

# Simultaneous Localization and Uncertainty Reduction on Maps (SLURM): Ear based Exploration

Ioannis Rekleitis

**Abstract**—Efficient exploration and accurate mapping are two conflicting goals. Efficient exploration requires minimizing traversal of previously mapped territory, accurate mapping necessitates that the robot goes through previously mapped areas to reduce the accumulated uncertainty. This problem has many parallels with the exploration versus exploitation problem. In this paper a new algorithm is proposed that explicitly aims to facilitate loop closure in a systematic way. The problem of localizing a camera sensor network by employing a mobile robot will be used to demonstrate the effect that different parameters of the ear-based exploration strategy have on the speed of exploration and the accumulated uncertainty. Simulation results using a realistic noise model are presented for different environments.

## I. INTRODUCTION

In this paper a new algorithmic approach to the exploration of unknown environments is presented. The proposed method explicitly guides the robot into trajectories that would facilitate loop closure and thus improve the quality of the resulting map in terms of accuracy. In many applications robot(s) enter an unknown environment and have to explore it while simultaneously constructing a map. One important question is how much time should be spent on improving the map quality versus exploring new areas. In this paper we use the paradigm [1] of simultaneously localizing a mobile robot and mapping a camera sensor network. The mobile robot navigates through an environment equipped with a camera sensor network; see Fig. 1. The robot's goal is to localize all the cameras in a global frame of reference, and to maintain its own pose estimate.

Planning trajectories in the face of conflicting goals such as accuracy and efficiency, in terms of distance travelled, in combination with the high dimensional uncertainty estimates of the underlying simultaneous localization and mapping (SLAM) solution results in a challenging problem. In earlier work we examined the problem of localizing the cameras [1] and also performing autonomous calibration [2]. This formulation of the problem eliminates the data association problem which, though relevant, is not at the centre of the decision between exploration and relocalization. In this scenario, the robot's motion through the network places the cameras in the same frame of reference of the robot. By maintaining an ongoing estimate of the robot's location, the position of any sensor that it observes with can be probabilistically estimated, (and updated), given the appropriate motion and measurement models. Furthermore, the environment is represented as a graph, where vertices are locations that can be seen by the cameras and the edges represent accessible paths

School of Computer Science, McGill University, 417-3480 University Street, Montréal, QC, Canada H3A 2A7 [yiannis@cim.mcgill.ca](mailto:yiannis@cim.mcgill.ca)

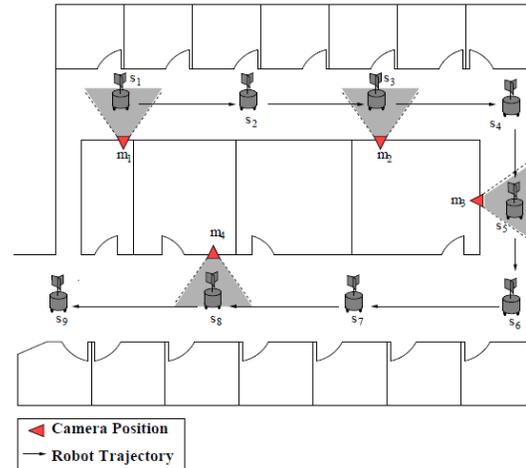


Fig. 1. The experimental setup used throughout this paper. The robot carries a calibration target which can be easily detected in images taken by the cameras in the network.

between these areas; the cameras have no overlapping fields of view.

In general, after an initial phase of exploration, which can be as short as moving to the nearest unexplored territory, the robot is in possession of a partial map that indicates the known areas, and the borders to unexplored territory, usually called frontiers [3]. In the camera sensor network localization, the known map is a graph  $G_t = [V_t, E_t]$  with an additional set of directed edges  $E_t^f$  that have a starting vertex but no ending vertex, because the end is unknown. The exploration and map construction process can be divided conceptually into three components:

- 1) *Explore or Relocalize*: In regular intervals choose between the two options.
  - **Explore**: Visit unexplored nodes.
  - **Uncertainty Reduction**: Go and relocalize a node that has high uncertainty.
- 2) *Where to go next?*
- 3) *How to go there?:* Use an A\* based algorithm that guides the robot to the goal via an optimal path in terms of distance and uncertainty; see [4].

The choice between exploration and uncertainty reduction resembles the strategies that provide an approximate solution to the multi-armed bandit problem [5]. In the robot exploration problem, the reward derives from reducing the map uncertainty, while exploring new edges on the map can provide shorter paths and reduce uncertainty by loop closing. The main difference is that the robot has to explore the complete map, and the degree of relocalization is flexible. In this paper we use an adaptive strategy that increases

the frequency of relocalization during the later phases of the exploration; a strategy similar to “*Epsilon-decreasing strategy*”.

A novel exploration strategy with the implicit goal to close loops is proposed in this paper. This strategy resembles the algorithm proposed in [6]; where a graph-like world is explored one face<sup>1</sup> at the time. The earlier work abstracted the problem of localization to a vertex labelling problem, and addressed it by dropping a pebble as a marker. The strategy of exploring one face at a time, is extended here to an actual loop closing strategy which drastically improves the accuracy of the map. A major difficulty of the earlier work was to identify when a vertex was revisited; as the vertices represent areas seen by cameras with known labels, the data association problem is solved, and thus the proposed algorithm has a clear stopping condition.

This paper is structured as follows. The next section discusses related work. In Section III the new exploration strategy is discussed. Next an outline of the different test environments is described together with extensive experimental results of the proposed exploration strategies. An analysis of the effect of the graph connectivity to the accuracy of the resulting graph is also included. The paper concludes with future work and a description of lessons learned.

## II. RELATED WORK

Localizing a sensor network of cameras with a mobile robot has many similarities with the Simultaneous Localization and Mapping (SLAM) problem. Both require estimating the poses of landmarks (in our case the pose of cameras) and the robot for sensor readings corrupted by noise. In order to achieve that several estimation approaches are feasible, for example particle filters, Kalman filters, information filters, and Markov Chain Monte Carlo techniques. The extended Kalman filter (EKF) [8] used for SLAM has been adapted for camera network localization and is used in this paper. Many other alternative solutions [9] are possible, but the EKF is used here for ease of analysis and computational efficiency. The problem of path planning through known areas in order to increase the accuracy of the map by collecting additional information is well studied, e.g. [10], [11], [12], [13].

Numerous authors have studied the problem of planning paths through the already known map in order to gather additional information and to increase mapping accuracy, e.g. [10], [11], [12], [13]. Many approaches have attempted to reduce the uncertainty in the map estimates [14], [15], [16]. Sim and Roy [10] presented two measures from information theory for which either the determinant or the trace of the covariance matrix provides an estimate for the uncertainty.

Early work proposed a greedy choice of the action, with one step look-ahead, which maximally minimizes the entropy because optimal planning of multi-step paths claims computational cost exponential in the path length. Consequently, Sim and Roy [10] have proposed pruning loops during

<sup>1</sup>For a planar graph a face is a area bounded by edges please note, there is also an outer infinitely-large face. Each face is described by a cycle, also called an ear [7].

breadth first search in order to ensure manageable complexity even when planning longer paths under conditions of idealized sensing and a rough initial estimate of landmark locations. In addition, [11] has proposed a simulation-based strategy which has the potential to produce multi-step paths at the cost of significant computation. More recently the more general solution was proposed using Gaussian processes which is computationally heavy for a robotic planning application [17]

In contrast, the proposed approach considers the more general problem of an unknown environment where the robot dynamically decides if more time should be spent improving positional accuracy by revisiting nodes of high uncertainty, or a route to unknown parts of the world should be selected. This is achieved by employing  $A^*$  search for efficient planning and by selecting future actions based on the condition of the map. Variations of the  $A^*$  search have been used in the past for path-planning in dynamic environments without any consideration for the resulting pose uncertainty, such as the  $D^*$  algorithm proposed by Stentz [18]. Uncertainty, was not considered in  $D^*$ , so our work extends this method by explicitly planning to reduce the uncertainty accumulated while mapping an environment.

As mentioned above, efficiency and accuracy are conflicting goals during exploration. In order to generate paths that balance between the goals, distance and uncertainty have to be combined into one cost function. Unfortunately, the two are incommensurable; that is, they lack common units for comparison, so care must be taken in combining their values. Makarenko *et al.* [15] have proposed previously a weighted linear combination of distance and uncertainty for path  $p$ :

$$C(p) = \omega_d \text{length}(p) + \omega_u \text{trace}(P(p)) \quad (1)$$

In this cost function,  $P$  is the covariance matrix resulting from the EKF and its trace is an approximation of the uncertainty in the map. The choice of weighting factors  $\omega_d$  and  $\omega_u$  represents the compromise between mapping uncertainty and distance travelled or *efficiency* versus *accuracy*. A flexible method is produced based on varying the one intrinsic parameter, the contribution of each quantity is normalized by a rough estimate of its maximum possible value. Once each quantity has been normalized, a single free parameter  $\alpha$  in the range  $[0, 1]$  is able to specify the contribution of each factor. Based on this formulation, the weights used in our cost function are:

$$\omega_d = \frac{\alpha}{\text{maxdistance}} \quad , \quad \omega_u = \frac{1 - \alpha}{\text{maxuncertainty}}$$

By setting  $\alpha$  to the two extremes, zero and one, it is possible to consider only one of the factors at a time: distance only, by setting  $\alpha = 1$ , and uncertainty only, by setting  $\alpha = 0$ . In [4] the effect of varying  $\alpha$  on the quality of the resulting paths was discussed.

Several authors have considered the collaboration between a sensor network and a mobile robot in different sensing

scenarios and in some cases with much more capable robotic agents [19], [20]. The importance of loop closing was identified early on [21] but the structure of the environment was not utilized to enforce loop closure. Theoretical exploration algorithms on graphs have been extensively studied [22] but without taking into account the positional uncertainty accumulated during edge traversals.

### III. SIMULTANEOUS LOCALIZATION AND UNCERTAINTY REDUCTION ON MAPS

There are different ways to measure the uncertainty of a node; in this work it is calculated by the map uncertainty in the 3D position (no orientation uncertainty) of the  $i^{\text{th}}$  camera. This uncertainty is encoded in the state covariance matrix  $\mathbf{P}$  and is represented by the trace of the covariance matrix:

$$\text{trace}(\mathbf{P}^{C_i}) = \sum_{j=3+6*(i-1)+1}^{3+6*(i-1)+3} P_{jj} \quad (2)$$

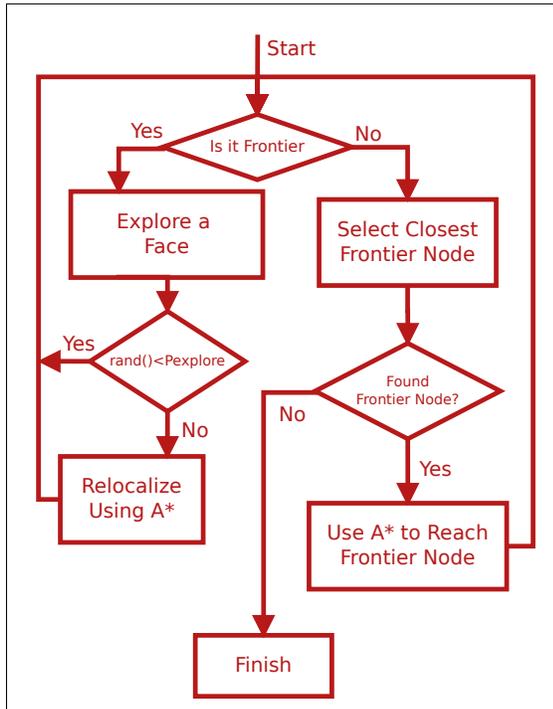


Fig. 2. The flowchart of the Ear based Exploration algorithm.

Figure 2 presents an overview of the proposed algorithm. Many different strategies can be used to decide between exploration and relocalization (explore vs. exploit). Among them the most common are to select with probability  $p_{\text{explore}}$  to explore and with probability  $1 - p_{\text{explore}}$  to exploit. In order to make this choice a random number is drawn uniformly distributed between zero and one. If the number is below  $p_{\text{explore}}$  then the robot proceeds to apply the exploration strategy of its choice; ear-based exploration. Otherwise, the robots selects a goal node and then plans a path through the known graph to the goal node using the A\* algorithm [23]. If the robot is at a node with no frontier edges, then the robot plans a path to the closest frontier node.

#### A. A\* based path planning

The A\* based path planning algorithm presented in [2], [23], [4] takes into account both distance travelled and accumulated uncertainty. Because the planning is happening through the known world, an EKF simulator is capable to predict the effect of different paths in the resulting map uncertainty. Dramatic improvements on the quality of the map were observed when uncertainty affected the cost function, even with a very small weight. In the results presented in this paper every time the robot travels to a new location, a frontier node or a node to relocalize, the A\* algorithm is used, thus its influence is omnipresent.

#### B. Explore vs. Exploit

The choice of the threshold  $p_{\text{explore}}$  is very important in order to control the amount of time spent relocalizing. Traditionally, four different strategies have been used [5]:

*Epsilon-greedy strategy:* The threshold  $p_{\text{explore}}$  is constant throughout the experiment. This has the disadvantage that at the beginning there is not enough information (known map is small) the robot chooses to relocalize with the same frequency as later.

*Epsilon-first strategy:* The robot first explores the world, and then moves through it to reduce uncertainty. This strategy is very hard to justify because errors in the map can result in inconsistencies that would be very hard to correct later on.

*Epsilon-decreasing strategy:* This is the preferred strategy used in this paper. The robot adapts the threshold  $p_{\text{explore}}$  as a function of the percentage of the map covered. We know the total number of cameras in the environment, so this adaptation is modelled as the ratio of mapped over the total number of cameras.

$$p_{\text{exploit}} = \frac{|V_{\text{mapped}}|}{|V_{\text{total}}|} (1 - p_{\text{explore}}) \quad (3)$$

*Adaptive epsilon-greedy strategy based on value differences (VDBE):* The threshold  $p_{\text{explore}}$  changes as a function of the performance. This strategy was not selected because there is no way to know if exploring a new edge would lead to a loop closure, drastically reducing the uncertainty.

A key difference is the fixed number of trials in the bandit problem, and the fixed nature of rewards. In the exploration and map refinement problem one exploration path could drastically reduce the uncertainty by facilitating an early loop closure.

When the robot chooses to relocalize it travels through the known graph using the A\* based algorithm with  $\alpha = 0.01$  that is only one percent of the cost is contributed by distance and the remaining by the uncertainty reduction. The goal node is also important; selecting the node with the highest uncertainty resulted in high uncertainty reduction at high distance costs. We chose the node that minimized a distance and maximized uncertainty:

$$V_i = \text{argmax} \left( \left( 1 - \frac{D_i}{D_{\text{max}}} \right) + \frac{\text{trace}(P_i)}{\text{Uncertainty}_{\text{max}}} \right) \quad (4)$$

where the  $D_i$  and  $D_{max}$  are calculated through the known map (mapped edges) and the  $Uncertainty_{max} = \text{trace}(P_k) : \text{trace}(P_k) \geq \text{trace}(P_l), l = 1, \dots, N_{known}$ .

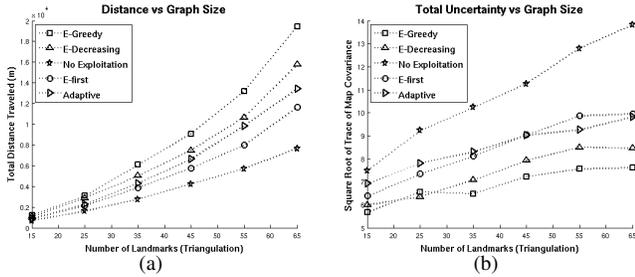


Fig. 3. Results averaged over 15 trials: Distance travelled (a) and map uncertainty (b) for different number of nodes, different number of edges and for different exploitation strategies. We compare the results for applying a version of: Epsilon Greedy, Epsilon-Decreasing, Epsilon first, Adaptive epsilon-greedy, and no exploitation.

Figure 3 presents a comparison for the above strategies, for complete triangulations of fifteen to sixty five nodes. The results labelled “No Exploitation” are shown as a baseline when no exploitation (relocalization) was performed. The exploration problem does not have a fixed number of trials, as the bandit problem has, as such the Epsilon-first strategy applied, run the relocalization procedure for half the total number of nodes. Given that each exploration run on average explored more than one node this biased the results in favour of the Epsilon-first strategy; however, as can be seen in Fig. 3b the uncertainty reduction was minimal.

### C. Ear-based Exploration

Different strategies can be selected for exploring the unknown part of the environment. Selecting the closest frontier node; the node with the highest, or lowest uncertainty; or even a random frontier node are all possible strategies. In this work we propose a strategy that takes into account the planar nature of the environment, and guides the robot into exploring trajectories of multiple nodes, instead of a single node, which end with loop closure when possible. This algorithm was first proposed for mapping an abstract graph-like world [6] with minimal information available. The robot left a marker in the starting node and then followed edges by always selecting the next edge clockwise (or counter-clockwise) until it returned to the node with the marker. This method explored the graph one face at a time, by following a series of cycles termed “ears” [7]. Because of the limited available information there was a specific configuration that resulted into incorrect representations.

In the current paradigm each location (vertex) is associated with an observing camera with a unique label. As such, the data association problem is absent. For a known map  $G_t = [V_t, E_t]$  with a frontier edge set ( $E_t^f$ ) the modified algorithm behaves as follows: starting from a frontier node  $V_i$ , select a frontier edge ( $V_{ij} \in E_t^f$ ); follow the frontier edge; if the robot arrives to a known vertex ( $V_j \in V_t$ ) then stop; otherwise add  $V_j$  to the map and then select the next edge clockwise; if  $V_j$  has degree one, stop.

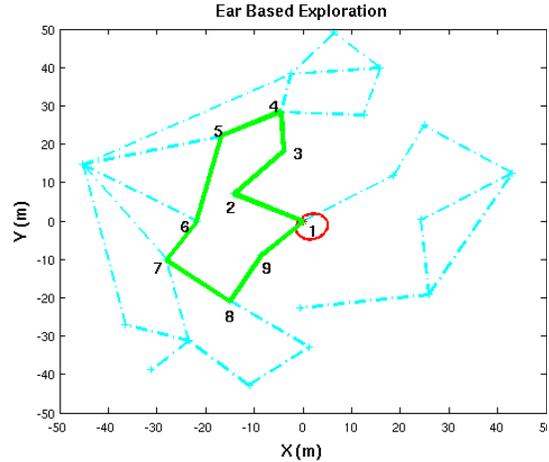


Fig. 4. Ear based exploration. Robot after mapping Camera 1, explores an ear adding the nodes 1-9 to the map.

This algorithm traverses an ear until loop closure or until the robot reaches a leaf (a node of degree one). Figure 4 presents the path taken at the first step; the robot started at node 1 and then followed a nine edge path, adding eight new vertices into the map. An alternative terminating strategy would be to continue traversing the next edge clockwise until the return to the starting vertex. However, in this case the robot would be travelling though explored territory, which is not the most efficient strategy.

## IV. EXPERIMENTAL RESULTS

### A. Environment Representations

Environment representations based on graphs are quite common in robotics, e.g., visibility graphs [24]; generalized Voronoi graphs [25]; Reeb graph [26]; triangulations [27]. In particular, in many cases random graphs can, and have, been used to validate proposed algorithms and also to test scalability and robustness. In this paper we employ a family of random planar graphs with varying density. As a starting point a set of  $N$  random points are chosen on a plane. In order to eliminate the effect of the starting point on the exploration the first point is selected at the origin and the rest are uniformly distributed on the  $x$  and  $y$  axis, ensuring they are inside a fixed radius. Another consideration is to reject any points that are closer than a threshold, 10 m in the results presented in this paper. The rationale for this rejection is that decisions and landmarks that are located too close together are not really effective. The choice of points inside a radius was selected in order to ensure that starting from the origin there will be no bias when moving in any direction.

From the set of random points the Delaunay triangulation [28] is generated. The resulting triangulation represents the denser graph used in this experiments; see Fig. 5d. In order to produce random graphs of varying density, each edge in the triangulation is assigned a random weight and then a minimum spanning tree (MST) is constructed; number of edges  $|E| = N - 1$ . The graph density is measured as a function of the number of vertices  $|V| = N$ , ranging from the sparsest (MST), to the triangulation  $|E| = 3 * |V| - 3 - k^2$ .

<sup>2</sup>Where  $k$  is the number of vertices on the convex hull [28]

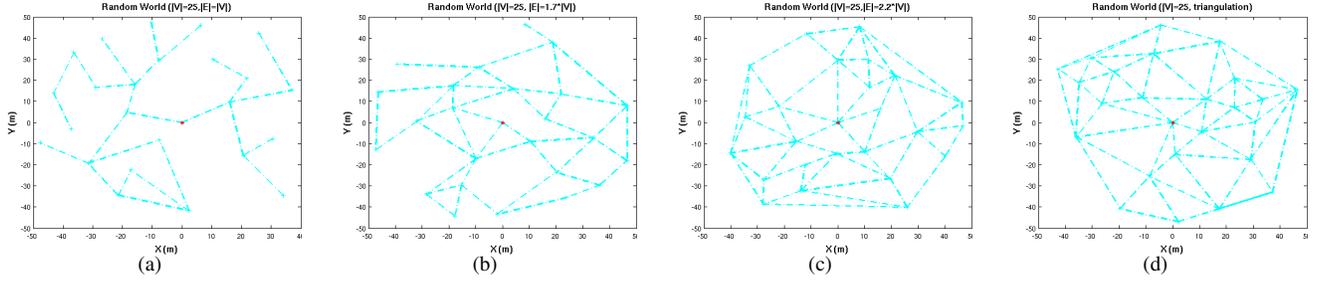


Fig. 5. Random graphs based on the Delaunay triangulation with 25 vertices, and varying density: (a) Almost a spanning tree ( $|V| = 25, |E| = |V|$ ); (b) ( $|V| = 25, |E| = 1.7 * |V|$ ); (c) ( $|V| = 25, |E| = 2.2 * |V|$ ); (d) the complete triangulation.

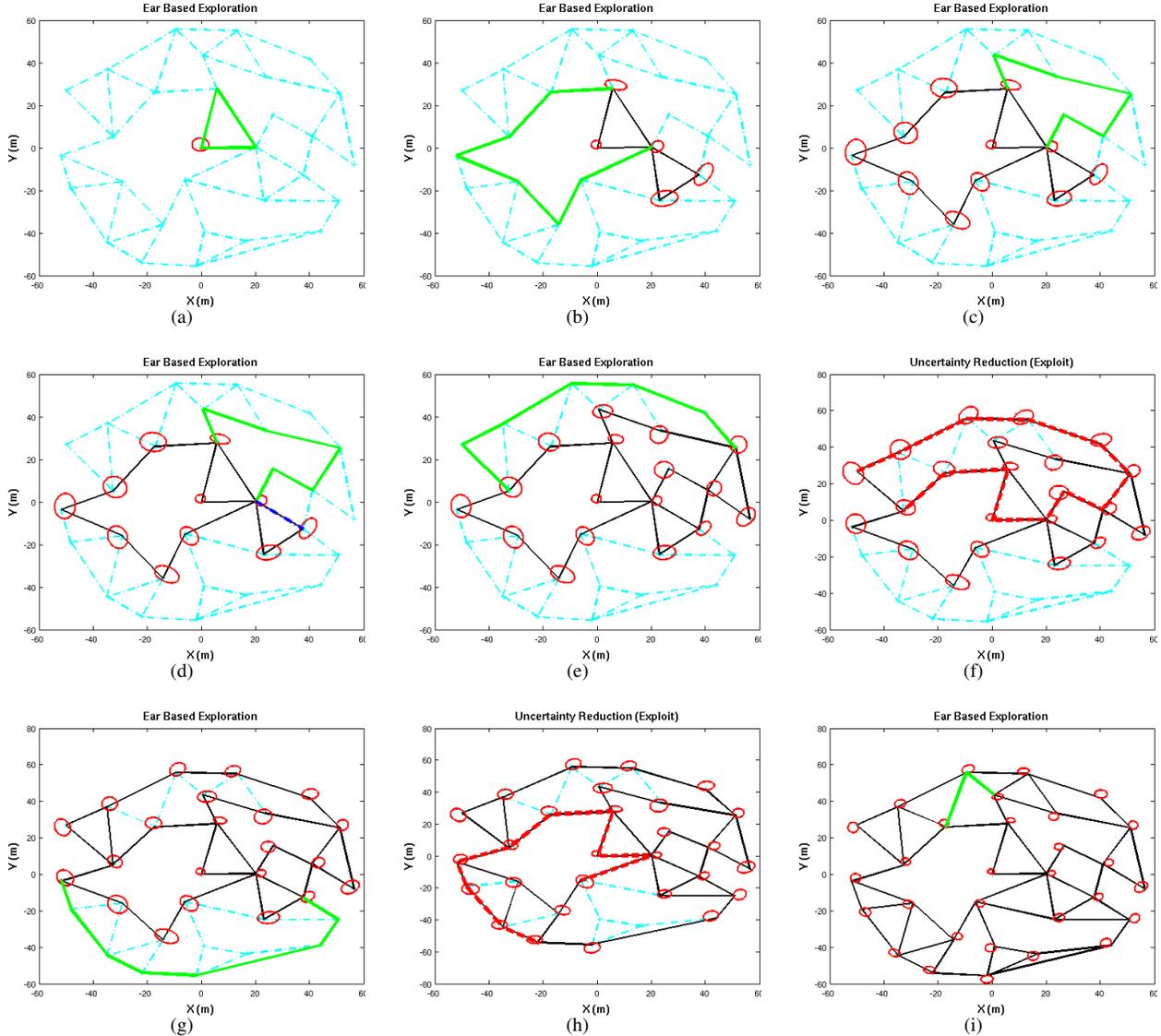


Fig. 6. Exploring a 30 vertices, 51 edges graph: (a) First face (ear) mapped; (b) third ear mapped; (c) continuing the exploration (fourth ear); (d) goal node has no more frontier edges, travelling to the closest frontier node; (e) exploring part of the outside face until encountering a mapped vertex; (f) reducing uncertainty by travelling through the known graph; (g) exploring the remaining part of the outside face; (h) most of the graph is explored, reducing uncertainty by travelling through the known graph (exploit); (i) the whole environment is explored.

### B. Ear based exploration

This section presents an illustrative example of the ear-based exploration proposed in this paper. In Figure 6a the beginning of the exploration can be seen; the world is a graph (dash-dotted thin lines)  $G = (V, E) : |V| = 30, |E| = 51$ . The

robot started at the origin and explore one face, by traversing an ear forming a triangle. After two steps the robot explores a bigger face, outlined by a seven-edge ear; see Fig. 6b. The exploration continues until in Fig. 6d the robot terminated at a vertex with no more frontier edges; the closest frontier

edge was selected, and the robot travelled there (dash-dotted line).

It is worth noting that the outside part of the graph represents also one face, in Fig. 6e the robot followed an ear that marks the outside of the graph until it arrived at a known vertex, at which point the ear-exploration halted. With probability  $p = 0.5 * \frac{|V_{explored}|}{|V_{total}|}$  the robot selects to exploit the known map by relocalizing; see Fig. 6f dashed red line. The uncertainty reduction procedure selects the target node based on a combination of distance and uncertainty of that node, in this case the goal node was an immediate neighbour. The uncertainty reduction A\*-planning algorithm [4] though selected the best route that minimized the overall map uncertainty.

Exploration continued in Fig. 6g with the robot travelling through the other half of the perimeter of the graph. In Fig. 6h the robot traversed through the graph in order to reduce the uncertainty; please note that the resulting path passed through the first node thus further reducing the uncertainty. Finally, the robot completed the exploration by traversing through every edge and visiting every vertex several times; see Fig. 6i.

### C. Uncertainty versus Distance evaluation

Uncertainty and distance measurements are presented here for graphs of different sizes and also for different edge densities. In all the plots in Figure 7 the number of vertices varied from ten to eighty, in increments of ten; for a specific number of vertices ten random graphs were constructed. The plots show average distance travelled (averaged over the ten random graphs each time) and average uncertainty in the form of the square root of the trace of the map covariance:

$$E = \sqrt{\text{trace}(P)} \quad (5)$$

The graph density was also varied as a function of the number of vertices. Figure 7a,b present results from very sparse graphs where the number of edges was slightly above the number of vertices  $|E| = 1.1 * |V|$ . The next plots present results from two times the number of vertices, see Fig. 7c,d; and finally for the full triangulation  $O(3 * |V| - 3 - k)$  order of edges where  $k$  is the number of vertices on the convex hull; see Fig. 7e,f.

For every random graph two basic exploration strategies were executed, together with the ear-based exploration strategy run while varying the exploitation threshold, over three different values. The simplest strategy is to always select a random frontier node, and then plan an uncertainty reduction path to it. This strategy, quite often, sends the robot across the graph and the A\* algorithm ensures that low uncertainty nodes are visited, as a result in all cases this strategy produced the most accurate maps but at a high cost on the distance travelled, more than twice the distance travelled than the other strategies.

The second strategy is to go to the closest frontier node (including the current node) and then take a random frontier edge. This greedy algorithm produced the shortest paths, but

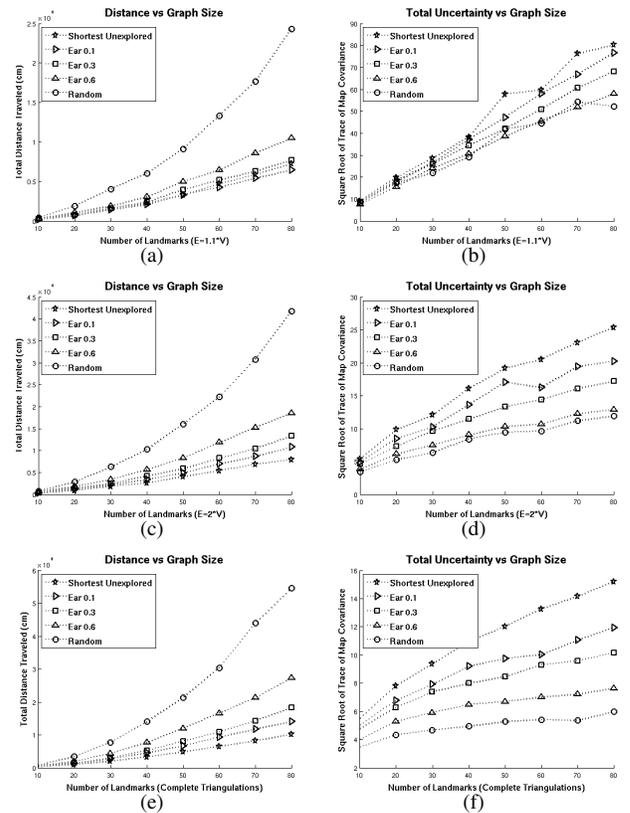


Fig. 7. Ten trial averages of distance travelled (a,c,e) and map uncertainty (b,d,f) for different number of nodes, different number of edges and for different exploration strategies.

with the highest uncertainty. The three ear-based exploration strategies differ in the frequency they perform uncertainty reduction traversals. In all cases exploration was favoured at the early stages and exploitation at the later part. As noted earlier, with probability  $1 - p_{explore}$  at the end of each ear-exploration, an uncertainty reducing path is selected. The three strategies use  $[0.1, 0.3, 0.6] * \frac{|V_{explored}|}{|V_{total}|}$  probabilities. As expected the more often the robot performed uncertainty reducing traversals the lower the final uncertainty but the higher the overall distance travelled. The severity of the distance penalty varied though with the density of the graph. In very sparse environment with not many loops, the uncertainty reduction produced low uncertainty with half the distance travelled. Even better were the results for medium sparsity graphs, more similar to indoor environments; see Fig. 7c,d.

Figure 8 presents similar results from single trials for much bigger graphs. The size varies from twenty five to one hundred seventy five vertices, for edge densities of  $1.3 * |V|$  and  $2.0 * |V|$ . As can be seen as the number of vertices increases, the efficiency of random strategy degrades rapidly, while the ear based exploration, even with frequent relocalizations is much more efficient. For 175 vertices the distance travelled of the ear-based exploration is one third of the random node strategy; while uncertainty stays at similar low levels.

## V. CONCLUSION

In this paper we present a new algorithm for exploring an unknown environment. The contributions of this work

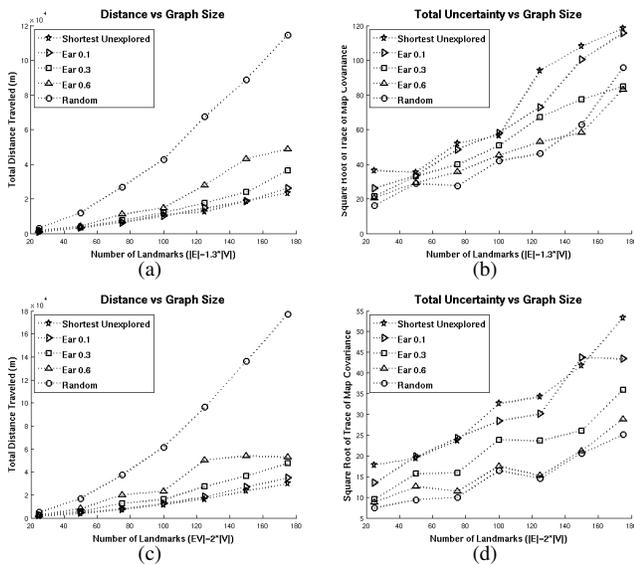


Fig. 8. Single trial results: Distance travelled (a,c) and map uncertainty (b,d) for different number of nodes, different number of edges and for different exploration strategies.

are twofold: first a systematic way of addressing the problem of efficient exploration and uncertainty reduction was introduced drawing from the classical *exploration versus exploitation* paradigm; second a pure exploration strategy that explicitly forces loop closure during exploration was implemented.

Augmenting a standard frontier-based [3] or generalized Voronoi graph (GVG)-based [25] exploration strategy with the algorithm presented in this paper is in the immediate future plans. We are currently working on such an implementation using a laser equipped iCreate robot running robot operating system (ROS) <sup>3</sup>.

We presented distance travelled and uncertainty accumulated results from random graphs with varying edge densities, illustrating the effect the number of vertices and the edge density has on the frequency of map refinement. In addition, the scalability of the approach was tested by running the proposed algorithm on graphs of different sizes and of different densities.

#### ACKNOWLEDGEMENTS

The author is grateful to the Natural Sciences and Engineering Research Council of Canada (NSERC) for their generous support. This work was performed at the Mobile Robotics Lab of Professor Gregory Dudek. Many thanks to David Meger for stimulating discussions and for providing the early implementations of the EKF and the A\* algorithm.

#### REFERENCES

- [1] I. M. Rekleitis and G. Dudek, "Automated calibration of a camera sensor network," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Aug. 2-6 2005, pp. 401-406.
- [2] I. Rekleitis, D. Meger, and G. Dudek, "Simultaneous planning, localization, and mapping in a camera sensor network," *Robotics and Autonomous Systems*, vol. 54, no. 11, pp. 921-932, November 2006.

- [3] B. Yamauchi, "Frontier-based exploration using multiple robots," *Autonomous Agents*, 1998.
- [4] D. Meger, I. Rekleitis, and G. Dudek, "Heuristic search planning to reduce exploration uncertainty," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, Nice, France., 2008, pp. 3382 - 3399.
- [5] L. Kaelbling, M. Littman, and A. Moore, "Reinforcement learning: A survey," *J. of Artificial Intelligence Research*, pp. 237 - 285, 1996.
- [6] I. M. Rekleitis, V. Dujmović, and G. Dudek, "Efficient topological exploration," in *Proc. of Int. Conf. in Robotics and Automation*, Detroit, USA, May 1999, pp. 676-681.
- [7] 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, Nov. 1986.
- [8] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," *Autonomous Robot Vehicles*, pp. 167 - 193, 1990.
- [9] D. P. Meger, D. Marinakis, I. Rekleitis, and G. Dudek, "Inferring a probability distribution function for the pose of a sensor network using a mobile robot," in *IEEE Int. Conf. on Robotics and Automation*, Kobe, Japan, May 2009, pp. 756-762.
- [10] R. Sim and N. Roy, "Global a-optimal robot exploration in slam," in *Int. Conf. on Robotics and Automation*, 2005, pp. 661 - 666.
- [11] R. Martinez-Cantin, N. de Freitas, A. Doucet, and J. Castellanos, "Active policy learning for robot planning and exploration under uncertainty," in *In Proc. of Robotics: Science and Systems (RSS)*, 2007.
- [12] T. Kollar and N. Roy, "Using reinforcement learning to improve exploration trajectories for error minimization," in *In Proc. of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, Orlando, 2006.
- [13] D. Fox, W. Burgard, and S. Thrun, "Active markov localization for mobile robots," *Robotics and Autonomous Systems*, 1998.
- [14] C. Stachniss, D. Haehnel, and W. Burgard, "Exploration with active loop-closing for fastslam," *Int. Conf. on Intelligent Robots and Systems*, 2004.
- [15] A. Makarenko, S. Williams, F. Bourgault, and H. Durrant-Whyte, "An experiment in integrated exploration," *Int. Conf. on Intelligent Robots and Systems*, 2002.
- [16] S. Hang, N. Kwok, G. Dissanayake, Q. Ha, and G. Fang, "Multi-step look-ahead trajectory planning in slam: Possibility and necessity," *Int. Conf. on Robotics and Automation*, 2005.
- [17] N. Srinivas, A. Krause, S. Kakade, and M. Seeger, "Gaussian process optimization in the bandit setting: No regret and experimental design," in *Int. Conf. on Machine Learning*, 2010.
- [18] A. Stentz, "Optimal and efficient path planning for unknown and dynamic environments," *Int. J. of Robotics and Automation*, vol. 10, no. 3, 1995.
- [19] M. Batalin and G. S. Sukhatme, "Coverage, exploration and deployment by a mobile robot and communication network," *Telecommunication Systems J., Special Issue on Wireless Sensor Networks*, vol. 26, no. 2, pp. 181-196, 2004.
- [20] P. Corke, R. Peterson, and D. Rus, "Localization and navigation assisted by cooperating networked sensors and robots," *Int. J. of Robotics Research*, vol. 24, no. 9, 2005.
- [21] C. Stachniss, D. Haehnel, W. Burgard, and G. Grisetti, "On actively closing loops in grid-based fastslam," *ADVANCED ROBOTICS*, vol. 19, p. 2005, 2005.
- [22] P. Fraigniaud, D. Ilcinkas, G. Peer, A. Pelc, and D. Peleg, "Graph exploration by a finite automaton," *Theoretical Computer Science*, vol. 345, no. 2-3, pp. 331-344, 2005.
- [23] D. Meger, "Planning, localization, and mapping for a mobile robot in a camera network," *Master of Science Thesis - supervisors Ioannis Rekleitis and Gregory Dudek*, 2007.
- [24] G. Dudek and M. Jenkin, *Computational Principles of Mobile Robotics*. Cambridge University Press, 2010.
- [25] H. Choset and J. Burdick, "Sensor-based exploration: The hierarchical generalized voronoi graph," *The Int. J. of Robotics Research*, vol. 19, no. 2, pp. 96-125, 2000.
- [26] E. U. Acar, H. Choset, A. A. Rizzi, P. N. Atkar, and D. Hull, "Morse decompositions for coverage tasks," *The Int. Journal of Robotics Research*, vol. 21, no. 4, pp. 331-344, April 2002.
- [27] I. Rekleitis, G. Dudek, and E. Milios, "Experiments in free-space triangulation using cooperative localization," in *Int. Conf. on Intelligent Robots and Systems*, 2003, pp. 1777-1782.
- [28] F. P. Preparata and M. I. Shamos, *Computational Geometry: An Introduction*. New York, NY: Springer-Verlag, 1985.

<sup>3</sup><http://www.ros.org>