Sparse Planning Graphs for Information Driven Exploration

Erik Nelson, Vishnu Desaraju, John Yao
The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15217
{enelson, rajeswar, johnyao}@cmu.edu

***Abstract*—. . .**

I. Introduction

Exploration is a key capability that enables robotic vehicles to operate in unknown environments. In this project we develop an active perception policy for robotic exploration. Active perception exploration formulations choose control actions which optimize an information-theoretic objective function such as Shannon's mutual information or entropy [**?**], [**?**] over the robot's map, given a sensor measurement model. Other common exploration techniques, such as Frontier exploration [], use geometric reasoning to infer explorative paths. While these strategies work well in practice, they operate on a maximum likelihood estimate of the map, and apply heuristics to determine the most uncertain locations in the environment. In contrast, active perception strategies do not utilize geometric or maximum likelihood assumptions, and instead interpret the map as a binary random variable, choosing actions which directly minimize the random variable's uncertainty. Julian et al. prove that maximizing mutual information between a robot's map and expected future map naturally yields explorative behaviors [**?**].

Active perception formulations seek to optimize information-theoretic objectives. While this optimization is real-time for short planning horizons, these metrics are often expensive to compute online, requiring double integration over possible future robot states and measurements, or Monte Carlo sampling from the distribution of measurements. These expensive per-pose computations inhibit online dense evaluation over a configuration space. In this project, we aim to develop an efficient active perception exploration strategy which evaluates the information-theoretic objective in a sparse, but well-chosen set of poses across the configuration space. This strategy evaluation the objective function a limited number of times, while still generating paths that sufficiently explore the space.

To achieve real-time active perception exploration, we use a Rapidly-Exploring Random Tree (RRT) to generate sets of dynamically feasible actions over a finite planning horizon [**?**]. RRT planners trade trajectory optimality for efficiency, allowing for evaluation of many potential future locations in the configuration space during a single planning step. In addition, RRT planners are anytime, and generate potential trajectories for a pre-specified amount of time before evaluating the most optimal sampled trajectory. Our strategy evaluates each RRT leaf-node using the information-theoretic objective function, and stores the resulting reward in the tree. After planning for a specified amount of time, the maximum reward leaf-node is chosen as the optimal location to visit, and the RRT is traversed to generate a dynamically feasible trajectory to that location.

In addition to the efficiency gains from using an RRT, a recent work by Charrow et al. [**?**] has proposed the Cauchy-Schwarz Quadratic Mutual Information (CSQMI) as an efficient information-theoretic objective function. CSQMI is theoretically well-motivated: it is derived from Renyi's Quadratic Entropy, a generalization of Shannon's entropy. However, in contrast to Shannon's mutual information (which is derived directly from Shannon's entropy), CSQMI is shown to have superior computational efficiency.

The contribution of this work is an exploration framework to enable online exploration using CSQMI in sparse planning graphs, such as RRTs. We demonstrate an implementation of our approach in simulation, and provide analysis of experiments in which a mobile ground robot must explore an unknown space using a laser scanner range sensor. We discuss the formulation and implementation of the CSQMI metric, RRT, and a controller and Unscented Kalman Filter (UKF) that were developed to enable trajectory tracking and state estimation. Finally, we discuss the implementation of these capabilities on a real ground robot.

This paper is structured in the following manner: Section II gives a brief overview of occupancy grid mapping which is necessary for the CSQMI information metric. Section III details the CSQMI information-theoretic cost objective and its similarities to Shannon's mutual information. Section IV discusses the measurement model that was chosen to evaluate expected future measurements. Sections V and VI cover the RRT and UKF formulations used in our implementation. Finally, Sections VII and VIII give results and analysis of our implementation in simulation, and describe future work towards implementing our algorithms on a ground robot.

II. Occupancy Grid Mapping

In order to develop an information-theoretic reward surface, we model the robot's map as a binary random variable. We therefore discretize the space, and represent the map as an occupancy grid - a common environmental representation for robotic mapping.

As a basis for the core formulation in the following sections, we provide a brief overview of occupancy grid mapping. Occupancy grids are a common and useful We represent the map as an occupancy grid, which consists of a set of cells: $m = \{m^i\}_{i=1}^{N}$. The probability that an individual cell is occupied is given by $p\left(m^i \mid x_{1:t}, z_{1:t}\right)$, where $x_{1:t}$ denotes the history of states of the vehicle, and $z_{1:t}$ denotes the history of range observations accumulated by the vehicle. We assume that cell occupancy probabilities are independent of one another: $p\left(m \mid x_{1:t}, z_{1:t}\right) = \prod_i p\left(m^i \mid x_{1:t}, z_{1:t}\right)$. For notational simplicity we write the map conditioned on random variables $x_{1:t}$ and $z_{1:t}$ as $p_t\left(m\right) := p\left(m \mid x_{1:t}, z_{1:t}\right)$. Additionally, unobserved grid cells are assigned a uniform prior of being occupied.

We represent the occupancy status of grid cell m^i at time t with a log odds expression

$$l_t := \log \frac{p\left(m^i \mid z_{1:t}\right)}{p\left(\bar{m}^i \mid z_{1:t}\right)} \tag{1}$$

where $\bar{m}^i$ denotes the probability that m^i is unoccupied. When a new observation z_t is obtained, the log odds update is given by

$$l_t = l_{t-1} + \log \frac{p\left(m^i \mid z_t\right)}{p\left(m^i\right)} - \log \frac{p\left(\bar{m}^i \mid z_t\right)}{p\left(\bar{m}^i\right)} \tag{2}$$

where the last two terms represent the inverse sensor model.

III. Information-Theoretic Objective

IV. Measurement Model

V. Planning with RRT

To make use of this information cost function to guide exploration, we consider a sampling-based planning approach that can evaluate the predicted information gain throughout the environment. In addition, we wish to use the occupancy grid that is being updated online (as described in Sect. **??**) to guide the vehicle around obstacles in the environment. The rapidly-exploring random tree (RRT) algorithm is well suited to planning paths through these types of large environments, and works as follows. The planner starts from the vehicle's current state and samples a point x in the environment. Using the occupancy grid, we can reject samples that lie in cells with a sufficiently high probability of being occupied. If the sample is valid, we find the closest node in the tree of paths (initially just the vehicle state), where closeness is measured in terms of Euclidean distance, and add a new edge to the tree connecting the sample point to the nearest node. Then a new sample is drawn and the process repeats to grow a tree of path segments through the environment. This tree-growing process terminates after a specified time, and the minimum cost path is returned.

Figure **??** shows snapshots of the system planning through the environment while updating the occupancy grid. The edges in the tree are smooth since they are generated by forward simulating the closed-loop vehicle dynamics toward the sample point, resulting in a variant of the RRT algorithm known as Closed-loop RRT (CL-RRT) [**?**]. This approach is traditionally used to ensure dynamic feasibility and dense collision checking. However, the forward simulation also means we have full state information for the system at the end of each segment. This allows us to evaluate the predicted information gain at that point and assign a corresponding cost to candidate trajectory.

To change CL-RRT from a goal-directed planner to an exploration-driven planner, we first define the sampling distribution to be a Gaussian centered about the root of the tree, with no bias toward any direction (unlike standard sampling-based planners that will sample the goal some small probability). We also define the cost of each branch segment to just be the information metric computed at its endpoint (as opposed to a more traditional setup where the cost is the total distance traveled from the root plus a cost to go based on an admissible heuristic, such as Euclidean distance to a goal). Finally, since there is no goal to guide the selection of the best branch from the tree, we simply select the branch with the minimum cost endpoint in the entire tree. This enables the planner to compute paths that aim to maximize the predicted information gain.

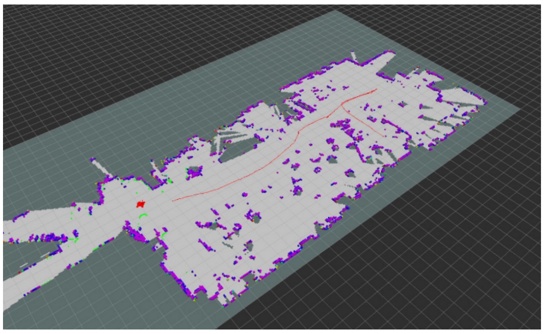
Algorithm 1 CL-RRT: Tree Expansion

Sample point x_sample from the environment State Identify nearest node N_near in tree
$k \leftarrow 0$
$\hat{x}(t+k) \leftarrow$ last state of N_near
while $\hat{x}(t+k) \in \mathcal{X}_free(t+k)$ and $\hat{x}(t+k)$ has not reached x_sample **do**
  Compute reference input $\hat{r}(t+k)$ from x_sample
  Compute control input $\hat{u}(t+k)$ from control law
  Compute next state $\hat{x}(t+k+1)$ from propagation model
  $k \leftarrow k+1$
end while$N \leftarrow r_final$
for each feasible node N produced **do**
  Update cost estimates for N
  Add N to tree
end for

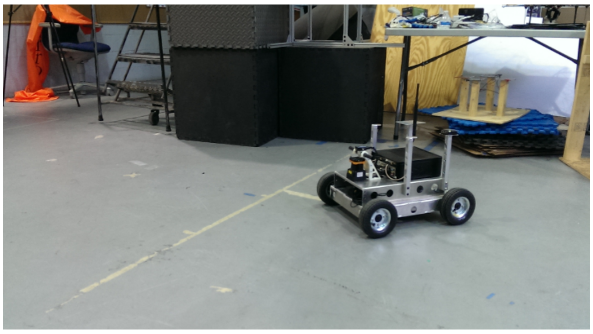
VI. Unscented Kalman Filter

VII. Results

VIII. Conclusion and Future Work



(a) Map and exploration trajectory



(b) Ground robot

Fig. 1: A ground robot exploring and mapping a cluttered environment.