The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

Path Planning with Variable-Fidelity Terrain Assessment

Braden Stenning and Timothy D. Barfoot

Abstract— **Terrain assessment and path planning are intrinsically linked. There exist a variety of terrain-assessment algorithms and these methods follow the trend of low-fidelity at low-cost and high-fidelity at high-cost. We present a modular path-planning algorithm that uses a hierarchy of terrain-assessment methods; from low-fidelity to high-fidelity. Using all the available sensor data, the visible terrain is assessed with the low-fidelity, low-cost method. The decision to assess a piece of terrain with the high-fidelity, high-cost method is made considering potential path benefits and the cost of assessment. The result is a lower combined cost of the path and terrain assessment that exploits the capabilities of the robot chassis where prudent. We demonstrate the technique on a large number of simulated path-planning problems using fractal terrain, as well as provide preliminary results from an experimental field test carried out on Devon Island, Canada.**

I. MOTIVATION

The next wave of planetary exploration is going to include rovers that need to travel great distances without constant supervision by Earthbound operators. For Mars exploration, the rover will require onboard autonomy to allow it to move beyond its sensing horizon many times between command cycles from Mission Control [1]. It is therefore imperative to limit the time the terrain assessment and short-range motion planning are under direct control of operators on Earth.

There are a significant number of terrain-assessment methods available to a mobile robot, ranging from simple methods that can run onboard and in real-time, to more complex simulations that include vehicle kinematics and terrain properties. These simulations can run onboard, but typically not in real-time. Most costly is a call home to ask for human intervention, and this is how much of the Mars Exploration Rovers (MERs) operations are carried out [2]. The trend to these terrain-assessment methods is low-fidelity at low-cost and high-fidelity at high-cost, where we consider fidelity to be a measure of how closely the assessment method is able to model the costs of driving over any patch of ground. We recognize that not all terrain requires a sophisticated assessment to determine traversability. For instance, plane-fit methods such as those used on the MERs [3], can confidently classify smooth, flat ground as traversable, and tall cliffs as not drivable. This is an example of a low-fidelity assessment method as it is only accurate in certain types of terrain. The difficulty arises when the rover must decide whether or not it can pass over and into areas that are cluttered or steep; this is exactly where much of the interesting science is located. The challenging terrain includes areas where geological forces and meteor impacts have naturally excavated the terrain and revealed many interesting scientific targets.

Fig. 1: Mobile robots can operate in terrain of varying difficulty. Some areas are easily assessed (as in the top image), while others require more advanced methods before the traversability can be determined (bottom image).

The next robotic explorers may have many ways of determining terrain traversability, but there will always be a trade-off between the cost of terrain assessment (computational or communication time) and the fidelity of the assessment method. We can also see that there is a fundamental relationship between path planning and terrain assessment; the path depends on the traversability of the terrain and, ideally, only the regions on the path need to be assessed.

In this paper, we present a novel, modular path-planning framework that uses a suite of terrain-assessment algorithms, limiting the application of the highest-cost (but highest-fidelity) methods only to regions that may be on the optimal path and that require a high-fidelity assessment.

II. RELATED WORK

The path planner must find an efficient route to an objective (scientific target, waypoint, etc.), if such a path exists. In order to do this, the planner needs traversability information for the surrounding terrain, including the traversal cost; the planner therefore has an inherent dependence on the terrain-assessment capabilities. In what follows we focus on the planning aspect of this problem, acknowledging that there are a variety of appropriate terrain-assessment techniques,

This work was supported by the Canada Foundation for Innovation and the Canadian Space Agency.

The authors are with the University of Toronto, Institute for Aerospace Studies, 4925 Dufferin Street, Toronto, Ontario, Canada M3H 5T6 braden.stenning@robotics.utias.utoronto.ca, tim.barfoot@utoronto.ca

but only explaining those examples in the test system.

Of particular note are some of the graph-based planning techniques. A graph is a set of states connected by edges. In the typical rover path-planning context, the states represent the locations in a world at which a robot might exist, while the edges contain the associated cost of driving between two adjacent states. The start state corresponds to the current position of the robot and the goal state is the desired robot position. The A* search algorithm [4] is an intuitive optimal planner, and when applied to a static, completely known graph, the performance is quite good. However, when the graph is unknown or changing, the A* algorithm is not suitable as it must completely re-plan after graph updates. The D* family of algorithms [5, 6], is designed to function efficiently with dynamic graphs. The algorithms make local repairs to the path as the graph is updated, thereby maintaining an optimal path according to the known terrain model. One extension, Field D* [7], allows for more direct routes by planning a path between two adjacent states using linear interpolation to determine the edge cost. This algorithm has been successfully implemented on the MERs [8, 9].

The idea of using two different terrain-assessment methods has been developed in the Terrain Adaptive Navigation (TANav) system [10]. TANav is designed for operation in areas where the robot may experience unpredictable wheel slippage. The initial method of assessment identifies terrain as definitely traversable, definitely not traversable, or uncertain. The path is planned on a goodness map created by this assessment. If the path goes through an uncertain region the second level of assessment, High-Fidelity Traversability Analysis (HFTA), is used to determine the cost of that section. The cost of the HTFA is never considered, though it is acknowledged that it is more computationally expensive.

Multiresolution path-planners [11] limit detailed map representation to complex areas of the path. They do not consider the representation cost during planning since it is small, unlike the cost of higher-fidelity terrain assessment.

Nabbe and Herbert [12] have work that deals with planning beyond the sensor horizon or through areas where insufficient data has been collected. When determining a path, the cost of acquiring data is considered during path planning. The actual assessment cannot be carried out until sufficient data is collected. However, when a robot has already acquired a great deal of data in these complex areas, the decision is whether to spend time doing a costly assessment.

The PAO* algorithm for planning with hidden state introduces the idea of *pinch points* [13]. Pinch points are areas of uncertain traversability that may significantly impact the optimal path. Here the option of performing a deeper assessment with a more capable method is not considered.

To the best of our knowledge, there are no path-planning frameworks that account for the cost of terrain assessment at the path-planning stage.

III. METHODOLOGY

In this paper we present our new path-planning framework called the Second Opinion Planner (SOP). The SOP framework has two major components: (i) a hierarchy of terrain-assessment methods, and (ii) the planning framework itself. In Section III-A we briefly discuss the assessment hierarchy. In Section III-B we present the theory of the SOP framework before discussing the SOP algorithm itself in Section III-C.

A. A Hierarchy of Terrain-Assessment Methods

We wish to construct a hierarchy of terrain-assessment methods. For illustrative purposes we use a two-level hierarchy, but in practice there can be a great number of terrain-assessment methods as long as they are ordered by increasing cost and fidelity. At the bottom is the low-cost, low-fidelity method. This method is applied to all the data and the output must be a cost graph. The assessment must be able to mark uncertain areas (areas where the method cannot ascertain if the terrain is traversable or not) and give the corresponding edges in the cost graph a probability of being an obstacle.

The highest-fidelity assessment method is at the top of the hierarchy and is the highest-cost method. The idea is for the planner to prudently use this method only in areas that: (i) are labeled uncertain by the low-fidelity method, and (ii) potentially lie on the optimal path. We assume that the assessment methods are consistent. For example, a high-fidelity assessment on an area deemed traversable by the low-fidelity method must also find the area to be traversable.

We can consider the extreme cases of how the two methods may be used. If the high-fidelity terrain assessment is never used, the result is a long path as we only allow paths through certainly traversable terrain. If the high-fidelity assessment is used on all the terrain the robot will spend a huge amount of time processing, and the shorter path will unlikely be worth the effort. Therefore we require a method to select where to use the high-fidelity terrain assessment.

B. Theoretical Foundations of the SOP

We have already established that not all terrain needs to be assessed with the same level of fidelity in order to determine if it can be successfully traversed by a mobile robot. This suggests the use of a framework that judiciously uses expensive techniques. The SOP is a novel, modular framework that uses various terrain-assessment methods and a path planner in an efficient package. SOP is also formulated to facilitate a probabilistic extension that is still under development.

We have used a graph-based planning paradigm, where an edge can be one of three types: *viable*, *uncertain* or *obstacle*. Viable edges can be traversed for a known cost, uncertain edges have an unknown cost (possibly infinite) and obstacle edges cannot be traversed, giving them an effective weight of infinity. We will omit obstacle edges from graphs and plots. The start state corresponds to the current location of the robot, while the goal state is the desired location.

1) Determining the Distance Matrices: Consider a *high-resolution graph*, $G_H := (V_H, E_H)$, that encodes traversability information for all the visible terrain based on the low-fidelity terrain-assessment results. The states, V_H, and the edges, E_H are defined as

$$V_H := V_L \cup V_{H-L}, \quad \text{and} \quad E_H := E_C \cup E_U, \qquad (1)$$

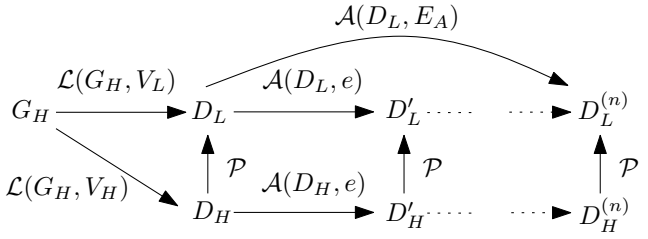

Fig. 2: A summary of the available operators, their commutativity and the option for batch updates.

where E_C is the set of edges with known weight, $w(e)$, and E_U is the set of edges with uncertain weight. These disjoint sets contain all the edges in the graph G_H. The *lumped states*, V_L, are the set of states consisting of the start, v_S, the goal, v_G, and all states touched by at least one edge in E_U. The remaining states, V_{H-L}, are the states of V_H not in V_L.

We can solve the all-pairs-shortest-paths (APSP) problem for G_H and store the results in D_H, which we will call the *high-resolution distance matrix*. A *distance matrix* contains the shortest-path weight between all states. The operation to create the distance matrix for the graph is represented by a *state-lumping operator*, denoted by $\mathcal{L}$, acting on the graph and the states that are to be included. We introduce another distance matrix, the *lumped-edge distance matrix*, D_L, which can be constructed in a similar manner to D_H, but starting with V_L instead of V_H:

$$D_H := \mathcal{L}(G_H, V_H), \quad \text{and} \quad D_L := \mathcal{L}(G_H, V_L). \tag{2}$$

Since $V_L \subseteq V_H$, all path-weights contained in D_L are also in D_H. This means that an equivalent way to find D_L is to take the relevant elements from D_H using a projection matrix P, which is determined based on the set of lumped states V_L. The $\mathcal{P}$ operator is introduced to represent this operation:

$$D_L = PD_HP^T = \mathcal{P}(D_H). \tag{3}$$

The *lumped-edge distance graph*, G_L, is just another way of representing D_L, where the cost of an edge connecting two states is given by the corresponding entry in D_L.

2) Assessing Edges of Uncertain Weight: Uncertain edges cannot be traversed and are therefore assigned a naïve weight of infinity, as would an edge representing an obstacle. However, the weights of uncertain edges may be determined by assessment with a higher-fidelity method. The assessment operator, $\mathcal{A}$, denotes the process of updating a distance matrix by reassessing edge $e(v_a, v_b)$ connecting states v_a and v_b. Therefore, D_H is updated as

$$D'_H = \mathcal{A}(D_H, e). \tag{4}$$

3) Updating the Lumped-Edge Distance Matrix: If the size difference between V_H and V_L is large, and it will be if the ratio of uncertain edges is relatively low, it is highly desirable to directly update D_L.

Theorem 1:: $\mathcal{P}$ and $\mathcal{A}$ commute, $\mathcal{P} \circ \mathcal{A} = \mathcal{A} \circ \mathcal{P}$, therefore D_L can be updated directly.

Proof: The weight being updated, $w'(e)$, by $\mathcal{A}$ always decreases but remains non-negative. The weight is associated with an edge, $e(v_a, v_b)$ that joins states v_a and v_b in V_L where $V_L \subseteq V_H$, so

$$D'_L = \mathcal{A}(D_L, w'(e)) = \mathcal{A}(\mathcal{P}(D_H), w'(e)). \tag{5}$$

Let $\delta(v_x, v_y)$ be the weight of the lowest cost path from v_x to v_y for every pair of vertices (v_x, v_y) in V_L so that,

$$\delta(v_x, v_a) + w'(v_a, v_b) + \delta(v_b, v_y) < \delta(v_a, v_b). \tag{6}$$

Then update the weight according to,

$$\delta'(v_x, v_y) = \delta(v_x, v_a) + w'(v_a, v_b) + \delta(v_b, v_y). \tag{7}$$

This guarantees D'_L represents the true lowest weight paths. Since taking the alternate approach yields

$$D'_L = \mathcal{P}(\mathcal{A}(D_H, (x, y))) \tag{8}$$

and also guarantees that lowest weight paths are represented, $\mathcal{P}$ and $\mathcal{A}$ commute. ■

The commutativity can be seen as part of Figure 2. It is also desirable to be able to directly update D_L with the results from multiple high-fidelity assessments of uncertain edge-weights. Let E_A be the set of n edges of the form $e(v_a, v_b)$, that are to be reassessed. Since only uncertain edges can be updated, $E_A \subseteq E_U$. Instead of updating D_L with each edge weight individually to get $D_L^{(n)}$, where superscript (n) denotes the n^{th} update, the $\mathcal{A}$ operation can directly produce $D_L^{(n)}$. This leads to the following theorem, whose proof is omitted due to space constraints.

Theorem 2:: Batch updates to D_L are possible, so multiple edge assessment results can be incorporated at one time.

4) Selecting Appropriate Edge Weights to Reassess: Potential path improvements can be detected by checking if the total path cost from start to goal is improved if uncertain edges are traversable. Uncertain edges are assigned a heuristic minimum possible cost, h, augmented by the cost of the next highest-fidelity assessment method, c_a. The distance matrix D_L, which contains the solution to the APSP problem is then updated using these new weights. If the path cost from start to goal in D'_L, is less than that in D_L then there are possible path improvements. We then state the following theorems, omitting proofs due to space constraints.

Theorem 3:: If after reassessment of edges contained in the path considered in $D'_L(v_S, v_G)$, the total path cost is equal to $D'_L(v_S, v_G) - \sum c_a$, then the shortest path from the start to goal has been found.

The order that uncertain cells are reassessed influences the number of assessments and therefore the cost of the path and terrain-assessment combination.

Theorem 4:: Assuming there is a potential improved path that uses n uncertain edges ($n > 1$), a minimum average path cost occurs when the next-highest-fidelity edge assessments are ordered according to

$$p_1 \geq p_2 \geq \cdots \geq p_{n-1} \geq p_n, \tag{9}$$

where p_i is the probability that the i^{th} edge, assessed at the next level of fidelity, is an obstacle.

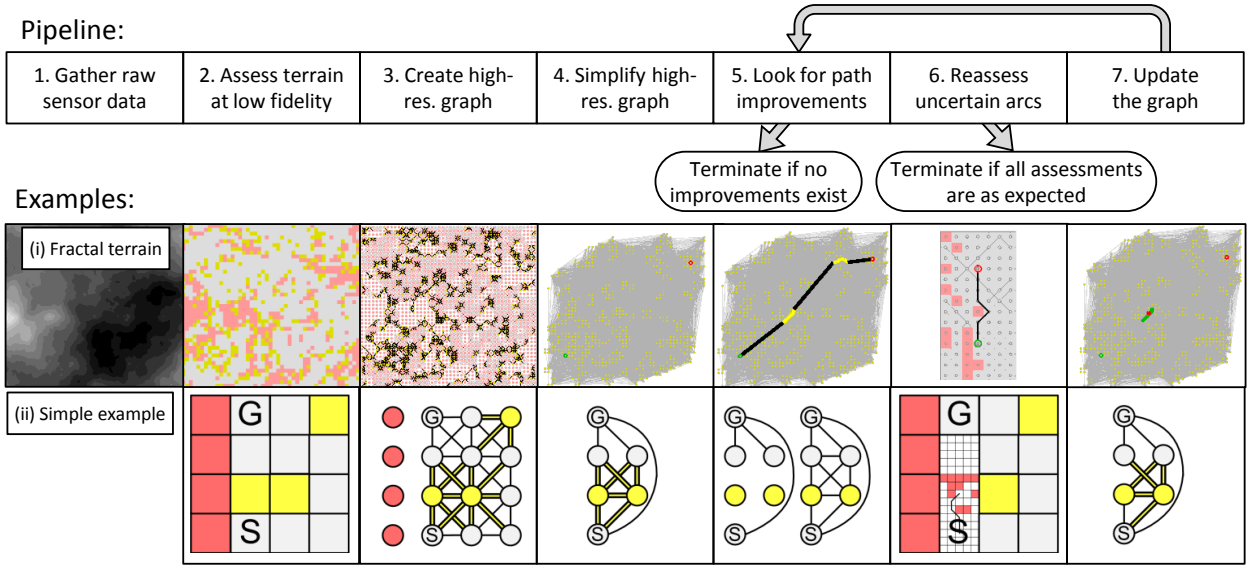

Fig. 3: The pipeline view of the Second Opinion Planner algorithm with examples showing the data product at each step: (i) on the top is an example using simulated fractal terrain (using the same terrain map as in Figure 4), and (ii) the bottom shows a simple case to better illustrate the steps of the algorithm. The start and goal states, corresponding to the current and desired positions of the robot, are shown as green and red circles or S and G, respectively. Gray areas are viable, yellow areas are uncertain, and red areas are obstacles. Each step is a major stage of the SOP algorithm.

C. The Optimistic Second Opinion Planner Algorithm

The pipeline of the algorithm is shown in Figure 3. There are two examples: (i) a case using simulated fractal terrain on the top, and (ii) on the bottom, a simple case used to more clearly show the steps. Step 1 of the pipeline is to gather raw sensor data. Step 2 is to use all the data to assess the terrain using the low-fidelity assessment method. The result is a map where all regions are either known or uncertain. The known areas are either viable (traversable) or an obstacle. Step 3 is to use the low-fidelity assessment results to create a high-resolution cost graph, G_H, where edges are viable or uncertain. Each uncertain edge also has probability, $p > 0$, of being an obstacle. For the viable edges $p = 0$. In this version of the framework, we discretize the terrain as regular, square cells such that G_H is an eight-connected graph.

Step 4 is to simplify the high-resolution graph, G_H, to create a graph that can be used to both speed up re-planning and to efficiently look for potential path improvements. The simplified graph is the lumped-edge distance graph, G_L, and it contains the subset of states connected by paths of uncertain cost, plus the start and goal. The edges between the states of G_L are the edges of uncertain cost and the edges representing the cost to travel between the states along edges of known cost in G_H. Part of this step is to find the shortest path from the start to the goal going through only viable edges and states, this is the *naïve path* and its cost is c_n.

A large G_L can greatly increase the computational effort required to run SOP. Therefore, at many points in the algorithm we take steps to keep G_L small. For example here, during the construction of G_L, we use c_n as an upper limit when deciding the states and edges to include. The minimum cost of using an edge must be less than c_n, and the minimum cost of using an edge from state v_a to v_b is

$$h(v_a, v_b) + c_a + h(v_S, v_a) + h(v_b, v_G), \tag{10}$$

where $h(v_a, v_b)$ is the heuristic minimum cost of traversing the edge, and c_a is the cost of terrain assessment at the next level of fidelity. The minimum possible cost of getting to v_a from the start state, v_S, is $h(v_S, v_a)$. Similarly, the minimum cost of traveling from the edge to the goal state, v_G is $h(v_b, v_G)$. We use the Euclidean distance between states v_x and v_y for the heuristic distance, $h(v_x, v_y)$. If a state other than the start or goal is not directly connected by an uncertain edge, it is not added to G_L. In this paper we assume a naïve path always exists, if this is not the case, G_L will contain more states and the computation will be more taxing.

Step 5 has the planner look for path improvements. To do this, consider the two extreme cases of what G_L may represent: (i) the *pessimistic* G_L where all the uncertain edges are not traversable, or (ii) the *optimistic* G_L where all the uncertain edges are traversable at the minimum heuristic cost plus the cost of carrying out the terrain assessment at the next highest level of fidelity. If the start-to-goal path cost in the optimistic G_L is less than the lowest cost start-to-goal path in the pessimistic G_L, then there are uncertain edges that may be part of the optimal path and terrain-assessment combination. If the start-to-goal path changes when the path is allowed to go through uncertain regions, there must be uncertain regions that are on the optimistic optimal path.

The pessimistic start-to-goal path cost is available in the pessimistic distance matrix as the entry corresponding to the start and goal states, $D_L(s, g)$. To find the optimistic path we can use the pessimistic G_L and the uncertain edges to find the start-to-goal entry of the optimistic D_L. Recall that D_L is a

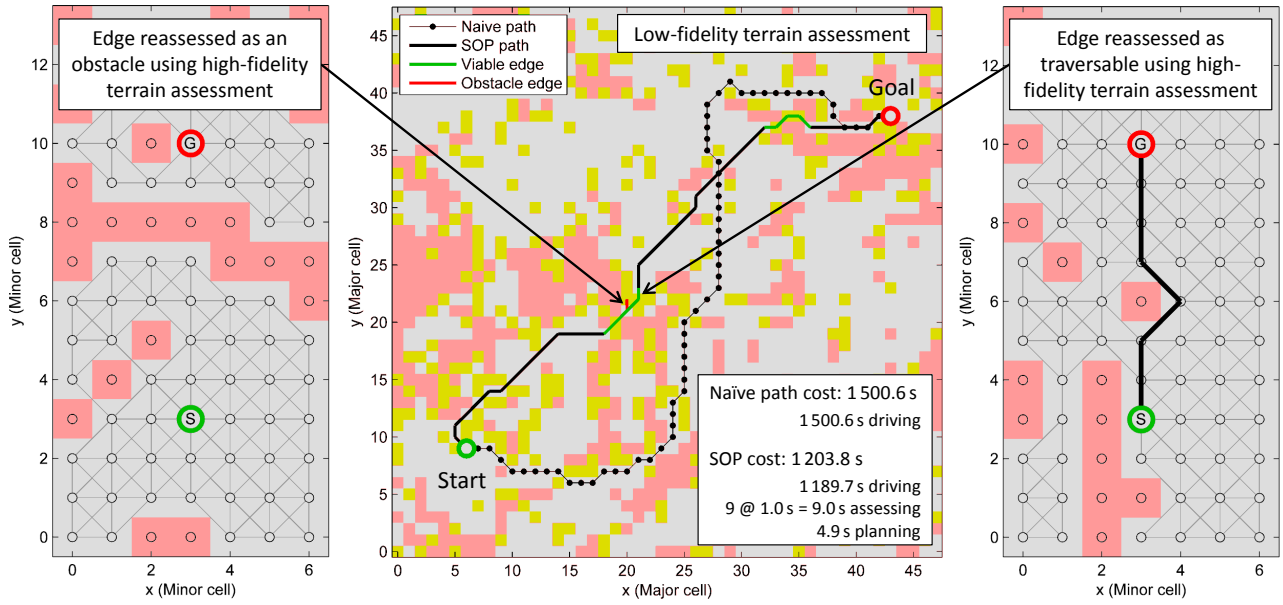

Fig. 4: A SOP plan on simulated fractal terrain. In the center is the low-fidelity terrain assessment with the resulting paths and high-fidelity assessment locations. The naïve path (dashed) goes through only viable regions (gray) while the SOP path (solid) crosses through areas that were uncertain (yellow) according to the low-fidelity assessment. High-fidelity assessments of edges that were found to be obstacles are red while viable edges are green. Sample high-fidelity assessments of an obstacle and traversable edge are shown on the left and right respectively. The true shortest path is the same as the final SOP path.

representation of the graph G_L, where the states correspond to the rows and columns of D_L, and the cost of the edge connecting state v_i to state v_j is the value of entry $D_L(i,j)$. The uncertain edges (with augmented cost) can be added to the pessimistic G_L and the planner can find a path from v_s to v_g on this new graph. The path is the optimistic start-to-goal path. If this is different than the pessimistic G_L path, there are uncertain edges that could yield path improvements.

Step 6 is to reassess the uncertain edges in the optimistic path (in order of decreasing p) using the terrain-assessment method at the next-highest level of fidelity. The result is that we now know the actual cost of the reassessed edge. If the reassessment result is that the edge is traversable (as hoped for in the optimistic D_L), then the shortest path has been found. If an edge is assessed as an obstacle, the assessments are immediately stopped, even if there are other assessments pending, as the optimistic path is now not traversable.

When necessary, step 7 is to update the graph G_L with the reassessed edge costs. We then return to step 5 and the planner can again look for further improvements. The planner will continue to seek a second opinion on pieces of terrain until the assessment costs outweigh the possible path benefit.

IV. RESULTS

A. *Tests on Simulated Fractal Terrain*

We generated fractal terrain (a planner test method similar to that of Stentz [5]) to test the performance of the algorithm and to compare it to other potential methods. Each generated terrain model is 1008×1008 pixels, with each pixel value corresponding to the local terrain height. The low-fidelity terrain-assessment method divides the terrain model into coarse, square cells that are 21 pixels per side. The traversability of a cell is estimated based on the roughness (variance of the height) of the pixels in the cell. There are two roughness thresholds: (i) a limit below which a cell is definitely traversable, and (ii) a limit above which the cell is an obstacle. If the roughness is between the two thresholds, the cell is deemed uncertain. The states in the high-resolution graph correspond to the cell centers. The edges, based on an eight-connected graph, are added if the associated states are not obstacles and if the difference between the mean heights is below the maximum step threshold. If both cells are viable, the edge is viable; otherwise, the edge is marked as uncertain. The probability that an edge is an obstacle is modeled using a function based on the roughness.

The high-fidelity assessment operates on an uncertain edge in the high-resolution graph. The pixels associated with the edge are grouped into square cells of three pixels per side, breaking the low-fidelity assessment cells into a grid of 7×7 smaller cells. A local cost graph is created using the same method as the low-fidelity assessment, except there is no uncertain option since the high-fidelity method is at the top of the assessment hierarchy. On the local cost graph we search for a path from the tail to the head of the uncertain edge. Recall that this is simply an example of a high-fidelity assessment method. Other examples include simulation with vehicle kinematics, or manual assessment.

Figure 4 shows a low-fidelity assessment with a detailed look at two examples of high-fidelity assessment. Also shown is a SOP path plan, which is an improvement over the naïve path plan. The SOP plan uses uncertain cells that required assessment using the high-fidelity assessment method.

TABLE I: Results for three cases corresponding to three different values of the cost of assessment, c_a. Case 1, 2 and 3 use high-fidelity assessment costs of 0.03, 3.5 and 100 s/assessment respectively. The methods considered are: (i) low-fidelity assessment used everywhere, (ii) high-fidelity assessment used on all uncertain edges, (iii) high-fidelity assessment applied to the uncertain edges in G_L, (iv) the cost of assessment is not considered during planning, (v) the Second Opinion Planner, and (vi) the best possible case which is a fortuitous lower bound created if high-fidelity assessment is only carried out on uncertain edges on the final path (this is not achievable on average). The table presents the average cost of each contribution and case over all 1 089 fractal terrain maps and gives one standard deviation.

	Path Length (pixels)	High-fidelity Assessments (#)	Planning Time (s)	Cost of Case 1 (s)	Cost of Case 2 (s)	Cost of Case 3 (s)
(i) Low-fidelity	$1\,253.2 \pm 177.3$		0.029 ± 0.0079	$1\,253.2 \pm 177.3$	$1\,253.2 \pm 177.3$	$1\,253.2 \pm 177.3$
(ii) High-fidelity-all-uncertain	$1\,146.9 \pm 111.3$	$1\,843.9 \pm 118.9$	0.031 ± 0.0091	$1\,202.2 \pm 110.9$	$7\,600.7 \pm 419.0$	$185\,540 \pm 11\,876$
(iii) High-fidelity-on-useful						
Case 1: $c_a = 0.03$ s	$1\,146.9 \pm 111.3$	893.0 ± 459.8	0.046 ± 0.012	$1\,173.9 \pm 120.4$		
Case 2: $c_a = 3.5$ s	$1\,146.9 \pm 111.3$	876.0 ± 468.9	0.046 ± 0.013		$4\,228.1 \pm 1\,717.1$	
Case 3: $c_a = 100$ s	$1\,146.9 \pm 111.3$	556.0 ± 639.8	0.045 ± 0.013			$56\,843 \pm 64\,074$
(iv) Ignoring cost of assessment	$1\,146.9 \pm 111.3$	15.4 ± 12.3	0.031 ± 0.0091	$1\,147.4 \pm 111.5$	$1\,200.8 \pm 135.0$	$2\,685.9 \pm 1\,279.3$
(v) Second Opinion Planner						
Case 1: $c_a = 0.03$ s	$1\,146.9 \pm 111.3$	8.7 ± 7.2	4.7 ± 4.6	$1\,151.5 \pm 113.5$		
Case 2: $c_a = 3.5$ s	$1\,147.0 \pm 111.4$	7.1 ± 5.6	4.6 ± 4.6		$1\,173.2 \pm 120.1$	
Case 3: $c_a = 100$ s	$1\,220.8 \pm 142.2$	0.29 ± 0.94	2.2 ± 3.4			$1\,249.1 \pm 169.7$
(vi) Best possible						
Case 1: $c_a = 0.03$ s	$1\,146.9 \pm 111.3$	4.8 ± 2.9		$1\,147.1 \pm 111.3$		
Case 2: $c_a = 3.5$ s	$1\,147.0 \pm 111.4$	4.4 ± 2.8			$1\,162.1 \pm 113.3$	
Case 3: $c_a = 100$ s	$1\,220.8 \pm 142.2$	0.21 ± 0.69				$1\,242.1 \pm 152.9$

The start and goal were placed in locations similar to those shown in the sample, and if there was not a naïve path (i.e., all certainly traversable edges from the start to the goal) the map was not used. We ran the SOP algorithm on 1089 fractal terrain maps and compared the results with the some alternative options that are described later. The results are summarized in Table I, it provides the average and standard deviation of the costs across all the maps for three different values for the cost of high-fidelity terrain assessment. The total cost is broken down into the three main components: (i) path length, (ii) number of high-fidelity assessments, and (iii) planning time. We also provide a total-time cost that combines the three components by assuming a driving speed and time for each high-fidelity assessment. The speed of the robot is 1 pixel/s and three cases are considered with different costs of assessment, c_a. The cases use c_a as: (1) 0.03 s/assessment, (2) 3.5 s/assessment, and (3) 100 s/assessment. The cost has a dramatic influence on the choices made by the SOP algorithm. The tests were carried out on one core of an Intel®Core™2 Duo 2.4 GHz processor with 3 GB of RAM. The average distance between the start and the goal is 959 pixels with a standard deviation of 32.6 pixels.

In Table I, the *low-fidelity* assessment case employed the low-fidelity assessment method everywhere and plans on the resulting graph without allowing the path to traverse uncertain edges (i.e., the path is the naïve path). As we would expect, the path length is relatively long and is the major factor in the total cost. The *high-fidelity-all-uncertain* assessment case begins with the same graph, but then uses the high-fidelity assessment method on all the uncertain edges. The result is that the path taken is the shortest possible path, but there is a huge cost associated with carrying out all the high-fidelity assessments. In the *high-fidelity-on-useful* case, the high-fidelity assessment is only carried out on the smaller set of uncertain edges included in G_L. The result is again the shortest possible path, but with fewer high-fidelity assessments than the high-fidelity-all-uncertain case. The number of assessments varies with c_a because c_a is used when selecting the states and edges to include in D_L. For *ignoring the cost of assessment* the cost of high-fidelity terrain assessment is neglected during planning. Therefore the planner will plan along uncertain edges if that is the shortest path. There is no consideration for the eventual cost of assessment that must be incurred; this method is most similar to the TANav framework [10] mentioned previously. In all these cases the cost of planning is negligible.

SOP, on average, finds the lowest cost combination of path and terrain assessment, or is quite close for all cases. It is only slightly more costly at low c_a due to the SOP planning overhead. Of course, if the high-fidelity terrain assessment were this cheap it would be used as the low-fidelity method, allowing for even more capable methods higher in the assessment hierarchy. The SOP costs are 1151.5 s, 1173.2 s, and 1249.1 s, this corresponds to improvements of 101.7 s (8.12%), 80.0 s (6.38%), and 4.1 s (0.33%) for cases 1, 2 and 3 respectively. SOP uses the shortest path most of the time and uses very few high-fidelity assessments, though clearly to good effect. The cost of planning is small but not insignificant. SOP offers an advantage over the other options shown, finding what is quite close to the minimum possible path/assessment combination. The minimum possible path/assessment combination would be to only assess uncertain edges that are on the final path, but it is impossible to know in advance if an edge will turn out to be an obstacle or traversable after high-fidelity assessment.

Looking critically at the results there are some opportunities for improvement. The planning time can be reduced with a more efficient implementation and is dependent on the computing power. The number of assessments may be

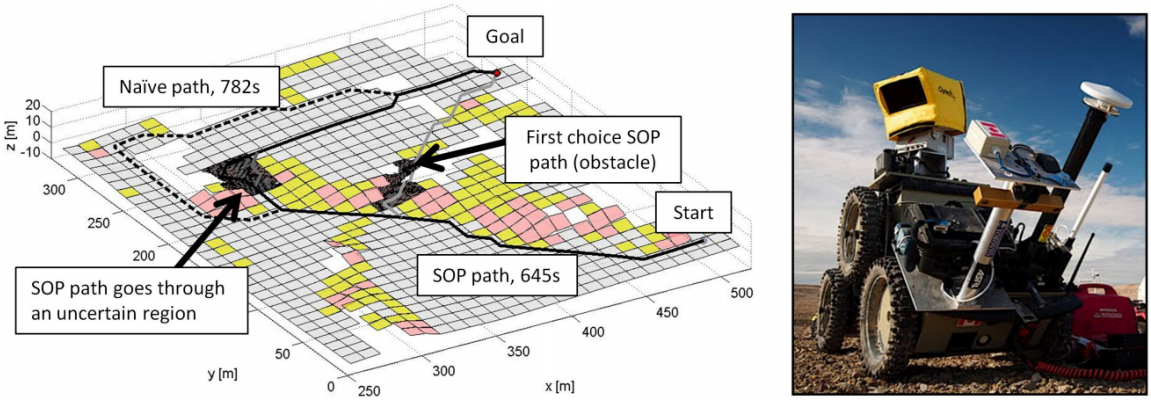

Fig. 5: A sample SOP plan at a planetary analogue site is shown on the left. A long-range point cloud was collected using an Optech ILRIS, shown mounted on the test robot at the right. Plane-fit terrain assessment was used for the low-fidelity assessment. The gray, yellow and red planes respectively show traversable, uncertain and obstacle regions of the terrain. The naïve (dashed) and SOP (solid) plans are shown. The initial reassessments requested by SOP yielded obstacles so an alternate path was proposed, assessed, and found to be traversable. Times are the total of all cost contributions.

also be reduced by moving away from an optimistic model and making better use of the probability that an edge is an obstacle. In contrast, the path length cannot be appreciably reduced since it is already quite close to the minimum. Further improvements are going to be relatively small, and in practice, only noticeable on large maps where the cost of high-fidelity terrain assessment is large (but not too large) relative to the driving speed. We have been considering a probabilistic extension to the SOP but further development may be at a point of diminishing returns.

B. SOP at a Planetary Analogue Site

We have used SOP on real data collected at a planetary analogue site. In Figure 5 we can see the terrain assessment and SOP plan using a long-range LIDAR scan from an area near the Haughton Crater, Devon Island, Nunavut, Canada. The low-fidelity assessment used 10×10 m cells and the high-fidelity method is the same as the short-range guidance system on the robot shown in Figure 1. The purpose of the field trials was to validate an assessment hierarchy. We carried out nearly 10 km of autonomous driving and have collected over 40 long-range LIDAR scans to achieve this objective. The initial results are promising and further testing using this data is ongoing.

V. CONCLUSIONS AND FUTURE WORK

We have presented a novel path-planning framework that considers the cost of terrain assessment in the planning process. It attempts to limit the use of high-fidelity terrain assessments to areas that can result in a shorter path (including the cost of assessment). We presented the theory behind the framework and results of using our implementation, the optimistic Second Opinion Planner, on simulated fractal terrain. The SOP plans are quite close to the minimum possible cost, though some improvements are possible. We are working on more efficient implementations of the algorithm and also a probabilistic version that uses the probability that an edge is not traversable to better model the actual driving costs before carrying out a costly high-fidelity terrain assessment.

VI. ACKNOWLEDGMENTS

The authors gratefully acknowledge the funding support from the Canada Foundation for Innovation and the Natural Sciences and Engineering Research Council of Canada; and the logistics support from the Canadian Space Agency and the Mars Institute. This support made our field trials on Devon Island possible.

REFERENCES

[1] R. Volpe. Rover Functional Autonomy Development for the Mars Mobile Science Laboratory. In *Proceedings of the 2003 IEEE Aerospace Conference*, volume Vol. 2-643, 2003.

[2] J.J. Biesiadecki, P.C. Leger, and M.W. Maimone. Tradeoffs Between Directed and Autonomous Driving on the Mars Exploration Rovers. *International Journal of Robotics Research*, 26(1):91–104, 2007.

[3] J.J. Biesiadecki and M.W. Maimone. The Mars Exploration Rover Surface Mobility Flight Software Driving Ambition. In *2006 IEEE Aerospace Conference*, Big Sky, MT, USA, 2006.

[4] P.E. Hart, N.J. Nilsson, and B. Raphael. A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions of Systems Science and Cybernetics*, SSC-4:100–107, 1968.

[5] A. Stentz. Optimal and Efficient Path Planning for Unknown and Dynamic Environments. Technical Report CMU-RI-TR-93-20, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 1993.

[6] S. Koenig. Fast Replanning for Navigation in Unknown Terrain. *IEEE Transactions on Robotics*, 21(3):354–363, 2005.

[7] J. Carsten, A. Rankin, D. Ferguson, and A. Stentz. Global Path Planning On Board the Mars Exploration Rovers. In *IEEE Aerospace Conference*, pages 1–11, Big Sky, MT, March 2007.

[8] J.J. Biesiadecki, M.W. Maimone, and P.C. Leger. Overview of the Mars Exploration Rover Autonomous Mobility and Vision Capabilities. In *IEEE International Conference on Robotics and Automation*, 2007.

[9] 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*, 26(4):337–357, 2009.

[10] D. Helmick, A. Angelova, and L. Matthies. Terrain Adaptive Navigation for Planetary Rovers. *Journal of Field Robotics*, 26(4):391–410, 2009.

[11] S. Kambhampati and L.S. Davis. Multiresolution Path Planning for Mobile Robots. *IEEE J. Robot. Autom.*, RA-2(3), pages 135–145, 1986.

[12] B. Nabbe and M. Hebert. Extending the Path-Planning Horizon. *The International Journal of Robotics Research*, 26(10):997–1024, 2007.

[13] D. Ferguson, A. Stentz, and S. Thrun. PAO* for Planning with Hidden State. *IEEE Intl. Conf. on Robot. and Autom.*, pages 2840–2847, 2004.