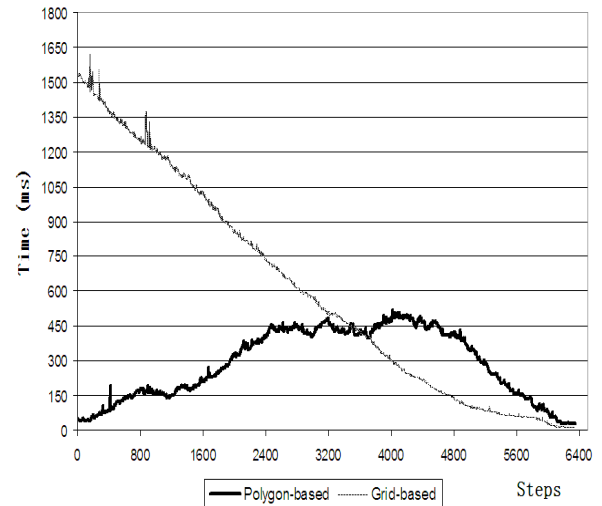

Fig. 10. Partitioning times during the exploration of a big-size blank map of 400x400 cells with 2 robots.

been shown in the eight graphs of Fig. 7, 8, 9, 10, 11, 12, 13, 14 are compared in both tables.

V. CONCLUSION

This paper describes and evaluates the extension of a previously proposed multi-robot exploration algorithm and shows that by replacing the original world model (an occupancy grid) with a more compact and flexible polygonal representation, the new approach significantly increases the efficiency of the most expensive stage of the original algorithm, which is the partitioning of unknown areas into as many regions as robots. The original K-Means clustering algorithm applied over grid cells is substituted for a Voronoi-based partitioning algorithm applied to polygons.

The current algorithm has been tested in simulation. Further work will involve improving the global efficiency of the multi-robot coordinated algorithm and evaluating it on a real team
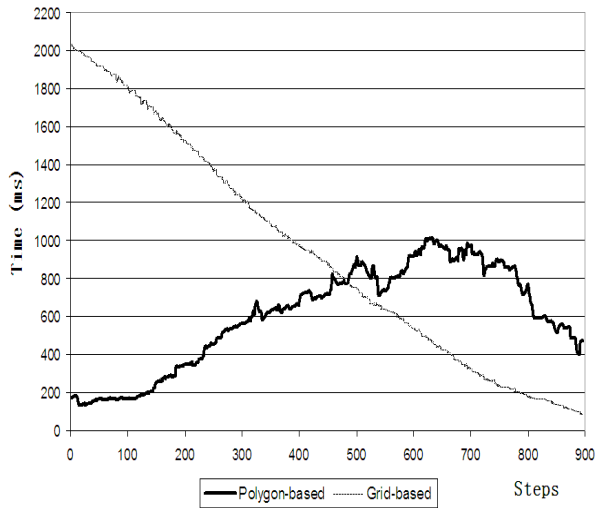
Fig. 11. Partitioning times during the exploration of a medium-size blank map of 320x240 cells with 6 robots.
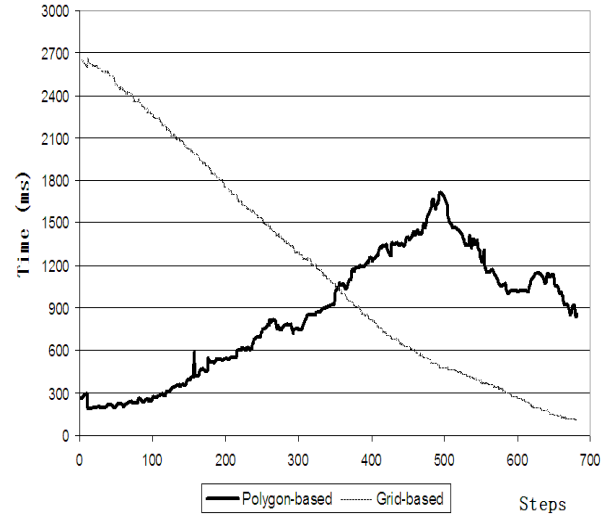

Fig. 13. Partitioning times during the exploration of a medium-size blank map of 320x240 cells with 8 robots.
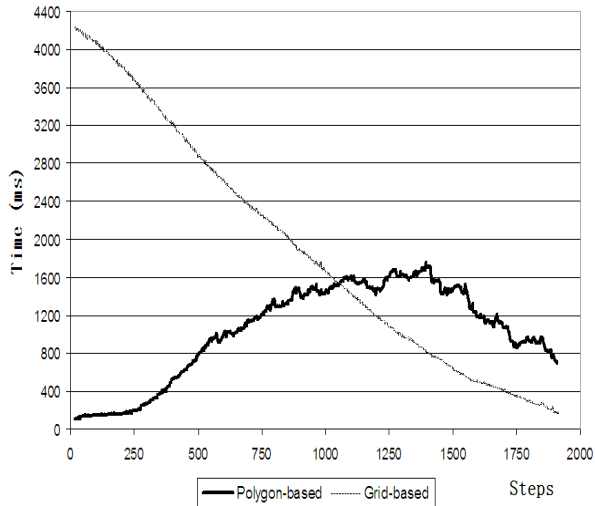

Fig. 12. Partitioning times during the exploration of a big-size blank map of 400x400 cells with 6 robots.
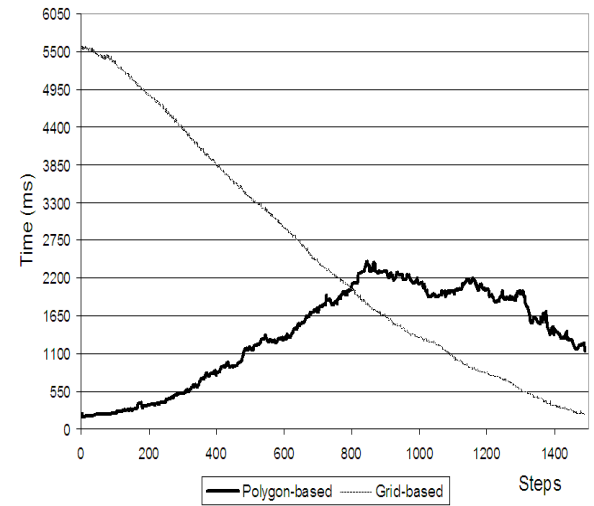

Fig. 14. Partitioning times during the exploration of a big-size blank map of 400x400 cells with 8 robots.

of robots. We also aim at extending the current technique to unbounded environments.

REFERENCES

[1] W. Burgard, M. Moors, D. Fox, R. G. Simmons, and S. Thrun, *Collaborative multi-robot exploration*. In Proc. IEEE Int. Conf. on Robotics and Automation (ICRA), pages 476–481, 2000.

[2] H. Choset and K. Nagatani, *Topological simultaneous localization and mapping (slam): toward exact localization without explicit localization*. IEEE Transactions on Robotics and Automation, vol. 17, no. 2, pp. 125–137, April 2001.

[3] A. Elfes, *Using occupancy grids for mobile robot perception and navigation*. Computer, vol. 22, no. 6, pp. 46–57, 1989.

[4] G. Fang, G. Dissanayake, and H. Lau, *Behaviour-based optimization strategy for multi-robot exploration*. In Proc. of IEEE Conf. on Robotics, Automation and Mechatronics, vol. 2, pages 875–879, Singapore, Dec 2004.

[5] S. Fortune, *A sweepline algorithm for voronoi diagrams*. Algorithmica, vol. 2, no. 1, pp. 153–174, March 1987.

[6] D. Fox et al., *Distributed multirobot exploration and mapping*. In Proc. of IEEE, vol. 94, July 2006.

[7] K. Holwerda, *Boolean: Gdsii viewer/editor + (boolean) operations on sets of 2d polygons*. Available: http://boolean.klaasholwerda.nl/bool.html

[8] B. Lisien et al., *The hierarchical atlas*. IEEE Transactions on Robotics, vol. 21, no. 3, pp. 473–481, June 2005.

[9] H. Moravec and A. E. Elfes, *High resolution maps from wide angle sonar*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), pp. 116–121, March 1985.

[10] I. M. Rekleitis, G. Dudek, and E. Milios, *Multi-robot collaboration for robust exploration*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), 2000.

[11] A. Solanas and M. A. Garcia, *Coordinated multi-robot exploration through unsupervised clustering of unknown space*. In Proc. of

IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, vol. 1, pp. 717–721, Sendal, Japan: 2004.

[12] C. Stachniss and W. Burgard, *Exploring unknown environments with mobile robots using coverage maps*. In Proc. of IEEE Int. Conf. on Artificial Intelligence (IJCAI), 2003.

[13] C. Trevai et al., *Cooperative exploration of mobile robots using reaction-diffusion equation on a graph*. In Proc. of IEEE Int. Conf. Robotics and Automation (ICRA), vol. 2, pp. 2269–2274, 2003.

[14] R. M. Zlot, A. T. Stentz, M. B. Dias, and S. Thayer, *Multi-robot exploration controlled by a market economy*. In Proc. of IEEE Int. Conf. on Robotics and Automation (ICRA), vol. 3, pp. 3016–3023, May 2002.