Single Robot Exploration: Simultaneous Localization and Uncertainty Reduction on Maps (SLURM)

Ioannis Rekleitis
School of Computer Science
McGill University
Montreal, Canada
Email: yiannis@cim.mcgill.ca

***Abstract*—During exploration of an unknown environment by a single robot, the robot is driven by two conflicting goals: to explore as fast as possible; and to produce the most accurate map. While fast exploration necessitates minimizing traversal of already mapped territory, accurate mapping requires that the robot passes over previously explored areas to reduce the localization and map uncertainty. This problem has been labeled as exploration versus exploitation. In this paper the problem of mapping a camera sensor network by a mobile robot has been used to demonstrate the effect that different exploration strategies have on uncertainty and speed of exploration. Simulation results using a realistic noise model are presented for different environments and for different strategies.**

***Keywords*-Mobile Robotics; Sensor Networks; Localization; Uncertainty Reduction.**

I. Introduction

In this paper the problem of exploring an unknown environment and at the same time maintaining the quality of the resulting map is addressed. In many applications robot(s) enter an unknown environment and have to explore it while at the same time constructing a map. One key question is how much time they should spend improving the quality of the map versus exploring new territory. We use the paradigm of a mobile robot navigating through an environment equipped with a camera sensor network; see Fig. 1. The robot's goal is to produce a map with the locations of all the cameras, and to maintain its own pose estimate. Such a scenario is quite common in the case where a service robot (vacuuming, patrolling, etc.) operates inside a building equipped with security cameras. The pose of each camera is unknown; the robot is carrying a target that can be detected when it enters a cameras field of view. In addition, by moving in-front of the camera, the robot is able to perform autonomous calibration [1] and also recover the six degree of freedom coordinate transformation between the target and the camera [2], thus localizing the camera in the robot's frame of reference. This formulation of the problem eliminates the data association problem which, though relevant, it is not at the centre of the decision between exploration and relocalization. In this scenario, the robot's motion through the network facilitates localization by explicitly transferring pose information between sensor locations. By maintaining an ongoing estimate of the robot's location, the position of any sensor that it interacts with can be probabilistically estimated, (and updated), given the appropriate motion and measurement models. Planning trajectories in the face of conflicting goals such as efficiency and accuracy, in combination with the high dimensional uncertainty estimates of the underlying SLAM solution results in a challenging problem. 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 between these areas; the cameras have no overlapping field of views.

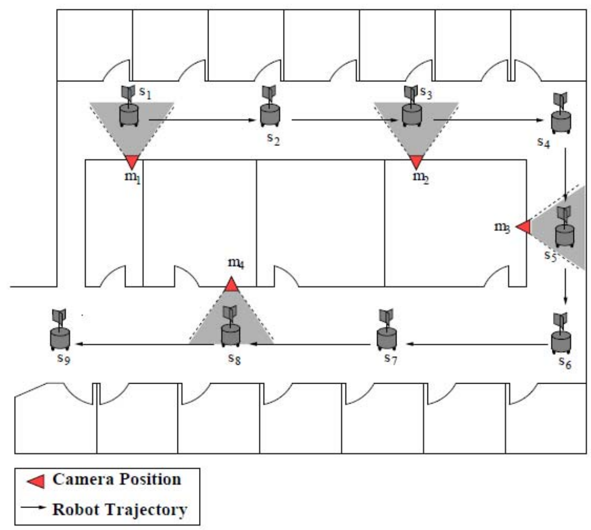

Figure 1. The experimental setup used throughout this paper. The robot carries a calibration target which can be easily detected in images taken by the cameras in the network.

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 represented by a graph $G_t = [V_t, E_t]$ where each vertex $v_i \in V_t$ represents the centre of

978-0-7695-4683-4/12 $26.00 © 2012 IEEE
DOI 10.1109/CRV.2012.36

a camera's field of view where the robot would visit. Every edge $e_{ij} \in E_t$ represents a path between the vertices v_i and v_j observed by camera C_i and C_j respectively. In addition we consider an additional set of directed edges E_t^f that have a starting vertex but no ending vertex termed frontier edges, these are unexplored edges. The exploration process can be divided into two parts: first, "*where to go next?*" and second, "*how to go there?*". In [1], [4] an A* based algorithm was proposed that addressed the question of "*how to go there?*" by returning a path through G_t. It was shown [5] that by allowing uncertainty to influence the cost function of A*, big improvements in the uncertainty of the resulting map were realized even when uncertainty had a small influence. In this paper we build on top of the A* path-planner to answer the question "*where to go next?*". In general, there are three options when selecting the next action:

1) **Pure Exploration**: Follow the closest unexplored edge $e \in E_t^f$.
2) **Uncertainty Driven Exploration**: Select a frontier vertex based on the vertex's uncertainty; then follow a random unexplored edge originating from that vertex.
3) **Uncertainty Reducing Path Planning**: Pay a visit to a vertex with high uncertainty in order to reduce its uncertainty.

The above choices, resemble the strategies that provide an approximate solution to the multi-armed bandit problem [6]. In particular selecting to explore, and then exploit falls under the general term of "*Epsilon-first strategy*", while electing with probability $p_{explore}$ to explore a random vertex is the "*Epsilon-greedy strategy*". In the localization problem discussed in this paper, the reward stems from reducing the map uncertainty, while exploring new edges on the map can provide shorter paths and reduce uncertainty by closing the loop.

This paper is structured as following. The next section discusses related work. In Section III the different exploration strategies are presented. 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

The problem of localizing a camera sensor network using a mobile robot is similar to Simultaneous Localization and Mapping (SLAM) since both problems require estimating the pose of the robot and the positions of environment features (landmarks or camera nodes) from acquired sensor data. Hence, numerous similar estimation approaches are appropriate. In this paper, the extended Kalman filter (EKF) as described in [7] for SLAM is adapted for camera network localization. The EKF computes the mean μ and covariance P for each map quantity. Many other solutions are possible, but the EKF is used here for computational simplicity and ease of analysis.

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. [8]–[11]. Many approaches have attempted to reduce the entropy in the map estimates [12]–[14], which is the measure of the uncertainty in a distribution and is defined as:

$$H(p(\xi)) \equiv - \int p(\xi) \log(p(\xi)) d\xi \qquad (1)$$

For the Gaussian distributions used by an EKF representation of the environment, entropy can be expressed in closed form. Sim and Roy [8] discuss two different measures from information theory for which either the trace or the determinant of the covariance matrix provides the final measure for entropy.

Early work proposed a single-step, greedy choice of the action which maximally minimizes the entropy because optimal planning of multi-step paths requires computational cost exponential in the path length. Recently, Sim and Roy [8] have proposed pruning loops during 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, [9] has considered a simulation-based approach which has the potential to generate multi-step paths at the cost of significant computation.

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, in the form of the D^* algorithm proposed by Stentz [15]. 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 earlier, accuracy and efficiency are conflicting goals during exploration. In order to produce paths that compromise between the goals, distance and uncertainty have to be combined into a single 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.* [13] have previously proposed a weighted linear combination of distance and uncertainty for path p:

$$C(p) = \omega_d \; length(p) + \omega_u \; trace(P(p)) \quad (2)$$

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 ω_d and ω_u represents the compromise between distance travelled and mapping uncertainty or *accuracy* versus *efficiency*. We would like to produce a flexible method based on varying the one intrinsic parameter, so we normalize the contribution of each quantity by a rough estimate of its maximum possible value. Once each quantity has been normalized, a single free parameter α 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:

$$\omega_d = \frac{\alpha}{maxDistance} \quad , \quad \omega_u = \frac{1-\alpha}{maxUncertainty}$$

By setting α 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 [5] the effect of varying α 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 [16], [17].

III. Exploration versus Exploitation

As discussed in [5], the localization and map uncertainty reduction process involves inferring the positions of each sensor node m_i, which is part of the map of the sensors $m^n = [m_1 \; m_2 \; ... \; m_n]$, based on measurements obtained by a robot; see Fig. 1. These positions can only be measured relative to the position of the robot at a given time, s_t, which is the most recent component of the robot's path, $s^t = [s_1 \;\; s_2 \;\; ... \;\; s_t]$, and so both quantities must be estimated simultaneously. The measurements available are the position of a sensor node relative to the robot at time t, denoted z_t and the position of the robot at time t relative to its position at time $t-1$, denoted u_t. Different state estimation algorithms have been used in the past, Rao-Blackwellized Particle Filters (RBPF) and Markov chain Monte Carlo (MCMC) [18], as well the classical extended Kalman filter [19], with varying degrees of accuracy and efficiency; MCMC being the slower but most accurate. In this work the EKF filter is used both for efficiency's sake and for the clear description of uncertainty resulting from the map's covariance matrix. In the EKF formulation the state vector contains the pose of the robot and the map with the poses of the cameras, it has the form of:

$$\mathbf{x}_t = [\mathbf{x}^{robot}, \mathbf{x}_1^C, \ldots, \mathbf{x}_N^C]^T \quad (3)$$

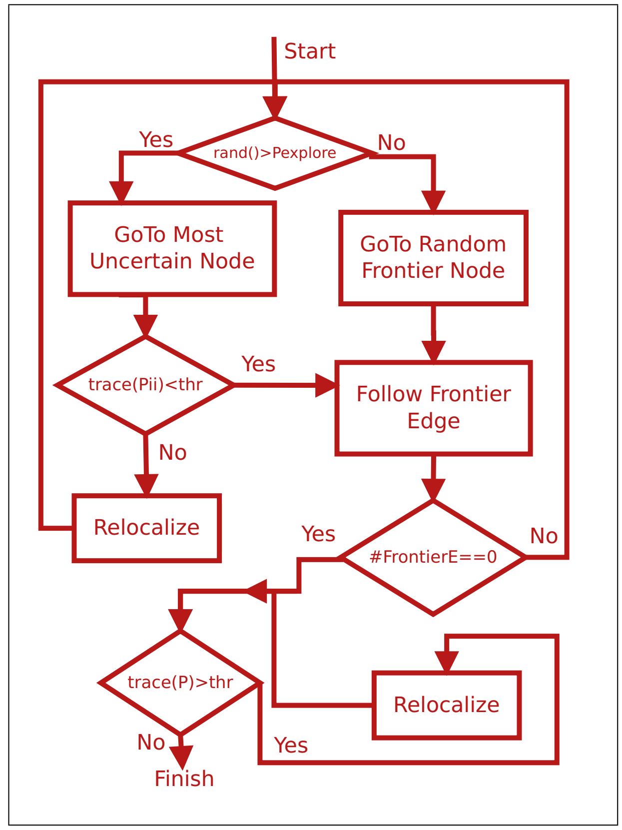

Figure 2. The flowchart of the Simultaneous Localization and Uncertainty Reduction on Maps (SLURM) algorithm.

where $\mathbf{x}^{robot} = [x_t^r, y_t^r, \theta_t^r]$ is the position and orientation of the robot in 2D, and $\mathbf{x}_i^C = [x_t^i, y_t^i, z_t^i, \theta_t^i, \phi_t^i, \psi_t^i]^T$ is the pose of the camera in 3D where x_t^i, y_t^i, z_t^i represents the position of the camera and $\theta_t^i, \phi_t^i, \psi_t^i$ are the roll, pitch, yaw Euler angles respectively.

For the rest of the paper, the measure of uncertainty for node i is calculated by the map uncertainty in the 3D position of the i^{th} camera. This uncertainty is encoded in the state covariance matrix $\mathbf{P}$ and is represented by the trace of the covariance matrix

$$trace(\mathbf{P}^{C_i}) = \sum_{j=3+6*(i-1)+1;}^{3+6*(i-1)+3} P_{jj} \quad (4)$$

Please note, the trace only uses the position uncertainty and not the attitude in order to ensure unit compatibility. As mentioned earlier during exploration there are two major planning processes running in turns. First select what to do next, that is where to go next (select goal); and second, plan a path through known space to reach the selected goal. For planning through known space, the A* algorithm presented in [5] is used with different values of α, which determines the trade-off between accuracy and efficiency.

Figure 2 presents an outline of the proposed algorithm. At the beginning with probability $p_{explore}$ exploration of a random frontier edge is selected. Otherwise, the frontier node with the most uncertainty is selected. The robot uses the A* path planning method to reach there favouring a path that minimizes the cumulative uncertainty. If the uncertainty on that node on arrival is more than an acceptable threshold, then, instead of exploring, the robot plans a path to another high uncertainty node, thus reducing the total map uncertainty. The relocalization process continues as long as the frontier node with the most uncertainty keeps having high uncertainty. The random exploration decision at the beginning of the loop ensures that, with probability $p_{explore}$, new edges are explored, and the robot is not stuck on a local minima of the search space. Finally, when all edges have been explored the robot goes through a relocalization process to ensure that uncertainty is kept at acceptable levels.

IV. Experimental Results

A. Environment Representations

Graph based representations are quite common in robotics, e.g., visibility graphs [20]; Spatial Semantic Hierarchy [21], [22]; generalized Voronoi graphs [23]; Reeb graph [24]; triangulations [25]; and networks of reusable paths [26]. In many cases random graphs have been used to validate the proposed algorithm and also to test scalability and robustness. In the localization problem discussed in this paper different types of random graphs have been used in order to explore the trade-offs between exploration and exploitation. In all cases we start with planar embedding of a graph such that the graph representation used is compatible with the physical properties of the robots used. For example, there is no point trying to simulate the motions of an one meter square robot inside an environment of radius one. First we create the vertices, two different approaches can have been explored: a set of random points, Fig. 3a, or points on a regular grid. Next, an initial graph is generated either as a triangulation or ,when the vertices are arranged in a regular grid as a four or eight connected complete grid. Finally, edges are randomly removed while ensuring that the graph stays connected. In addition, obstacles can be utilized that restrict the vertex and edge placement; see Fig. 3b. When the starting graph is a grid then the resulting graph is termed the “Montreal Graph” [27]; see Fig. 3c. The results presented here are based on triangulation graphs (with edges removed). Preliminary results on the other graphs indicated similar performance. The noise statistics for the odometry and the camera estimates used in the simulations were experimentally derived from [5].

B. Simultaneous Localization and Uncertainty Reduction on Maps

Figure 4 presents an illustrative example of the proposed algorithm. The environment consists of 15 cameras (nodes) which are connected by 22 edges. At the beginning the robot is performing exploration while always selecting a frontier node of high uncertainty, after a sufficiently large number of nodes are detected, the robot selectively performs relocalization when a node’s uncertainty rises above a certain threshold. Please note that the uncertainty ellipses are plotted where the cameras are located, which are approximately a meter off the robot’s position. The robot started at [0,0] then moved to the next node [15,12] (increased uncertainty) and then planed a path (green dashed line) back to the starting node from where moved and explored a new edge (solid cyan line). In Fig. 4b the robot followed a new edge and closed the first cycle (uncertainty reduced). The robot continued exploring new edges and mapped 8 nodes; see Fig. 4c. It is worth noting that when the robot travels to the next node A* is called with $\alpha = 0.01$, see eq. 3, which means the cost is mainly influenced by the uncertainty build up. This is apparent in the path taken which passes through the most accurate nodes, even though it is longer. The uncertainty grew and the robot reverted to an uncertainty reduction procedure by traversing repeatedly through the known graph targeting nodes with high uncertainty; see Fig. 4d. The A* is called with $\alpha = 0.01$ giving priority to uncertainty reductions, as can be seen from the resulting path. The majority of the environment is explored, there are only three remaining unexplored edges, again the A* planner guided the robot (dashed green line) in a path that minimized uncertainty; see Fig. 4e. Finally after all edges are explored the robot moves through the known map, reducing the uncertainty to a pre-specified value; see Fig. 4e.

C. Performing exploration with bounded Uncertainty

A major advantage of the proposed methodology is the ability to produce maps of bounded uncertainty by setting a specific threshold for the relocalization procedure. During simultaneous exploration and uncertainty reduction, when any landmark’s uncertainty rises above a prespecified threshold, the robot is guided through a pure relocalization procedure. Relocalization is achieved by moving to the nodes with the highest uncertainty via trajectories that improve the localization accuracy. As can be seen, even though the number of nodes increased the uncertainty was maintained almost constant, at a level similar or lower than the set threshold. There is a clear cost for the higher accuracy as can be seen in figure 5a, where maintaining the lowest uncertainty resulted in an ever-increasing distance.

D. Distance and Uncertainty Results

In this section uncertainty and distance measurements are plotted for different number of landmarks (vertices in the graph) and also for different density of edges. In all the plots in figure 6 the number of vertices varied from ten to hundred, 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

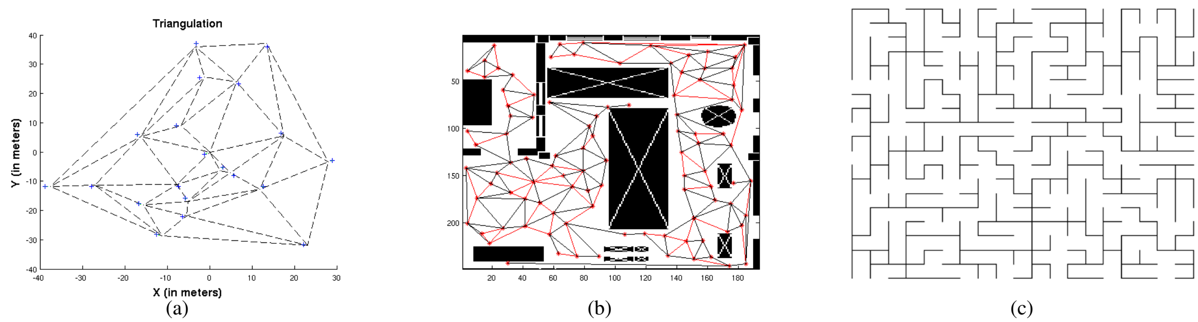

Figure 3. Different types of planar graphs: (a) Random dense graph based on the Delaunay triangulation; (b) sparser graph with obstacles; (c) Montreal graph.

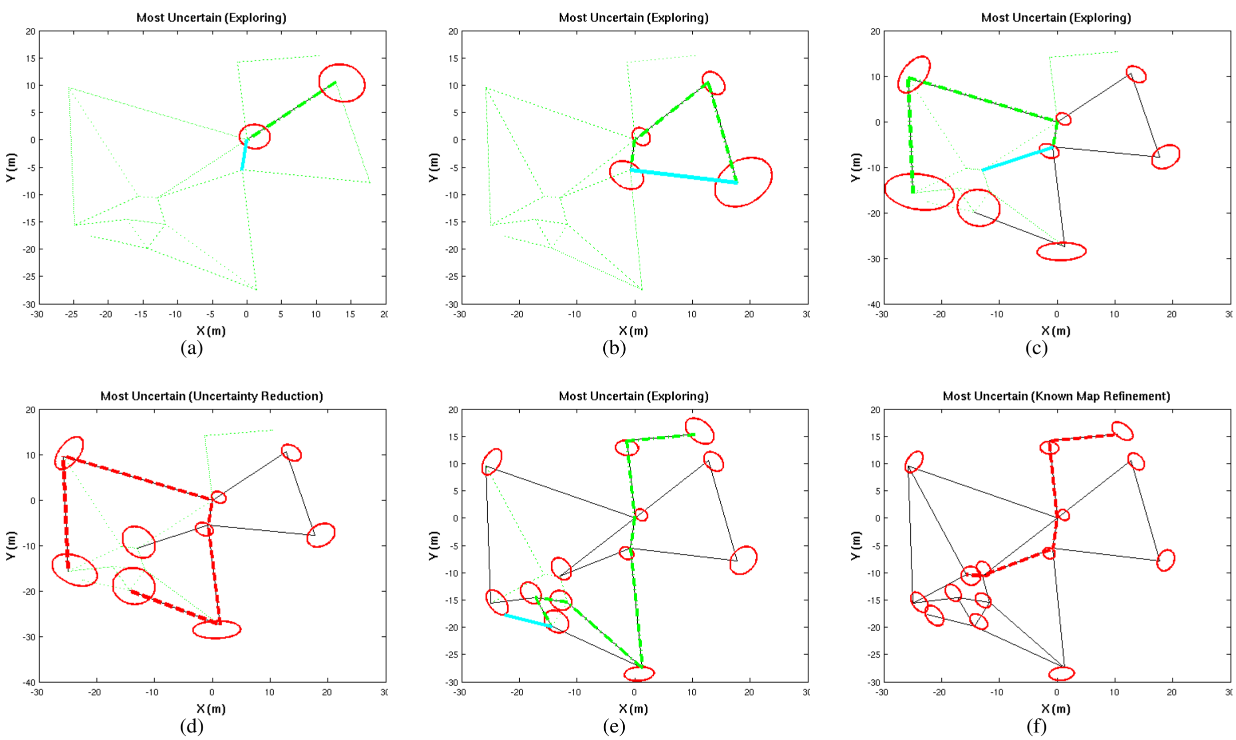

Figure 4. Exploring a 15 vertices, 22 edges graph: (a) Two nodes mapped, exploring the third edge; (b) first loop closure; (c) continuing the exploration; (d) reducing uncertainty by travelling through the known graph; (e) most of the graph is explored, some areas have high uncertainty; (f) reducing uncertainty by travelling through the completed map.

time) and average uncertainty in the form of the square root of the trace of the map covariance:

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

In addition the density of the graph was also varied as a percentage of the number of vertices. Figure 6a,b present results from very sparse graphs where the number of edges was equal to the number of vertices; this was one edge more than the minimum requirement of a spanning tree. The following plots present results from 1.3 times the number of vertices, see Fig. 6c,d; 2.1 times the number of vertices, see Fig. 6e,f; and finally a full triangulation $O(3N-3-k)$ order of edges where N is the number of vertices and k is the number of vertices on the convex hull; see Fig. 6g,h.

For every random graph four different exploration strategies were selected to run as a way for comparing. The simplest strategy is to always select a random frontier node, and then plan an uncertainty reduction path to it. The second is to go to the closest frontier node (including the current node) and then take a random frontier edge. The third is to select the frontier node with the minimum uncertainty and then follow a random frontier edge from it. Finally, the fourth is to select the maximum uncertainty node and

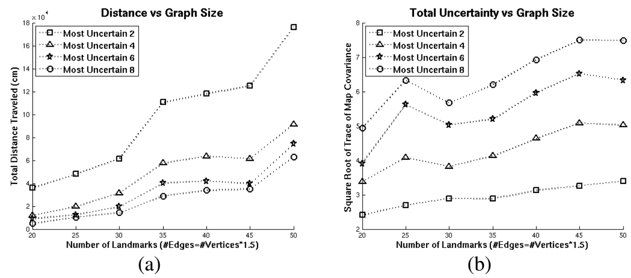

Figure 5. Distance travelled (a) and Map uncertainty (b) for different number of nodes $\{20,25,\ldots,50\}$ using the bounded uncertainty strategy for different uncertainty thresholds ($\{2,4,\ldots,8\}$ meters). The number of edges is 1.5 times the number of vertices, e.g., $G_{20} = [V, E] : |V| = 20, |E| = 30$.

follow the algorithm described in section III. As expected the fourth algorithm gave the most accurate results albeit at ever increasing costs. To a large extend the majority of the time was spend relocalizing to maintain the uncertainty at the predefined threshold, thus it is not a fair comparison with the rest of the algorithms as they did not perform any relocalization explicitly. The rationale behind selecting as a departing point to the unknown the node with the minimum uncertainty was to start each time the exploration from the most accurate position. The surprising result was that selecting a random frontier node was similar in performance with selecting the node with minimum uncertainty. In practise, most of the time the random node resulted in a good relocalization path and improved the overall accuracy of the map.

V. Conclusion

In this paper a systematic way of addressing the problem of localization and uncertainty reduction was presented. By drawing from the multi-armed bandit problem solutions, we proposed an algorithm for systematically exploring an unknown environment while at the same time maintaining map uncertainty at the desired levels.

We presented results from random graphs with varying densities, illustrating the effect 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.

Currently, the proposed method is being tested on a variety of random graphs. Of particular interest are graphs embedded in a physical environment with obstacles because the choice of where to explore next can lead away from low uncertainty areas. Furthermore we are extending this work on the multi-robot domain by examining the benefits of distinct roles (explorer versus localizer).

The problem of maintaining bounded uncertainty while exploring an unknown environment is crucial for enhancing the autonomy capabilities of robotic agents. Though the problem is challenging the analysis presented in this paper will provide significant guidelines for future research.

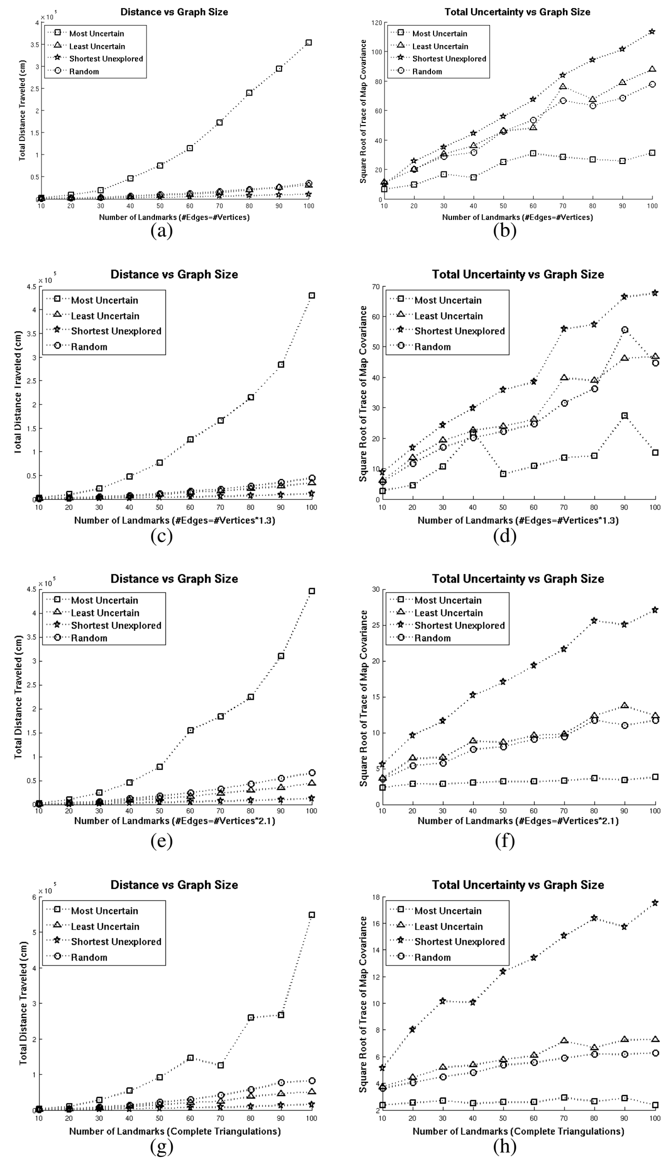

Figure 6. Distance travelled (a,c,e,g) and Map uncertainty (b,d,f,h) for different number of nodes, different number of edges and for different exploration strategies.

Acknowledgment

The author would like to thank the Natural Sciences and Engineering Research Council of Canada (NSERC) for their generous support. This work was done at the Mobile Robotics Lab of Professor Dudek. Many thanks to David Meger for stimulating discussions and for providing the early implementations of the EKF and A* algorithms.

References

[1] 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.

[2] I. M. Rekletis and G. Dudek, "Automated calibration of a camera sensor network," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton Alberta, Canada, Aug. 2-6 2005, pp. 401–406.

[3] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Monterey, CA, July 1997, pp. 146–151.

[4] 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.

[5] D. Meger, I. Rekleitis, and G. Dudek, "Heuristic search planning to reduce exploration uncertainty," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS),*, Nice, France,, 2008, pp. 3382 – 3399.

[6] L. Kaelbling, M. Littman, and A. Moore, "Reinforcement learning: A survey," *Journal of Artificial Intelligence Research*, pp. 237 – 285, 1996.

[7] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," *Autonomous Robot Vehicles*, pp. 167 – 193, 1990.

[8] R. Sim and N. Roy, "Global A-optimal robot exploration in SLAM," in *International Conference on Robotics and Automation*, 2005, pp. 661 – 666.

[9] R. Martinez-Cantin, N. de Freitas, A. Doucet, and J. Castellanos, "Active policy learning for robot planning and exploration under uncertainty," in *In Proceedings of Robotics: Science and Systems (RSS)*, 2007.

[10] T. Kollar and N. Roy, "Using reinforcement learning to improve exploration trajectories for error minimization," in *In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Orlando, 2006.

[11] D. Fox, W. Burgard, and S. Thrun, "Active markov localization for mobile robots," *Robotics and Autonomous Systems*, 1998.

[12] C. Stachniss, D. Haehnel, and W. Burgard, "Exploration with active loop-closing for FastSLAM," *International Conference on Intelligent Robots and Systems*, 2004.

[13] A. Makarenko, S. Williams, F. Bourgault, and H. Durrant-Whyte, "An experiment in integrated exploration," *International Conference on Intelligent Robots and Systems*, 2002.

[14] S. Hang, N. Kwok, G. Dissanayake, Q. Ha, and G. Fang, "Multi-step look-ahead trajectory planning in SLAM: Possibility and necessity," *International Conference on Robotics and Automation*, 2005.

[15] A. Stentz, "Optimal and efficient path planning for unknown and dynamic environments," *International Journal of Robotics and Automation*, vol. 10, no. 3, 1995.

[16] M. Batalin and G. S. Sukhatme, "Coverage, exploration and deployment by a mobile robot and communication network," *Telecommunication Systems Journal, Special Issue on Wireless Sensor Networks*, vol. 26, no. 2, pp. 181–196, 2004.

[17] P. Corke, R. Peterson, and D. Rus, "Localization and navigation assisted by cooperating networked sensors and robots," *International Journal of Robotics Research*, vol. 24, no. 9, 2005.

[18] 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 International Conference on Robotics and Automation*, Kobe, Japan, May 2009, pp. 756–762.

[19] D. Meger, I. Rekleitis, and G. Dudek, "Autonomous mobile robot mapping of a camera sensor network," in *The 8th International Symposium on Distributed Autonomous Robotic Systems (DARS)*, Minneapolis, Minnesota, July 2006, pp. 155–164.

[20] G. Dudek and M. Jenkin, *Computational Principles of Mobile Robotics*. Cambridge University Press, 2010.

[21] B. J. Kuipers and Y.-T. Byun, "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representation," *Journal of Robotics and Autonomous Systems*, vol. 8, p. 4763, 1991.

[22] B. Kuipers, "An intellectual history of the spatial semantic hierarchy," in *Robot and Cognitive Approaches to Spatial Mapping*, M. Jefferies and A. W.-K. Yeap, Eds. Springer Verlag, 2008.

[23] H. Choset and J. Burdick, "Sensor-based exploration: The hierarchical generalized Voronoi graph," *The International Journal of Robotics Research*, vol. 19, no. 2, pp. 96–125, 2000.

[24] E. U. Acar, H. Choset, A. A. Rizzi, P. N. Atkar, and D. Hull, "Morse decompositions for coverage tasks," *The International Jounal of Robotics Research*, vol. 21, no. 4, pp. 331–344, April 2002.

[25] I. Rekleitis, G. Dudek, and E. Milios, "Experiments in free-space triangulation using cooperative localization," in *IEEE/RSJ/GI International Conference on Intelligent Robots and Systems*, Las Vegas, NV, Oct. 27-31 2003, pp. 1777–1782.

[26] K. Y. K. Leung, T. D. Barfoot, and L. H. H. T., "Distributed and decentralized cooperative simultaneous localization and mapping for dynamic and sparse robot networks," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Shanghai, China, 2011, p. 38413847.

[27] G. Dudek, M. Jenkin, E. Milios, and D. Wilkes, "Map validation and robot self-location in a graph-like world," *Robotics and autonomous systems*, vol. 22, no. 2, pp. 159–178, 1997.