

# Robust Environment Mapping Using Flux Skeletons

M. Rezanejad<sup>†</sup>, B. Samari<sup>†</sup>, I. Rekleitis<sup>‡</sup>, K. Siddiqi<sup>†</sup>, and G. Dudek<sup>†</sup>

**Abstract**—We consider how to directly extract a road map (also known as a topological representation) of an initially-unknown 2-dimensional environment via an on-line procedure which robustly computes a retraction of its boundaries. While such approaches are well known for their theoretical elegance, computing such representations in practice is complicated when the data is sparse and noisy. In this paper we present the online construction of a topological map and the implementation of a control law for guiding the robot to the nearest unexplored area. The proposed method operates by allowing the robot to localize itself on a partially constructed map, calculate a path to unexplored parts of the environment (frontiers), compute a robust terminating condition when the robot has fully explored the environment, and achieve loop closure detection. The proposed algorithm results in smooth safe paths for the robot's navigation needs. The presented approach is an any-time-algorithm which allows for the active creation of topological maps from laser-scan data, as it is being acquired. The resulting map is stable under variations to noise and the initial conditions. The key idea is the use of a flux-based skeletonization algorithm on the latest occupancy grid map. We also propose a navigation strategy based on a heuristic where the robot is directed towards nodes in the topological map that open to empty space. The method is evaluated on both synthetic data and in the context of active exploration using a Turtlebot 2. Our results demonstrate complete mapping of different environments with smooth topological abstraction without spurious edges.

## I. INTRODUCTION

This paper addresses on-line topological mapping and navigation using a robust skeletonization mechanism that is computationally efficient and provides smooth maps without spurious edges. Our approach is based on computing a flux based skeleton from two dimensional, dense, laser data. This fundamentally one dimensional structure (embedded in 2D) constitutes an efficient and elegant representation that can be used for a range of navigation and localization tasks.

Topological representations have been proposed and employed in robotics for over 25 years [1], [2], [3] because of the potentially simple ensuing control laws, their relevance to human cognitive mapping, and their compactness. At the core of many approaches to extracting such abstractions from real environments is the calculation of points that are locally maximally distant from sensed obstacles, yielding the medial axis. Such topological structures result in safe areas where robots can navigate without collisions. Indoor environments with long corridors, like those found in malls and office



Fig. 1. The experimental platform used, a Turtlebot 2, with a Hokuyo laser range finder. This image is taken at the junction next to the triangular obstacle in the center of the map; see Fig. 4.

buildings, underground mines, and dry or underwater caves, are ideal candidates for using a topological representation as are waterways such as in the Florida Everglades.

Traditionally, pragmatic topological mapping approaches have been proposed using local sensor-based data [4] to guide the robot from one vertex to the next. As such, loop closure and global reasoning is challenging requiring additional information for verification [5], [6], [7]. The main contribution of this paper is an any-time-algorithm that returns the topological map based on all the available information up to execution time, enabling reliable loop closure. The second contribution of this paper is the development of an active exploration strategy based on the heuristic that at any given time the robot will be guided towards a node, computed from the present skeleton, whose local neighborhood remains relatively unexplored. In addition, an effective skeleton graph pruning strategy is used which leads to topological maps that are far simpler than those obtained from related Voronoi based methods.

In the following section we review related work. Section III presents the flux-based skeleton algorithm for the online construction of a topological map. The exploration strategy of the robot for the complete mapping of an unknown environment is outlined in Section IV. Experimental results from simulated and real environments are presented in Section V. We then summarize our contributions and discuss directions for future work in Section VI.

## II. BACKGROUND

Medial axes are among the most fundamental geometric structures in computer vision and robotics since they relate

<sup>†</sup> M. Rezanejad, B. Samari, G. Dudek and K. Siddiqi are with the School of Computer Science and the Centre for Intelligent Machines, McGill University, Montreal, Canada. [morteza, babak, dudek, siddiqi]@cim.mcgill.ca

<sup>‡</sup> I. Rekleitis is with the Computer Science and Engineering Department, University of South Carolina, 315 Main St. Columbia, SC, 29208, USA yiannis@cs.cse.sc.edu

to local axes of mirror symmetry. One of the first formal definitions for the medial axis was introduced by Blum based on the analogy of a “grassfire” [8]. Here, the boundary is set on fire and as the front advances inward at a constant speed, skeletal points are created where fire fronts meet and quench one another [9]. It turns out that this formalism is equivalent to computing skeletons by finding local maxima of the Euclidean distance function to the boundary [10] or by finding locations where its gradient is multi-valued [11]. Other than these two classes of approaches, there are a number of methods that use Voronoi diagrams for skeleton computation [12], [13], [14] due to the theoretical relationship between them [15]. In particular, the vertices of the Voronoi diagram of a set of boundary points converges to the exact skeleton as the sampling rate increases under some appropriate smoothness conditions [16].

The major weakness with traditional skeletonization algorithms is that they suffer from high sensitivity to noise (perturbations of the boundary data). A small perturbation in the sensed data can drastically change the skeleton structure and its abstraction. This has led to a number of different results on the stability of medial axis, including approaches which try to remove those skeletal branches that are likely to be generated due to boundary noise [17], [18], [19]. In the present article we opt for a Hamilton-Jacobi formulation of the eikonal equation (grassfire transform) [11] because it yields a direct and robust manner for finding skeletal points, since it utilizes an average outward flux computation involving integrals rather than derivatives. Here skeletal points are associated with locations where the average outward flux of the gradient of the Euclidean distance function through a shrinking circular neighborhood is non-zero.

The appeal of topological representations for mapping, exploration and human-robot interaction has been noted by several authors who suggested they be used directly, computed from 2D data [1], [2]. The Voronoi diagram, a classic structure in computational geometry that has appeared in many fields, and the Generalized Voronoi Graph (GVG) was exploited in robotics as a mechanism for computing topological maps [20], [21]. Pure topological results have been studied in [22], [3], [23] for mapping a graph-like world with minimal sensor input. More recent work includes [24] on exploration strategies on a graph-like world.

The full employment of the GVG in a SLAM framework was proposed in [25], and extended for use in hybrid metric/topological maps in [26], [27]. Tully et al. [28] recommends a hypothesis tree method for loop-closure where the branches that are considered to be unlikely based on topological and metric GVG information are pruned. Among the pruning tests, it is worth mentioning the planarity test which ensures that when a loop-closure has been decided, the resulting GVG graph remains planar. The utility of this test has been examined extensively in [29]. The exploration of a hybrid metric-topological map based on the GVG taking into account both the uncertainty of the map and the distance traveled was accomplished using the A\* algorithm in [30]. The purely topological variant of these approaches had been

previously examined on [23].

Kuipers et al. [31] recommend the use of a framework, termed a hybrid spatial semantic hierarchy, where the incremental construction of topological large-scale maps is employed in conjunction with metric SLAM methods for the creation of maps of small-scale. No use of a global frame of reference is made and a multi-hypothesis approach is used to represent potential loop-closures.

### III. MAPPING ENVIRONMENTS USING FLUX SKELETONS

Our system takes laser scanned data from a 2D laser line scanner and generates an abstraction of the scanned environment. This is done through a number of modules: GMapping/binarization, flux skeleton computation, pruning and simplification, and path planning for further exploration; see Fig. 2. These modules are executed in a serial pipeline where the output of each module is the input to the next module.

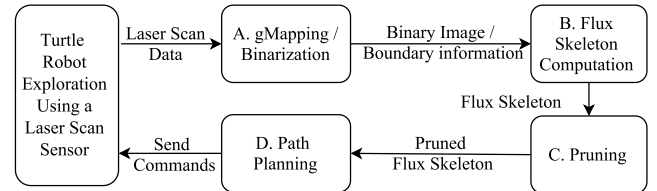


Fig. 2. System overview. The system consists of four independently running modules along with a robot which is exploring the environment. Each of these modules is a component of a feedback chain system. The main four modules are explained in detail in Subsections: III-A, III-B, III-C, and IV.

#### A. GMapping and Binarization

The system first receives 2D laser scan data in a format where each scan is a single line containing range measurements. These laser scan data serve as input to the GMapping module. GMapping is one of the most used laser-based SLAM algorithms [32]. It takes raw laser scan range data and odometry and produces gridmaps of the considered environment, where each gridmap is a probability distribution of cells (regions) being covered by the laser scan. The algorithm uses a highly efficient Rao-Blackwellized particle filter in which each particle has an individual map of the environment. The generated gridmap at the end of this stage is an intensity image where higher intensities show higher probabilities of being covered by the laser scanner (white regions), grey cells with lower intensities represent points which have not been covered yet by the robot, and where black cells usually represent walls where the range scanner has faced a physical obstacle. The top row of Fig. 4 shows an example of a gridmap obtained after several scans have been incorporated.

Gridmaps must be binarized before they can be fed as input to our flux based skeletonization algorithm. To do this we apply the following sequence of steps: a) all pixels on gridmaps that are not scanned (grey regions) are set to background regions. Pixels that have high probability of being obstacles (e.g. walls - black pixels) are stored as the

foreground regions. b) Gaussian blurring is applied to smooth the structure that remains. c) The smoothed image is then thresholded to yield a binary one. d) The contours of all foreground regions in this image are extracted and sorted according to their area; regions having very small area are considered outliers. During this process, we keep track of the transformation needed to translate the final output to world coordinates. The second row in Fig. 4 depicts a binarized version of the grid map in the top row.

### B. Flux Skeletons

We now review the computation of the skeleton using the average outward flux of the gradient of the Euclidean distance function to the boundary through a limiting circular region as developed in [11], [33]. We begin with some definitions.

*Definition 1:* A 2-D object  $\Omega$  is a set of geometrical loci that the projection of a 3D object occupies in 2-D space. Its boundary,  $\partial\Omega$ , consists of a finite number of mutually disjoint closed curves, each being a connected path in this space that does not intersect itself.

The Euclidean distance between two  $n$ -dimensional points  $P = (p_1, p_2, \dots, p_n)$  and  $Q = (q_1, q_2, \dots, q_n)$  is the length of the line segment that connects these two points, and the *Euclidean metric*  $d(P, Q) : \mathbb{R}^n \times \mathbb{R}^n \rightarrow \mathbb{R}$  is a function that represents this distance:  $d(P, Q) = \|Q - P\| = \sqrt{\sum_{i=1}^n (q_i - p_i)^2}$ . For each point  $P$ , and a given object  $\Omega$ , a distance metric,  $d_\Omega(P)$ , can be defined as follows:  $d_\Omega(P) = \inf_{Q \in \partial\Omega} d(P, Q)$ .

The signed Euclidean distance function within  $\Omega$  specifies how close a given point  $P$  is to the boundary  $\partial\Omega$ :

$$D_\Omega(P) = \begin{cases} d_\Omega(P) & \text{if } P \text{ is inside } \Omega \\ 0 & \text{if } P \in \partial\Omega \\ -d_\Omega(P) & \text{if } P \text{ is outside of } \Omega \end{cases}$$

Let us define the projection  $\Pi(P)$  as the set of closest points on the boundary  $\partial\Omega$  to  $P$ , i.e.,  $\Pi(P) \triangleq \{P' \in \partial\Omega : \|P - P'\| = \min\{\|P - P'\| \mid P' \in \partial\Omega\}\}$ . Assume that on the boundary  $\partial\Omega$ , there exists only one point  $Q$  of minimum distance to  $P$  ( $\Pi_\Omega(P) = \{Q\}$ ). The *distance function gradient vector* for point  $P$  is  $\dot{\mathbf{q}}_\Omega(P) = \frac{Q-P}{\|Q-P\|}$ .

In the case  $|\Pi_\Omega(P)| > 1$ , there is more than one closest boundary point and the distance function gradient vector can not uniquely be defined. In fact the Euclidean distance function gradient vector  $\dot{\mathbf{q}} = \nabla D$  is multivalued at skeletal points. Except at skeletal points,  $\dot{\mathbf{q}}$  is continuous everywhere on its domain and it satisfies the equation:  $|\dot{\mathbf{q}}| = 1$ . Figure 3 shows an example of the distance function gradient vector.

By exploiting the relationship between the integral of the divergence of a vector field within a simply-connected region and the outward flux of that vector field through the boundary of that region, a modified divergence theorem leads to characterization of skeletal points by *average outward flux* [11], [33]. Let  $R$  be a region with boundary  $\partial R$  a simple closed curve, and let  $\mathbf{N}$  be the outward normal at each point on  $\partial R$ .

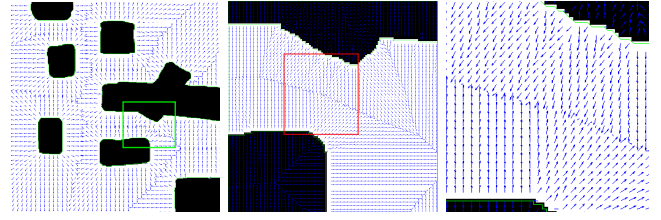


Fig. 3. An illustration of the Euclidean distance function gradient vector field  $\dot{\mathbf{q}}$  for a sample environment where the black regions represent obstacles.

*Definition 2:* The *outward flux* of  $\dot{\mathbf{q}}$  through  $\partial R$  is defined as  $\int_{\partial R} \langle \dot{\mathbf{q}}, \mathbf{N} \rangle ds$ , and the *average outward flux* of  $\dot{\mathbf{q}}$  through  $\partial R$  is defined as  $\mathcal{AOF} = \frac{\int_{\partial R} \langle \dot{\mathbf{q}}, \mathbf{N} \rangle ds}{\int_{\partial R} ds}$ .

It can then be shown that the limiting behaviour of the average outward flux of the distance function gradient field through a circular region, as the radius of that region shrinks to zero, is zero at non-skeletal points and non-zero at skeletal points. Furthermore, at skeletal points the value of the average outward flux is related to the angle made between the tangent to the skeleton and the vector in the direction of the closest boundary points on each side, as illustrated in Fig. 3.

### C. Pruning the Skeleton

#### Algorithm 1 Pruning Algorithm

```

1: procedure ITERATIVE_PRUNING(Skeleton S, BinaryImage I)
2:   list: source.node  $\leftarrow \emptyset$ 
3:   size  $\leftarrow S.nodes.SIZE$ 
4:   for  $\forall$  EndPoint E  $\in S$  do
5:     if IS_FEASIBLE(S, E) == false then
6:       S.nodes.REMOVE_END_POINT(E)
7:     end if
8:   end for
9:   if S.nodes.SIZE == size then  $\triangleright$  No reduction has been made
10:    return
11:   end if
12:   ITERATIVE_PRUNING(S, I)
13: end procedure
14:
15: procedure IS_FEASIBLE(EndPoint E, BinaryImage I)
16:   T  $\leftarrow$  EmptyImage
17:   for  $\forall$  ContourPoint P  $\in I$  do
18:     if !P.IS_OBSTACLE & E.DISTANCE.TO(P) <  $\tau$  then
19:       T.ADD_POINT(P)
20:     end if
21:   end for
22:   C  $\leftarrow$  T.GET_ALL_CONTOURS
23:   for  $\forall$  c  $\in C$  do
24:     if c.AREA >  $\tau'$  then  $\triangleright \tau$  and  $\tau'$  are certain thresholds.
25:       return true
26:     end if
27:   end for
28:   return false
29: end procedure

```

As illustrated in Fig. 4 (third row), the skeletonization process yields some branches that can be pruned without altering the skeleton's topology. To prune such branches with the goal of topological mapping, we suggest a fairly simple but effective algorithm where the robot explores unseen regions and avoids getting too close to obstacles. Algorithm

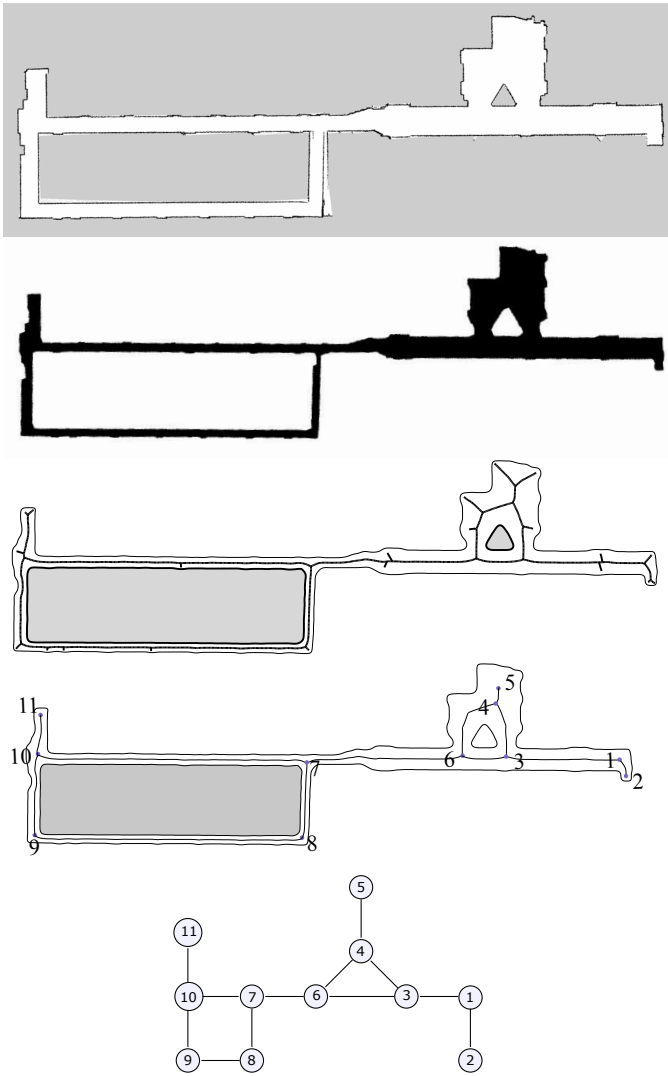


Fig. 4. Top row: an example of a grid map of an environment. Second row: the binarization of the grid map in the top row. Third row: the full skeletonization process applied to the binarization of the environment. Although, the skeleton is very smooth, there are still branches that can be removed without altering its topology. Fourth row: the skeleton in the third row is pruned and simplified in a way that makes robot navigation safe. Bottom row: the topological map resulting from the abstraction in row four.

1 summarizes this process which works as follows: it looks for all branches that have one branch point connected to an end point. If that end point is surrounded by walls of obstacles then there is no possibility for further exploration in that direction. In such a case, the branch connecting the branch point to the end point is removed and the skeleton is simplified.

#### IV. PATH PLANNING

The last step of the proposed process is to guide the robot through the environment to explore new territory. As we showed in subsection III-C, at each time, there is a partial abstraction of the environment generated by utilizing the GMapping package, discretizing the output map of GMapping, and then extracting the flux-based skeleton. These steps

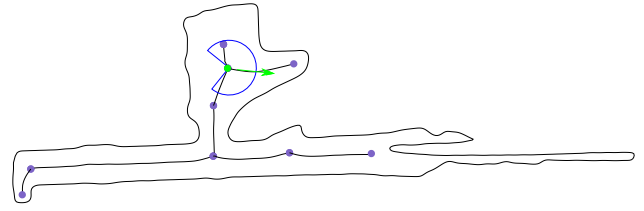


Fig. 5. The environment has been partially explored and the robot now selects an edge (green) leading into unexplored space.

produce an up to date topological map of the environment at each time step. The system keeps track of the visited nodes in a list. This enables the system to explore novel territory for its next move. At each time step as the robot moves from one of the visited nodes to a frontier node the new nodes traversed on-route to the frontier node are added to the list of visited nodes. Each connecting edge between nodes is then weighted by the length of the path through the skeleton. This weighting strategy results in a selection of good candidates for future exploration. The algorithm then selects the direction towards the nearest frontier node to the current node. To find the nearest frontier node, we use the *Bellman Ford* algorithm which computes shortest paths from a single node to all of the other nodes in a weighted directed graph; see Fig. 5.

#### V. EXPERIMENTAL RESULTS

Several experiments were performed, both in simulation and with a real robot. The proposed methodology was implemented under the ROS framework<sup>1</sup>. During the non-simulation experiments the Turtlebot 2 platform was used with a Hokuyo laser range finder; see Fig. 1. The laser sensor has a range of 30 m and a 270° field of view, and returns a dense cloud of 1080 coplanar points. During simulated experiments the Stage simulator<sup>2</sup> was used with a different environment.

Figure 6 presents an experiment in the Stage *cave* simulated world. The robot started at the middle of the environment, and created a skeleton based on the current information it had (Fig. 6a). After moving to the nearest frontier node, more of the environment became visible and the topological map was updated (Fig. 6b). The lower left obstacle was not fully mapped, however, enough information was available to produce a loop. The robot then proceeded to explore the top left corner (6c) and then continued exploration towards the branching nodes at the right side of the environment (Fig. 6d,e). Finally, the robot finished with a complete topological map of the environment, as illustrated in Fig. 6f.

Figure 7 presents the proposed algorithm in action using the Turtlebot 2 robot within the corridors of the fourth floor of the McConnell Engineering building at McGill

<sup>1</sup><http://www.ros.org/>

<sup>2</sup><http://wiki.ros.org/stage>

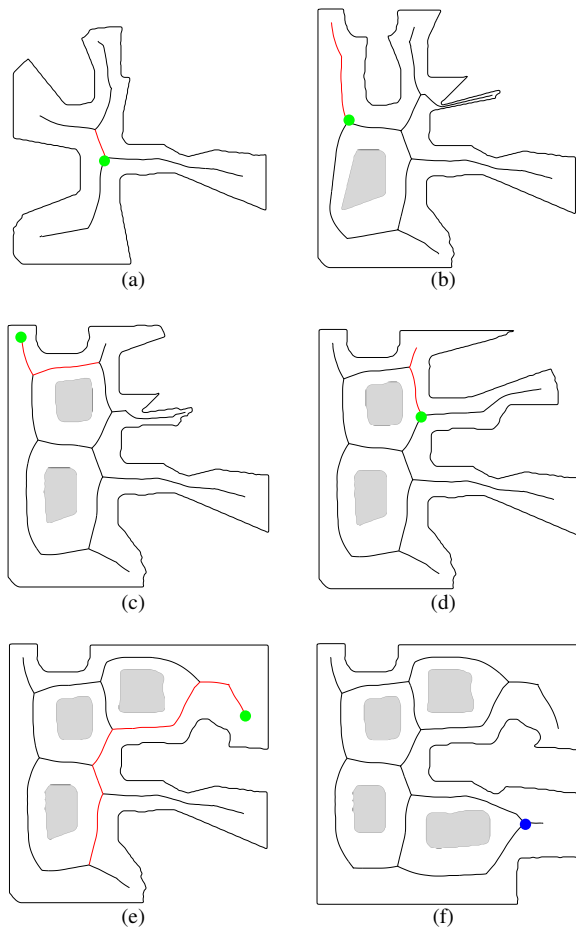


Fig. 6. Six steps of the exploration algorithm, using the Stage cave simulated world, are shown here. At each step the robot's position, the skeleton of the mapped environment, obstacles, and the future path is shown. The green disk represents the robot, and the red path is where the robot will traverse next. (f) The pose of the robot is drawn in blue to indicate that the robot has now fully explored the map.

University. Here the scale of the map changes as the explored environment grows. The robot starts with a very limited view of the environment and the resulting skeleton is a simple curve, the concave part results from the limited field of view of the laser sensor (Fig. 7a). The robot identifies one side as a dead-end and proceeds down the corridor (Fig. 7b), until it detects a junction (Fig. 7c) where the robot decides to follow the right side. Figure 7d shows the robot closing a loop, and then continuing down the corridor selecting the left edge, based on proximity (Fig. 7e). Finally, Fig. 7f presents the completed map of the environment.

## VI. CONCLUSION

A new methodology for the exploration and mapping of an unknown environment was presented in this paper. The algorithm belongs to the family of sensor-based topological maps. In contrast to the Voronoi based topological representations, the proposed approach employs flux-based skeletons which have a smoothing effect on boundary noise, resulting in the elimination of spurious edges. Utilizing all

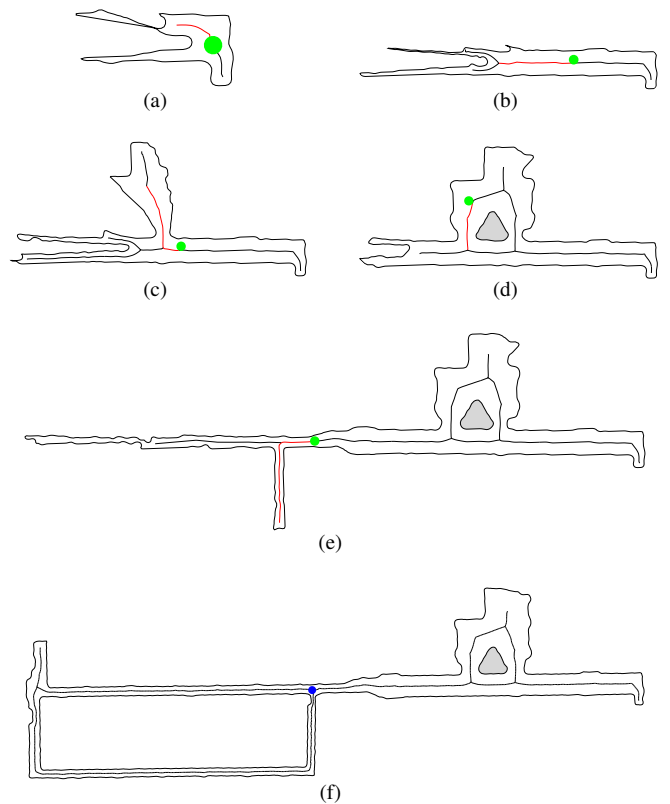


Fig. 7. Six snapshots from an exploration in the corridors of the McConnell Engineering Building at McGill University. This experiment was conducted using the Turtlebot 2 robot. Similar to Fig. 6, the green disk indicates the position of the robot and the red curves the planned trajectory. The blue disk indicates the robot's pose upon termination and successful construction of the skeleton-based topological map.

the recorded data up to the current step results in efficient loop closures and elimination of the side effects of noise. Experimental results from synthetic as well as live data from an exploring robot demonstrated the efficiency and robustness of the proposed framework. The proposed algorithm utilizes all available information collected up to the point of execution, and as such, loop closure is accomplished via the laser mapping step. As an anytime algorithm, the resulting roadmap can be used to guide the exploration to frontier areas while avoiding spurious edges that lead to dead-ends.

We are currently investigating 3D structure reconstruction using a stereo camera inside an underwater cave, depicted in Fig. 8. The 3D structure will be further utilized for the construction of a topological map which will aid in cave exploration and mapping. The flux skeleton method proposed in this paper is very promising as it extends to 3D and results in smooth reconstructions that emphasize the major bounding surfaces in the environment while ignoring debris and small obstacles.

Future extensions of this work will consider the adaptation of the motion planning technique to deploy on aerial vehicles, such as quadrotors, where smoothness of the trajectory is of paramount importance.



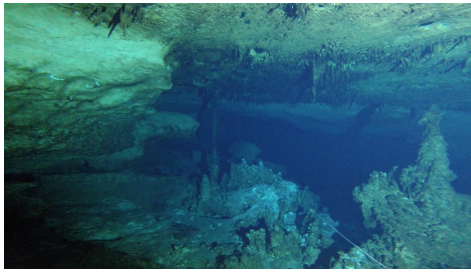


Fig. 8. Left image from a stereo pair of an underwater cave in Mexico. Future work includes online construction of the skeleton of its interior.

## ACKNOWLEDGMENTS

The authors would like to thank the Natural Sciences and Engineering Research Council of Canada (NSERC) for their generous support. We are grateful to Hokuyo Automatic Co., LTD., for the loan of the laser range finder used in the experiments.

## REFERENCES

- [1] B. Kuipers and Y.-T. Byun, "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations," *Robotics and Autonomous Systems*, vol. 8, pp. 46–63, 1991.
- [2] R. Chatila and J. Laumond, "Position referencing and consistent world modelling for mobile robots," in *IEEE Conference on Robotics and Automation*, 1985, pp. 138–170.
- [3] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes, "Robotic exploration as graph construction," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 6, pp. 859–865, 1991.
- [4] H. Choset, K. Nagatani, and A. Rizzi, "Sensor based planning: Using a honing strategy and local map method to implement the generalized voronoi graph," in *SPIE Mobile Robotics*, 1997.
- [5] E. Brunskill, T. Kollar, and N. Roy, "Topological mapping using spectral clustering and classification," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2007, pp. 3491–3496.
- [6] A. Tapus and R. Siegwart, "A cognitive modeling of space using fingerprints of places for mobile robot navigation," in *IEEE Int. Conf. on Robotics and Automation*, 2006, pp. 1188–1193.
- [7] Q. Zhang, D. Whitney, F. Shkurti, and I. Rekleitis, "Ear-based exploration on hybrid metric/topological maps," in *IEEE Conference on Intelligent Robots and Systems*, Sept 2014, pp. 3081–3088.
- [8] H. Blum, "Biological shape and visual science: Part I," *Journal of theoretical Biology*, vol. 38, no. 2, pp. 205–287, 1973.
- [9] F. Leymarie and M. D. Levine, "Simulating the grassfire transform using an active contour model," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 14, no. 1, pp. 56–75, 1992.
- [10] C. Arcelli and G. Sanniti di Baja, "Ridge points in euclidean distance maps," *Pattern Recognit. Lett.*, vol. 13, no. 4, pp. 237–243, Apr. 1992.
- [11] K. Siddiqi, S. Bouix, A. Tannenbaum, and S. W. Zucker, "Hamilton-Jacobi skeletons," *International Journal of Computer Vision*, vol. 48, no. 3, pp. 215–231, 1999.
- [12] R. L. Ogniewicz, "Discrete voronoi skeletons," Ph.D. dissertation, Diss. Techn. Wiss. ETH Zürich, Nr. 9876, 1992. Ref.: O. Kübler; Korref.: T. Pun, 1992.
- [13] D. J. Sheehy, C. G. Armstrong, and D. J. Robinson, "Shape description by medial surface construction," *IEEE Trans. Vis. Comput. Graph.*, vol. 2, no. 1, pp. 62–72, Mar. 1996.
- [14] E. C. Sherbrooke, N. M. Patrikalakis, and E. Brisson, "An algorithm for the medial axis transform of 3D polyhedral solids," *IEEE Trans. Vis. Comput. Graph.*, vol. 2, no. 1, pp. 44–61, Mar. 1996.
- [15] R. Fabbri, L. F. Estrozi, and L. D. F. Costa, "On voronoi diagrams and medial axes," *Journal of Mathematical Imaging and Vision*, vol. 17, no. 1, pp. 27–40, 2002.
- [16] M. Schmitt, "Some examples of algorithms analysis in computational geometry by means of mathematical morphological techniques," in *Geometry and Robotics*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 1989, pp. 225–246.
- [17] F. Chazal and R. Soufflet, "Stability and finiteness properties of medial axis and skeleton," *J. Dyn. Control Syst.*, vol. 10, no. 2, pp. 149–170, 2004.
- [18] D. Attali, J.-D. Boissonnat, and H. Edelsbrunner, "Stability and computation of medial axes - a State-of-the-Art report," in *Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Exploration*, ser. Mathematics and Visualization. Springer Berlin Heidelberg, 2009, pp. 109–125.
- [19] F. Chazal and A. Lieutier, "The  $\lambda$ -medial axis," *Graph. Models*, vol. 67, no. 4, pp. 304–331, Jul. 2005.
- [20] H. Choset and J. Burdick, "Sensor based planning, part ii: Incremental construction of the generalized voronoi graph," in *Int. Conf. on Robotics and Automation*, 1995, pp. 1643 – 1648.
- [21] H. Choset, "Incremental construction of the generalized voronoi diagram, the generalized voronoi graph and the hierarchical generalized voronoi graph," in *CGC Workshop on Computational Geometry*, 1997.
- [22] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes, "Using multiple markers in graph exploration," in *Proc. Symposium on Advances in Intelligent Robotics Systems: Conf. on Mobile Robotics*, 1989.
- [23] G. Dudek, P. Freedman, and S. Hadjres, "Using local information in a non-local way for mapping graph-like worlds," in *Int. Joint Conf. Artificial Intelligence*, 1993, pp. 1639–1647.
- [24] H. Wang, M. Jenkin, and P. Dymond, "Enhancing exploration in graph-like worlds," in *Conf. on Computer and Robot Vision*, 2008, pp. 53–60.
- [25] H. Choset and K. Nagatani, "Topological simultaneous localization and mapping (SLAM): Toward exact localization without explicit localization," *IEEE Tran. on Robotics and Automation*, vol. 17, pp. 125–137, 2001.
- [26] S. Thrun, "Learning metric-topological maps for indoor mobile robot navigation," *Artificial Intelligence*, vol. 99, no. 1, pp. 21–71, 1998.
- [27] B. Lisien, D. Morales, D. Silver, G. Kantor, I. M. Rekleitis, and H. Choset, "The hierarchical atlas," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 473–481, June 2005.
- [28] S. Tully, G. Kantor, H. Choset, and F. Werner, "A multi-hypothesis topological SLAM approach for loop closing on edge-ordered graphs," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009, pp. 4943–4948.
- [29] F. Savelli, "Loop-closing and planarity in topological map-building," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, 2004, pp. 1511–1517.
- [30] Q. Zhang, I. Rekleitis, and G. Dudek, "Uncertainty reduction via heuristic search planning on hybrid metric/topological map," in *12th Conference on Computer Robot Vision*, Halifax, Nova Scotia, Canada, June 2015, pp. 222–229.
- [31] B. Kuipers, J. Modayil, P. Beeson, M. MacMahon, and F. Savelli, "Local metrical and global topological maps in the hybrid spatial semantic hierarchy," in *IEEE Int. Conf. on Robotics and Automation*, 2004, pp. 4845–4851.
- [32] G. Grisetti, C. Stachniss, and W. Burgard, "Improving grid-based slam with rao-blackwellized particle filters by adaptive proposals and selective resampling," in *IEEE International Conference on Robotics and Automation*. IEEE, 2005, pp. 2432–2437.
- [33] K. Siddiqi and S. Pizer, *Medial representations: mathematics, algorithms and applications*. Springer, 2008.