An Information-Driven Navigation Strategy for Autonomous Navigation in Unknown Environments

Redouane Boumghar
CNRS; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse, France
Université de Toulouse ; UPS , INSA , INP, ISAE ; LAAS ; F-31077
Magellium Toulouse, 24 rue Hermès, BP12113, 31521 Ramonville Saint-Agne, France
Simon Lacroix
CNRS; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse, France
Université de Toulouse ; UPS , INSA , INP, ISAE ; LAAS ; F-31077
Olivier Lefebvre
Magellium Toulouse, 24 rue Hermès, BP12113, 31521 Ramonville Saint-Agne, France

***Abstract*— This paper presents a strategy for a rover navigation in initially unknown or poorly known environments. The strategy consists in determining the areas in which information is relevant to gather for the rover to reach the goal. The approach relies on a probabilistic reasoning on the currently available information on the environment, and on the models of the vehicle perception and motion abilities. The interest of perceiving a given area is assessed by analysing the way navigation costs are propagated during the search of the optimal path to reach the goal, taking into account the vehicle perception and motion models. Results illustrate the proposed strategy, and show its pertinence when compared to more classical navigation strategies. The extension towards multiple-robot coordination is straightforward, and is exemplified in the case where an aerial robot assists the rover by acting as a scout, aiming at optimizing the overall mission time.**

I. Introduction

The problem of path finding for a robot in an unknown or partially known environment is a key one in robotics. In disastrous situations, few information are available on the environment, and the difficulty of the path planning comes from the fact that required information to define optimal paths, namely *terrain traversability*, is known with a varying degree of certainty. At planning time the terrain information is not completely known, and it will be updated during navigation, which may lead path re-planning. Hence the efficacy of a given path not only depends on the information known at planning time, but also on the information on the environment the path will allow to acquire: shortest paths may be less efficient than longer ones that allow gathering additional information. The path finding problem to reach a given goal can thus be turned into "finding the areas of the environment where information is required in order to reach the goal".

a) Related work: Several studies deal with the navigation problem in partially known terrains by maximizing the information gain. [1] proposes a methodology for exploration where frontier information, traversability and reachability certainty are used to produce paths maximizing the expected information gain. [2] proposes an approach for perception and path planning in which confidence and utility of perception are exploited to drive a robot in partially known environments. Similarly, a utility measure is defined in [3] to extract points of interest by introducing "hallucinated maps". These points are planned to be sensed to extend the path planning horizon in most promising areas. The approach proposed in [4] adds the notion of "localizability" to further constrain the path definition.

Other work exploit the available information in the context of multiple robot missions [5, 6]. [7] proposes an algorithm for multi-robot exploration using target points on the frontiers between known and unknown areas, selected on the basis of the cost to reach them and their utility. The concept of sub-target utility allows to distribute them among robots to avoid simultaneously sharing the same target point. The utility of a target point is initially set to 1, and to 0 when selected by a robot. The utility of each of its neighborhood target points is reduced by the probability of inter-visibility with the previously selected target, defined by an analysis of the visible distances distribution in the partially known environment. No sensor quality model is explicitly considered, which may result in assigning robots to areas where they can not augment the information on the terrain. [8] also guides the exploration with goal points placed on frontiers between perceived and unknown areas, and selects them by choosing the lowest cost provided by the A* algorithm called for each frontier point considered. The utility of a point is defined by the sum of its distances from other robots, and is used to rule the points distribution. This utility does not take into account that some robots may never be able to reach a point, even if they are spatially close to it. To plan most-likely paths, [9] uses Gaussian processes in order to replace the classical scalar navigation cost by a density of probabilities. The terrain is segmented into n classes, and the solution, illustrated in the three classes case (obstacle - scrub - road), provides a distribution of paths that vary with local variations in terrain cost – but it does not assess which path to select.

b) Problem statement: We consider the case of a rover that is endowed with N motion modes $M_1, \ldots, M_N$ (*e.g.*

fast navigation on flat terrains, slower navigation on rough terrains), to each of which is associated a unitary distance cost c_i that represents time. The navigation approach is made according to a two-stages itinerary/trajectory approach [10, 11, 12]: navigation is decomposed into *(i)* an itinerary planning step, that defines the waypoints to reach and the motion modes to apply, and *(ii)* a trajectory planning step, that defines the elementary motion commands to reach the waypoints, according to the selected motion mode.

We focus in this paper on the definition of the itinerary, which we refer to as "path planning". The fine trajectory planning step exploits dedicated terrain models (*e.g.* digital elevation map to navigate on uneven terrains, or a binary obstacle/free representation to navigate on easier terrains). The way such finer models are built and exploited to define the motions to apply are out of the scope of this paper – only the knowledge of the speed of each motion mode is used to define the cost of a path.

c) Approach: Our approach relies on an information-based decision process, that exploits the available information on the environment and models of the robot motion and perception actions. The environment is modeled using a traversability uncertainty grid that expresses the probabilities that a given navigation mode can be applied, plus a probability that it is an untraversable obstacle. We use the **A*** algorithm to find optimal paths, but more importantly, we use its resulting calculations, coupled with the known traversability information on the environment, to extract areas of interest for the navigation mission. Perceiving these areas comes to extend the viewing horizon as in [3], but using an approach that explicitly reasons on the available information on the environment, on the robot perception models and on its motion abilities to define the interest of perceiving a given area.

d) Outline: The next section depicts the information available on the terrain, the means to gather such information, and the way it is structured into a graph. Section III is the heart of the paper: it introduces the way to determine the areas that are relevant to perceive for the rover to reach its goal. Section IV analyses various simulation and experimental results, and introduces an extension to the air/ground cooperative navigation problem, in which a UAV is directed to perceive the areas relevant for the rover to reach its goal.

II. Environment model

A. Traversability map

The proposed path finding approach exploits a probabilistic representation of the environment that assesses the motion modes that can be applied, the *traversability map*. It is a scaled georeferenced raster structure. This representation is akin to a probabilistic occupancy grid: to each pixel is associated a probability distribution that represents the applicability of the possible N motion modes, plus the probability that it is not traversable (obstacle). This distribution corresponds to the probabilistic labelling of the pixels into N_c classes: $\mathcal{C} = \{C_1, \dots, C_{N_c}\}$, where C_{N_c} denotes the *obstacle* class ($N_c = N + 1$). So to each pixel N_c partial probabilities are associated: $\{P(C_1), \dots, P(C_{N_c})\}$, where $\forall C_i \in \mathcal{C}, P(C_i) \in [0, 1]$ and $\sum_{i=1}^{N_c} P(C_i) = 1$. The unperceived pixels have an uniform probability distribution – unless a priori information on the environment is known, in which case it is represented by a non-uniform distribution.

Numerous contributions in the literature propose means to build such a representation on the basis of range data (*e.g.* using a Bayesian classifier [11], or analysing digital terrain maps). In the context of this paper where the traversability information is assessed by a UAV (see section IV), we use the method presented in [13] that assesses the planarity of areas perceived by monocular vision using homography estimates (figure 1) – a similar approach can be found in [14].

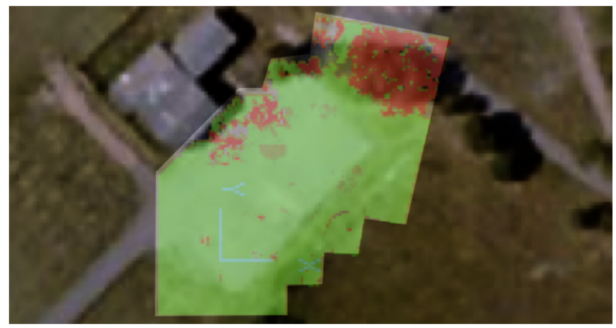

Fig. 1. Probabilistic environment model built from a sequence of images acquired by a UAV, superimposed on an aerial view of the area. The model here contains two classes: *(flat,obstacle)*, the probabilities being represented according to a green/red color scale – transparency has been added to the colored map layer to exhibit the mapped area.

B. Navigation Graph

A navigation graph is built upon the cells of a quadtree decomposition of the raster world. The quad-tree is defined on the information encoded in each pixel. It is built according to a bottom-up scheme, and stops when a cell holds homogeneous data or reaches the maximum cell size – this latter constraint being set according to the desired itinerary resolution.

a) Quad-tree decomposition: The path planner exploits the graph defined upon the raster traversability map to produce a path described by a sequence of way points and to assess the relevant areas to perceive. The resolution of the way point sequence (its spatial frequency) does not need to be very fine, and to save computation time the graph must be of reasonable size: the raster traversability map is therefore first structured into a quadtree.

The homogeneity of a cell is not defined by a strict state equality of all its pixels but by a similarity criterion based on the comparison of each pixel with the mean and standard deviation of the cell it belongs to.

In the quadtree cell number k containing N_k pixels, the mean value of a terrain class C_i is noted μ_i^k. Its corresponding standard deviation is noted σ_i^k.

The cell k is homogeneous if all its pixels j are close to the mean state by respecting the following (1):

$$\forall i \in [1, N_c], \forall j \in [1, N_k],\ |P_j^k(C_i) - \mu_i^k| < F_\sigma\ \sigma_i \qquad (1)$$

The standard deviation factor denoted F_σ can be used to relax or restrain the decomposition constraint, and the limit $F_\sigma \;\; \sigma_i$ is inspired from the Chebyshev's inequality which ensures that 75% of the data in a normal distribution is closer than two standard deviations from its mean ($F_\sigma = 2$).

b) Navigation graph: A navigation graph is built upon the quadtree cells: graph nodes are set at the middle of each cell border, and each node is connected to all the other nodes of the two neighboring cells. Thus each edge of the graph traverses only one cell of the environment and holds one consistent information of traversability.

The use of the quadtree decomposition reduces the search domain. The number of vertices and edges in the graph depends on the quadtree segmentation of the world map, as illustrated Figure 2: the use of a regular grid map without any quadtree decomposition produces a navigation graph that contains about 6 times more edges than the graph built upon a quadtree decomposition with a loose similarity criterion.

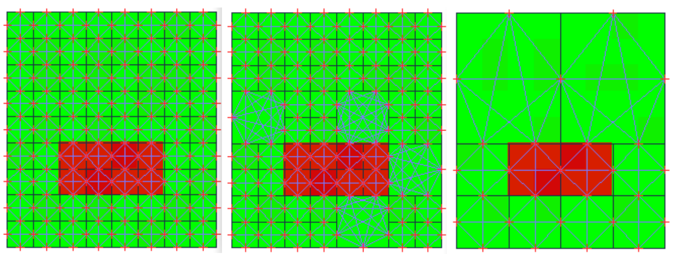

Fig. 2. Three navigation graphs defined on the same environment model. Left: graph defined without any quadtree decomposition; center: quadtree decomposition using a strict homogeneity criterion ($F_\sigma = 0$); right: quadtree decomposition with $F_\sigma = 2.0$. The number of nodes/edges are respectively 161/432, 142/414 and 27/66.

Special care has to be taken while choosing the F_σ value. A too high value will cause the gathering of terrains of too different types in the same quadtree. Generally a value of $F_\sigma = 2.0$ gives good decompositions. It is coherent with the Chebyshev inequality and with the approximation done while affecting the same terrain attributes to all edges in a same cell.

III. Assessing relevant areas to perceive

The main idea here is to use the planning algorithm itself to highlight the most needed information for path planning optimization. Our approach to assess areas where information is required relies on the two following points:

- information is useful to gather at places where it is low: information entropy expresses the amount of information in each map cell,
- information must be useful for the path planner: the proposed approach exhibits alternate potentially optimal paths, from which areas which are relevant to perceive for the navigation task are defined.

A. Navigation cost evaluation

The navigation cost f_c of a path in the graph is the cumulative sum of the cost of each graph edge in the path. Edges' states are represented by the probability distribution over the terrain classes, The real estimated cost to traverse a region of class C_i (*i.e.* $P(C_i) = 1.0$) is defined by c_i (cf. section I).

We define the cost function by separating the obstacle class from other traversable terrain classes: the cost function is expressed as the product of two contributions f_t and f_o: $f_c = f_t * f_o$.

- f_t defined in equation (2) accounts each traversable class $C_{j \neq N_c}$
- f_o defined in equation (4) instances the contribution of the obstacle class N_c taken into account by using $1 - P(C_{N_c})$ as the expected chance to traverse a zone.

$$f_t = w_0 + \sum_{i=1}^{N_c} \big(w_i(1 - \sum_{j=1}^{i} P(C_j))\big) \qquad (2)$$

Equation 2 shows a sum of hyperplanes each weighted by w_i representing the influence of each corresponding class C_i on the hyperplanes sum. The weights w_i are defined recursively from the real estimated costs according to (3).

$$w_0 = c_1, \quad c_i = \sum_{j=0}^{i-1} w_j \qquad (3)$$

To take into account the obstacle class effect f_o, a solution in (4) based on the logarithm function allows to get a greater influence on the cost sooner than an hyperbole. A polynomial adjustment is used to accentuate obstacles impact on probabilities greater than 0.5. A safety limit on the obstacle probability $P^{lim}(C_{N_c})$ is also used to consider all $P(C_{N_c}) > P^{lim}(C_{N_c})$ as totally untraversable terrains and thus affecting maximum cost to those classes.

$$\forall P(C_{N_c}) < P^{lim}(C_{N_c})$$
$$f_o = 1 + log((1 - \left(\frac{P(C_{N_c})}{P^{lim}(C_{N_c})}\right)^2)^4)^2 \qquad (4)$$

Terrain classes are considered in order of their influence to the cost. The most traversable classes are considered first. Each terrain class probability is taken into account within the cost function by considering all its previous classes in the mentioned order. This permits the navigation cost to be combined coherently with the vehicle's motion model.

For the **A*** search, the heuristic function h is defined as the euclidean distance to the goal.

B. Navigation Success Potential - NSP

The planned path is defined by the start and goal locations, and the information on the terrain state. In some cases as illustrated by Figure 3 a slightly different start condition can generate large differences in the path solution. In this figure only half of the world is well known and a wall in the middle forces the robot to choose a solution trajectory on the left or on the right. These two alternatives will lead the robot into two geographically very different zones.

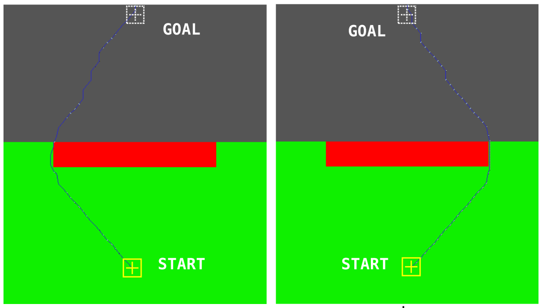

Fig. 3. Two slightly different start points lead to two very different solutions. (a) slight left offset of the start point. (b) slight right offset of the start point.

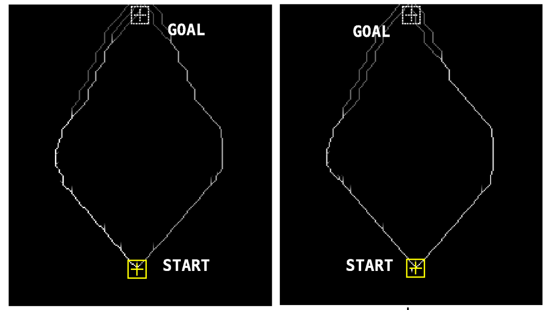

Fig. 4. Navigation Success Potential on the same examples (a) and (b) of figure 3

Zones behind information frontiers being in the way of path alternatives have to be perceived to help the navigation process. In order to plan perception tasks over these zones, we need to evaluate if the choice of these zones is worth for the navigation mission. The **A*** search provides the graph with a cost for each visited node. The visited nodes, noted as $\mathcal{N}_i$, form an area bordered by a list of open nodes noted as $\mathcal{N}_j^o$. The set of all open nodes is defined as $\mathcal{G}_o$ and the set of all visited nodes is noted as $\mathcal{G}_i$. Open nodes are the points where the **A*** search stopped before finding the goal itself. They all have an evaluated cost. We introduce the navigation success potential (NSP) to evaluate how promising are path alternatives after having calculated the nominal path.

The NSP noted as $\mathcal{P}$ is calculated by accounting on each node $\mathcal{N}_i$ how many other nodes back point to it in order to patch up with the start position. This means counting how many path alternatives each node $\mathcal{N}_i$ offers. Each count for an alternative is weighted by the sum of the inverse A* cost $f_c(\mathcal{N}_j^o)$ and the heuristic $h(\mathcal{N}_j^o)$ of the open node it leads to. Thus the NSP is described on each node i as $\mathcal{P}_i$ thanks to (5). The potential is initially set to zero for all the nodes. The set $\mathcal{G}_o^i$ is the set of all open nodes having path to the start that passes by the visited node $\mathcal{N}_i$.

$$\forall \mathcal{N}_i \in \mathcal{G}_i, \quad \mathcal{P}_i = \sum_{j=\{\mathcal{G}_o^i\}} \frac{1}{f_c(\mathcal{N}_j^o) + h(\mathcal{N}_j^o)} \tag{5}$$

Figure 4 illustrates (5) on the environment illustrated by the figure 3. On each case the NSP greatly highlights the second alternative.

Nodes not belonging to the nominal path and that hold a strong NSP are very likely to be used if the nominal path turns to be impassable after further discovery of the environment. They can be given to other robots for perception task in order to proceed for path clearance. For that purpose we need to define which of these points should be perceived first.

C. Perception task selection

To integrate the environment knowledge we need to cross the NSP with the probabilistic information from the map. The cooperation task is mainly an information gathering operation, perception. For a perception task to be beneficial it must take into account:

- the navigation potential of the zone to be perceived
- what would be the information gain if a given zone is perceived by a given sensor

The first point is covered by the NSP built in previous section III-B. The second point is the evaluation of the entropy shift of the information on a piece of terrain observed with a specific sensor.

c) Probabilistic Sensor Model.: The perception abilities of each sensor are represented by all the conditionnal probabilities across the N_c terrain classes. It helps us evaluating the gain of information a given sensor can produce on a given piece of terrain.

The probabilistic model of each sensor is defined as the matrix M_s of all conditionnal probabilities among terrain classes. For each terrain class i we have an array of conditionnal probabilities $P(C_j|C_i)$, $\forall j \in 1..N_c$. Each array represent a column of the model matrix. For each of these column the conditionnal probabilities fulfill the following (6):

$$P(C_i|C_i) = 1 - \sum_{j=1,j!=i}^{N_c} P(C_j|C_i) \tag{6}$$

The general form of matrix M_s is given in (7) where the perfect sensor would be represented by the identity matrix of size N_c.

$$M_s = \begin{pmatrix} P(C_1|C_1) & \dots & P(C_1|C_{N_c}) \\ \dots & P(C_i|C_i) & \dots \\ P(C_{N_c}|C_1) & \dots & P(C_{N_c}|C_{N_c}) \end{pmatrix} \tag{7}$$

This model can be used to predict a terrain state $\mathbf{P}$ knowing its current a priori state $\overline{P} = (P(C_1), \dots, P(C_{N_c}))$. Equation 8 depicts the definition of $\mathbf{P}$ with a normalization parameter noted η.

$$\mathbf{P} = \eta M_s \overline{P} \tag{8}$$

The quality of the perception is a function of the distance between the positions of the perceived cell and the sensor. It expresses the confidence of the overall perception [2], which degrades as the distance increases. This quality falloff is modeled by a function $\mathcal{F}_q$ that expresses the rate by which the entropy grows causing the overall information quality to fall. A full sensor model is defined by the set of a probabilistic perception matrix and a quality falloff function $\{M_s, \mathcal{F}_q\}$. The figure 5 illustrates the perception modelled through a 3-terrain-class perception matrix (similar to (9)) on three different cases.

$$M_e = \begin{pmatrix} 0.8 & 0.1 & 0.1 \\ 0.1 & 0.8 & 0.1 \\ 0.1 & 0.1 & 0.8 \end{pmatrix} \tag{9}$$

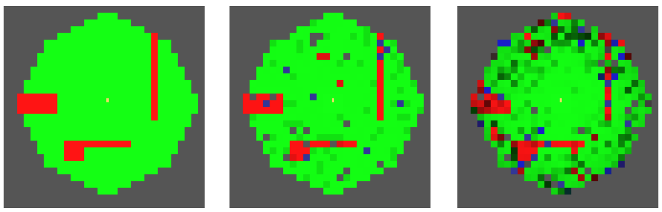

Fig. 5. Simulation of different bird-view perceptions. The vehicle is in the center of each view, obstacles are in red, free zones in green, rough terrain in blue and unknown terrain in grey. From left to right: perfect sensor, sensor modelled with sensor model matrix M_e (defined in (9)) with no quality loss, and same sensor with a polynomial entropy quality falloff.

d) Path selection: Once a navigation target is defined the UGV calculates the best initial path to reach it according to its current knowledge about the environment. Then a utility map is built by fusing the NSP with the information gain provided by the model of the cooperative vehicle's sensor. The information gain I_g is evaluated by comparing the entropy $\mathcal{H}$ of the current state to its predicted state as in (10).

$$I_g = \mathcal{H}(\overline{P}) - \mathcal{H}(\mathbf{P}) \tag{10}$$

The information gain I_g from (10) and the NSP defined in (5) are used to specify the navigation utility $\mathcal{U}$ as written in (11).

$$\mathcal{U} = \mathcal{P} \times I_g \tag{11}$$

Thus the utility permits to highlight zones that are likely good to explore for navigation and on which perception is the most helpful. To extract a path that maximises the utility map we need to define its start and goal points. The start point is the frontier point with the greatest utility that is not included in the UGV path. The goal point is defined to be the last point the UGV is sure to reach with its best initial path. It can also be considered as the meeting point for communication.

The figure 6 shows the UGV paths on the left and the resulting path selection made on the utility map on the right that is to be given for execution to a cooperative vehicle.

IV. Analysis of the Proposed Strategy

A. Extension to an air/ground cooperative navigation approach

We consider here the problem of long range rover navigation in poorly known environments where maps can not be trusted anymore after a catastrophe. The rover is assisted by an UAV (cf. figure 7) to sense the environment ahead. The navigation

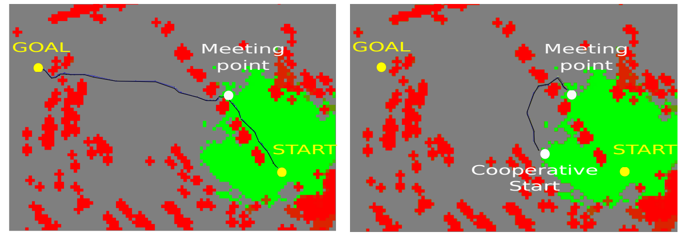

Fig. 6. Example of a region map built upon Velodyne laser scanner to show nominal path (left) and path selection (right) to maximize utility.

problem in this case comes to concurrently find sub-goals to reach for the rover and relevant areas to explore for the UAV. As an improvement of the UAV task in [15] where it is virtually tethered above the UGV, our approach explicitly drives the UAV to the most useful zones to perceive. In [15], the UGV evolves in a sparse vegetated off road environment where the UAV is mainly used to build an elevation map around the UGV in order to detect negative obstacles that are sorely detectable from the ground. As in [3] the use of a mid-range sensor or perception abilities from an UAV comes to extend the planning horizon and helps replanning before the UGV pushes itself into dead ends. The main differences

Fig. 7. UGV and UAV on the start point.

between the UAV and the UGV are reported in table I. The three points related to terrain perception, and the fact that an UAV can quickly move over obstacles make it an ideal remote explorer for the UGV.

TABLE I
UAV and UGV important differences

UGV	UAV
slow motion	fast motion
narrow sensor foot print	large sensor footprint
high-quality perception	low-quality perception
do not see behind obstacles	see all around obstacles

In our approach the UAV is directed to perceive areas on the basis of an optimisation of a utility / cost ratio, while satisfying communication constraints to ensure that it can transfer the information it acquires back to the rover. The strategy consists on a sequence of the following tasks: path planning for the UGV, utility map building, path selection for the UAV and communication of the newly perceived data at the meeting point where the cooperation loop restarts.

B. Comparison with different approaches

Table II shows the simulation results on two different maps shown in Fig. 8, comparing four different navigation strategies:

- Mono-UGV: The UGV by itself with an A* planner.
- UGV/UAV Overhead: the UAV supports the UGV by providing overhead perception.
- UGV/UAV Scouting: the UAV supports the UGV by scouting 50 meters ahead of the current optimal path.
- UGV/UAV Cooperative Approach: the UAV uses the utility $\mathcal{U}$ (cf. (11)) to plan observation paths.

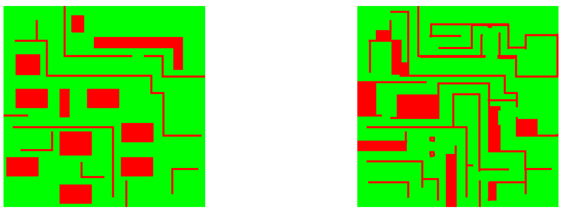

Fig. 8. Two large maps used in simulation. Left: an environment with easy detours, right: an environment with many dead-ends. The start point is on the bottom left corner and goal point on the top right corner of each map.

TABLE II

COMPARISON OF FINAL TRACK LENGTH AVERAGED OVER 100 RUNS OF DIFFERENT NAVIGATION STRATEGIES ON THE TWO MAPS OF FIG. 8.

	Mono-UGV Simple Planner	UGV/UAV Overhead	UGV/UAV Scout	UGV/UAV Cooperative Approach
Final Track Length (m)				
Detour map	27930	12933	11906	**6179**
Dead-ends map	53183	18577	14293	**9372**
Explored Zone Percentage (%)				
Detour	14.9	56.2	41.0	**65.5**
Dead-ends map	27.4	56.6	48.0	**79.5**

V. CONCLUSION

Results from the table II show that our approach optimizes the mission time with paths that are 48% shorter than a comparable simple scouting strategy. It also yields paths 77% shorter than the mono-robot planner which is bounded to scout for itself, thus making numerous backtrack trajectories. We have also noticed that the paths followed by the UAV in our approach are shorter than those made by the UAV following a simple scout strategy. Mission time reduction, total surface of explored zone and energy saving are shown to be the main benefits of using our approach.

Our global path planner uses A* but D* is a straightforward alternative that is more compatible with the fact that the map is incrementally updated. The solution provided by D* is the same as A* but it is more efficient while replanning when navigation costs change. Our current trajectories are filtered to avoid irregularities due to the fact the navigation graph is fitted to a quadtree structure. As measured in [15] there are slight differences between these trajectories and the true optimal paths. Another way to cope with path irregularities is to position the graph nodes more loosely on the quadtree edges making them converge to the true optimal. Field D* [16] and the sub-optimal Theta* path planner [17] are good alternatives to provide solutions closer to the true optimal solutions. However deeper considerations are needed to have graphs that fulfill our requirements. Indeed path planning has to visit a large part of the map in order to produce navigation costs far enough to highlight alternatives paths thanks to the NSP. This latter is required to provide a meaningful utility map for the navigation cooperation.

REFERENCES

[1] S. J. Moorehead, R. Simmons, and W. L. Whittaker, "Autonomous exploration using multiple sources of information," in *IEEE International Conference on Robotics and Automation, ICRA*, vol. 3. Carneige Mellon University, USA, 2001, pp. 3098 – 3103 vol.3.

[2] J. Gancet and S. Lacroix, "Pg2p: A perception-guided path planning approach for long range autonomous navigation in unknown natural environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, Las Vegas (USA)*, Oct. 2003, pp. 2992–2997.

[3] B. Nabbe and M. Hebert, "Extending the path-planning horizon," *The International Journal of Robotics Research*, 2007.

[4] J. Ko, B. Stewart, D. Fox, K. Konolige, and B. Limketkai, "A practical, decision-theoretic approach to multi-robot mapping and exploration," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, vol. 4, oct. 2003, pp. 3232 – 3238 vol.3.

[5] F. Bourgault, T. Furukawa, and H. Durrant-Whyte, "Coordinated decentralized search for a lost target in a bayesian world," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, vol. 1, Oct. 2003, pp. 48 – 53 vol.1.

[6] A. Singh, A. Krause, C. Guestrin, W. J. Kaiser, and M. A. Batalin, "Efficient planning of informative paths for multiple robots," in *IJCAI*, 2007, pp. 2204–2211.

[7] W. Burgard, D. Fox, M. Moors, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," *ICRA*, 2000.

[8] A. Marjovi, J. Nunes, L. Marques, and A. de Almeida, "Multi-robot exploration and fire searching," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2009, pp. 1929–1934.

[9] L. Murphy and P. Newman, "Planning most-likely paths from overhead imagery," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, May 2010, pp. 3059 –3064.

[10] A. Stentz and M. Hebert, "A complete navigation system for goal acquisition in unknown environments," *Autonomous Robots*, vol. 2, no. 2, Aug. 1995.

[11] S. Lacroix, A. Mallet, D. Bonnafous, G. Bauzil, S. Fleury, M. Herrb, and R. Chatila, "Autonomous rover navigation on unknown terrains: Functions and integration," *International Journal of Robotics Research*, vol. 21, no. 10-11, pp. 917–942, Oct-Nov. 2002. [Online]. Available: http://homepages.laas.fr/simon/publis/LACROIX-IJRR-2002.pdf

[12] J. Carsten, A. Rankin, D. Ferguson, and A. Stentz, "Global planning on the mars exploration rovers: Software integration and surface testing," *Journal of Field Robotics*, vol. 26, no. 4, pp. 337–357, April 2009.

[13] G. L. Besnerais, "Vision for (small) UAVs," in *2nd European Conference For Aero-Space Sciences, Brussels (Belgium)*, July 2007.

[14] S. Bosch, S. Lacroix, and F. Caballero, "Autonomous detection of safe landing areas for an UAV from monocular images," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, Beijing (China)*, Oct. 2006.

[15] A. Kelly, O. Amidi, H. Herman, T. Pilarski, A. Stentz, N. Vallidis, and Y. Warner, "Toward reliable off road autonomous vehicles operating in challenging environments," *The International Journal of Robotics Research*, vol. 25, pp. 5–6, 2006.

[16] D. Ferguson and A. Stentz, "The field d* algorithm for improved path planning and replanning in uniform and non-uniform cost environments," Tech. Rep., 2005.

[17] A. Nash, K. Daniel, S. Koenig, and A. Felner, "Theta*: Any-angle path planning on grids," in *AAAI Conference on Artificial Intelligence*, 2007.