Ear-based Exploration on Hybrid Metric/Topological Maps

Qiwen Zhang[1], David Whitney[1], Florian Shkurti[1], and Ioannis Rekleitis[1]

***Abstract*— In this paper we propose a hierarchy of techniques for performing loop closure in indoor environments together with an exploration strategy designed to reduce uncertainty in the resulting map. We use the generalized Voronoi graph to represent the indoor environment and an extended Kalman filter to track the pose of the robot and the position of the junctions (vertices) of the topological graph. Every time a vertex is revisited, the robot re-localizes and updates the uncertainty estimate accordingly. Finally, since the reduction of the map uncertainty remains one of the main concerns, the robot will optimize its schedule of revisiting junctions in the environment in order to reduce the accumulated uncertainty. Experimental results from a mobile robot equipped with a laser range-finder and results from realistic simulations that validate our approach are presented.**

I. Introduction

This paper presents a solution to the problem of mapping an indoor environment by combining local sensor data into a hybrid topological/metric representation. The presented approach includes an exploration strategy, an environment representation, and a hierarchical approach to the loop closure problem. More specifically, it is assumed that the environment is static and that the robot has no prior knowledge of its starting position or the space layout. The environment is completely GPS-denied and the only sensory information available to the robot are scans from a laser range-finder and odometry information; see Fig. 1. The proposed algorithm presents a robust way of mapping an unknown environment by performing loop closure in a systematic manner. The main building block of this work is the efficient on-line construction of the 2D generalized Voronoi graph (GVG) [1] based on local laser scans taken by the robot.

Many different environments can be mapped adequately using a topological (graph) representation. The most common ones are indoor areas with long corridors, such as those found in office buildings, but underground mines and cave systems are also good candidates. The main advantage of topological representations is that they encode path-planning guidance for the vehicle through a roadmap representation, without using high-dimensional representations of the environment. In other words, the robot can move through the environment by simply following a path in the roadmap graph. In addition, the nodes in the graph provide distinct places which can be used for localization. In this paper, the topological representation used is the generalized Voronoi graph (GVG), a graph that encodes all the points in free space that are equidistant to at least two obstacles. Nodes in the graph represent points that are equidistant to three or more obstacles, whereas edges contain points that are equidistant to exactly two obstacles. As the robot navigates through space, it can employ any known technique [2], [3], [4], [5] to keep track of its pose; when it reaches a vertex it will perform a high-level location matching in order to further reduce the pose uncertainty. In this work we employ an ear-based exploration strategy[1], mapping the environment and populating the graph one cycle at a time. This method ensures frequent loop-closure, thus further reducing the localization error.

Fig. 1. The experimental setup used throughout this paper. The robot, a TurtleBot 2, is equipped with a Hokuyo laser range-finder used for navigation. The robot was used to explore the corridors of the Centre for Intelligent Machines, McGill University.

In the proposed exploration framework, where uncertainty reduction is one of the fundamental concerns, different choices are made in order to balance accuracy and efficiency. In general, when a robot explores an unknown environment to produce a map, the following decisions are made:

- **Choice between exploring new territory or re-localizing**: After the robot completes an exploration or re-localization step, the robot chooses to explore new territory with probability $p_{explore}$ or to re-localize with probability $1 - p_{explore}$. In addition, when the robot's pose uncertainty grows above a threshold, the robot will return to a distinctive location whose uncertainty is low in order to re-localize and reduce the robot's pose uncertainty.
- **Destination selection**
  - *Explore*: If the current location is a frontier node, the robot explores new territory. Otherwise, the closest frontier node is selected for exploration from the current location. In this work, explo-

[1]Qiwen Zhang, David Whitney, Florian Shkurti, and Ioannis Rekleitis are with the School of Computer Science, McGill University, Montreal, Canada, [qzhang32,dwhitney]@cim.mcgill.ca [florian,yiannis]@cim.mcgill.ca

[1]An ear is a cycle in a graph that only contains consecutive edges, i.e. no edges in the area enclosed by the cycle [6].

ration of new territory is done via the ear-based exploration strategy [7], facilitating frequent loop-closures by always choosing the left-most edge; see Section IV.
  - *re-localization*: Select a nearby node with high uncertainty as the next destination to have the maximum uncertainty reduction, which is a technique that has been explored in [8]. By re-visiting a node with high uncertainty the robot will attempt to reduce the uncertainty of that node.
- **Plan a path to destination**: If the robot is re-localizing, we compute the shortest path, using Dijkstra's algorithm, through the known graph to guide the robot to the chosen destination. Otherwise, we need to compute the shortest path through the known graph to reach the closest frontier node to further explore new territory.

II. Related Work

The use of topological maps can be traced back to the seminal paper by Kuipers and Byun [9]. Later work includes the first GVG presentation [1] and the pure topological results in [10], [12], [11]. More recent work includes [13] on exploration strategies on a graph-like world. The use of a hybrid topological and metric representation of the world in order to achieve loop-closure is not a novel idea in the field of robotics. Werner et al. [14] proposed a particle filtering technique for loop-closure whereby a robot traverses a GVG representation of the environment and attempts to disambiguate its location by using a sequence of signatures corresponding to the appearance of the visited GVG meetpoints. The approach assumes prior knowledge of the environment in order to compare the sequence of signatures of the recently visited meetpoints with the local GVG neighborhood around the estimated robot position. Furthermore, the paper contains no mention of how to construct signatures of the meetpoints since it depends largely on the available sensory information.

A similar idea was explored by Tapus and Siegwart [15], who overlaid a rich gamut of sensory information, such as corners extracted from laser scans, visual features and color information on top of a topological representation of the environment. As a result, their approach provides highly distinctive fingerprints of distinct locations, which they managed to exploit in their topological SLAM algorithms. Essentially, this approach turns the loop-closure problem into a sequence alignment problem of the fingerprints of visited locations.

Tully et al. [16] proposed a hypothesis tree approach for loop-closing where the branches that are deemed to be unlikely based on metric and topological GVG information are pruned. Among their pruning tests, it is worthwhile to mention the planarity test which ensures that once a loop-closure has been accepted, the resulting GVG graph has to be planar. The utility of this test has been studied extensively by Savelli [17]. The purely topological variant of these approaches had been previously examined by [18]

Kuipers et al. [19] proposed the use of a framework, referred to as hybrid spatial semantic hierarchy, whereby metric SLAM methods for the creation of small-scale maps are combined with the incremental construction of topological large-scale maps. Their approach makes no use of a global frame of reference and uses a multi-hypothesis approach to represent possible loop-closures.

Brunskill et al. [20] followed a different approach from the ones described above. Spectral clustering was used in order to separate a graph representation of the world into distinct sub-graphs to maximize the connectivity among nodes that are similar and to minimize connectivity among those that are different. Using an AdaBoost-trained classifier (constructed off-line), they managed to compare the sub-graph resulting from recent laser scans against the existing sub-graphs obtained from spectral clustering, thus doing sub-graph matching on-line. The classifier and the combination of metric and topological features used were inspired by the work of Mozos et al. [21].

Carlone and Lyons [22] have approached the problem of autonomous exploration of an unknown environment by considering a simple linear framework to estimate the robot's position. Their formulation leads to a mixed-integer linear program (MILP) which they have shown to plan effective collision-free trajectories.

Choset and Nagatani [23] used the GVG representation of the environment by overlaying metric information at meetpoints and used a number of heuristics to eliminate candidate GVG meetpoints when attempting to do loop-closure. Thus, the approach followed in their paper is most similar in spirit to the work done in this project.

The work of Choset et al. [1], [24], [25] has been the main proponent of the GVG in the existing literature. Beeson et al. [26] extended their work in cases where the operating range of the robots sensors is too small compared to the distance of the objects in environment. The GVG has been also used to guide rendezvous strategies for multiple robots, using the visible area as the deciding factor [27].

III. Generalized Voronoi Graph

In two-dimensional environments, the GVG is the locus of points that are equidistant to at least two distinct objects in the environment. The points that are equidistant to exactly two distinct objects form GVG edges, while the ones that are equidistant to more than two distinct objects define the GVG vertices, also called "*meetpoints*". In addition to representing the environment, the GVG also dictates the rules for robot navigation from which simple control laws can be formulated. For instance, the navigation between meetpoints simply follows the equidistant point between the two distinct obstacles. As the robot constructs the GVG, it will check if it has previously visited the meetpoints that it encounters. If the current meetpoint has never been visited, a new vertex is created and added to the representation. The meetpoint encodes the degree of the vertex and the direction from which the robot arrived. Essentially, we are adding bi-directional edges to the graph, connecting the last meetpoint visited to the current one. The remaining unvisited directions are marked as frontier edges and similarly, the

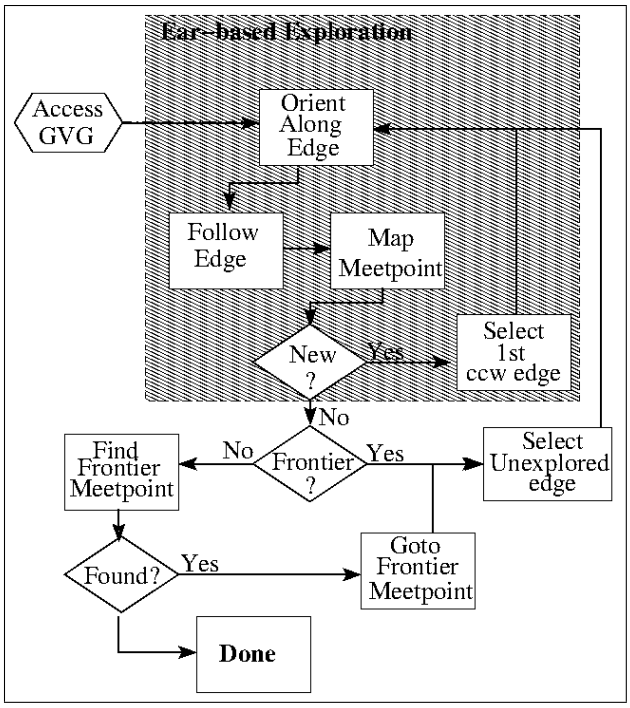


Fig. 2. Ear-based exploration using the GVG representation.

meetpoint is marked as a frontier vertex. In other words, the robot knows that new areas can be visited by exploring the frontier edges. During the robot's navigation, if the current meetpoint has been fully explored, then the vertex is marked as completed. From that location, the robot searches through the graph to find the closest frontier vertex to explore next. Since the chosen frontier vertex has previously been visited, there exists a path through the existing graph to the frontier vertex. If there are no frontier vertices left, then the whole environment has been mapped and the algorithm terminates.

From a purely topological standpoint, the GVG is a graph consisting of vertices and edges which provides a topological representation of the environment. However as the robot traverses through a real environment, additional metric information can be associated with the topological representation to facilitate loop-closure. The flowchart of the algorithm for the exploration and mapping of an unknown environment using the GVG representation is presented in Fig. 2. The basic operations of the on-line construction of the GVG are the following:

Access GVG: This is the initial part where the robot has to reach a position on a GVG edge. First, the robot detects the closest obstacle point. Then, the robot heads towards the opposite direction of that point until it arrives at the midpoint between the two obstacles.

Orient Along Edge: this method assumes that the robot starts on a GVG edge, but its orientation may not be aligned with the orientation of the GVG edge. Therefore, this procedure rotates the robot towards the direction that is perpendicular to the line joining the two closest obstacles. Intuitively, this direction of heading can be regarded as the tangent vector of the curve that defines the GVG edge.

Follow Edge: this method is the backbone of the navigation algorithm. It operates under the assumption that the robot is on the GVG and is properly oriented with respect to a GVG edge. It commands the robot to traverse the GVG edge (without making U-turns) by continuously locating the current pair of distinct closest obstacles and guiding the robot to stay equidistant to them. PD-controllers are employed to reduce the robot's distance from the GVG and to continuously fix the robot's orientation. The obstacle detection is performed by detecting the closest laser point at distance d_{min}. Then all the laser points that are further than $d_{min}+\varepsilon$ are filtered out. The remaining points inside the annulus centered at the laser range finder, are radially ordered and grouped in distinct obstacles. When only two obstacles are detected the robot is using the two closest points to guide the navigation along the GVG edge. When three or more obstacles are detected, the algorithm checks for the existence of a meetpoint. When the robot approaches a meetpoint, the robots velocity is reduced to facilitate its detection.

Detect Meetpoint: This method is constantly evaluated to check if there are at least three distinct and equidistant obstacles from the current position of the robot. If it is, then the meetpoint is mapped and added to the graph. Using the laser scans, all the different edge-orientations are calculated and stored.

Select Edge: Using the ear-based exploration, this method selects the next frontier edge; see Section IV.

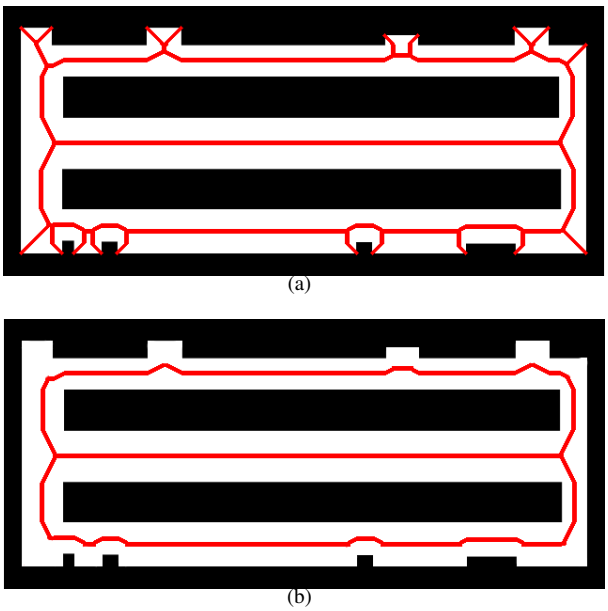


Fig. 3. (a) The pure Voronoi graph of an indoor-like environment. (b) The generalized Voronoi Graph with spurious edges eliminated.

Detect Endpoint: while the robot follows a GVG edge, the robot may reach a dead end. Endpoints are identified by three equidistant obstacles which are fully connected with nowhere for the robot to traverse. Connectivity is detected when consecutive points are too close to each other for the robot to pass through. Whenever the robot reaches an endpoint, it will store information in the graph, turn around

and head away from the endpoint. Endpoints are treated just like meetpoints; hence, they are considered as GVG vertices with degree one. In addition to marking the endpoints, this method also eliminates spurious edges in the pure Voronoi graph.

One of the fundamental weaknesses of Voronoi-based representations is the generation of spurious edges. Small structures or small amounts of sensor noise in the environment may give rise to meetpoints with edges leading into the wall.

Figure 3 illustrates the elimination of noisy edges; the pure Voronoi graph of an indoor-like environment contains up to twenty edges leading to blocked areas (obstacles); see Fig. 3a. By eliminating these edges, the topological representation contains only two meetpoints; see Fig. 3b. In practice, we avoid spurious edges by checking if the openings between the obstacles are less than the diameter of the robot. In that situation, we ignore the edges that lead into obstacles and follow the GVG through the unobstructed edges.

IV. Ear-based Exploration

Fig. 4. Ear-based exploration: The robot accessed the GVG (from start) then followed the edge up to the first meetpoint V_0. It selected the first edge clockwise and visited the meetpoints $V_1 - V_4$. At meetpoint, V_4 started on the first edge clockwise which will lead it back to V_0. In light blue is the unexplored graph for illustration.

Ear-based exploration was first used for mapping an abstract graph-like world [28] with minimal information available. A marker was left by the robot on the starting vertex and then the robot followed edges by always selecting the next edge clockwise until it returned to the vertex with the marker. This method explored the graph one face at a time by following a series of cycles termed "ears" [6]. The algorithm was also employed to map a camera-sensor network [29] deployed in an indoor environment with the help of a mobile robot [7]. In this paper, the robot traverses through the GVG edges and, upon arriving at a meetpoint, always selects the next edge clockwise; see Fig. 4. The main idea behind ear-based exploration is that the robot will explore frontier nodes in a systematic way until it arrives at a previously mapped vertex, ensuring that loop-closures occur on regular intervals.

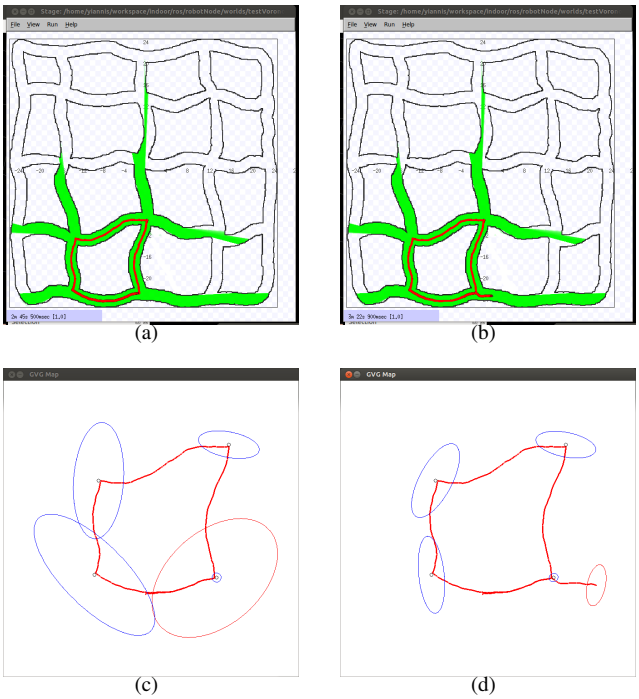


Fig. 5. Loop closure and the effect on uncertainty: (a) The robot approaches the previously visited node. (b) After re-localizing on the node, the robot selects the remaining frontier edge and continues the exploration. (c) Just before closing the loop, the robot's uncertainty has grown. In this example, only the odometry measurements of the robot were used. (d) After re-localization, the uncertainty reduces both for the explored meetpoints and for the robot. Please note that the first node discovered has very little uncertainty because it is considered the start of the map, as such only the sensing uncertainty is used.

During the exploration of the GVG, when the robot selects an unexplored edge to follow, it always performs the selection in the same (clockwise) order, which guarantees a return to a mapped area; see Fig. 5. When the robot arrives to a new vertex, the first operation is to check if this vertex already exists in the graph. This check for loop-closure relies on the accuracy of localization and will be discussed next.

V. Localization

One important aspect of the GVG-based navigation is its reactive nature. The robot is guided by the current sensor readings, moving on a trajectory equidistant to the nearest obstacles. The graph encodes the possible trajectories and the robot localizes at the meetpoints. The range data used for navigation can be also used to improve the estimate of the state of the robot. In this paper, we have not used a laser-based localization in order to verify the efficiency of the proposed approach in the face of large odometric error. Any additional information improving the accuracy of the localization will also further improve the accuracy of the proposed algorithm. Laser based localization can be achieved using any of the different graph-SLAM algorithms available, for example iSAM2 [30] or g^{2}o [31]. in this paper the extended Kalman filter algorithm is used in order to provide an efficient measure of the uncertainty accumulation in the form of the trace of the uncertainty Covariance. Future extensions of this work will use the EKF to simulate

uncertainty build-up in potential paths, selecting the path that minimizes a combined metric of distance and uncertainty, as in [32]. During the traversal of each edge, the EKF estimates the uncertainty accumulation using only the odometry information by repeatedly applying equation Eq. 1.

$$
\begin{aligned}
\mathbf{X}_{t+1} &= \begin{bmatrix} x_{t+1} \\ y_{t+1} \\ \theta_{t+1} \end{bmatrix} = \begin{bmatrix} x_t + v_{t+1} dt \cos\theta_t \\ y_t + v_{t+1} dt \sin\theta_t \\ \theta_t + \omega dt \end{bmatrix} \\
\mathbf{P}^r_{t+1} &= \Phi_{t+1} \mathbf{P}^r_t \Phi^T_{t+1} + G_{t+1} Q G^T_{t+1} \\
\text{where} & \\
\Phi_{t+1} &= \begin{bmatrix} 1 & 0 & -v_{t+1} dt \sin\theta_t \\ 0 & 1 & v_{t+1} dt \cos\theta_t \\ 0 & 0 & 1 \end{bmatrix} \\
G_{t+1} &= \begin{bmatrix} -dt\cos\theta_t & 0 \\ -dt\sin\theta_t & 0 \\ 0 & -dt \end{bmatrix} \\
Q &= \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_\omega^2 \end{bmatrix}
\end{aligned} \tag{1}
$$

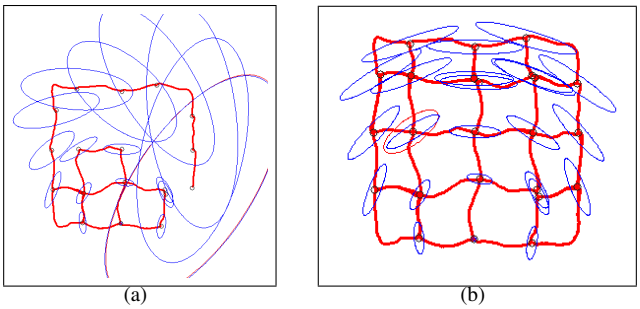
The first time a meetpoint (mtp) is encountered, the filter stores its position $\mathbf{X}_i^{mtp} = [x_i^{mtp}, y_i^{mtp}]^T$ together with the uncertainty of the robot at that point $\mathbf{P}^r_t$. Since there could be uncertainty about the orientation of the meetpoint based on the orientation of the adjacent edges, only the position of the meetpoint is used in the current implementation. Every time meetpoint i is revisited, an update is performed using the update equation in Eq. 2.

$$
\begin{aligned}
S &= H\mathbf{P}H^T + R \\
H &= [-C^T(\theta) - C^T(\theta) J (\mathbf{X}_i^{mtp} - \mathbf{X}_{t+1})] \\
C(\theta) &= \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \\
J &= \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \\
K &= \mathbf{P}H^T S^{-1} \\
\mathbf{X} &= \mathbf{X} + Kr \\
\mathbf{P} &= \mathbf{P} - KSK^T
\end{aligned} \tag{2}
$$

where r is the difference between the recorded value of the meetpoint and the latest estimate and R is the noise characteristic from the laser sensor.

VI. Loop-closure

A hierarchical approach to loop-closure in combination with the exploration strategy results in a robust algorithm for identifying self-similar environments. For instance, in an abstract graph, a vertex is characterized by only one value: its degree. In GVG terms, that is equivalent to the number of objects equidistant from the meetpoint. This view ignores the additional information that can be exploited; for example, the distance of the meetpoint to the obstacles or the angles between GVG edges. Maintaining this sort of information proves to be useful in creating more distinctive signatures/ characterizations for each meetpoint, which helps



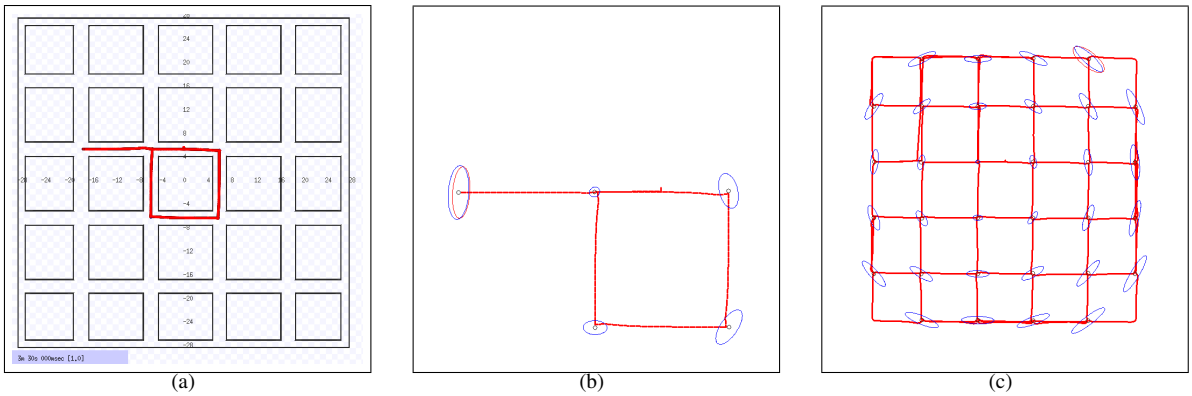
(a) (b)

Fig. 6. Ear-based exploration with delayed loop-closure and the effect on uncertainty; see Fig.5a for the environment. (a) The robot makes left hand turns, thus following the outer perimeter of the environment, accumulating large amounts of uncertainty. Hence, the robot has visited 8 meetpoints consecutively without closing the loop, causing no uncertainty reduction through an EKF update operation. (b) The final map after the whole environment is explored. Uncertainty is reduced. Taking into account the uncertainty build-up addresses the above problem.

to answer the question: "Have I seen this GVG vertex before?". In addition, each meetpoint is a unique location in metric space and there are only a finite number of them distributed in the environment. Therefore, when the robot approaches a meetpoint, the robot's spatial certainty (the area inside the robot's uncertainty ellipse as computed from the covariance matrix) reduces the candidate meetpoints to a very small subset, usually containing only one meetpoint. When more than one candidates are present, the potential candidates undergo a hierarchy of tests to determine the correct matching vertex. Those tests currently consist of a set of manually-tuned thresholds using different metrics such as the total distance of the previous edge and the orientations of the different obstacles at the current location. The following tests are employed for loop-closure:

- Existing meetpoint is inside the 3σ ellipse describing the uncertainty of the robot's pose estimate.
- The potential meetpoint candidates have the same degree.
- The meetpoint distance to obstacles is similar, up to a noise factor.
- Departing angles of the GVG edges, or equivalently the relative bearing to the closest obstacles.
- Edge signature. The minimum information used is the total distance travelled to traverse the edge. The shape of the edge as recorded by the odometry provides another weak indication. In our experiments, each vertex was uniquely identified before reaching this level of distinction.

The last item requires further discussion: when the robot traverses an edge, it records the actual GVG point on that edge. In addition, the starting and ending orientation can also be used in the signature. Finally, the distribution of distances to the closest obstacles as recorded along the edge traversal provide a much richer signature. However, it has not been used in this paper. Additional information can be used to augment a vertex's signature. Potential information includes: Wi-Fi signal strength, laser scan signature at the meetpoint, visual data. In particular, Wi-Fi signal strength and laser scan data are currently under investigation.



(a) (b) (c)

Fig. 7. The exploration of a regular grid with many self-similar locations; the exploration algorithm depended mainly in the uncertainty criterion in order to select and eliminate nodes. (a) The simulation environment with the robot path after closing the first loop. (b) Partial map of the regular grid world; loop-closure. (c) Complete map of the exploration. The blue ellipses represent the positional uncertainty of each meetpoint and the red ellipse represents the pose uncertainty of the robot.

VII. Experimental Results

A. Experimental setup

The software framework for the proposed approach was implemented using the ROS[2] environment. The simulated tests were performed in the Stage[3] simulation package. The experiments were executed using a TurtleBot 2[4] mobile robot equipped with a Hokuyo[5] laser range finder with 30 m range.

B. Simulated world

Several different simulated worlds were constructed for testing the proposed approach. Figure 5a presents an irregular grid-like world with high connectivity. The irregularity, similar to a cave environment, resulted in a few places to have two meetpoints very close to each other. In each case, the use of the extended signature facilitated the loop-closing. A regular grid is presented in Fig. 7. The majority of the meetpoints are self-similar. The simulated robot used the criterion of inclusion inside the 3σ uncertainty ellipse to enable vertex matching. The further the robot navigates away from the starting node, the higher the uncertainty. It is worth noting that self-similar worlds present the biggest challenge in loop-closure, thus the grid-worlds were select to challenge our approach.

C. Real world experiments

Two different floors of the McConnell Engineering Building, McGill University were used as the testing ground for our approach. Both floors are approximately 30 by 50 meters. The first experiment Fig. 9 was conducted on the 3rd floor housing the School of Computer Science. This environment consisted of two small loops near the center of the environment, and two long branches. Figure 9a presents the odometric traces together with the resulting map. Due to the localization update operations at the meetpoints, the odometry is drastically corrected when the same place is revisited. As can be seen from the resulting maps, while the odometric estimates drifted over the long traversals, the meetpoint detection successfully closed the loops present in the environment. Figure 9b presents the resulting map overlaid on top of a metric map constructed off-line using the gmapping package from ROS. The error at the end of the long corridors was large (approximately $2m$) compared with the cycles at the center of the environment. The 4th floor housing the Centre for Intelligent Machines was mapped in Fig. 8. This floor contains three larger loops and a single long branch. In Fig. 8 and 10b, the odometry estimates are plotted in red, demonstrating the drift over time. Every time the robot revisits an explored node the odometry is reset to the correct value.

In Fig. 8a the robot traversed a loop and returned at the starting node. The robot's estimate of its position was erroneous, however, still inside the uncertainty ellipse. The loop-closure update of the EKF operating on the meetpoints of the environment reset the pose of the robot and reduced the uncertainty. The red line represents the odometric estimates and thus is reset after re-localization. Figure 8b illustrates the effect of travelling long distances without no re-localization. Note that the large uncertainty ellipse at the right side of the environment together with the erroneous trajectory estimate was quickly corrected when the robot returned to the middle of the environment. In the final part, the robot moved on a single corridor and returned. The uncertainty accumulated during the long traversal was not feasible to be reduced due to the absence of nearby loops. While the odometry-based trajectories are unchanged, the meetpoint locations are updated and thus drift during the experiment. The result is two different trajectories for the edge connecting the center part with the rightmost loop, and the meetpoints at the top of the map to be away from the odometric trace.

During the experiments the laser and odometric information was also recorded. The results were processed off-line using the gmapping software package and a map of the environment was produced for comparison. Figure 10a presents the resulting map and is given here for comparison with the GVG-based topological map; see Fig. 10c. Please note that in the EKF formulation, only the location of the

[2]http://wiki.ros.org/
[3]http://wiki.ros.org/stage
[4]http://turtlebot.com/
[5]http://www.hokuyo-aut.jp/

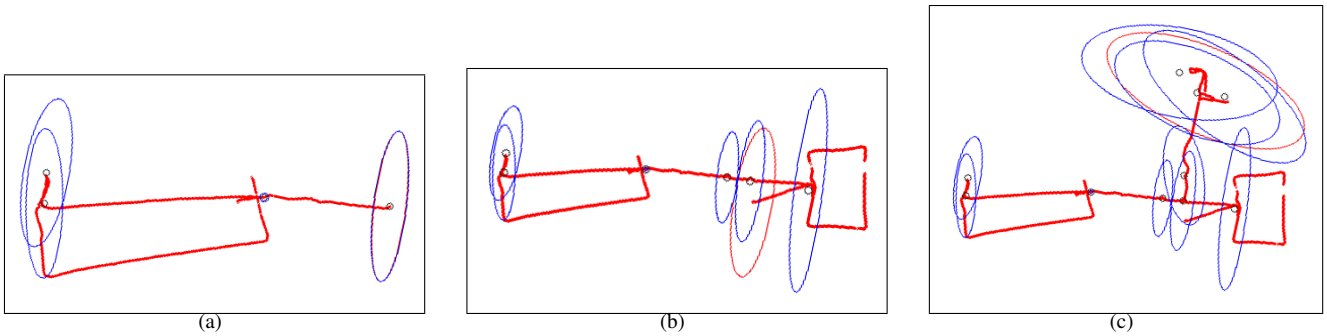


Fig. 8. Illustrating the progress of the exploration process, by plotting the meetpoints poses and uncertainty and the odometry trace. (a) The first loop of the environment was closed. The robot traversed along a long corridor and explored a second loop. (c) The last corridor is explored. The final map is presented in Fig. 10b.

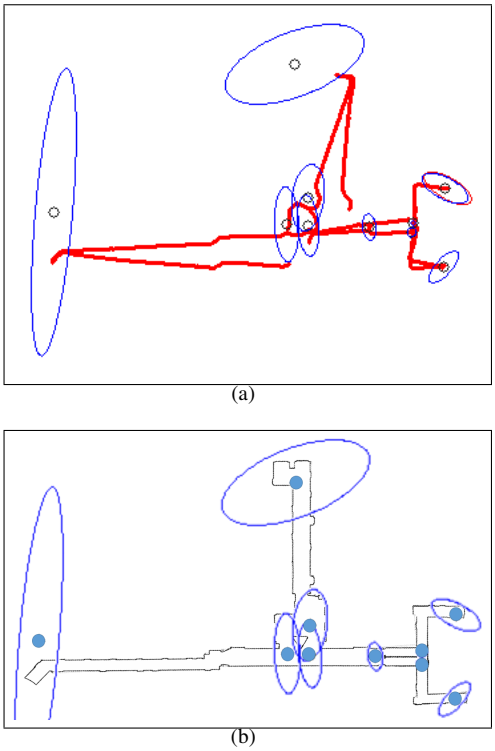


Fig. 9. Exploring the main floor of the School of Computer Science, McGill University. (a) The vertices of the topological map with the odometry traces (in red). (c) The meetpoint locations with uncertainty ellipses overlaid on top of a metric map constructed by gmapping for visualization purposes.

meetpoints is recorded and updated. Moving along an edge is robustly handled by the GVG navigation algorithm using local sensing information only, thus no edge information is recorded. In Fig. 10c the edges are drawn connecting the meetpoints for illustration purposes only (in dotted lines). It is worth noting that similar accuracy was achieved by employing loop-closure only at the meetpoints while navigating through the environment as when computationally expensive scan matching operations were employed throughout the exploration.

A comparison between figures 9b and 10c demonstrates the improvement resulting from the frequent loop-closure, when possible from the environment.

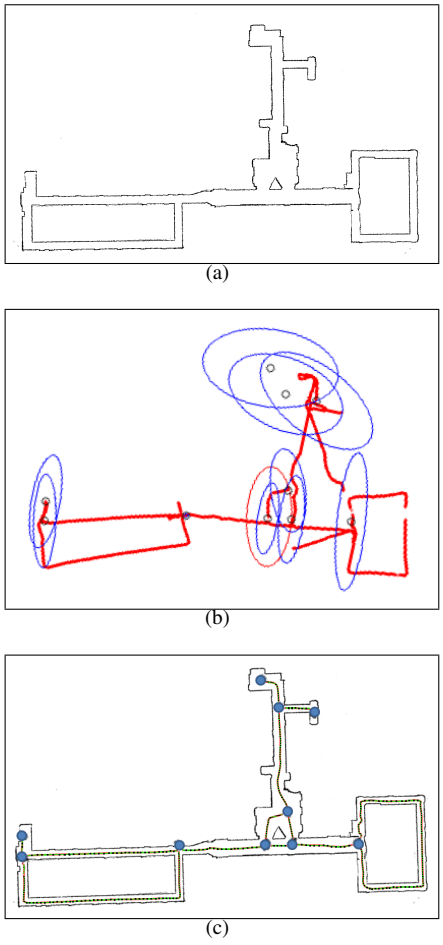


Fig. 10. (a) The map of the Centre for Intelligent Machines, McGill University constructed using the ROS package gmapping. This map serves as a comparison to the GVG based map presented in subfigure 10c. (b) The vertices of the topological map with the odometry traces (in red). (c) The final topological map of the Centre for Intelligent Machines constructed using the GVG exploration algorithm presented in this paper.

VIII. Conclusion

This paper presented a complete solution for the exploration and mapping of an unknown indoor environment using a hybrid metric/topological representation. The map is based on the generalized Voronoi graph, and the ear-based exploration strategy is used to perform loop-closure in regular intervals. Finally, place identification restricted to the

meetpoints is achieved by measuring a hierarchy of similarity measures among candidate nodes.

Extending the Simultaneous Localization and Uncertainty Reduction on Maps (SLURM) applied to a camera sensor network [8], [7] to operate on the topological structure of the GVG is currently a work in progress. In particular, making a choice between exploring new areas and revisiting already mapped territory in order to re-localize will increase the accuracy of the resulting map and improve the robustness of the approach. Deploying the proposed algorithm to multiple robots is another extension currently under consideration.

The proposed approach can be easily extended to other robotic platforms that have similar sensors and capabilities. In this paper, we presented results from simulation and from the TurtleBot 2 platform, but we have successfully deployed our method on the Husky[6] robot, in the same indoor environments presented here. The integration of our method on the new system was effortless, thanks to the modularity of ROS and ease of code re-use. The code developed for this work is available on-line [7].

The developed solution provides an efficient alternative to the frontiers based exploration on occupancy grid maps. Mapping the corridors of the Centre for Intelligent Machines, McGill University with a laser carrying robot illustrates our approach.

ACKNOWLEDGEMENT

The authors would like to thank the Natural Sciences and Engineering Research Council of Canada (NSERC) for their generous support. We would like to gratefully acknowledge the loan of the laser range finder used in the experiments by Hokuyo Automatic Co., LTD. This work was done at the Mobile Robotics Lab of Professor Dudek.

REFERENCES

[1] 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.

[2] G. Dudek and P. MacKenzie, "Model-based map construction for robot localization," in *Vision Interface*, 1993.

[3] D. Haehnel, D. Fox, W. Burgard, and S. Thrun, "A highly efficient fastslam algorithm for generating cyclic maps of large-scale environments from raw laser range measurements," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2003.

[4] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with rao-blackwellized particle filters," *IEEE Tran. on Robotics*, 2006.

[5] M. Kaess, A. Ranganathan, and F. Dellaert, "isam: Incremental smoothing and mapping," *IEEE Tran. on Robotics*, vol. 24, no. 6, pp. 1365–1378, 2008.

[6] Y. Maon, B. Schieber, and U. Vishkin, "Parallel ear decomposition search (EDS) and st-numbering in graphs," *Theoretical Computer Science*, vol. 47, no. 3, pp. 277 – 298, 1986.

[7] I. Rekleitis, "Simultaneous localization and uncertainty reduction on maps (slurm): Ear based exploration," in *IEEE Int. Conf. on Robotics and Biomimetics*, 2012, pp. 501–507.

[8] ——, "Single robot exploration: Simultaneous localization and uncertainty reduction on maps (slurm)," in *Conf. on Computer and Robot Vision*, 2012, pp. 214–220.

[9] B. Kuipers and Y.-T. Byun, "A robot exploration and mapping strategy based on a semantic hierachy of spatial representations," *Robotics and Autonomous Systems*, vol. 8, pp. 46–63, 1991.

[10] 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.

[11] G. Dudek, P. Freedman, and S. Hadjres, "Using local information in a non-local way for mapping graph-like worlds," in *Int. Conf. on Artificial Intelligence*, 1993, pp. 1639–1645.

[12] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes, "Robotic exploration as graph construction," *Tran. on Robotics and Automation*, vol. 7, no. 6, pp. 859–865, 1991.

[13] H. Wang, M. Jenkin, and P. Dymond, "Enhancing exploration in graph-like worlds," in *Conf. on Computer and Robot Vision*, 2008, pp. 53–60.

[14] F. Werner, F. Maire, J. Sitte, H. Choset, S. Tully, and G. Kantor, "Topological SLAM using neighbourhood information of places," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009, pp. 4937–4942.

[15] 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.

[16] 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.

[17] F. Savelli, "Loop-closing and planarity in topological map-building," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems*, 2004, pp. 1511–1517.

[18] 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.

[19] 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.

[20] 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.

[21] O. Martinez, M. Cyrill, and S. W. Burgard, "Mozos. supervised learning of places from range data using adaboost," Tech. Rep., 2004.

[22] L. Carlone and D. Lyons, "Uncertainty-constrained robot exploration: A mixed-integer linear programming approach," in *IEEE Int. Conf. on Robotics and Automation*, 2014.

[23] 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.

[24] 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.

[25] 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.

[26] P. Beeson, N. K. Jong, and B. Kuipers, "Towards autonomous topological place detection using the extended voronoi graph," in *IEEE Int. Conf. on Robotics and Automation*, 2005, pp. 4373–4379.

[27] M. Meghjani and G. Dudek, "Multi-robot exploration and rendezvous on graphs," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2012, pp. 5270 – 5276.

[28] I. Rekleitis, V. Dujmović, and G. Dudek, "Efficient topological exploration," in *Int. Conf. in Robotics and Automation*, 1999, pp. 676–681.

[29] I. Rekletis and G. Dudek, "Automated calibration of a camera sensor network," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2005, pp. 401–406.

[30] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the bayes tree," *Int. Journal of Robotics Research*, vol. 31, pp. 217–236, 2012.

[31] R. Kummerle, G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard, "G2o: A general framework for graph optimization," in *IEEE Int. Conf. on Robotics and Automation*, 2011, pp. 3607–3613.

[32] D. Meger, I. Rekleitis, and G. Dudek, "Heuristic search planning to reduce exploration uncertainty," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2008, pp. 3382 – 3399.

[6] http://www.clearpathrobotics.com/husky/

[7] https://github.com/QiwenZhang/gvg.git