Exploration Planning for Structural Inspection

LUKE YODER, Carnegie Mellon University

In this paper, I extend traditional frontier based exploration planning to 3D environments. In addition, I propose an integrated exploration exploitation strategy that trades off frontier based exploration planning with prediction of unseen structure and optimal path planning. Considering highly repetitious architectural structures, I propose exploiting observed structure to predict unobserved structure, using these predictions to plan and execute locally optimal paths. Unfortunately, I found that my proposed technique performed equal to or worse than the greedy exploration planning in terms of path cost.

1. INTRODUCTION

Large structures like bridges must be inspected periodically for safety. Currently, inspections are done manually—a dull, dirty, dangerous, and expensive job. In this paper I propose to partially automate structural inspection with a small flying vehicle. The proposed vehicle would autonomously build a complete map of a structure, storing images and geometric data of every structural component for offline analysis by engineers. Automating the data acquisition associated with structural inspection has two benefits. First, the efficiency and safety of inspection teams would improve by removing workers form dull, dirty, dangerous tasks. Second, the amount and consistency of inspection data archived would increase—providing engineers with better tools for insuring public safety.

The contribution of this paper is an exploration planning approach for completely observing an a priori unknown structure with a flying vehicle. Real world inspection of unknown environments is a challenging problem. To build an accurate map of an environment, a robot must visit all locations within the environment necessary for it to observe every surface. In two dimensions this problem has been solved in a range of environments. In 3D, however, large scale real time exploration planning is still an open research problem. In this paper, I borrow known exploration strategies and extend these strategies in a simple 3D simulation. My approach involves two complementary algorithms:

- Frontier based exploration path planning
- Prediction of unseen structure based on observed structure, followed by optimal local path planning

These two algorithms can be thought of as exploration and exploitation, where a robot begins by exploring a structure, then exploits what is has learned about a structure to choose a shortest path required as it continues to observe the structure.

2. RELATED WORK

There is a large body of related work surrounding frontier based exploration planning. This work attempts to answer the question: without prior knowledge of a structure, how can a shortest possible path be planned incrementally to completely observe an environment? This question has been answered numerous times with respect to a range of applications.

Traditional exploration techniques were pioneered by a frontier based exploration path planner proposed by Yamauchi (1997). Later, frontier based exploration was extended to include entropy and information gain approaches. All of these approaches select exploration goals that minimize either map uncertainty or path cost to the next frontier. Traditional frontier based exploration techniques

Graduate Artificial Intelligence
Final Project Report
5/5/2013

rely on dense occupancy grid maps that record all known free, known occupied, and unknown cells in the map. Because of this, naive extensions from 2D to 3D are computationally expensive. One such extension is proposed by Dornhege (2011). This approach has not been proven to run in real time on a computationally limited robot.

Next best view (NBV) algorithms were developed for digitizing objects. This family of algorithms tries to answer the question: where should the sensor move next to maximize information gain? NBV algorithms have been extended for use with mobile robots, as described Foissotte (2008) and Torabi (2011); however their computational expense also makes them impractical for real-time use with mobile robots in large environments.

Less complex exploration algorithms have been implemented on computationally limited micro aerial vehicles, namely by Shen (2012). These results are promising for exploring indoor environments. Because they focus on expanding free space instead of surface coverage, however, they are not effective at exploring large convex architectural structures.

There have been no proposed methods for predicting unobserved environment during exploration. This is likely explained by the fact that very few environments have enough repetition to consistently make correct predictions. In these environments a low prediction success rate may not justify complex and computationally expensive prediction algorithms.

3. EXPLORATION

Exploration planning answers the question: where should the robot move next if the goal is to build a complete map? Although there are many techniques, simply moving towards the closest occupied cell that borders an unknown cell (frontier) yields a short overall path at minimal computational cost. Compared to other techniques, this greedy approach is shown to produce an order of magnitude shorter paths in some environments, as shown by Juli´a (2012). Because this approach is the most effective at finding the shortest path, it is used exclusively throughout the rest of this paper.

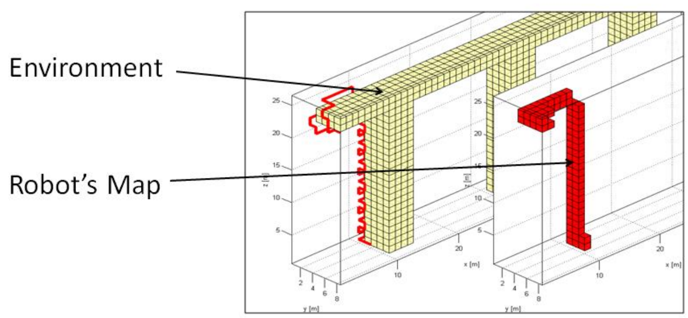

Fig. 1. A visualization of a partially observable environment. The foreground is a map built incrementally by a robot observing the environment shown in the background.

The major shortcoming of existing 3D frontier based exploration techniques is the high computational cost. This computational cost is partially due to the dense occupancy grid representation of the world. This dense representation is required because many algorithms require unknown space in the world to be defined as either occupied or free before exploration terminates.

When exploring architectural structures one can assume that the structure is fully connected—there are no floating bodies. This means that if we define a volume in space to be searched, it is unnecessary to observe all free space in the search volume. For example, in the extreme case that the search volume is void of all objects, the robot only needs to search the edges of the search volume to prove that the world is empty. In this paper, I propose defining frontiers as unknown cells bordering known, occupied cells. This simple change means that the algorithm can use a sparse occupancy grid that only records occupied cells and free cells that border the occupied cells. Then, traditional closest frontier exploration is applied, computing the paths to a frontier by running Dijkstra's algorithm starting from the robot's position. Dijkstra's algorithm terminates when the first frontier is found, and this shortest path to a frontier is sent to the motion planner.

An example environment and map (with free space hidden) are shown in Fig. 1.

4. PREDICTION AND EXPLOITATION

Nearly all architectural structures have repetition. My algorithm takes advantage of this repetition in an attempt to reduce overall path length. The algorithm predicts unobserved structure based on segments of the observed structure. Then, a locally optimal path is planned around the prediction by solving a version of the traveling salesperson algorithm (TSP).

My approach always begins by exploring the environment for a given distance. After this initialization period, the algorithm samples features from the partially complete occupancy grid map. Sampled features are centered at occupied cells, and include cells in a fixed volume around the occupied cell. Features do not only include occupied cells in the fixed volume, but also include free cells that will later be used to compute a locally optimal path. After sampling a fixed number of features, the algorithm slides the features through the occupancy grid looking for a match. The goal of this step is to select a repeating feature then match it to a region of the map that is currently being explored. Each feature is matched to the map and the match quality Q is computed using

$$R = \ 0.5 - abs\left(\frac{\sum map\ region \cap feature}{\sum feature} - 0.5\right) \tag{1}$$

Equation 1 is maximum for features that have 50% overlap with the current map, allowing for the selection of predictions that have a high confidence but also provide a large prediction.

After choosing the best prediction, a near optimal path is generated using the opt-2 version of the traveling salesperson problem created by Croes (1958). The traveling salesperson problem is computed over the observation locations that were saved with chosen feature, excluding any observation locations that have already been visited. The robot's current position is included in the goal locations. The output of the TSP algorithm is not necessarily a feasible path, but it is executed by the motion planner until a difference between the prediction and reality is observed.

It is important to have a strategy for trading off between exploration and exploitation. If the prediction portion of exploitation fails due to insufficient map data, exploration resumes for a set distance. If unpredicted occupied cells are observed when carrying out the locally optimal path, exploration resumes for a set distance. In this way, the robot constantly trades off between exploration and exploitation until the map is complete. Although the exploitation algorithm is not complete, frontier based exploration planning is complete. Because of the tradeoffs described, the overall algorithm is therefore complete.

5. IMPLEMENTATION

To test my exploration planning approach I implemented a simple simulator. This simulator provides the robot with perfectly known pose and perfect sensor measurements within a set radius of the robot's position. The simulator handles ray tracing so that the robot cannot observe occluded cells and so that the robot can define cells between it and occupied cells as free. A simple bridge-like structure was created in the world map, while a separate map is created incrementally as the robot explores the world. A simplified diagram of my exploration planning approach, as described in the previous sections, is shown in Fig. 2.

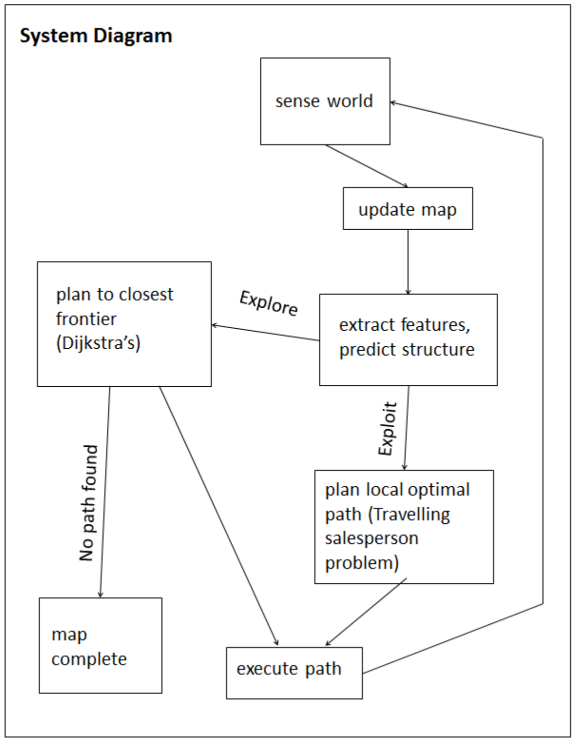

Fig. 2. This system diagram depicts my approach at a high level. The algorithm either chooses to explore the environment or exploit the knowledge it has recorded during exploration.

6. RESULTS

Despite its simplicity, the frontier exploration technique always found an equal or shorter path when compared to the exploitation algorithm. Fig. 3 compares total path length for three techniques that were tested. Fig 4, 5, and 6 show paths planned by the near optimal, exploration, and exploration exploitation algorithms, respectively. The exploration exploitation path shown in Fig 6 clearly shows an incorrect prediction inefficiency just left of center, an efficient path similar to the optimal path just right of center, and a backtracking inefficiency that occurred due to a series of locally optimal paths missing several cells. The optimal local planning strategy suffered from several inefficiencies:

1) It missed small groups of cells that had to be returned to later on
2) Predictions were occasionally wrong, especially when little data had been accumulated from exploration.
3) Traveling from one locally optimal path to another often increased path length but did not yield new observations

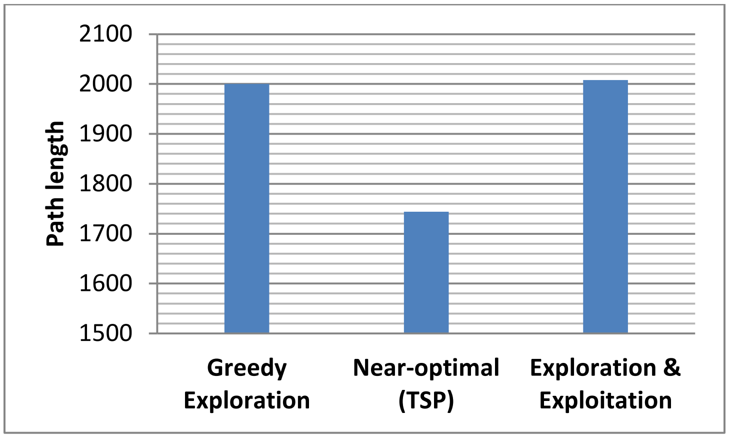

Fig. 3. The results for the two approaches implemented compared to the optimal path computed with full domain knowledge

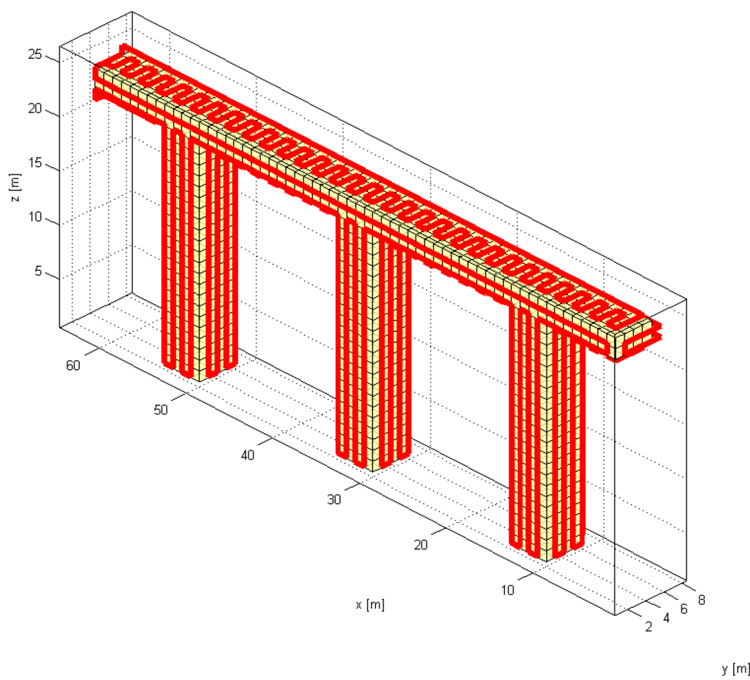

Fig. 4. The near-optimal path computed with a TSP algorithm

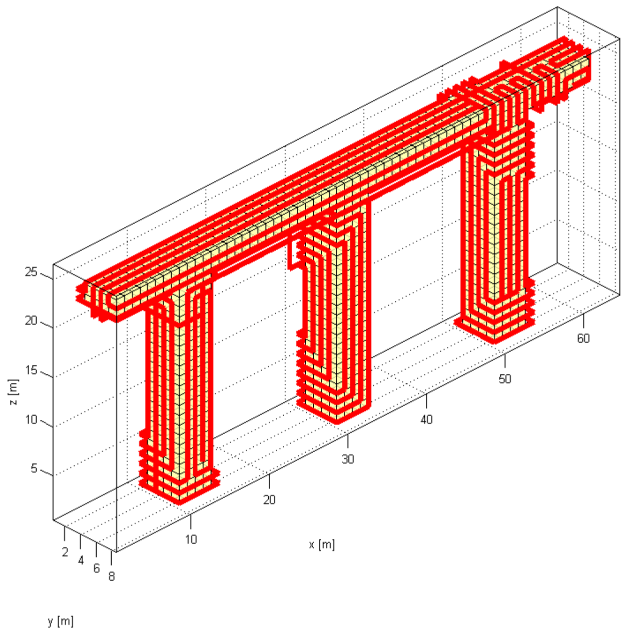

Fig. 5. The path incrementally found using the nearest frontier exploration algorithm

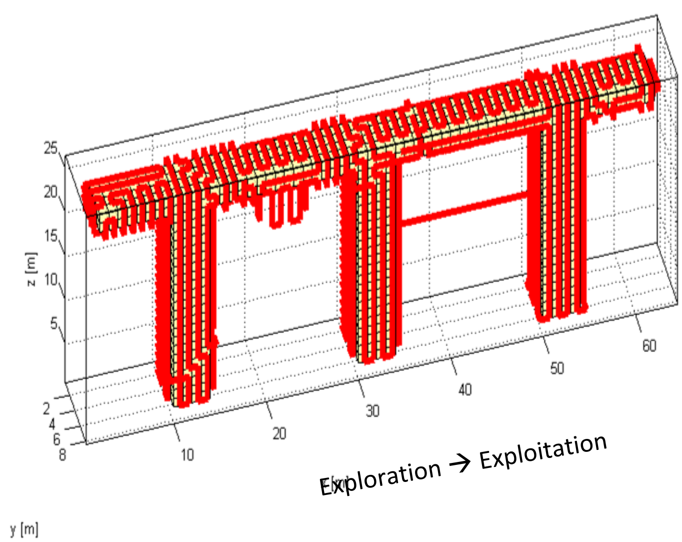

Fig. 6. The path found by trading off between exploration and exploitation. Note the incorrect prediction inefficiency just left of center, the efficient path similar to the optimal path just right of center, and the backtracking inefficiency that occurred due to a series of locally optimal paths missing a small surface.

7. FUTURE WORK

The goal of combining exploration with exploitation was to create a planner in a partially observable world that over time converged to the optimal path length possible with full domain knowledge. I expect that these results are still achievable; however improved techniques for feature extraction and transitioning between locally optimal paths are needed.

REFERENCES

G. Croes. A method for solving traveling salesman problems. Operations Res. 6. 1958. pp., 791-812.

Cyrill Stachniss. Robotic Mapping and Exploration. Springer Tracts in Advanced Robotics, Vol. 55, 2009.

Dunn E, Van den Berg J and Frahm J-M (2009) Developing visual sensing strategies through next best view planning. In: Proceedings of the IEEE/RSJ international conference on intelligent robots and systems, St Louis, MO, October 2009, pp.4001-4008

Dornhege, C.; Kleiner, A. A frontier-void-based approach for autonomous exploration in 3d, Safety, Security, and Rescue Robotics (SSRR). 2011 IEEE International Symposium on , vol., no., pp.351,356, 1-5 Nov. 2011

Foissotte, T.; Stasse, O.; Escande, A.; Kheddar, A. A next-best-view algorithm for autonomous 3D object modeling by a humanoid robot, Humanoid Robots, 2008. Humanoids 2008. 8th IEEE-RAS International Conference on , vol., no., pp.333,338, 1-3 Dec. 2008

Juli´a, M.; Gil, A.; Reinoso, O. A comparison of path planning strategies for autonomous exploration and mapping of unknown environments. Auton. Robot. 2012, 33, 427–444

Liila Torabi, Kamal Gupta. An autonomous six-DOF eye-in-hand system for in situ 3D object modeling. The International Journal of Robotics Research. January 2012 31: 82-100, Oct. 31, 2011

Shade, Robbie, and Paul Newman. Choosing where to go: Complete 3D exploration with stereo. 2011 IEEE International Conference on Robotics and Automation 2011 2806-2811.

S. Shen, N. Michael, and V. Kumar. A stochastic differential equation-based exploration algorithm for autonomous indoor 3D exploration with a micro-aerial vehicle. Intl. J. Robot. Research, 31(12):1431–1444, Oct. 2012

Yamauchi B (1997) A frontier-based approach for autonomous exploration. In Proceedings of IEEE Symposium on Computational Intelligence in Robotics and Automation, Monterey, CA, pp. 146–151.