Title Goes Here

Erik Arthur Nelson
April 22, 2015

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, Pennsylvania

Thesis Committee:
Nathan Michael, *Chair*
Artur Dubrawski
Third person

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Robotics.

Abstract

Acknowledgements

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Robots are emerging from controlled factories and laboratories into our homes, workplaces, roads, and public airspaces. Alongside their transition into these unstructured and transient environments comes their need to be able to explore, characterize, and catalog their surroundings. Mobile robot autonomy is generally accomplished by referring to a map - a 2D or 3D probabilistic representation of the locations of obstacles in the robot's workspace. With access to a map, robots can localize to determine their position, plan collision-free trajectories to goals, locate objects for interaction, and make decisions by reasoning about the geometry and dynamics of the world. Given that a robot's map is of critical importance for most autonomy tasks, robots that find themselves initialized without a priori access to a map should be capable of autonomously, efficiently, and intelligently creating one.

The exercise of choosing and executing actions that lead a robot to learn more about its own map is known as *active perception* or *exploration*, and is the central topic of this thesis. Active perception has previously been studied with a multitude of sensor models, environment representations, and robot dynamics models. The active perception task itself can be dissected into two components [25]:

Figure 1.1: A household service robot awakes in an unknown environment. Prior to accomplishing its main functionalities, it will require a map of its surroundings. What sequence of actions should it take to minimize the time it spends exploring?

component 1: Identifying regions in the environment that, when visited, will spatially extend or reduce uncertainty in the current map

component 2: Autonomously navigating to the aforementioned regions, while simultaneously localizing to the map and updating it with acquired sensor measurements

A motivating example is depicted in Fig. 1.1, where a household service robot is initialized in an unknown environment. Prior to accomplishing tasks that a human might ask it to perform, the robot must learn its surroundings and build a map of the house. Ideally this phase of initialization would be fast, as it is a prerequisite to the main functionality of the robot, and also might be required when furniture is moved or household objects are displaced. Where should the robot travel to observe the most of the environment in the shortest amount of time? Virtually any autonomous robot operating in an unknown environment will require a map-building initialization phase, welcoming strategies that enable high-speed and intelligent exploration.

This thesis introduces an assortment of information-theoretic optimizations that increase the efficiency of active perception when using a beam-based sensor model (e.g. LIDAR, time-of-flight cameras, structured light sensors) and an occupancy grid map [10]. Applying these optimizations during exploration allows a robot to consider a significantly larger number of future locations to move towards in its partially observed environment, regardless of the planning strategy used. Additionally, this thesis presents a method for analyzing the complexity of the local environment and adapting the robot's map resolution, planning frequency, movement speed, and exploration behaviors accordingly. By adapting these properties online, an autonomously exploring robot is able to speed up through areas with open expanses or where the map is well-known, and slow down when the local environment requires careful maneuvering or more thorough investigation.

1.1 Previous Work

Prior approaches to mobile robot active perception fall into two broad categories: *geometric* approaches that reason about the locations and presence of obstacles and free space in the robot's map [1, 7, 8, 27, 30, 31], and more recently, *information-theoretic* approaches that treat the map as a multivariate random variable and choose actions that will maximally reduce its uncertainty [2, 6, 9, 11, 15]. Both categories of approaches solve **component 1** of active perception, and assume that a planner and Simultaneous Localization and Mapping (SLAM) framework are available to accomplish **component 2**.

1.1.1 Geometric Exploration Strategies

Many successful geometric exploration approaches build upon the seminal work of Yamauchi [31], guiding the robot to *frontiers* - regions on the boundary between

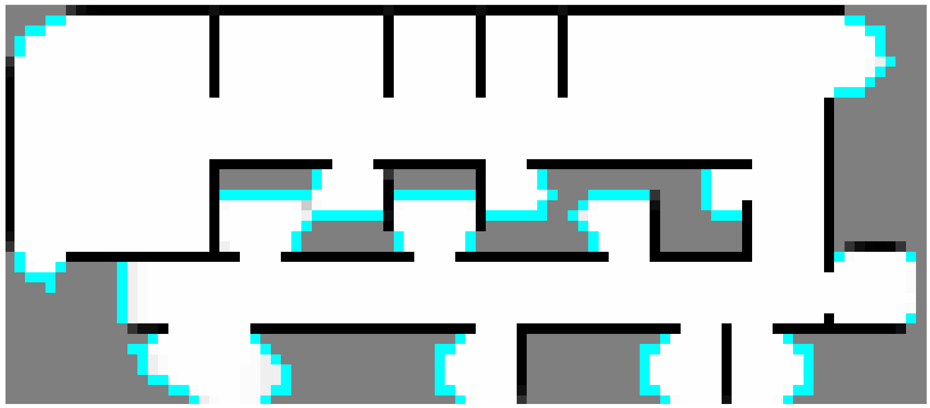


Figure 1.2: A partially explored map with frontiers between free and unknown space highlighted in blue.

free and unexplored space in the map (Fig. 1.2). Since multiple frontiers often exist simultaneously in a partially explored map, a variety of heuristics and spatial metrics can be used to decide which frontier to travel towards [20]. For example, an agent may decide to visit the frontier whose path through the configuration space from the agent's current position has minimum length, or requires minimal time or energy input to traverse. Similarly, an agent may decide to only plan paths to locations from which frontiers can be observed by its onboard sensors.

While effective in 2D environments, frontier exploration algorithms have several restrictive qualities. First, the naïve extension of frontier exploration from 2D to 3D maps poses a non-trivial challenge; as the dimensionality of the workspace increases, frontiers are distributed more densely throughout the environment due to occlusions, sensing resolution, and field-of-view, resulting in poor exploration performance [25]. Second, planning a path to a frontier does not imply that the path itself will be information-rich. Trajectory optimization techniques that consider information acquired by the robot's sensors along a planned path can be used as extensions to improve exploration performance [18, 26]. Finally, although the robot is guaranteed to

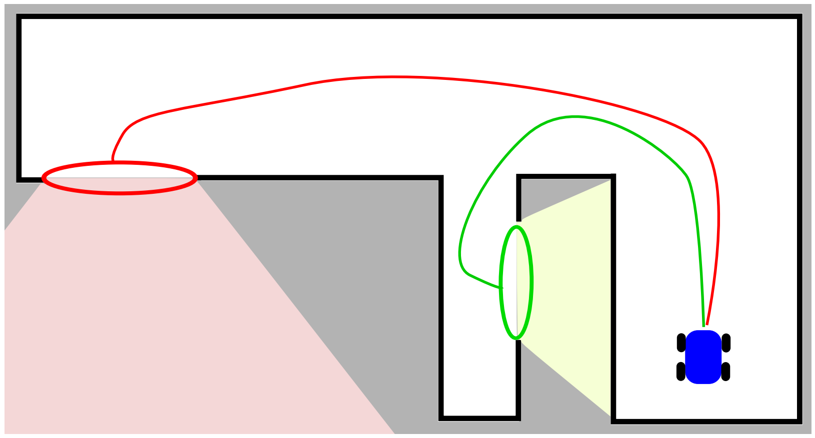


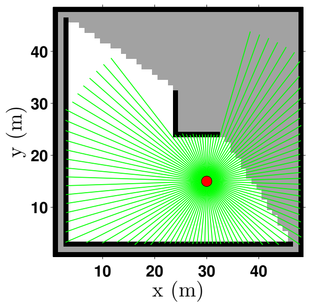
Figure 1.3: Traditional frontier exploration would visit the green location first because it is closest. A simple extension involves simulating sensor measurements from frontiers and examining their informativeness [12]. Applying this extension would cause robot to visit the red frontier first, since that location will provide more information about the map per unit time.

learn new information upon reaching a frontier, the amount of information learned is dependent on the robot's sensor model, which is not considered when identifying frontiers. It may therefore be more efficient to visit a frontier that is suboptimal according to heuristics such as path length if the robot's sensors will provide more information from that location (Fig. 1.3). This limitation was first overcome by evaluating the informativeness of simulated sensor measurements taken from frontier locations [12], and was the original motivation for developing a category of information-theoretic exploration strategies.

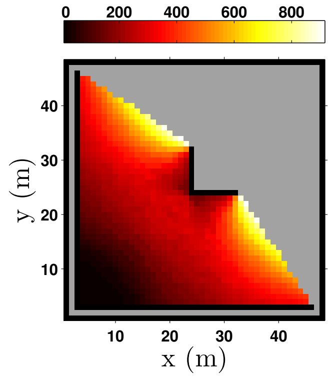
More thorough surveys of frontier exploration algorithms and heuristics are provided by Basilico et al. [5] and Holz et al. [14].

1.1.2 Information-Theoretic Exploration Strategies

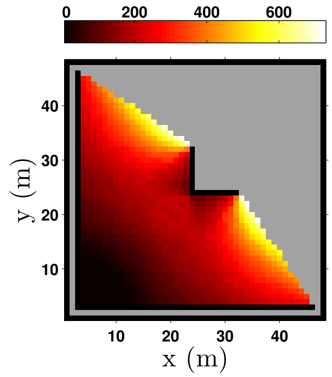
Information-theoretic exploration strategies cast the active perception task as an optimization, and choose actions for the robot that maximize an information-based objec-



(a)



(b)



(c)

Figure 1.4: Two variants of mutual information (Fig. 1.4b: Shannon; Fig. 1.4c: Cauchy-Schwarz Quadratic) densely computed in free space over an occupancy grid (Fig. 1.4a) using a 100-beam omnidirectional 2D laser with 30 m range. An exemplary sensor measurement is depicted in Fig. 1.4a. Controlling the robot towards locations that maximize either variant of mutual information would attract the robot to locations from which it could observe unknown areas of the map.

tive function such as Shannon's entropy or mutual information [6, 9, 15, 17] (Fig. 1.4). Entropic measures like these are appealing because unlike geometric methods, they capture the relationship between sensor placement and uncertainty in the map. In addition, they can be computed without a maximum likelihood map estimate, and therefore do not discard probabilistic information known to the robot. Control policies that maximize mutual information have been proven to guide robots towards unexplored space [15], and weighting frontiers by the expected mutual information between the map and a sensor measurement acquired at the frontier location has been shown to result in more efficient exploration behaviors than traditional frontier exploration [9]. The exact same calculation can be used to evaluate information-theoretic objective functions in 2D and 3D environments.

The utility afforded by information-theoretic approaches comes at the cost of computational inefficiency. As a point of comparison, frontiers and other geometricallydefined landmarks need only to be computed once per map update, and can be computed (at worst, using a brute force search) with time complexity linear in the number of cells in the robot's map. One may alternatively choose to identify and cache frontiers every time a sensor measurement is used to update the map, yielding a constant time frontier identification step that is bounded by the number of map voxels within the maximum sensor range. By contrast, information-theoretic objective functions typically consider the probabilistic uncertainty associated with the sensor and environment models, and therefore require expensive sensor-related operations such as raycasts or sampling a large number of times from the distribution of possible future measurements. Approximations to mutual information between a map and beam-based sensor measurements can be evaluated with time complexity linear in the number of map voxels intersected by a sensor's beams [9, 16, 21]. This already-expensive operation must be performed for every future location that the robot might wish to travel to. Julian et al. report that densely calculating mutual information over the free space in a large 1500 m^2 map requires approximately ten seconds with a parallelized implementation using a 3.4 Ghz quad-core CPU and NVIDIA GTX 690 graphics card [15].

1.2 Thesis Problem

Although inefficient, information-theoretic solutions to the active perception task are superior to geometric solutions in many ways. However, modern computers cannot densely evaluate information-theoretic objective functions over a robot's map in real-time. Any strategy that makes the evaluation of information-theoretic objective functions more efficient will allow a robot to consider more future locations prior to taking an action towards one, thereby increasing the speed at which the robot is able to explore an environment.

Currently, the most efficient information-theoretic exploration algorithms are too slow for the motivating scenario described in Fig. 1.1. For example, a highly-efficient recent approach by Charrow et al., requires eleven minutes to explore a 17 m $\times$18 m $\times$3 m building with a quadrotor - enough time for the quadrotor's batteries to deplete twice [9]. This thesis addresses the inefficiencies of information-theoretic exploration, summarized in the following statement:

> **Thesis Problem:** Solutions to the mobile robot active perception task that involve optimization of information-theoretic cost functions are too computationally expensive for high-speed exploration in complex environments.

1.3 Thesis Statement

This thesis proposes occupancy grid compression as a solution to the computational inefficiencies of information-theoretic exploration.

1.4 Outline

Chapter 2

Foundations

This thesis draws upon prior research from the robotics, information theory, and signal processing domains to develop its formulations. Sections 2.1 - 2.3 review relevant topics within robotics including occupancy grid mapping, active perception as an optimization, and several planning strategies that are suitable for the exploration task. These foundational topics will be used to develop a theory of optimal occupancy grid compression, and methods for guiding a robot to explore uncertain areas of its map efficiently. The formulations developed in Chapters 3 - 5 will also borrow heavily from information and rate distortion theory. These domains are frequently concerned with evaluating the effect one random variable (e.g. a sensor measurement) has on another (e.g. a map) or with compressing a random variable to a reduced representation in such a way that the compressed form preserves the structure of the uncompressed form. Section 2.4 reviews concepts from these domains that will be used when developing theories for optimal map resolution selection, and for adapting robot exploration behaviors to the resolution with which its environment is modeled.

2.1 Occupancy Grid Mapping

Occupancy grids (OGs) are a common and useful proabilistic map model for representing and reasoning about an unknown environment. The remainder of this thesis assumes that the robot's environment is represented as an OG. Figures 1.2, 1.3 and 1.4a depict occupancy grids, where black cells represent areas of the environment occupied by an obstacle, white cells represent areas that do not contain obstacles, and grey cells represent locations with unknown occupancy status.

OGs decompose the robot's workspace into a discrete set of 2D or 3D cells with a specified resolution. The presence or absence of obstacles within these cells is modeled as a K-tuple binary random variable, $\mathbf{m} = \{m_i\}_{i=1}^{K}$, with support set $\{\texttt{EMP}, \texttt{OCC}\}$. The probability that an individual cell is occupied is given by $p\left(m_i \mid \mathbf{x}_{1:t}, \mathbf{z}_{1:t}\right)$, where $\mathbf{x}_{1:t}$ denotes the robot's history of states, and $\mathbf{z}_{1:t}$ denotes the history of range observations accumulated by the robot. The OG representation treats cells as independent from one another, allowing one to express the probability of a specific map as the product of individual cell occupancy values:

$$p\left(\mathbf{m} \mid \mathbf{x}_{1:t}, \mathbf{z}_{1:t}\right) = \prod_i p\left(m_i \mid \mathbf{x}_{1:t}, \mathbf{z}_{1:t}\right). \tag{2.1}$$

For notational simplicity, the map conditioned on random variables $\mathbf{x}_{1:t}$ and $\mathbf{z}_{1:t}$ will henceforth be written as $p\left(\mathbf{m}\right) \equiv p\left(\mathbf{m} \mid \mathbf{x}_{1:t}, \mathbf{z}_{1:t}\right)$, and the probability of occupancy for a grid cell i as $o_i \equiv p\left(m_i = \texttt{OCC} \mid \mathbf{x}_{1:t}, \mathbf{z}_{1:t}\right)$. Unobserved grid cells are assigned a uniform prior such that $\{o_i = 1 - o_i = 0.5\}_{i=1}^{K}$. This implies that the robot is initially unaware of its surroundings prior to accumulating sensor measurements. To prevent numerical precision issues, the occupancy status of a grid cell m_i is represented by the log-odds ratio

$$l \equiv \log \frac{o_i}{1 - o_i}. \tag{2.2}$$

The log-odds ratio maps from occupancy probabilities existing on $[0, 1]$ to $\mathbb{R}$, which is more suitable for floating-point arithmetic. In addition, the log-odds ratio makes updates to a cell occupancy probability additive rather than multaplicative. When a new measurement $\mathbf{z}_t$ is obtained, cell occupancy values may be updated with

$$l \leftarrow l + L\left(m_i \left| \mathbf{z}_t\right.\right), \tag{2.3}$$

where the term $L\left(m_i \left| \mathbf{z}_t\right.\right)$ represents the robot's inverse sensor model [28].

2.2 Active Perception

Active perception is the idea that a machine should continually guide itself to states in which it is able to acquire better sensor measurements [3, 4]. Active perception draws inspiration from biological sensors that adapt in response to external stimuli. The human eye, for example, has muscles that constrict the pupil in response to bright light (adaptation), and others that distort the curvature of its lens to focus on nearby or far-away objects (accomodation). Adaptation and accomodation allow humans to see light varying nine orders of magnitude in brightness, and focus on objects an infinite distance away. Similarly, a man-made sensor such as a camera should not passively collect and report incoming photons, but should adapt its aperture, CMOS gains, and shutter speed based on the properties of the incoming light.

To extend this idea to mobile robotics, one must consider the robot system itself as a sensor that is able to move and actuate for the purpose of collecting better sensor measurements. From this perspective, the robot's task is to choose and execute *actions* that optimize the quality of its sensor measurements. An action can be defined as a sequence of configurations $\mathbf{x}_\tau \equiv (\mathbf{x}_{t+1}, \ldots, \mathbf{x}_{t+T})$ that the robot will achieve over a future time interval $\tau \equiv (t+1, \ldots, t+T)$. From configurations $\mathbf{x}_\tau$ the robot will acquire future sensor measurements $\mathbf{z}_\tau \equiv (\mathbf{z}_{t+1}(\mathbf{x}_{t+1}), \ldots, \mathbf{z}_{t+T}(\mathbf{x}_{t+T}))$. This thesis is concerned primarily with ground robots constrained to $SE(2)$, and will therefore use $\mathbf{x}_i$ to refer to a pose in 2D space: $\mathbf{x}_i \equiv (x_i, y_i, \theta_i)^T$.

In the context of exploring an unknown environment, the active perception problem can be framed as an optimization over possible future actions that the robot can take:

$$\mathbf{x}_\tau^* = \underset{\mathbf{x}_\tau \in \mathcal{X}}{\operatorname{argmax}} \ \mathcal{J}\left(\mathbf{m}, \mathbf{z}_\tau(\mathbf{x}_\tau)\right), \tag{2.4}$$

where $\mathcal{J}\left(\mathbf{m}, \mathbf{z}_\tau(\mathbf{x}_\tau)\right)$ is a reward function expressing the new information learned by sequentially moving the robot to configurations $\mathbf{x}_\tau$, collecting sensor measurements $\mathbf{z}_\tau$, and updating the map $\mathbf{m}$. $\mathcal{X}$ is the set of all collision-free and dynamically feasible actions that the robot can take. In addition to evaluating the pure information content of $\mathbf{z}_\tau$, $\mathcal{J}$ commonly incorporates the length of time or energy expenditure required to carry out the action $\mathbf{x}_\tau$.

Unfortunately, the active perception optimizaton faces the *curse of dimensionality*; the size of $\mathcal{X}$ grows exponentially with the length of the time horizon τ. As τ increases in size, it quickly becomes infeasible to evaluate $\mathcal{J}$ over all possible actions in $\mathcal{X}$. This inefficiency motivates generating a fixed-size set candidate actions that are likely to be informative prior to optimizing (2.4).

2.3 Action Generation

The primary concern of action generation is to suggest a fixed-size set of feasible actions $\mathcal{X}$ that are likely to be informative. A suitable choice of $\mathcal{J}$ can be evaluated on these actions to choose an optimal exploration action using (2.4). Several action generation options exist.

2.3.1 Frontier Seeding

Recent works by Charrow et al. [9] and Vallvé et al. [29] suggest seeding information-theoretic exploration by identifying frontiers and then evaluating a reward function from frontier locations. Because frontier identification is efficient, this two-pass approach is useful for locating potentially informative locations prior to performing the comparatively more expensive reward evaluation step. This strategy has the added benefit that frontiers are computed globally across the robot's map, guaranteeing that the robot will never become trapped in a dead-end or a location where its local map is already fully explored.

Identifying frontiers before planning to them avoids planning feasible trajectories to many future locations. Frontiers can be ranked by the information-theoretic reward offered from their locations, and the resulting sorted list of frontiers can be iterated through until a dynamically feasible and collision-free trajectory is found. Planning from an initial state to a goal state subject to dynamic and obstacle constraints becomes especially expensive in high-dimensional configuration spaces, and should be performed as few times as possible.

After selecting a location that will yield high reward, one may use a real-time pathfinding algorithm such as A* [13], RRT [19], or their many variants to generate a trajectory from the robot's initial state.

2.3.2 Forward-Arc Motion Primitives

Actions can also be generated by sampling from a set of pre-computed motion primitives. A simple strategy for generating motion primitives for a ground vehicle constrained to $SE(2)$ involves simulating the robot's path when moving at a constant linear and angular velocity for a specified amount of time. Actions resulting from this approach form arcs of a circle with a radius that is a function of the specified linear and angular velocity (Fig. 2.1).

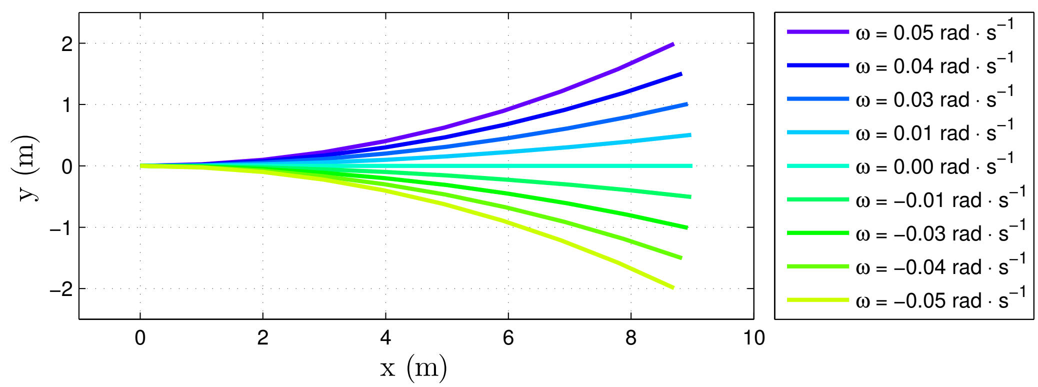


Figure 2.1: Nine motion primitives generated with $\omega = \{-0.05, -0.04, \ldots, 0.05\}$ rad/s, $v = 1.0$ m/s.

Consider a robot following the arc of a circle with velocity v and rotational velocity ω. Assuming the robot's current position is given by $\mathbf{x}_t = (x_t, y_t, \theta_t)^T$, forward-arc motion primitives can be generated by specifying the future robot state, $\mathbf{x}_{t+T}$, as a function of v and ω for a sequence of uniformly varying times $T \in \mathbb{R}^+$. These paths are described by a set of nonlinear differential equations:

$$\dot{\mathbf{x}}_{t+T} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix}_{t+T} = \begin{bmatrix} v\cos(\theta_{t+T}) \\ v\sin(\theta_{t+T}) \\ \omega \end{bmatrix}, \tag{2.5}$$

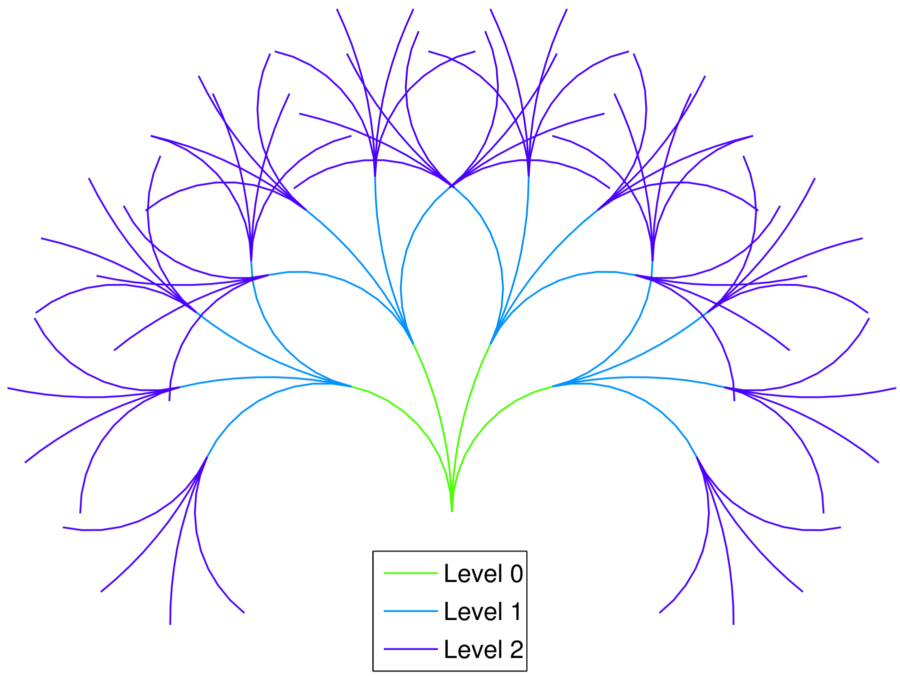


Figure 2.2: A primitive dictionary with a depth of three constructed from a library of four motion primitives.

the solution of which is given by

$$\mathbf{x}_{t+T} = \begin{bmatrix} \frac{v}{\omega}\left(\sin\left(\omega T + \theta_t\right) - \sin\left(\theta_t\right)\right) \\ \frac{v}{\omega}\left(\cos\left(\theta_t\right) - \cos\left(\omega T + \theta_t\right)\right) \\ \omega T \end{bmatrix} + \mathbf{x}_t. \tag{2.6}$$

Sequentially incrementing T in (2.6) produces a sampling of poses lying along an arc parameterized by the robot's velocity and angular velocity, with origin $\mathbf{x}_t$.

A sampling of actions with varying v and w values (such as that depicted in Fig. 2.1) is referred to as a primitive *dictionary*. To generate more actions, one can construct a primitive *library*. This is accomplished by forming a tree with nodes corresponding to poses at the endpoints of actions. The tree is initialized by adding the robot's current position as the root node. Then, a dictionary of motion primitives is rotated and translated to leaf nodes in the tree until a specified depth is reached. A primitive library is shown in Fig. 2.2.

Forward-arc motion primitives are pre-computed prior to deployment into an unknown environment, making them an efficient choice for real-time exploration. Collision checking involves stepping along actions during a breadth-first or depth-first search and pruning all nodes (actions) that lie below those that contain a collision.

2.3.3 Lattice Graph Motion Primitives

A third method for generating actions defines a discrete set of goal states, and solves Boundary Value Problems (BVPs) to find trajectories from ($\emptyset$) to each goal [22–24] (Fig. 2.3). The resulting set of motion primitives can be rotated and translated to the robot's current position at run-time, and sampled from to produce candidate actions. Like forward-arc motion primitives, lattice graph motion primitives can be pre-computed and are therefore a suitable choice for real-time exploration. Collision checking for motion primitives in the lattice graph involves stepping along the action

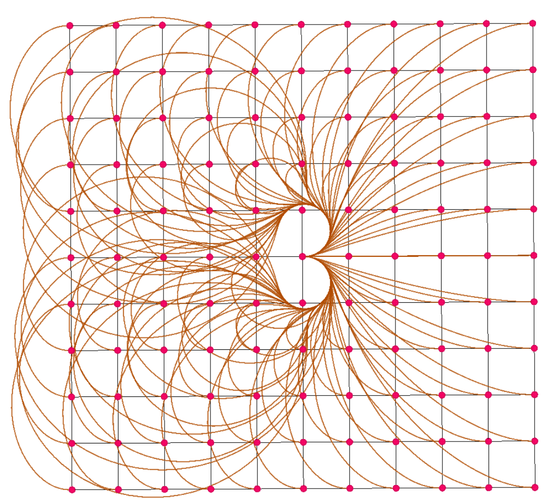


Figure 2.3: An $11 \times 11 \times 1$ lattice graph generated by solving a BVP from the robot's initial pose (middle, facing right) to a lattice of final poses (with final angle equal to initial angle) subject to linear and angular velocity constraints. No solution exists for the nodes immediately above and below the robot's initial position.

and checking for poses that lie outside of the robot's configuration space.

2.4 Information Theory

Information theory is a branch of mathematics concerned with describing the information content of and between random variables. The remainder of this thesis borrows heavily from information theory. This section briefly reviews foundational elements of information theory, including definitions of core concepts and more abstract extensions that are relevant to the formulations in Chapters 3-5.

2.4.1 Entropies, Divergences, and Mutual Information

The concepts of entropy, divergence, and mutual information are the most basic building blocks of information theory.

2.4.2 Rényi's α-entropy

2.4.3 Cauchy-Schwarz Quadratic Mutual Information

2.5 Summary of Foundations

Chapter 3

Information-Theoretic Map Compression

3.1 The Principle of Relevant Information

We formulate the OG compression problem as an information-theoretic optimization using the Principle of Relevant Information (PRI). PRI is a technique for learning a reduced representation $\hat{X}$ of a random variable X such that both the entropy of $\hat{X}$ and the divergence of $\hat{X}$ with respect to the original data are minimized.

$$J(\hat{X}) = \min_{\hat{X}} (\mathrm{H}_\alpha(\hat{X}) + \lambda \mathrm{D}_\alpha(X||\hat{X})). \tag{3.1}$$

The two terms of the PRI cost function are Rényi's α-entropy, which describes the amount of uncertainty in its argument, and Rényi's α-divergence, which is a distance measure describing distortion between $p(x)$ and $p(\hat{x})$. These terms simplify to the more common Shannon entropy and Kullback-Leibler divergence for $\alpha = 1$. The variational parameter λ controls the amount of distortion in the compressed data.

To complement the Information Bottleneck optimization described later (Sec. **??**), we choose to minimize the H_2 entropy and Cauchy-Schwarz divergence. For discrete random variables X and $\hat{X}$,

$$\mathrm{H}_2(\hat{X}) = -\log \sum_i p^2(\hat{x}_i), \tag{3.2}$$

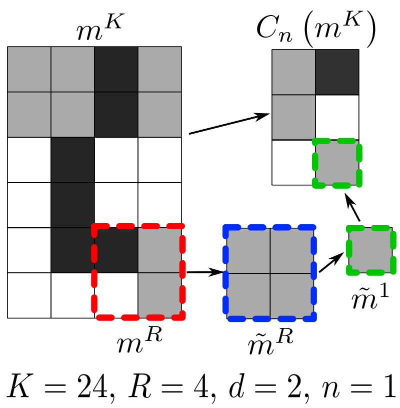
$$\mathrm{D}_{\mathrm{CS}}(X||\hat{X}) = \log \frac{\sum_i p^2(x_i) \sum_i p^2(\hat{x}_i)}{\left(\sum_i p(x_i)p(\hat{x}_i)\right)^2}. \tag{3.3}$$

The cost function in (3.1) is then:

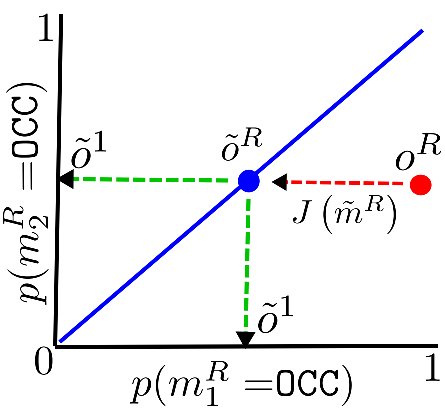
$$(1-\lambda)\,\mathrm{H}_2(\hat{X}) - \lambda 2 \log \sum_i p(x_i)p(\hat{x}_i) - \lambda \mathrm{H}_2(X). \tag{3.4}$$

The third term has no influence on the minimization over $\hat{X}$, and can be ignored. We choose to give equal weight to the entropy and divergence, and optimize for $\lambda = 1$. Noting that logs and quadratic functions increase monotonically for positive arguments, and noting that the summand in the second term of (3.4) must be positive, the optimization can be simplified to:

$$J(\hat{X}) = \max_{\hat{X}} \sum_i p(x_i)p(\hat{x}_i). \tag{3.5}$$



(a) OG compression sequence



(b) Probability space of the top two grid cells in m^R in 3.1a

Figure 3.1: For each square (cubic in 3D) region m^R in the uncompressed OG m^K, the PRI optimization finds a random variable $\tilde{m}^R$ that minimizes (3.1) and is constrained to have uniform occupancy probability $\tilde{o}^R = (\tilde{o}^1, \ldots, \tilde{o}^1)$.

3.2 Framing Occupancy Grid Compression as an Optimization

To apply the PRI optimization to OG compression, let X be an OG m^K with K cells. The problem must be constrained in three ways. First, because OGs encode a 2D or 3D geometry, $\hat{X}$ must represent X well in local regions. Compression over the map can therefore be accomplished by performing compression in many small independent square (cubic in 3D) regions $m^R \subseteq m^K$, assuming individual grid cell occupancies are independent. Second, we consider only the set of compressions that reduce OG cell count in each dimension by factors of two. Therefore an OG m^K will be compressed to an OG $m^{2^{-dn}K}$, where d is the OG dimension and n is the number of $2\times$ compressions in each dimension. The set of compressions with this property can be expressed as:

$$C_n(m^K) \equiv m^{2^{-dn}K}, \quad n \in \mathbb{N}_0, \tag{3.6}$$

where superscripts denote cell count and where a compression of $n = 0$ gives the original OG: $C_0(m^K) = m^K$. Both m^K and $C_n(m^K)$ will have the same metric dimensions, but will have different cell edge lengths and cell counts when $n \geq 1$. Finally, we enforce that $\hat{X}$ must also be an OG. Since $\mathrm{D_{CS}}(X||\hat{X})$ may only be computed for two random variables with the same support set, we use the PRI to find a random variable $\tilde{m}^R$ that has uniform occupancy probabilities, and then reduce its dimension to one, yielding a single grid cell $\tilde{m}^1$ (Fig. 3.1). Combining the single-cell $\tilde{m}^1$ variables from independent regions yields the compressed OG $C_n(m^K)$.

Rather than directly maximizing (3.5) over $\tilde{m}^R$, we are interested in finding the distribution $p(\tilde{m}^R)$ corresponding to the maximum. $p(\tilde{m}^R)$ is a Bernoulli distribution,

Table 3.1: Contingency table for a compression from the OG region m^R to $\tilde{m}^R$. `O` and `E` stand for `OCC` and `EMP`.

		$\tilde{m}^R$				
		`E, E, ..., E`	`E, E, ..., O`	...	`O, O, ..., E`	`O, O, ..., O`
m^R	`E, E, ..., E`	$w_2 \cdot (1-\tilde{o}^1) \cdot \prod_{i=1}^R (1-o_i^R)$	0	...	0	$w_3 \cdot \tilde{o}^1 \cdot \prod_{i=1}^R (1-o_i^R)$
	`E, E, ..., O`	$w_1 \cdot (1-\tilde{o}^1) \cdot o_1^R \cdot \prod_{i=2}^R (1-o_i^R)$	0	...	0	$w_4 \cdot \tilde{o}^1 \cdot o_1 \cdot \prod_{i=2}^R (1-o_i^R)$
	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$
	`O, O, ..., E`	$w_1 \cdot (1-\tilde{o}^1) \cdot (1-o_1^R) \cdot \prod_{i=2}^R o_i^R$	0	...	0	$w_4 \cdot \tilde{o}^1 \cdot (1-o_1^R) \cdot \prod_{i=2}^R o_i^R$
	`O, O, ..., O`	$w_1 \cdot (1-\tilde{o}^1) \cdot \prod_{i=1}^R o_i^R$	0	...	0	$w_4 \cdot \tilde{o}^1 \cdot \prod_{i=1}^R o_i^R$
	Total	$1-\tilde{o}^1$	0	...	0	$\tilde{o}^1$

and is completely determined by its single parameter $\tilde{o}^1 = p(\tilde{m}^R = \{\texttt{OCC}, \dots, \texttt{OCC}\}) = 1 - p(\tilde{m}^R = \{\texttt{EMP}, \dots, \texttt{EMP}\})$. Substituting the described variables into (3.5),

$$\tilde{o}^1_* = \underset{\tilde{o}^1}{\operatorname{argmax}} \sum_{M^R} p\left(m^R = M^R\right) p\left(\tilde{m}^R = M^R\right). \tag{3.7}$$

3.3 Solving the Optimization

Table 3.1 shows a contingency table for a compression from the OG region m^R to $\tilde{m}^R$. The middle columns of the contingency table have zero probability, since the $\tilde{m}^R$ must have a uniform cell probability to be able to reduce it to $\tilde{m}^1$ (i.e. $\tilde{o}_i^R = \tilde{o}_j^R = \tilde{o}^1 \, \forall i, j \in 1, \dots, R$). In this section we are only interested in the marginal distributions (bottom-most row and right-most column), which are needed to determine (3.7). Substituting these,

$$\tilde{o}^1_* = = \underset{\tilde{o}^1}{\operatorname{argmax}} \left((1-\tilde{o}^1) \prod_{i=1}^R (1-o_i^R) + \tilde{o}^1 \prod_{i=1}^R o_i^R \right), \tag{3.8}$$

which is satisfied for

$$\tilde{o}_*^1 = \begin{cases} 0 & \text{if } \prod_{i=1}^R \frac{o_i^R}{1-o_i^R} < 1 \\ 1 & \text{if } \prod_{i=1}^R \frac{o_i^R}{1-o_i^R} > 1 \\ \frac{1}{2} & \text{otherwise} \end{cases}, \tag{3.9}$$

where the last case applies in the limit as $\lambda \to 1^+$.

The PRI solution gives us a simple compression rule: if the product of cell occupancy likelihoods in a given region is greater than 1, set the occupancy of the cell corresponding to that region in the compressed OG to 1. Likewise set the value to 0 if the product of likelihoods is less than 1, and to 0.5 if the product of likelihoods is 1. Pragmatically, it is more reasonable to use the map's occupancy and free thresholds rather than 1.0 and 0.0. This variation corresponds to optimizing for λ slightly greater than one, favoring minimal distortion to minimal entropy. Additionally, one may introduce a heuristic to increase the fraction of occupied cells that are preserved through compression by multiplying the right-hand sides of the inequalities in (3.9) by $\eta \in (0, 1)$. As η decreases, occupied cells will be preserved through compression with higher frequency. For any application involving raycasting, it is especially important to include this heuristic, as vanishing occupied cells lead to poor ray termination.

Denoting $\pi^R \equiv \prod_{i=1}^R \frac{o_i^R}{1-o_i^R}$ and applying these modifications gives the $\sqrt{R} \times \sqrt{R} \to 1 \times 1$ (or $\sqrt[3]{R} \times \sqrt[3]{R} \times \sqrt[3]{R} \to 1 \times 1 \times 1$ in 3D) compression rule for each region m^R:

$$\tilde{o}_*^1 = \begin{cases} \frac{1}{2} & \text{if } \pi^R = \eta \vee \pi^R = 1 \\ p_{\text{free}} & \text{if } \pi^R < \eta \wedge \pi^R \neq 1 \\ p_{\text{occ}} & \text{if } \pi^R > \eta \wedge \pi^R \neq 1 \end{cases}, \tag{3.10}$$

where p_{occ} and p_{free} are the thresholds for occupied and free space in the OG implementation, respectively.

3.4 Occupancy Grid Pyramids

3.5 Results

3.6 Chapter Summary

Chapter 4

Balancing Map Compression with Sensing Accuracy

The PRI strategy in Sec. ?? determines an optimal compression given a desired OG resolution. However, Fig. ?? suggests that one should also reduce the resolution of the OG as much as possible to increase efficiency. In this section we formulate a second optimization based on the Information Bottleneck (IB) method [?] that chooses a grid resolution minimizing both the redundancy between m^K and $C_n(m^K)$, and loss of mutual information with respect to a sensor measurement z.

4.1 The Information Bottleneck Method

IB is a widely used technique in signal processing for finding the optimal reduced representation $\hat{X}$ of a random variable X that preserves maximum information about a second random variable Y:

$$\min_{\hat{X}} \mathrm{I}(X;\hat{X}) - \beta \mathrm{I}(\hat{X};Y). \tag{4.1}$$

IB resembles PRI, but considers the effects of compression on the information between two datasets, as opposed to one. Similar to λ in the PRI optimization, β is a design parameter that trades compression for conservation of information. As $\beta \to 0$, the optimization tends towards the trivial lossy compression $\{\hat{X}\} = 0$, whereas when $\beta \to \infty$, $\hat{X}$ approaches its original representation X [?]. The two terms in the argument of (4.1) can equivalently be thought of as the information loss incurred by describing $\hat{X}$ with Y instead of with X [?].

Most importantly for OG compression, when combined with the PRI approach in Section ??, the IB method can be used to find an optimal compression n^*:

$$n^* = \underset{n \in \mathbb{N}_0}{\operatorname{argmin}} \; \mathrm{I}_{\mathrm{CS}}\left(m^K; C_n(m^K)\right) - \beta \mathrm{I}_{\mathrm{CS}}\left(C_n(m^K); z\right). \tag{4.2}$$

4.2 Optimizing Map Resolution for Sensing

The second term can be computed using (??), and is $2^{1\times n}$ times more efficient than computing (??) with respect to the uncompressed map m^K (where $d = 1$ because $\mathrm{I}_{\mathrm{CS}}(m; z_\tau)$ is computed using 1D raycasts). Since $\mathrm{I}_{\mathrm{CS}}(m^K; C_n(m^K))$ describes the divergence between the distributions $p(m^K, C_n(m^K))$ and $p(m^K)p(C_n(m^K))$, the first term in (??) can be computed by substituting these for $p(x_i)$ and $p(\hat{x}_i)$ in the definition of Cauchy-Schwarz divergence (3.3).

However, the joint distribution $p(m^K, C_n(m^K))$ is underdetermined by two variables, and must be constrained before computing (4.2). While the remaining two degrees of freedom make the IB cost function arbitrary for a single resolution, fixing the joint distribution and using it to compute CSQMI across different grid resolutions still yields a meaningful optimization. To constrain the extra degrees of freedom we first decompose the joint distribution into independent regions $r \in m^K$:

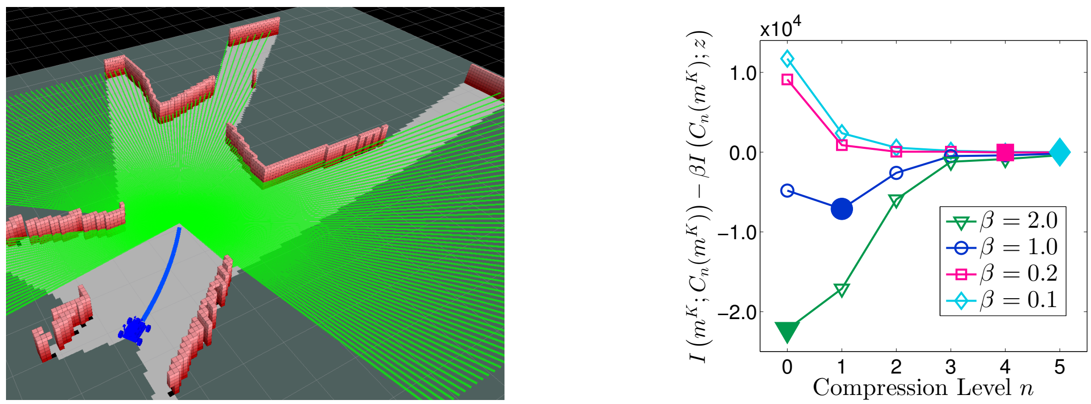
$$
\begin{aligned}
p(m^K, C_n(m^K)) &= \prod_{r \in m^K} p(m_r^R, C_n(m_r^R)) \\
&= \prod_{r \in m^K} p(m_r^R, \tilde{m}_r^R),
\end{aligned}
\tag{4.3}
$$

where the second equation holds by noting $C_n(m_r^R)$ has a dimension of one, and that that all cells in $\tilde{m}^r$ are completely determined by knowing $C_n(m_r^R)$. Then, for each region r, we choose the joint distribution $p(m_r^R, \tilde{m}_r^R))$ to be a product of the marginals $p(m_r^R)$ and $p(\tilde{m}_r^R))$, weighted by four extra coefficients $w_{1:4}$ (Table 3.1). Similar to the reasons that η in Sec. **??** is chosen to preserve occupied cells through compression, the constant $c_1 \in (0, 1)$ downweighs the probability of the event that $\tilde{m}^R$ is $\{\texttt{EMP}, \ldots, \texttt{EMP}\}$ if any grid cells in m^R are occupied. The remaining three constants balance the effects of w_1 such that the conditional distributions over the rows and columns of Table 3.1 all sum to the marginal distributions on the bottom-most row and right-most column.

Figure 4.1 displays the influence of β on the IB optimization for a multi-beam measurement captured from a planned future location. The optimization favors no compression when β is large, and maximum compression when β is small.

4.3 Results

4.4 Chapter Summary



(a) (b)

Figure 4.1: 4.1a shows an uncompressed map and measurement taken from a planned future position. With this map and expected laser scan, the optimal compression level (filled markers) computed with (4.2) decreases as β increases, favoring preservation of information about the measurement as opposed to compression (4.1b).

Chapter 5

Compressed Maps for Active Perception

5.1 Adapting the Map Resolution Online

5.2 Adapting Exploration Behavior to the Map Resolution

5.3 Results

5.4 Chapter Summary

Chapter 6

Summary, Contributions, and Future Work

6.1 Thesis Summary

6.2 Contributions

6.3 Future Work

6.4 Conclusions

Bibliography

[1] Ercan U Acar and Howie Choset. Sensor-based coverage of unknown environments: Incremental construction of morse decompositions. *The International Journal of Robotics Research*, 21(4):345–366, 2002.

[2] Francesco Amigoni and Vincenzo Caglioti. An information-based exploration strategy for environment mapping with mobile robots. *Robotics and Autonomous Systems*, 58(5):684–699, 2010.

[3] Ruzena Bajcsy. Active perception. *Proceedings of the IEEE*, 76(8):966–1005, 1988.

[4] Ruzena Bajcsy and Mario Campos. Active and exploratory perception. *CVGIP: Image Understanding*, 56(1):31–40, 1992.

[5] Nicola Basilico and Francesco Amigoni. On evaluating performance of exploration strategies for autonomous mobile robots. In *Proceedings of the performance evaluation and benchmarking for intelligent robots and systems workshop at IEEE/RSJ international conference on intelligent robots and systems*, 2008.

[6] Frederic Bourgault, Alexei A Makarenko, Stefan B Williams, Ben Grocholsky, and Hugh F Durrant-Whyte. Information based adaptive robotic exploration. In

Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on, volume 1, pages 540–545. IEEE, 2002.

[7] Wolfram Burgard, Mark Moors, Dieter Fox, Reid Simmons, and Sebastian Thrun. Collaborative multi-robot exploration. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 1, pages 476–481. IEEE, 2000.

[8] Kwong-Fai Chan and Tak Wah Lam. An on-line algorithm for navigating in an unknown environment. *International Journal of Computational Geometry & Applications*, 3(03):227–244, 1993.

[9] Ben Charrow, Sikang Liu, Vijay Kumar, and Nathan Michael. Information-theoretic mapping using cauchy-schwarz quadratic mutual information. In *Robotics and Automation, 2015. Proceedings., 2015 IEEE International Conference on*. IEEE, 2015.

[10] Alberto Elfes. *Occupancy grids: a probabilistic framework for robot perception and navigation*. PhD thesis, Carnegie Mellon University, 1989.

[11] Hans Jacob S Feder, John J Leonard, and Christopher M Smith. Adaptive mobile robot navigation and mapping. *The International Journal of Robotics Research*, 18(7):650–668, 1999.

[12] Hector H Gonzalez-Banos and Jean-Claude Latombe. Navigation strategies for exploring indoor environments. *The International Journal of Robotics Research*, 21(10-11):829–848, 2002.

[13] Peter E Hart, Nils J Nilsson, and Bertram Raphael. A formal basis for the heuristic determination of minimum cost paths. *Systems Science and Cybernetics, IEEE Transactions on*, 4(2):100–107, 1968.

[14] Dirk Holz, Nicola Basilico, Francesco Amigoni, and Sven Behnke. A comparative evaluation of exploration strategies and heuristics to improve them. In *ECMR*, pages 25–30, 2011.

[15] Brian J Julian, Sertac Karaman, and Daniela Rus. On mutual information-based control of range sensing robots for mapping applications. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 5156–5163. IEEE, 2013.

[16] Brian John Julian. *Mutual information-based gradient-ascent control for distributed robotics*. PhD thesis, Massachusetts Institute of Technology, 2013.

[17] Thomas Kollar and Nicholas Roy. Efficient optimization of information-theoretic exploration in slam. In *AAAI*, volume 8, pages 1369–1375, 2008.

[18] Thomas Kollar and Nicholas Roy. Trajectory optimization using reinforcement learning for map exploration. *The International Journal of Robotics Research*, 27(2):175–196, 2008.

[19] Steven M LaValle. Rapidly-exploring random trees a ew tool for path planning. Technical report, Compute Science Department, Iowa State University, 1998.

[20] Steven M LaValle. *Planning algorithms*. Cambridge university press, 2006.

[21] Erik Nelson and Nathan Michael. Optimal information-theoretic occupancy grid compression for high-speed information-based exploration. In *IEEE/RSJ In-*

ternational Conference on Intelligent Robots and Systems, 2015. (IROS 2015). IEEE, 2015.

[22] Mihail Pivtoraiko and Alonzo Kelly. Generating near minimal spanning control sets for constrained motion planning in discrete state spaces. In *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005. (IROS 2005).*, pages 3231–3237. IEEE, 2005.

[23] Mihail Pivtoraiko, Ross A Knepper, and Alonzo Kelly. Differentially constrained mobile robot motion planning in state lattices. *Journal of Field Robotics*, 26(3):308–333, 2009.

[24] Mihail Pivtoraiko, Daniel Mellinger, and Vijay Kumar. Incremental micro-uav motion replanning for exploring unknown environments. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 2452–2458. IEEE, 2013.

[25] Shaojie Shen, Nathan Michael, and Vijay Kumar. Autonomous indoor 3d exploration with a micro-aerial vehicle. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 9–15. IEEE, 2012.

[26] Robert Sim, Gregory Dudek, and Nicholas Roy. Online control policy optimization for minimizing map uncertainty during exploration. In *Robotics and Autkollar2008trajectoryomation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 2, pages 1758–1763. IEEE, 2004.

[27] Camillo J Taylor and D Kriegman. Exploration strategies for mobile robots. In *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, pages 248–253. IEEE, 1993.

[28] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT press, 2005.

[29] Joan Vallvé and Juan Andrade-Cetto. Dense entropy decrease estimation for mobile robot exploration. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 6083–6089. IEEE, 2014.

[30] Pengpeng Wang. *View planning with combined view and travel cost*. PhD thesis, School of Engineering Science-Simon Fraser University, 2007.

[31] Brian Yamauchi. A frontier-based approach for autonomous exploration. In *Computational Intelligence in Robotics and Automation, 1997. CIRA'97., Proceedings., 1997 IEEE International Symposium on*, pages 146–151. IEEE, 1997.