Mobile Robot Exploration with Potential Information Fields

Joan Vallvé and Juan Andrade-Cetto

Abstract— **We present a mobile robot exploration strategy that computes trajectories that minimize both path and map entropies. The method evaluates joint entropy reduction and computes a potential field in robot configuration space using these joint entropy reduction estimates. The exploration trajectory is computed descending on the gradient of these field. The technique uses Pose SLAM as its estimation backbone. Very efficient kernel convolution mechanisms are used to evaluate entropy reduction for each sensor ray, and for each possible robot orientation, taking frontiers and obstacles into account. In the end, the computation of this field on the entire C-space is shown to be very efficient computationally. The approach is tested in simulations in a common publicly available dataset comparing favorably both in quality of estimates and execution time against another entropy reduction strategy that uses occupancy maps.**

I. Introduction

We consider the problem of autonomous mobile robot exploration. The problem is posed as that of reducing both localization and map uncertainties. Exploration strategies driven by uncertainty reduction date back to the seminal work of Whaite [1] for the acquisition of 3-D models of objects from range data. Within the context of SLAM, it is the work of Feder et al. [2], who first proposed a metric to evaluate uncertainty reduction as the sum of the independent robot and landmark entropies with an exploration horizon of one step to autonomously produce occupancy maps. Bourgault et al. [3] alternatively proposed a utility function for exploration that trades off the potential reduction of vehicle localization uncertainty, measured as entropy over a feature-based map, and the information gained over an occupancy grid. In contrast to these approaches, which independently consider the reduction of vehicle and map entropies, Vidal et al., [4] tackled the issue of joint robot and map entropy reduction, taking into account robot and map cross correlations for the Visual SLAM EKF case.

Action selection in SLAM can also be approached as an optimization problem using receding horizon strategies [5], [6], [7]. Multi-step look ahead exploration in the context of SLAM makes sense only for situations in which the concatenation of prior estimates without measurement evidence remain accurate for large motion sequences. For highly unstructured scenarios and poor odometry models, this is hardly the case. In this work we compute trajectories descending on the gradient of a potential field computed from both path and map entropy reduction estimates.

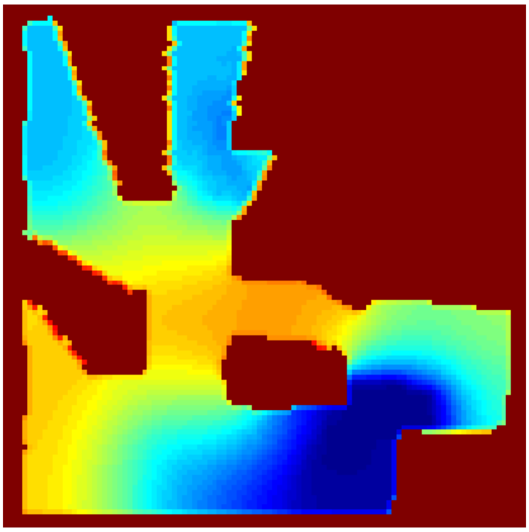


Fig. 1. Potential field values for one C-space orientation layer. The blue regions indicate competing exploratory and loop closure candidate configurations at that robot orientation.

One technique that tackles the problem of exploration in SLAM as a one step look ahead entropy minimization problem makes use of Rao-Blackwellized particle filters [8]. The technique extends the classical frontier-based exploration method [9] to the full SLAM case. When using particle filters for exploration, only a very narrow action space can be evaluated due to the complexity in computing the expected information gain. The main bottleneck is the generation of the expected measurements that each action sequence would produce, which is generated by a ray-casting operation in the map of each particle. In contrast, measurement predictions in a Pose SLAM implementation, such as ours, can be computed much faster, having only one map posterior per action to evaluate, instead of the many that a particle filter requires. Moreover, in [8], the cost of choosing a given action is subtracted from the expected information gain with a user selected weighting factor. In our approach, the cost of long action sequences is taken into consideration during the selection of goal candidates, using the same information metrics that help us keep the robot localized during path execution.

In [10] our group proposed a solution to the exploration problem that maximizes information gain in both the map and path estimates. The method evaluates both exploratory and loop closure candidate trajectories, computing entropy

This work has been supported by the Spanish Ministry of Economy and Competitiveness under Project DPI-2011-27510 and by the EU Project ARCAS FP7-287617.

The authors are with the Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens Artigas 4-6, 08028 Barcelona, Spain. {jvallve,cetto}@iri.upc.edu.

reduction estimates from a coarse resolution realization of occupancy maps. The final trajectory is computed using A* in the occupancy grid, just as [11] does so over an initial reference trajectory. The computational bottleneck of [10] was in the estimation of the occupancy map. In this paper we present an alternative method, in which we compute directly the global entropy reduction estimate for each possible robot configuration. The use of very efficient kernel convolutions allow us to compute this estimate very fast and without the need to reduce the grid resolution.

To find candidate exploration paths, the entropy reduction grid in C-space is transformed into a potential field, taking into account frontiers and obstacles. The path is obtained by gradient descent on this field. Potential methods have been previously used for exploration [12], [13], but different than our approach, these methods directly evaluate boundary conditions on deterministic maps of obstacles and frontiers, without taking uncertainty into account. Our method combines both the idea of gradient descent to a desired exploratory or loop closing location, and the minimization of joint map and path entropy.

II. Pose SLAM

The proposed exploration strategy uses Pose SLAM as its estimation backbone. In Pose SLAM [14], a probabilistic estimate of the robot pose history is maintained as a sparse graph. State transitions result from the composition of motion commands u_k to previous poses,

$$x_k = f(x_{k-1}, u_k) = x_{k-1} \oplus u_k \, , \tag{1}$$

and the registration of sensory data also produces relative motion constraints, but now between non-consecutive poses,

$$z_{ik} = h(x_i, x_k) = \ominus x_i \oplus x_k . \tag{2}$$

Graph links indicate geometric relative constraints between robot poses, and the density of the graph is rigorously controlled using information measures. In Pose SLAM, all decisions to update the graph, either by adding more nodes or by closing loops, are taken in terms of overall information gain.

Pose SLAM does not maintain a grid representation of the environment. It only encodes relations about robot poses. The environment however, can be synthesized at any instance in time using the pose means in the graph and the row sensor data. The resolution at which the map is synthesized depends on the foreseen use of this map. For instance, in [10] occupancy grid maps at very coarse resolution are produced to evaluate the effect of candidate trajectories in entropy reduction. But in [15] for instance, there is no need to render a map to plan optimal trajectories in a belief roadmap.

In this paper, we use the Pose SLAM estimate and raw sensor data to evaluate entropy reduction at each cell of a robot configuration space grid.

III. Log Odds Occupancy Grid

The quality of the occupancy grid produced is a key element of our exploration strategy. The mapping of frontiers near obstacles in the presence of uncertainty might drive the robot to areas near collision, a situation we need to avoid. Moreover, there is a compromise between tractability and accuracy in choosing the resolution at which the occupancy cells are discretized. In [10], our group devised a strategy to decide exploratory actions independent of grid cell size. Information gain was measured as a scalar over the whole occupancy map, with little variation depending on cell size. The key was to treat localization uncertainty independently of the occupancy map. In that work however, the computation of the occupancy map, although made at low resolution, required repeated ray-casting at each iteration of raw sensor data for each robot pose in the Pose SLAM graph. This process is computationally expensive and was computed only for a number of nearest neighbor poses.

In this paper we provide a more accurate computation of occupancy maps, which is necessary for the proper computation of potential information fields. We do so for all poses in the Pose SLAM graph, and not only a limited number of them. Moreover, the resolution at which the occupancy grid map is computed is finer than what we were able to compute in [10]. Instead of repeating the ray-casting operation at each iteration, we store local log odds occupancy maps at each robot pose, and aggregate them efficiently for the computation of a global log odds occupancy map.

A. Grid map aggregation

Once for each robot pose x_k, the raw sensor data is ray-casted to accumulate evidence for each cell c_{ij} in a log odds occupancy grid in local coordinates

$$m_{ij} = \log \frac{p(c_{ij})}{1 - p(c_{ij})} \, . \tag{3}$$

Fig. 2 shows the local log odds occupancy grids computed for a number of robot poses. Negative values mean free space, and positive values mean obstacles. A value of 0 means unexplored. During open loop, each local map is aggregated into the global log odds occupancy map. To relate them in a common reference frame, each local map is rotated via sheers and translated using very efficient image processing routines. Only at loop closure, the occupancy map is recomputed from scratch using all previously stored local log odds maps but oriented according to the new estimated robot poses. The result is shown in Fig 3a.

B. Frontiers, obstacles, and free cells

Cell classification can be solved for directly from Eq. 3. Note however that map aggregation was computed only at the mean pose estimates. To smooth out misclassified and unobserved cells and to classify free cells, obstacles and frontier cells (unobserved close to a free cell) morphological opening and closing operations on the global log odds map are used. The resulting detection of frontiers, obstacles and free cells is exemplified in Fig. 3b.

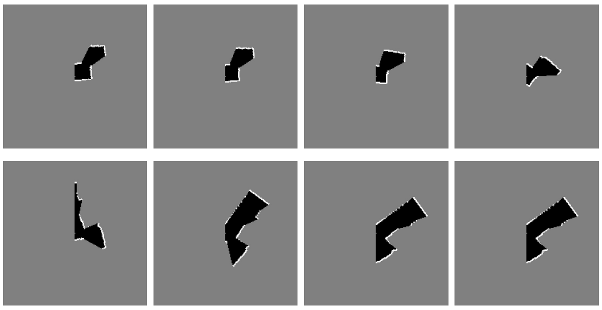
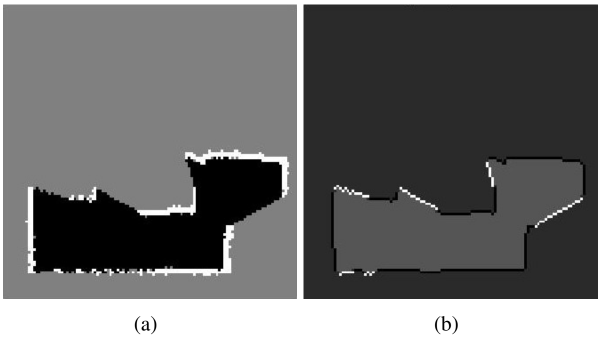


Fig. 2. A number of log odds occupancy maps in local coordinates.



(a)

(b)

Fig. 3. (a) Aggregated log odds occupancy map. (b) Classified frontier cells (white), obstacle cells (black), free cells (light grey) and unobserved cells (dark grey).

IV. Exploration with Potential Information Fields

Our purpose at each exploratory step is to find a path that drives the robot to those locations in the map that reduce the uncertainty in classification of free and occupied cells. That is, as in [10], to drive the robot to minimize the joint robot path and map entropies.

The objective is to find a scalar function ϕ defined over all C-space cells such that its gradient $\nabla\phi$ will consist of a path with largest joint path and map entropy decrease. Unfortunately, the entropy decrease at any given C-space cell is not independent of the path taken to arrive to such pose. The reason is that different routes induce different reduction of path and map entropies. Take for instance two different routes to the same C-space cell, one that goes close to previously visited locations, and one that does not. In the former, the robot would be able to close loops, and thus maintain bounded localization uncertainty. Conversely, an exploratory route would instead reduce the map entropy. To compute the reduction in both map and path entropies at any given cell in C-space, we assume that the robot can 'suddenly' appear at that location. This simplification will allow us to compute entropy variations over all C-space cells.

In contrast to our approach, [13] define a potential scalar function using attraction and repulsion fields on frontiers and the current robot pose, with some boundary conditions on obstacles. Choosing frontiers as attractors poses some challenges. Frontiers are unexplored areas next to free cells which have a significant probability of being yet unseen obstacles. The use of potential fields to reach frontiers produces perpendicular robot configurations at the arriving locations, thus making the robot face these new obstacles directly, with the consequent unavoidable collision. Other methods that select frontiers as goal locations during exploration that are not based on potential fields share the same inconvenience [16]. We instead set as attractors not the frontiers, but the robot configurations at which joint entropy reduction is maximized. These poses are not necessarily close to frontiers, but can be at any configuration in the free space. In addition, these attractors will also guarantee larger reductions in map entropy since more frontier cells can be observed from these locations than from the frontier. See Fig 4.

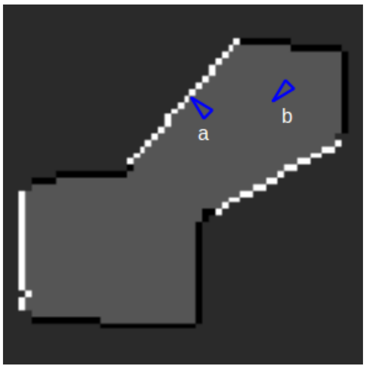


Fig. 4. Pose a: Frontiers method's exploratory goal. Pose b: Optimal map entropy reduction goal in C-space.

The joint state entropy is approximated, as in [10], as the sum of the entropy of the map m and the entropy of the path x, given all motions u and observations z,

$$\begin{aligned} H(x,m|u,z) &= H(x|u,z) + \int_x p(x|u,z)H(m|x,u,z)dx \\ &\approx H(x|u,z) + H(m|u,z). \end{aligned} \tag{4}$$

We evaluate joint entropy reduction on these two terms separately for each discretized robot configuration in C-space, treat this entropy reduction as an information field and smooth it to avoid discontinuities. We finally compute the exploration path as the gradient of this field.

A. Map entropy reduction

In contrast to [10], in which we compute the reduction in entropy for a limited set of final configurations, we now compute it for the entire discretized C-space. For a map with size cell l, its entropy can be computed as a scalar value.

$$H(m|u,z) = -l^2 \sum_{c\in m} (p(c)\log p(c) + (1-p(c))\log(1-p(c))). \tag{5}$$

The reduction in entropy that is attained after moving to a new location and sensing new data (u', z') depends basically on the number of cells that will change its status from unknown to discovered: either obstacle or free. Estimating the number of discovered free cells is impossible, we may predict how many cells will ve discovered. We are content

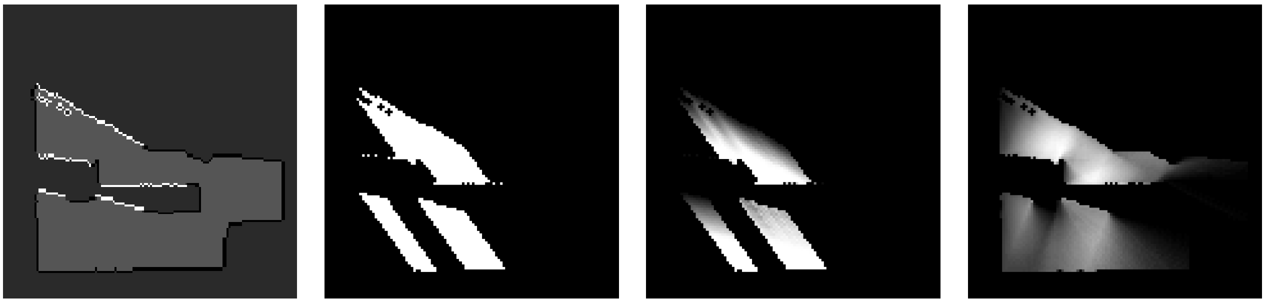


Fig. 5. (a) Occupancy map with obstacles (black), frontiers (white) and free cells (light grey). (b) Obstacle occlusion mask in one ray direction. (c) Correlation of map entropy decrease in one ray direction after variable resolution update. (d) Sum over entire sensor spread for one robot orientation.

with measuring only entropy reduction as the increase in the number of discovered frontier cells. This problem is as simple as convolving the sensor range at the hypothesized robot pose with the current frontier map.

We are able to compute this entropy change very efficiently with the following three steps:

1) *Obstacle occlusion mask.* We generate a 3-dimensional grid. Its dimensions are x, y, and the direction of each laser ray. For each ray orientation layer, an obstacle occlusion mask is created, i.e. a 3D binary mask annotated with neighbor obstacles along each ray direction. The mask is computed with a one-dimensional convolution with an inverse exponential motion kernel over a positive value for frontier cells and a negative value for obstacles. Binary thresholding these values effectively avoids the updating of entropy reduction behind obstacles. See Fig 5b.
2) *Frontier convolution.* For each ray orientation layer, a convolution is made with a one-dimensional motion kernel in all non-occlusion cells. This convolution spreads entropy reduction for observations of frontiers for each ray orientation. One frontier cell may receive different rays casts, thus it is necessary to compensate this in order not to overestimate the number of frontier cells observed. Ray cast density at each cell r is a function of the distance from the robot to that cell and the angle β between two consecutive sensor rays
$$n = \frac{1}{r \ \tan \beta} \ . \tag{6}$$
Thus, our convolution kernel compensate this overestimation of frontier ray castings at each cell. To this end, each cell in our kernel is weighted with $\min(1, r \ \tan \beta)$. The result of this frontier convolution is shown in Fig. 5c.
3) *Sum over entire sensor spread.* We now define a second 3D grid of C-space to annotate entropy reduction for each hypothetical robot pose. Once the frontier convolution layers for all ray directions have been computed, we sum all the layers within the sensor orientation range to update the corresponding cell in the C-space entropy reduction grid. The result of this step is shown in frame d of the same Figure.

Finally, since we already have localization uncertainties encoded in the Pose SLAM graph, these can be used to weight our entropy reduction estimates. The motivation behind this is that exploration activities that depart from well localized priors will be weighted higher than explorations that depart from uncertain locations. A noise free motion command gives a lower bound to the marginal path posterior equal to current marginal pose. Thus, it suffices to weight the entire entropy reduction map with this lower bound, the determinant of the current marginal covariance $|\mathbf{\Sigma}_{kk}|$.

B. Path entropy reduction

To compute the first term in Eq. 4, the entropy of the path could be approximated without taking into account correlation between poses [10], by averaging over the individual pose marginals
$$H(x|u,z) \approx \frac{1}{N} \sum_{i=1}^{N} \ln((2\pi e)^{(n/2)} |\mathbf{\Sigma}_{ii}|), \tag{7}$$

Evaluating this term is not necessary since we are interested in entropy change, i.e., information gain. And, as stated before, we are not evaluating entropy change for just one posterior pose, but for the whole discretized C-space. Assuming a noise free platform for the evaluation of the final leg in the path, the jump from the current pose to each configuration will produce the same marginal posterior, with zero information gain, except at loop closure.

For a configuration i in the C-space, the path entropy reduction of a loop closure with pose j is given precisely by its information gain.
$$\mathcal{I}_{ij} = \frac{1}{2} \ln \frac{\mathbf{S}_{ij}}{|\mathbf{\Sigma}_y|} \tag{8}$$
with $\mathbf{\Sigma}_y$ the sensor covariance, and $\mathbf{S}_{ij}$ the innovation covariance of the Pose SLAM update.

The parameter match area of the sensor is defined as the intervals in x, y and θ where loops can be closed by the sensor. Thus, a loop can be closed in each configuration in the C-space inside the match area of any previous pose of the trajectory. Instead of iterating over each cell in the C-space and searching for its loop closure candidates in the

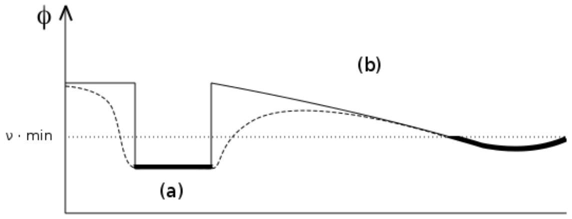


Fig. 6. The C-space entropy change grid is cropped at a desired value v and smoothed with a harmonic function to produce the desired information potential field. The zone (a) represents a region with steep entropy reduction within the sensor range to guarantee loop closure. Zone (b) represents an area worth exploring.

Pose SLAM graph, the iteration proceeds the other way. For each pose in the Pose SLAM graph, we annotate the cells inside their match area in the C-space with the corresponding information gain.

C. Potential field and gradient path

For our C-space entropy change grid (information grid) resulting from the sum of map and path entropy reduction previously explained,to become the scalar function ϕ, we still need one more step. To avoid long valleys with no entropy change, the grid is turned into a potential field, by cropping it first to a desired value v of 60% to define attraction areas. And then, smoothing it using a harmonic function of the form

$$\phi_{xy\theta} = \frac{1}{6}(\phi_{x^{\pm}y\theta} + \phi_{xy^{\pm}\theta} + \phi_{xy\theta^{\pm}}) \tag{9}$$

where the superscript ± is used to indicate neighbor cells in the C-space grid. See Figs 1 and 6.

D. Obstacle avoidance and boundary conditions

In our computation of the entropy grid we have considered obstacles to adequately propagate entropy change along sensor rays taking into account occlusions, but we have still not penalized configurations that get close to them. We resort to the use of boundary conditions

$$\nabla\phi(x) = g(x) \qquad \forall x \in \mathcal{S}_{\text{boundary}} \tag{10}$$

as in [13], with the difference that instead of using Neumann boundary conditions to guarantee flow parallel to obstacles, we still want some repulsive perpendicular effect from them. This effect can be achieved by mirroring weighted inner cell values near obstacles. A unity weight ($w = 1$) means parallel traverse along the obstacle boundary, and larger values of w induce repulsion. In the method reported here we start each each planning step with low repulsion w to avoid bottleneck local minima and increasing it and re-planning in case of a collision path. The final path is obtained traversing the gradient field from the current robot configuration to the robot configuration with largest joint entropy reduction. A C-space orientation layer of potential fields resulting of the method is shown in Fig.1.

	Active Pose SLAM	**Potential Information Fields**
Occupancy map	47.41 s	8.2 s
Path plan	56.98 s	28.52 s
Final map entropy:	149.33 nats	123.84 nats
Final path entropy:	−2.22 nats	−2.08 nats
Planning steps:	20.3	18.4

TABLE I

AVERAGE COMPARISON OF ACTIVE POSE SLAM AND THE PROPOSED EXPLORATION APPROACH.

V. SIMULATIONS

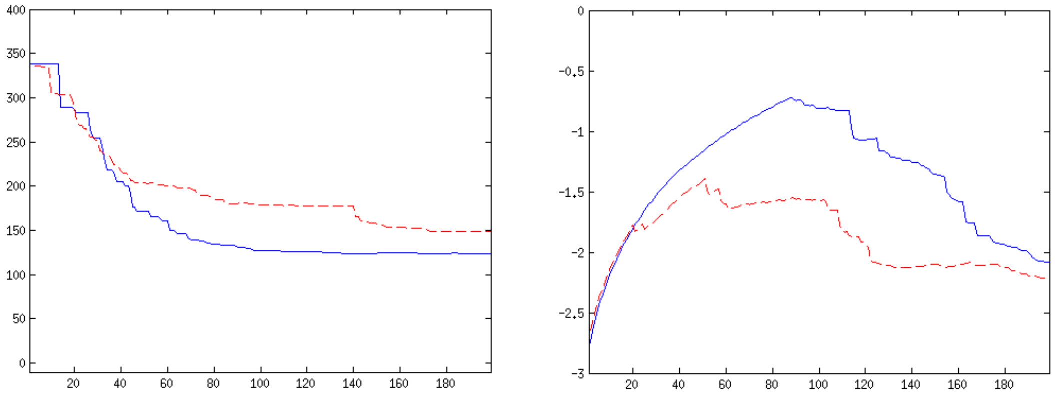
In order to evaluate the exploration strategy presented in this paper and to compare the results with those of [10], we simulated a robot exploring the widely used cave-like two-dimensional environment available from [17], scaled to a resolution of $20\,\text{m} \times 20\,\text{m}$. Robot motion was simulated with an odometric sensor with noise covariance $\boldsymbol{\Sigma}_u = \text{diag}(0.1\,\text{m}, 0.1\,\text{m}, 0.0026\,\text{rad})^2$. The robot is fitted with a laser range finder sensor with a match area of $\pm 3\,\text{m}$ in x and y, and $\pm 0.52\,\text{rad}$ in orientation. That is, this is the maximum range in configuration space for which we can guarantee that a link between two poses can be established. Relative motion constraints were measured using the iterative closest point algorithm. Measurement noise covariance was fixed at $\boldsymbol{\Sigma}_y = \text{diag}(0.05\,\text{m}, 0.05\,\text{m}, 0.0017\,\text{rad})^2$. Laser scans were simulated by ray casting over a ground truth gridmap of the environment using the true robot path. The initial uncertainty of the robot pose was set to $\boldsymbol{\Sigma}_0 = \text{diag}(0.1\,\text{m}, 0.1\,\text{m}, 0.09\,\text{rad})^2$. Informative loop closures were asserted at $\mathcal{I}$ = 2.5 nats.

Table I summarizes average computation times for map building and path planning, as well as final entropy values after 7 simulations of 200 odometry steps each for the two methods using the same simulation variables.

Plots a and b in Figure 8 show average map and path entropy values, respectively, for the 7 simulation runs. The plots show the reduction in map and path entropies for both methods. In continuous blue, the proposed approach; in dashed red, the result of Active Pose SLAM; the bold lines indicate map entropy, and the thin lines indicate path entropy.

From the plot, we observe that, in average, the proposed method attains a steady state in map entropy at a significantly lower number of simulation steps, without compromising localization estimates. Path entropy is about the same. One major advantage of the proposed approach is in execution time. As also shown in the Table, despite the increase in the dimensionality from 2D to 3D for the computation of the entropy occupancy grids, the use of kernel convolutions provides a significant reduction in computation speed compared to the traditional occupancy maps of Active Pose SLAM. Furthermore, the computation of the gradient descent path in our approach takes about half the time in average than the A* planning method of Active Pose SLAM.

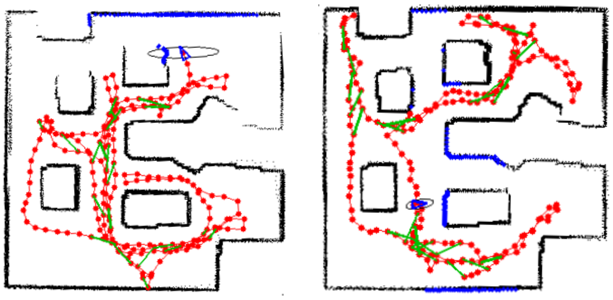
In all simulations, collision was detected 5 times or less per experiment realization. Boundary condition weights were



(a)

(b)

Fig. 7. Average entropies for the 7 simulation runs. In continuous blue, the proposed approach; in dashed red, the result of Active Pose SLAM. (a) Map entropy. (b) Path entropy.



(a)

(b)

Fig. 8. Final trajectories after two explorations of 200 simulation steps. In red the robot path, in green the loop closure links. (a) Exploration with Active Pose SLAM. (b) Exploration with Potential Information Fields.

reduced in those cases as explained in Sec. IV-D, and a new collision free trajectory was found.

Plots b and c in Figure 8 show one realization of the experiment. The red dots and lines indicate the executed robot trajectories, the green lines indicate loop closures, and the black dots render occupancy using the complete path estimate.

VI. Conclusions

We have presented a mobile robot exploration strategy that traverses a gradient descent on a C-space field based on path and map entropy reduction estimates. The technique makes use of very efficient convolutions first, to project boundaries along sensor rays, and secondly, to integrate entropy measures at independent robot orientation layers. The method outperforms exploration methods that drive the robot to frontiers and loop closures using the more conventional grid-based occupancy maps, both in map and path quality, and in computation speed.

In computing the gradient descent on the C-space field, the method assumes an holonomic robot platform. We plan to extend the method to account for non-holonomic restrictions. Future work also includes an implementation in ROS and comparison against competing approaches on real scenarios.

References

[1] P. Whaite and F. P. Ferrie, "Autonomous exploration: Driven by uncertainty," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 3, pp. 193–205, Mar. 1997.

[2] H. J. S. Feder, J. J. Leonard, and C. M. Smith, "Adaptive mobile robot navigation and mapping," *Int. J. Robot. Res.*, vol. 18, pp. 650–668, 1999.

[3] F. Bourgault, A. Makarenko, S. Williams, B. Grocholsky, and H. Durrant-Whyte, "Information based adaptative robotic exploration," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Lausanne, Oct. 2002, pp. 540–545.

[4] T. Vidal-Calleja, A. Sanfeliu, and J. Andrade-Cetto, "Action selection for single camera SLAM," *IEEE Trans. Syst., Man, Cybern. B*, vol. 40, no. 6, pp. 1567–1581, Dec. 2010.

[5] S. Huang, N. Kwok, G. Dissanayake, Q. Ha, and G. Fang, "Multi-Step look-ahead trajectory planning in SLAM: Possibility and necessity," in *Proc. IEEE Int. Conf. Robot. Autom.*, Barcelona, Apr. 2005.

[6] C. Leung, S. Huang, N. Kwok, and G. Dissanayake, "Planning under uncertainty using model predictive control for information gathering," *Robot. Auton. Syst.*, vol. 54, no. 11, pp. 898–910, Nov. 2006.

[7] C. Leung, S. Huang, and G. Dissanayake, "Active SLAM for structured environments," in *IEEE Int. Conf. Robot. Autom.*, Pasadena, 2008, pp. 1898–1903.

[8] C. Stachniss, G. Grisetti, and W. Burgard, "Information gain-based exploration using Rao-Blackwellized particle filters," in *Robotics: Science and Systems I*, Cambridge, Jun. 2005, pp. 65–72.

[9] B. Yamauchi, "A frontier-based approach for autonomous exploration," in *IEEE Int. Sym. Computational Intell. Robot. Automat.*, Monterrey, 1997, pp. 146–151.

[10] R. Valencia, J. Valls Miró, G. Dissanayake, and J. Andrade-Cetto, "Active Pose SLAM," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Vilamoura, Oct. 2012, pp. 1885–1891.

[11] A. Kim and R. M. Eustice, "Perception-driven navigation: Active visual SLAM for robotic area coverage," in *Proc. IEEE Int. Conf. Robot. Autom.*, Karlsruhe, May 2013, to appear.

[12] E. de Silva, P. Engel, M. Trevisan, and M. Idiart, "Exploration method using harmonic functions," *Robot. Auton. Syst.*, vol. 51, no. 1, pp. 25 – 42, 2002.

[13] R. Shade and P. Newman, "Choosing where to go: Complete 3D exploration with stereo," in *Proc. IEEE Int. Conf. Robot. Autom.*, Shanghai, May 2011, pp. 2806–2811.

[14] V. Ila, J. M. Porta, and J. Andrade-Cetto, "Information-based compact Pose SLAM," *IEEE Trans. Robot.*, vol. 26, no. 1, pp. 78–93, Feb. 2010.

[15] R. Valencia, M. Morta, J. Andrade-Cetto, and J. Porta, "Planning reliable paths with Pose SLAM," *IEEE Trans. Robot.*, 2013, to appear.

[16] B. Yamauchi, "Frontier-based exploration using multiple robots," in *Int. Conf. Autonomous Agents*, Minneapolis, 1998, pp. 47–53.

[17] A. Howard and N. Roy, "The robotics data set repository (Radish)," http://radish.sourceforge.net, 2003.