Safe Motion Planning under Uncertainty for Mobile Manipulators in Unknown Environments

by

Vinay Kumar Pilania

B.Tech., and M.Tech. Dual Degree, Indian Institute of Technology Kharagpur, 2009

Dissertation Submitted in Partial Fulfillment of the
Requirements for the Degree of
Doctor of Philosophy

in the
School of Engineering Science
Faculty of Applied Sciences

SIMON FRASER UNIVERSITY
Fall 2015

Approval

Name: **Vinay Kumar Pilania**

Degree: **Doctor of Philosophy**

Title: ***Safe Motion Planning under Uncertainty for Mobile Manipulators in Unknown Environments***

Examining Committee: **Chair:** Dr. Rodney Vaughan
Professor

Dr. Kamal K. Gupta
Senior Supervisor
Professor

Dr. Parvaneh Saeedi
Supervisor
Associate Professor

Dr. Carlo Menon
Supervisor
Associate Professor

Dr. Ahmad Rad
Internal Examiner
Professor, School of Mechatronic
Systems Engineering

Dr. Siddhartha Srinivasa
External Examiner
Associate Professor, RI, CMU
Pittsburgh, USA

Date Defended: December 8, 2015

Abstract

For a mobile manipulator to operate and perform useful tasks in human-centered environments, it is important to work toward the realization of robust motion planners that incorporate uncertainty inherent in robot's control and sensing and provide safe motion plans for reliable robot operation. Designing such planners pose a significant challenge because of computational complexity associated with mobile manipulator planning and planning under uncertainty. Current planning approaches for mobile manipulation are often conservative in nature and the uncertainty is largely ignored. In this thesis, we propose sampling-based efficient and robust mobile manipulator planners that use smart strategies to deal with computational complexity and incorporate uncertainty to generate safer plans. The first part of the research addresses the design of an efficient planner for deterministic case, where robot state is fully known, and then subsequent extension to incorporate base pose uncertainty. In the first part, we propose a Hierarchical and Adaptive Mobile Manipulator Planner (HAMP) that plans both for the base and the arm in a judicious manner - allowing the manipulator to change its configuration autonomously when needed if the current arm configuration is in collision with the environment as the mobile manipulator moves along the planned path. We show that HAMP is probabilistically complete. We then propose an extension of HAMP (HAMP-U) to account for localization uncertainty associated with the mobile base position. The advantages of our planners are illustrated and discussed. The second part of the research deals with the computational complexity involved in planning under uncertainty. For that, we propose localization aware sampling and connection strategies that help to reduce the planning time significantly with little compromise on the quality of path. In the third part, we learnt from the shortcomings of HAMP-U and took advantage of our smart strategies developed to combat the computational complexity. We propose an efficient and robust mobile manipulator planner (HAMP-BAU) that plans judiciously and considers the base pose uncertainty and the effects of this uncertainty on manipulator motions. It uses our localization aware sampling and connection strategies to consider only those nodes and edges which contribute toward better localization. This helps to find the same quality of path in shorter time. We also extend HAMP-BAU to incorporate task space constraints (HAMP-BAU-TC). Finally, in the last part of the work, we incorporate our planners (HAMP-BAU and HAMP-BAU-TC) within an integrated and fully autonomous system for mobile pick-and-place tasks in unknown static environments. We demonstrate our system both in simulation and real experiments on SFU mobile manipulator.

Keywords: Planning under uncertainty; Autonomous mobile manipulation; Sampling and Connection strategies; unknown environment; 3D exploration; mobile pick-and-place tasks

Dedication

To my parents, my brothers (Vivek, Vikas) and my wife (Varsha).

Acknowledgements

First, I would like to acknowledge my senior supervisor, Dr Kamal K. Gupta, for all the support that I received during the course of my PhD studies. I would like to thank him for educating me in writing good research papers and being patient during all the discussions we had.

I would like to thank my fellow graduate students for their constant support during this academic exploration. Thanks also go to my friends at the MENRVA lab, for helping me in constructing my final experiment setup and many more.

Finally, I would like to acknowledge the tremendous sacrifices that my mother *S. Pilania* and my father *R.S. Pilania* made to ensure that I get an excellent education. I would like to thank my younger brothers, *Vivek* and *Vikas*, for their incredible patience. It has been painful staying away from you all. I would like to thank my wife *Varsha* for her unconditional support during my studies. Whenever the PhD memories will be remembered, the episode of my final experiments, when my wife stood with me shoulder to shoulder for those continuous 48 hours experimental work (we hardly took rest for less than 5 hours), will be remembered first.

Table of Contents

List of Tables

List of Figures

Acronyms

ATACE	Alternate Task-space And C-space Exploration
BiRRT	Bi-directional Rapidly-exploring Random Tree
BRM	Belief Roadmap
BRM-LAS	Belief Roadmap with Localization Aware Sampling
CBiRRT	Constrained Bi-directional Rapidly-exploring Random Tree
EKF	Extended Kalman Filter
HAMP	Hierarchical and Adaptive Mobile manipulator Planner
HAMP-BAU	HAMP with Base uncertainty propagated to Arm motion Uncertainty
HAMP-BAU-TC	HAMP-BAU with Task Constraints
HAMP-RRT	HAMP with Rapidly-exploring Random Tree as underlying sub-planners
HAMP-BiRRT	HAMP with Bi-directional Rapidly-exploring Random Tree as underlying sub-planners
HAMP-U	HAMP with base pose Uncertainty
HERB	Home Exploring Robot Butler
IK	Inverse Kinematics
LAS	Localization Aware Sampling
LAC	Localization Aware Connection
Lazy-CPC-PRM	Lazy Collision Probability Constrained Probabilistic Roadmap
LQG	Linear Quadratic Gaussian
MCL	Monte Carlo Localization

MPV	Maximize Physical Volume
NBV-A	Next Best View of Arm
NBV-B	Next Best View of Base
OMPL	Open Motion Planning Library
POMDP	Partial Observable Markov Decision Process
PRM	Probabilistic Roadmap
ROS	Robot Operating System
RNG	Random Number Generator
RRBT	Rapidly-exploring Random Belief Tree
RRBT-TF	Rapidly-exploring Random Belief Tree Type Framework
RRBT-LAC	Rapidly-exploring Random Belief Tree with Localization Aware Connection
RRBT-LAS	Rapidly-exploring Random Belief Tree with Localization Aware Sampling
RRBT-LASC	Rapidly-exploring Random Belief Tree with Localization Aware Sampling and Connection
RRG	Rapidly-exploring Random Graph
RRT	Rapidly-exploring Random Tree
SLAM	Simultaneous Localization And Mapping
UAV	Unmanned Aerial Vehicle

Chapter 1

Introduction

1.1 Introduction

Robots that use mobility and manipulation capabilities, such as wheeled mobile manipulators or humanoid robots, can be seen as an attempt by robotics fraternity to design a machine to imitate various capabilities - mobility, manipulation, etc. - of a human being. The utility of such robots, generally called as mobile manipulators, has been demonstrated in a range of indoor and outdoor applications, for example, to assist elder people [1], helper in household chores [2], [3], for search and rescue operations [4], [5], and planetary exploration [6]. The Intel HERB mobile manipulation platform has demonstrated impressive capabilities ranging from pick-and-place objects (collecting the objects and loading them into a dishwasher rack) [2], [7] to push-based manipulation on tabletop environments [8]. One of the key components that imparts intelligence to mobile manipulators is motion planning where a planner plans a collision-free path from start to goal (base poses and manipulator configurations) by minimizing a cost metric, for example, path length. The mobile manipulator motion planning is the main focus of our research work. Even in motion planning, there can be two broad categories, one, the planners for deterministic case where robot state is fully known and second, the planners for stochastic case where robot state is partially known (or uncertainty in robot state). The second category is also popularly known as motion planning under uncertainty. Here, the word stochastic is used in context of uncertainty associated with the mobile base position. In this thesis, we contribute to both the categories. Our contributions are described in four phases and evaluated in context to wheeled mobile manipulators, however, at a certain level of abstraction, they can be extended to humanoid robots as well. First, we consider the mobile manipulator planners for deterministic case. Most work [9, 10, 11] usually takes a very conservative approach, which is, to fold the arm to some safe "home" configuration and then plan for a 2D projected footprint of the mobile manipulator in a projected 2D representation of the world from start base pose to goal base pose. Clearly, this approach has two main limitations: (i) the projection of the mobile manipulator with extended arm may have a large footprint, and may be in collision with 2D projected map, while the mobile manipulator is collision-free in 3D map, and more fundamentally (ii) it may not always possible to change the arm to a predefined home configuration at base's start pose because of physical constraints or there may be task constraints that prevent the arm being folded, e.g., if the robot is carrying a glass of liquid which needs to be kept vertical to avoid spillage. Another example is where the mobile manipulator is carrying a long payload, say a pole and it needs to continuously move the arm (and thereby the pole to avoid the pole colliding with walls and other objects in the environment) to navigate through the doors and hallways. In such scenarios, mobile manipulator with arm in start configuration can not reach the goal unless it changes the arm configuration several times along the path.

One possible solution to this motion planning problem is to use sampling based planners [12, 13] in full configuration (C-space) of the mobile manipulator. However, besides being somewhat computationally expensive, the computed path for the mobile manipulator may result into undesired and excessive motions for the manipulator. This is primarily because of the randomness associated with sampling based planners and persists even after applying a post processing smoothing filter. In most scenarios, there is no need to move manipulator except at certain base poses - the undesired arm motion (post smoothing) refers to this extraneous manipulator motion while the base is moving. We would like to avoid such undesired manipulator motions. Furthermore, it is generally difficult to ensure tight error bounds on the mobile base that are comparable to those for the arm and hence synchronizing controllers between the two can be difficult. Therefore, it is quite reasonable to execute arm and base motions sequentially, and within this overall paradigm, our proposed planner, as outlined below, is quite reasonable.

The first part of the research addresses the design of an efficient mobile manipulator planner for deterministic case. We propose a Hierarchical and Adaptive Mobile Manipulator Planner (HAMP) that plans both for the base and the arm in a judicious manner - allowing the manipulator to change its configuration autonomously when needed if the current arm configuration is in collision with the environment as the mobile manipulator moves along the planned path. Our planner first constructs a base roadmap (using Probabilistic Roadmap (PRM) in the base configuration space) and then for each node in the roadmap it checks for collision status of current manipulator configuration along the edges formed with adjacent nodes, if the current manipulator configuration is in collision, the manipulator C-space is searched for a new reachable configuration such that it is collision-free as the mobile manipulator moves along the edge and a path from current configuration to the new reachable configuration is computed. If no such manipulator configuration is found, then a new edge will be searched for in the base roadmap, and the process repeats.

Summarizing, HAMP searches in two sub-spaces (base sub-space and manipulator sub-space) in a novel way and on a "need to" basis, i.e., the search in manipulator space is invoked only for those points in the base-space where it is needed. Hence, HAMP searches a much smaller size of space, as a result, it computes paths in shorter time with higher success rate than a search in the full configuration space of the mobile manipulator, and more importantly, it also avoids unnecessary motion of the arm, as is the case for the full search. Both these key points are validated in our experiments. Our choice to use PRM as underlying core sub-planner (for base and for the manipulator) within the HAMP framework is primarily because when we incorporate base uncertainty (as explained later in this section) in the mobile manipulator paths, it allows us to optimize the paths with respect to the base uncertainty (at the goal). This would not be the case if we were to use tree versions of sampling based planners (such as RRT [12]) as core sub-planners within HAMP. However, RRT (in the absence of uncertainty) is generally more efficient in terms of planning time than PRM, especially the Bi-directional RRT [14]. Therefore, we also evaluated the tree versions of HAMP with RRT and Bi-directional RRT as the core underlying sub-planners. Moreover, we provide a mathematical proof to show that HAMP is probabilistically complete.

Safe execution of motion plans is of critical importance for many robotic tasks. As a result of uncertainty associated with a robot's motion and its sensory readings, the true robot state is not available. Deterministic planners (including HAMP) assume deterministic motion and leave the issues of uncertainty to the control phase in which the path is executed with a feedback controller. However, a mobile base inherently has localization uncertainty due to wheel slippage and other unmodeled errors. Therefore, a planning method must account for these uncertainties for safe and collision-free execution of motion plans. Partially observable Markov decision process (POMDP) [15] is a general framework to deal with motion and sensing uncertainty, however due to its significant complexity, solving realistic problems with large state spaces remains a challenge, even though progress has been made on the efficiency issues of these approaches [16, 17, 18, 19]. A class of methods that carries robot state and associated uncertainty is an approximation to POMDP. Among them, a sub-class [20, 21, 22] assumes the presence of landmark regions in the environment where accumulated motion uncertainty can be "reset". Another sub-class [23, 24, 25, 26, 27, 28, 29] uses sampling-based methods (graph-based and tree-based) where uncertainty is propagated from start to goal. We call this sub-class as sampling-based stochastic motion planners.

In this thesis, we address the sampling-based stochastic motion planners for mobile manipulators. Motion planning under uncertainty has made considerable progress over the past few years for mobile robots and unmanned aerial vehicles (UAVs) but still largely ignored for mobile manipulators. One key reason might be because most work usually take conservative approach, which is, to fold the arm to some "safe home configuration" and treat mobile manipulator essentially as a mobile robot where planners designed for them can be used to deal with uncertainty. Although, we assume that the motion of manipulator, in and of itself, is quite accurate (a reasonable assumption, give the joint encoders are quite precise), however, the uncertainty in mobile base position causes some repercussions on manipulator (robotic arm) motion and grasping task (robotic hand). We group the base pose uncertainty and its effects in three levels: Level 1 - the base pose uncertainty is not translated to manipulator motion and grasping task; Level 2 - the base pose uncertainty is translated to manipulator motion but not to grasping task; Level 3 - in addition to Level 2, the base pose uncertainty is also translated to grasping task. Please note that even for a fixed mobile base, there is uncertainty associated with grasping tasks [30], which arises mainly from errors in object localization and perceived shape. Most of the existing grasping approaches deal with uncertainty in the execution stage using reactive grasp execution controller where feedback from range or image sensors and tactile sensors is used to guide the gripper. In this thesis, we consider only Level 1 and Level 2 and leave the grasping uncertainty (Level 3) for future work.

The first part of the research also addresses the extension of HAMP to incorporate Level 1 base pose uncertainty. We propose a mobile manipulator planner, HAMP-U, that uses belief space planning to account for localization uncertainty associated with the mobile base position and ensures that the resultant path for the mobile manipulator has low uncertainty at the goal. A probability distribution over all possible states is referred as belief and the set of all possible distributions is called belief space. Our experimental results show that the paths generated by HAMP-U are less likely to result in collision and are safer to execute than those generated by HAMP (without incorporating uncertainty), thereby showing the importance of incorporating base pose uncertainty in our overall HAMP algorithm.

Although HAMP-U helped in generating paths which are less likely to result in collision, still these paths are not completely safe for execution. There are two main reasons for this: a) HAMP-U does not consider the effect of base pose uncertainty on manipulator motions, b) HAMP-U assumes the robot is at the mean position (of belief, represented by uncertainty ellipse) and checks collision from mean position to mean position. In both cases if the mobile base slightly deviates from its intended path then the arm (which was in some collision-free configuration along intended path but may not necessarily hold along the deviated path) could collide with the surrounding obstacles. Below, we mention the problems involved in addressing above mentioned issues, followed by the solutions.

It is important to note that addressing the shortcomings of HAMP-U requires involvement of uncertainty related techniques at different stages of path planning. This in turn requires costly operation of 3D collision checks, another possible reason why planning under uncertainty for mobile manipulators may be largely ignored. Designing a reliable mobile manipulator planner is difficult to realise for real time applications unless one addresses the issue of computational complexity involved with mobile manipulator planning. [31] deals with it by using a multi-layered 2D representation of both the robot and the environment. However, since the planning is still carried out only for the base and not the manipulator, their approach will fail in the scenarios where arm configuration needs to be changed while navigating from start to goal. Our strategy in HAMP helps to avoid unnecessary 3D collision checks without being overtly conservative. It first checks the 2D projected footprint of the base against the 2D representation of the world (obtained from projecting 3D range date up to certain height), and if it is collision-free then a 3D collision check is performed on the manipulator. Although, our strategy helps to reduce the planning time (from collision checks perspective), we need to find additional efficiencies for mobile manipulator planning under uncertainty as incorporating uncertainty further increases the computational time.

One way is to look at next level, i.e., after doing collision checks for the sampled point, do we really need to retain all the points (nodes) or the local paths (edges) connecting two points. A good decision at this level (before connecting the sampled point to the graph or tree) could help us improve the run time associated with mobile manipulator planning. Note that expensive 3D collision checking is not the only factor involved in computational complexity, there is another facet to it other than the high-dimensional full configuration space (which we handle in HAMP by planning in two different sub-spaces). It is important to understand why incorporation of uncertainty makes the mobile manipulator planning computationally expensive.

The sampling-based stochastic motion planners can be implemented either in an incremental (graph-based [28] or tree-based [24, 25, 27]) or in a non-incremental way (graph-based [23, 26, 27, 24]). These planners are computationally demanding as compared to their counterparts that do not consider uncertainty (deterministic motion planners). This is because they do not follow the "optimal substructure" property of paths, i.e., the incurred costs on different edges depend on each other. To compute the cost of an edge emanating from a node, the full knowledge of belief (robot pose and associated uncertainty) at the node is required, this in turn requires full knowledge of the history of observations and actions leading up to the node. [29] is an exception in the sense that the incurred costs on different edges do not depend on each other. This comes at the cost of some simplifying assumptions including holonomic robot and Gaussian belief for robot states with trivial dynamics. The computational cost further increases if an edge cost in these planners uses collision probability [25, 27, 29], computation of which depends on the beliefs along that edge. Furthermore, this cost will go up drastically if collision checks are carried out in 3D (for example, for mobile manipulators). Since the time consuming step in stochastic motion planners arises from the uncertainty propagation along the edges, incremental stochastic planners can be computationally more demanding as compared to non-incremental ones where search mechanism is carried out only once while in former, search mechanism is repeated every time a new sample is added to the roadmap. For real time applications, for instance to facilitate anytime planning [32], it is important to reduce this run time. At least part of this run time reduction can be achieved by "smart" sampling and connection strategies. Current stochastic motion planners [23, 24, 25, 26, 27, 28, 29] use traditional sampling and connection strategies which are designed for deterministic motion planners and address the issue of uncertainty at path search phase. These strategies add unnecessary nodes and edges that do not contribute to better localization. This leads to a dense roadmap which in turn increases the computational cost.

In the second part of our research, we propose efficient localization aware sampling and connection strategies to bring down the computational cost for sampling-based stochastic motion planners. The localization aware sampling strategy avoids putting large number of samples by considering the "localization ability" of a new sample relative to its neighbouring nodes. It puts more samples in regions where sensor data is able to achieve higher uncertainty reduction while maintaining an adequate number of samples in regions where uncertainty reduction is poor. This leads to a less dense roadmap that results in significant time savings in the path search phase. Note that localization of a robot at a point depends on 1) the path taken to reach the point and 2) on the update based on sensor model. However, at the sampling stage the path taken to a node is not available. We develop a new measure of "localization ability of a sample" that "extracts" how well a sensor observation at a sample point reduces uncertainty without explicitly knowing the path leading to it and use this measure to design a localization aware sampling strategy.

A key reason we use reduction in uncertainty as a measure is that higher uncertainty is more detrimental and hence has higher cost for many tasks. Nevertheless, one possible consequence of our sampling technique is that path quality (we use true localization uncertainty along the path as a quality metric) may suffer, if the path passes through regions where uncertainty reduction is poor. Via simulation results, we show that, at least empirically, there is little compromise in path quality. Furthermore, note that since at the sampling stage, true localization uncertainty is not available, a cost function metric using it can not be computed, hence can not be used. The best one can do is to use the uncertainty reduction ability of the sensor at the sample point, as we do. Note that in the search phase (where edges are added and uncertainty is propagate along the path), appropriate cost function is still minimized.

The localization aware connection strategy first connects the new sample to a nearest node (chosen based on an uncertainty metric and not on distance metric) and then to other neighbouring nodes. Connection from new sample to a neighbouring node is made only if the new path to that node reduces the uncertainty. Our efficient connection strategy eliminates the inefficient edges that would be created in current connection schemes but do not contribute toward better localization. As a result, it also reduces the number of search queue iterations needed to update the paths. This helps to find a well-localized path in shorter time with no compromise on the quality of path. Note that our strategy is applicable to graph-based incremental stochastic planners that maintain a single belief at a node and is not applicable to planners with multiple beliefs. Multiple beliefs at a node are needed for planners that optimize multiple objective functions since multiple paths to a node can not be completely ordered (as is the case for a single objective function, which can be a weighted sum of multiple costs) and need to be kept so as not to prematurely prune an optimal one, although domination criteria can be used to do some pruning (see [28, 33], the later is more specific to manipulator planning for fixed base). Of course, tree-based methods, by definition, have single belief since they have a unique path to any given node. Our simulation results showed that by using these smart strategies, the planning time can be reduced significantly with little compromise on the quality of path. Our simulations involved mobile robot (2D planning), therefore, it is expected that the savings in planning time will be even higher for mobile manipulator (as we show in our third part of research). The probabilistic completeness issues with our approaches are also discussed.

In the third part of our research, we integrated HAMP with the smart sampling and connection strategies to implement an efficient and robust mobile manipulator planner (HAMP-BAU) that plans judiciously and considers the base pose uncertainty and the effects of this uncertainty on manipulator motions (Level 2). It uses our localization aware sampling and connection strategies to consider only those nodes and edges which contribute toward better localization. Moreover, it respects the collision probability threshold along the path and uncertainty threshold at goal. We also propose an extension of HAMP-BAU to incorporate task space constraints (we call the resultant planner as HAMP-BAU-TC). We evaluated both the planners and show that our planners find a safer path as compared to other variants where uncertainty is not considered at different levels, for example, not incorporating base uncertainty on manipulator plans, not respecting collision probability threshold along the edges. We also show that the variants of these planners that do not use our localization aware sampling and connection strategies will take longer to find the same quality of path.

Finally, in the last part of work, we incorporate our planners (HAMP-BAU and HAMP-BAU-TC) within an integrated and fully autonomous system for mobile[1] pick-and-place tasks in unknown static environments. A key aspect of our integrated system is that the planner works in tandem with base and arm exploration (view planning) modules that explore the unknown environment. Note that we assume unknown areas of environment as obstacles and not free. The task of base exploration is to take the mobile manipulator (mainly mobile base) to next best view of the base (NBV-B), take a scan using a 3D sensor (Kinect, mounted on the mobile base) and then invoke arm exploration which scans the local region surrounding the manipulator using a 2D sensor (Hokuyo, mounted at end-effector acting as eye-in-hand) by reaching to next best views of arm (NBVs-A). From 2D sensor (or a 2D scan) we mean line scan while from 3D sensor (or a 3D scan) we mean area scan. Also note that since the eye-in-hand sensor can provide only line scans, therefore, at NBV-A, the sensor rotates to make an area scan by collecting all the line scans during rotation. Scans from both Kinect and Hokuyo sensors are inserted into a global Octomap [34]. The base exploration module works on a 2D occupancy grid map to compute a NBV-B and uses

[1]The world “mobile” emphasizes that the mobile manipulator is required to move from one location to another.

HAMP-BAU to plan a path for it. This 2D occupancy grid map is obtained by the fusion of two 2D maps - one is from down projection of global Octomap upto a certain height and other is from SLAM (simultaneous localization and mapping). There is a third sensor (LMS100, mounted at base bottom) which feed SLAM. Please see Figure **??** for system components (mobile manipulator, sensors and their mounting locations) and Figure **??** for how the sensor information is used and different maps are obtained. On the other hand, the arm exploration module works on a local Voxelmap (obtained from global Octomap) to compute NBVs-A and uses a manipulator planner (that considers base pose uncertainty) to reach there. After each scan, the arm exploration module updates the Voxelmap and repeat the procedure until the Voxelmap is fully explored. It is important to note that since sensor scans are not directly inserted into Voxelmap, therefore, the status (occupied, free, unknown) of each voxel cell in the Voxelmap is updated by communicating with global Octomap. We also want to state that in addition to scans taken by respective sensors at NBV-B and NBV-A, the scans collected during the mobile manipulator motion to reach NBV-B or the manipulator (end-effector) motion to reach NBV-A are also incorporated into the global Octomap. Our system is implemented both in simulation and on the actual SFU mobile manipulator. Please note that the pick and place modules that we use in our integrated system are very quick and an ad hoc attempt to be able to show the system for pick-and-place task. In future, it will be replaced by a systematic approach.

1.2 Related Work

In this section, we review the related work and place our research work into context. We consider the work concerning mobile manipulator motion planning, planning under uncertainty, sampling and connection strategies, mobile manipulator based autonomous systems in unknown environment.

1.2.1 Mobile Manipulator Motion Planning

Most of the previous work [35, 36, 37, 38] on mobile manipulation mainly deals with the coordination of the mobile base and the manipulator motion for following a given end effector trajectory. In motion planning related work, [10] and [11] use a compact 3D representation of the environment, but path planning is accomplished in a projected 2D environment representation with a 2D footprint of the mobile manipulator. Such an approach will fail, for example, where the mobile manipulator is required to push and store a cart under a table. [31] improved upon [10, 11] using a multi-layered 2D representation of both the robot and the environment. However, since the planning is still carried out only for the base and not the manipulator, their approach will fail in the scenarios where arm configuration needs to be changed while navigating from start to goal. [39] proposed an adaptive approach for efficient humanoid robot navigation, which allows for finding solutions for foot-step planning where planning based on a 2D grid fails. Our approach (HAMP) has a similar adaptive flavour, but it is in the context of mobile manipulation and not foot step planning. In the context of mobile manipulators, hierarchical strategies have been used to estimate reachable workspace [40]. An interesting use of adaptive dimensionality has recently been introduced in [41]. Their approach uses deterministic search (A* over discretized C-space) in a low dimensional end-effector C-space interleaved with tracking in the full mobile manipulator C-space. It is shown that the resulting planner outperforms a full dimensional RRT in a class of tasks where the end-effector is carrying a large payload. One could characterize this approach toward the "greedy" end of the spectrum since the search is, in effect, guided by a path for the end-effector. While this approach could be used in a relatively small region near the goal, as shown in the example tasks in the above mentioned paper, a key problem is that due to its deterministic search, it is not applicable to relatively large areas as is the case in our examples. Finally a genetic optimization based planner for a mobile manipulator that plans motions in real time in dynamic environments is presented in [42]. The planner takes advantage of redundancy in optimizing overall motion via randomly invoking a "stop" genetic operator that allows for either the base or the manipulator to remain stationary during a portion of the trajectory. Note that the performance of genetic optimization relies on maintaining a diverse population of trajectories that belong to different homotopic groups which is a significant challenge.

1.2.2 Sampling Based Planning Under Uncertainty

While standard motion planning algorithms often assume that a mobile base can track its position reliably during path execution stage (as is the case with HAMP), in reality, there is always some uncertainty associated with mobile base position. The uncertainty typically originates from three sources: (i) motion uncertainty - uncertainty in a robot's motion often caused by factors such as wheel slippage, (ii) sensor uncertainty - uncertainty in its sensory readings, and (iii) map uncertainty - uncertainty in the environment map or imperfect locations of features (information sources) in the environment.

Planning under uncertainty has made considerable progress over the past few years for mobile robots. For example, approaches in [26, 25, 24, 43, 23, 44], essentially add an uncertainty dimension(s) to the robot state and each belief state then is a combination of robot state and the associated uncertainty. An attractive aspect of Belief Roadmap (BRM) [26] is that, while it explicitly simulates measurements along candidate paths and then chooses the path with minimal uncertainty at the goal, it uses covariance factorization techniques to significantly reduce the computation burden of this process but with the assumption of maximum likelihood observation, i.e., the controller is capable of driving the state estimate back to the desired path. More recent approaches have also accounted for the controller in the planning stage, e.g., [27, 28, 29], however, there is significant increase in the computational cost. These planners only consider the motion and sensing uncertainty.

Approaches in [45, 46, 47, 48] consider the mapping uncertainty about the environment but not the motion and sensing uncertainty.

While mobile robotics literature (mobile base only) has extensively considered uncertainty (world is 2-dimensional in most of these cases, although some recent work has considered 3D world, but for SLAM and not planning), to the best of our knowledge, this uncertainty is largely ignored in mobile manipulation. [33] considered this, but for the case of fixed mobile base only.

1.2.3 Sampling Strategies

A large number of sampling schemes have been used with the standard (without uncertainty) sampling based planners (RRT or PRM) such as, sample around and near the obstacles, or in narrow corridors, medial axis sampling to sample far away from the obstacles, use visibility to reduce the number of samples, adaptive strategies such as restrict sampling to size-varying balls around nodes, entropy guided approaches, etc. [49] and [50] provide a survey of recent work in non-uniform sampling for PRMs. Above mentioned sampling approaches do not consider the uncertainty associated with robot and its sensors.

[45] proposed an approach where the sampling strategy incorporates mapping uncertainty (they do not consider localization uncertainty that we consider in this paper) in which the decision to accept or reject a sample is based on its collision probability (computed using each of the possible world model). However, the issue of "how good a sample would be in localizing the robot?", which we explicitly consider does not arise in their problem context. As mentioned earlier, computing the collision probability in the presence of localization uncertainty of a sample right at the sampling stage, i.e., before connecting it to the roadmap is not possible. Note that at sampling stage we consider only sensing uncertainty while for path search (where uncertainty is propagated from start) we consider both motion and sensing uncertainty. To the best of our knowledge, we are not aware of any other sampling approach that considers uncertainty. All sampling-based stochastic motion planners [23, 24, 25, 26, 27, 28, 29] use one of the sampling techniques from deterministic motion planners and address the motion and sensing uncertainty at path search phase by propagating uncertainty from start to goal.

Although not directly related to motion planning (or sampling techniques), the notion of uncertainty has been used in the past to select the best sample (the next best goal of robot) for search and exploration. For example: [51] first plans for each of the possible goal candidates and selects the one (as next best goal) which in addition to information maximization (unknown region), also has good localization along the path.

1.2.4 Connection Strategies

Connection strategies used in sampling-based deterministic motion planners simply connect the new sample to the neighbouring nodes within in a fixed size ball or size varying ball. A thorough discussion on these strategies can be found in [52, 53] while for more recent updates we refer to [49, 54]. These approaches do not account for uncertainty associated with robot and its sensors.

All [23, 24, 25, 26, 27, 28, 29] of the sampling-based stochastic motion planners that consider uncertainty inherit the connection strategy from deterministic motion planners. Among incremental planners, [28] is obliged to use traditional connection strategy as they optimized multiple objective functions, hence are required to maintain multiple paths to (hence multiple beliefs at) a node in order to guarantee not to prune an optimal path, although some pruning can be done via domination criteria. To the best of our knowledge, the work of [28] is the only roadmap (graph) based stochastic motion planner that works in an incremental fashion. Although it is designed for a set of beliefs, the same strategy also works for the case of single belief at a node. We call their algorithm (RRBT) with single belief as RRBT type framework (RRBT-TF). It minimizes the uncertainty at goal while respecting the chance-constraints (threshold on uncertainty) along the path. Planners in [23, 24, 25, 26, 27, 29] also use single belief at a node but they do not incrementally construct the roadmap.

The problem with the use of traditional connection strategy for incremental stochastic planners is that it even considers those edges which do not contribute toward better localization. With the inclusion of such edges, the planning time increases, however, the same quality of path can be find in lesser time if we eliminate these edges. This is exactly what our localization aware connection strategy does. It eliminates those edges which do not contribute toward better localization.

Similar to graph-based incremental stochastic planners, current tree-based stochastic motion planners [24, 25, 27] also inherit the connection strategy from tree-based deterministic motion planners. There the EXTEND step simply connects the sample to nearest node (distance based) and then propagate the uncertainty to it. However, this does not provide the least uncertain path to the sample. Instead, our connection strategy will connect the sample to a neighbouring node (within a ball) the uncertainty propagation from which gives minimal uncertainty at the sample. We use the additional “rewiring" notion of RRT* [54], albeit with uncertainty metric, to rewire the connections to the neighbouring nodes.

1.2.5 Mobile manipulator based autonomous systems in unknown environment

Please note that in this section we do not talk about individual exploration (view planning algorithms) techniques for either base or the arm. There is huge literature on that and a good review can be found in [55]. But we review the related work on integrated and autonomous systems that use mobile manipulator for some application in unknown environment. Note that we consider unknown regions of the environment as obstacles and not collision-free regions (which can be detrimental) as the assumption in [56]. Neither the work in [56] uses view planning to explore the environment. It just incorporates the sensor readings (from a 3D sensor mounted on the base and not acting as eye-in-hand) as the mobile manipulator moves along the path that follows end-effector trajectory. [57] searches for an object in the unknown environment using a planar range sensor mounted at end-effector but mobile base was fixed in their experiments. A system proposed by Lila Torabi [55, 58, 59] (an earlier Ph.D. Thesis work from our Robotic Algorithms and Motion Planning (RAMP) Lab) autonomously builds a 3D model of an object placed in unknown environment. The work considers decoupled approach for mobile manipulator planning and moreover, uncertainty is not considered. There is lot of work related to environment exploration and mapping both in 2D and 3D either using mobile base [51] or UAVs [60]. However, we have not come across any work where mobile manipulator is used to explore the unknown environment and achieve some tasks (for example, mobile pick-and-place). We believe that our system is first of its kind in many ways: a) it is the first integrated application that explores the unknown environment, picks the object (once the object is deemed to be in the known region) and then further explores the environment with object in hand and places it at target location only after the place location is deemed to be in the known region, b) it combines two different exploration schemes into one - uses frontier based exploration for the base and information gain maximization (in workspace) based exploration technique for the arm, and finally c) how the scans from multiple sensors are integrated and then used for base and arm view planning.

1.3 Contributions

There are two broad visions associated with this thesis. The first is to design an efficient and reliable mobile manipulator planner that plans judiciously both for the base and the arm and considers the base pose uncertainty and the effects of this uncertainty on manipulator plans. The second is to integrate such planner into a system that autonomously explores the unknown environment to complete an assigned task, for example, mobile pick-and-place task in our case. The key contributions of the thesis are listed below:

- We designed a novel mobile manipulator planner (HAMP) for deterministic case that plans both for the base and the arm in a judicious manner. We also evaluated the tree versions of HAMP with RRT and Bi-directional RRT as the core underlying sub-planners. A mathematical proof is also provided to show that HAMP is probabilistically complete. Furthermore, we extended HAMP to design a new planner (HAMP-U) that incorporates localization uncertainty associated with the mobile base position. These works have been published in [61] and [62].

- We designed novel efficient localization aware sampling and connection strategies for sampling based motion planning under uncertainty. For sampling, we developed a new measure of "localization ability of a sample" that "extracts" how well a sensor observation at a sample point reduces uncertainty without explicitly knowing the path leading to it. A mathematical proof is also provided to show the probabilistic completeness of our sampling strategy under some reasonable conditions on parameters. The sampling work has been published in [63], while the connection work has been published in [64].

- Integrating the above two components, we designed an efficient and robust mobile manipulator planner (HAMP-BAU) that plans judiciously and incorporates the base pose uncertainty and the effects of this uncertainty on manipulator plans. We also extended HAMP-BAU to incorporate task space constraints (HAMP-BAU-TC). This work is reported in [65].

- We incorporated HAMP-BAU and HAMP-BAU-TC in an integrated and fully autonomous system for mobile pick-and-place tasks in unknown static environments. The system is demonstrated both in simulations and real experiments on SFU mobile manipulator. This work is also reported in [65].

1.4 Outline of Thesis

The rest of the thesis is organised as follows. In Chapter 2 we present a mobile manipulator planner (HAMP) for deterministic case, its probabilistic completeness proof and extension to incorporate base pose uncertainty (HAMP-U). In Chapter 3 we present localization aware sampling and connection strategies for sampling based motion planning under uncertainty. In Chapter 4 we present a more advanced and safer mobile manipulator planner (HAMP-BAU) and its extension (HAMP-BAU-TC) to incorporate task space constraints. In Chapter 5 we describe a mobile manipulator based autonomous system for mobile pick-and-place tasks in unknown environment and demonstrate the system in simulations and real experiments on SFU mobile manipulator. We conclude in Chapter 6.

Chapter 2

Conclusions and Future Work

2.1 Conclusions

In this thesis, we proposed sampling-based efficient and robust mobile manipulator planners that use efficient and smart strategies to deal with computational complexity and incorporate uncertainty to generate safer plans. In the first part of research, we addressed the design of an efficient mobile manipulator planner for deterministic case, where robot state is fully known. For that, we proposed a Hierarchical and Adaptive Mobile Manipulator Planner (HAMP) that plans both for the base and the arm in a judicious manner - allowing the manipulator to change its configuration autonomously when needed if the current arm configuration is in collision with the environment as the mobile manipulator moves along the planned path. We showed that HAMP is probabilistically complete. We extensively evaluated HAMP in different scenarios with varying levels of complexity. We also evaluated the tree versions of HAMP with RRT and Bi-directional RRT (BiRRT) as the core sub-planners for searching for both the base and the manipulator, respectively called HAMP-RRT and HAMP-BiRRT.

In the second part, we proposed localization aware sampling and connection strategies (LAS and LAC, respectively) to consider only those nodes and edges which contribute toward better localization. Our novel sampling strategy judiciously places the samples using a new notion of "localization ability of a sample", i.e., it puts more samples in regions where sensor data is able to achieve higher uncertainty reduction while maintaining adequate samples in regions where uncertainty reduction is poor. Our simulation results showed that these strategies help to reduce the planning time significantly with little compromise on the quality of path. We also discussed probabilistic completeness and optimality issues associated with our strategies.

Third, integrating the above two components (LAS and LAC with in HAMP), we designed an efficient and robust mobile manipulator planner (HAMP-BAU) that plans judiciously and incorporates the base pose uncertainty and the effects of this uncertainty on manipulator plans. We also extended HAMP-BAU to incorporate task space constraints (HAMP-BAU-TC). We evaluated both planners in known environment and showed that our planners find a safer path as compared to other variants where uncertainty is not considered at different levels.

Finally, in the last part of the work, we incorporated our planners (HAMP-BAU and HAMP-BAU-TC) within an integrated and fully autonomous system for mobile pick-and-place tasks in unknown static environments. A key aspect of our integrated system is that the planner works in tandem with base and arm exploration modules that explore the unknown environment. We demonstrated our system both in simulation and real experiments on SFU mobile manipulator.

2.2 Future Work

Below, we suggest few directions to further extend our work:-

- HAMP can be enhanced in term of reduction in planning time. For example, one key enhancement could be a reachable manipulator configuration, serve as a goal for reconfiguration path, should be searched in task space instead of C-space. This was noticed during evaluation of HAMP-BAU-TC. Our experience tells that it is one of the time consuming steps. Furthermore, there is a scope to improve the path search phase.

- HAMP basically decomposes the full space into two sub-spaces, i.e., base sub-space and manipulator sub-space. In this particular example, the decomposition is quite natural and is motivated by the fact that in a majority of day to day indoor environments, it is often the case that when the base moves the arm does not need to move to avoid collisions except at few configurations where arm reconfiguration can take place while the base is stationary. In principle, the HAMP framework can be applied to any robotic system by decomposing it into two (or more) sub-spaces and then searching them in HAMP type manner. This approach may not always lead to efficient planners and an interesting question to explore would be under what conditions such a decomposition would lead to more efficient planners rather than searching the full space ?

- We believe that L_n can be extended to the multimodal distribution using Monte Carlo localization (MCL) [66] that uses a particle filter to represent the distribution of likely states, with each particle representing a possible robot state. The MCL algorithm works in two stages. First, it uses the motion model to shift the particles to predict its new state after the motion and the likelihood (weight) of each new particle is computed using sensor measurements. In the second stage, the particles are resampled based on how well the actual sensed data correlate with the predicted state.

To extend our L_n measure to multimodal distribution, we bypass the prediction of new particles based on the robot motion as we do not know the control commands at the sampling stage. The procedure to compute the localization ability of a sample is then as follows. We could assume a fixed distribution of particles around a sampled point with each particle assigned the same weight. This set of particles essentially serves the same role as M for the Gaussian case. The same distribution is used for all samples by appropriately transforming corresponding to the co-ordinates of the sample points. Sensor measurement step is then used to assign new weights to each particle followed by a resampling step as in standard MCL. This new set of particles essentially serves the same role as Σ_n for the Gaussian case. Kullback-Leibler divergence [67] that measures the information gain between two probability distributions can then be used as localization ability of a sample.

- There is ample scope of improvement in HAMP-BAU, especially the computation of reconfiguration paths by considering the base pose uncertainty. Presently, we use Lazy-CPC-PRM and this manipulator planner fails to find a path even in simple environment if base pose uncertainty is high (and also depending on the threshold used there). One possible solution is that the planner should give the best scenario path even if it fails to find one that satisfies collision probability threshold.
- Our integrated and autonomous system (described in Chapter 5) does not consider a systematic approach for grasping task, it assumes a given grasp pose. Therefore, first an appropriate grasp planner should be integrated with in the system by replacing the PICK module. Thereafter, the respective grasp planner can be extended to incorporate uncertainty associated with grasping tasks. So, there is a possibility of extending our work in that direction.

Bibliography

[1] A. Jain and C. Kemp, "EL-E: An assistive mobile manipulator that autonomously fetches objects from flat surfaces," *Autonomous Robots*, vol. 28, no. 1, pp. 45–64, 2010.

[2] S. Srinivasa, D. Ferguson, C. Helfrich, D. Berenson, A. Collet, R. Diankov, G. Gallagher, G. Hollinger, J. Kuffner, and M. VandeWeghe, "HERB: A home exploring robotic butler," *Autonomous Robots*, vol. 28, no. 1, pp. 5–20, 2010.

[3] M. Beetz, U. Klank, I. Kresse, A. Maldonado, L. Mosenlechner, D. Pangercic, T. Ruhr, and M. Tenorth, "Robotic roommates making pancakes," in *Proc.of the 11th IEEE-RAS International Conference on Humanoid Robots*, Bled, Slovenia, 2011, pp. 529–536.

[4] J. Stuckler, M. Schwarz, M. Schadler, A. Topalidou-Kyniazopoulou, and S. Behnke, "NimbRo Explorer: Semi-autonomous exploration and mobile manipulation in rough terrain," *to appear in Journal of Field Robotics*, 2015, (accessed on November 20, 2015). [Online]. Available: http://www.ais.uni-bonn.de/papers/JFR_2015_NimbRo_Explorer.pdf

[5] A. S. et al, "Chimp, the cmu highly intelligent mobile platform," *Journal of Field Robotics*, vol. 32, no. 2, pp. 209–228, 2015.

[6] M. Maimone, Y. Cheng, and L. Matthies, "Two years of visual odometry on the mars exploration rovers," *Journal of Field Robotics*, vol. 24, 2007.

[7] S. Srinivasa, D. Ferguson, J. M. Vandeweghe, R. Diankov, D. Berenson, C. Helfrich, and H. Strasdat, "The robotic busboy: Steps towards developing a mobile robotic home assistant," in *International Conference on Intelligent Autonomous Systems*, 2008, pp. 129–136.

[8] M. Dogar and S. S. Srinivasa., "Push-grasping with dexterous hands: Mechanics and a method," in *Proc.of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 2123–2130.

[9] D. Berenson, J. Kuffner, and H. Choset, "An optimization approach to planning for mobile manipulation," in *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2008, pp. 1187–1192.

[10] E. Marder-Eppstein, E. Berger, T. Foote, B. P. Gerkey, and K. Konolige, "The office marathon: Robust navigation in an indoor office environment," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2010, pp. 300–307.

[11] J. Scholz, S. Chitta, B. Marthi, and M. Likhachev, "Cart pushing with a mobile manipulation system: Towards navigation with moveable objects," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China, May 2011, pp. 6115–6120.

[12] S. M. Lavalle, "Rapidly-exploring random trees: A new tool for path planning," Computer Science Dept., Iowa State University, Tech. Rep. 98-11, 1998.

[13] L. E. Kavraki, P. Svestka, J.-C. Latombe, and M. H. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 4, pp. 566–580, 1996.

[14] J. Kuffner and S. LaValle, "RRT-connect: An efficient approach to single-query path planning," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, 2000, pp. 995–1001.

[15] L. Kaelbling, M. Littman, and A. Cassandra, "Planning and acting in partially observable stochastic domains," *Artificial Intelligence*, vol. 101, pp. 99–134, 1998.

[16] J. Pineau, G. Gordon, and S. Thrunn, "Point-based value iteration: An anytime algorithm for POMDPs," in *International Joint Conferences on Artificial Intelligence*, August 2003, pp. 1025–1032.

[17] H. Kurniawati, Y. Du, D. Hsu, and W. S. Lee, "Motion planning under uncertainty for robotic tasks with long time horizons," *International Journal of Robotics Research*, vol. 30, no. 3, pp. 308–323, 2011.

[18] H. Kurniawati, T. Bandyopadhyay, and N. Patrikalakis, "Global motion planning under uncertain motion, sensing, and environment map," *Autonomous Robots*, vol. 33, no. 3, pp. 255–272, 2012.

[19] H. Bai, D. Hsu, and W. S. Lee, "Integrated perception and planning in the continuous space: A POMDP approach," *The International Journal of Robotics Research*, vol. 33, no. 9, pp. 1288–1302, June 2014.

[20] B. Bouilly, T. Simeon, and R. Alami, "A numerical technique for planning motion strategies of a mobile robot in presence of uncertainty," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, May 1995, pp. 1327–1332.

[21] A. Lazanas and J.-C. Latombe, "Motion planning with uncertainty: a landmark approach," *Artificial Intelligence*, vol. 76, no. 1-2, pp. 287–317, 1995.

[22] T. Fraichard and R. Mermond, "Path planning with uncertainty for car-like robots," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, May 1998, pp. 27–32.

[23] A. Lambert and D. Gruyer, "Safe path planning in an uncertain-configuration space," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, Roma, Italy, September 2003, pp. 4185–4190.

[24] N. A. Melchior and R. Simmons, "Particle RRT for path planning with uncertainty," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, Roma, Italy, April 2007, pp. 1617–1624.

[25] Y. Huang and K. Gupta, "RRT-SLAM for motion planning with motion and map uncertainty for robot exploration," in *Proc.of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, September 2008, pp. 22–26.

[26] S. Prentice and N. Roy, "The belief roadmap: Efficient planning in belief space by factoring the covariance," *The International Journal of Robotics Research*, vol. 28, no. 11-12, pp. 1448–1465, July 2009.

[27] J. van den Berg, P. Abbeel, and K. Goldberg, "LQG-MP: Optimized path planning for robots with motion uncertainty and imperfect state information," *The International Journal of Robotics Research*, vol. 30, no. 7, pp. 895–913, 2011.

[28] A. Bry and N. Roy, "Rapidly-exploring random belief trees for motion planning under uncertainty," in *Proc.of the IEEE Int. Conf. on Robotics and Automation (ICRA)*, May 2011, pp. 723–730.

[29] A. Agha-mohammadi, S. Chakravorty, and N. Amato, "FIRM: Sampling-based feedback motion planning under motion uncertainty and imperfect measurements," *The International Journal of Robotics Research*, vol. 33, no. 2, pp. 268–304, February 2014.

[30] R. Glashan, K. Hsiao, L. P. Kaelbling, and T. Lozano-Pérez, "Grasping POMDPs: Theory and experiments," *Robotics Science and Systems Manipulation Workshop: Sensing and Adapting to the Real World*, 2007, (accessed on November 20, 2015). [Online]. Available: http://www.willowgarage.com/sites/default/files/paper__grasping_pomdps_theory_and_experiments___glashan.pdf

[31] A. Hornung, M. Phillips, E. G. Jones, M. Bennewitz, M. Likhachev, and S. Chitta, "Navigation in three-dimensional cluttered environments for mobile manipulation," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2012, pp. 423–429.

[32] S. Karaman, M. Walter, A. Perez, E. Frazzoli, and S. Teller, "Anytime motion planning using the RRT*," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 1478–1483.

[33] Y. Huang and K. Gupta, "Collision-probability constrained PRM for a manipulator with base pose uncertainty," in *Proc.of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2009, pp. 1426–1432.

[34] A. Hornung, K. M. Wurm, M. Bennewitz, C. Stachniss, and W. Burgard, "OctoMap: An efficient probabilistic 3D mapping framework based on octrees," *Autonomous Robots*, vol. 34, no. 3, pp. 189–206, April 2013.

[35] H. G. Tanner and K. J. Kyriakopoulos, "Nonholonomic motion planning for mobile manipulators," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, April 2000, pp. 1233–1238.

[36] J. Tan and N. Xi, "Unified model approach for planning and control of mobile manipulators," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2001, pp. 3145–3152.

[37] Y. Yamamoto and X. Yun, "Coordinating locomotion and manipulation of a mobile manipulator," *IEEE Trans. Autom. Control*, vol. 39, no. 6, pp. 1326–1332, 1994.

[38] Y. Yang and O. Brock, "Elastic roadmaps - motion generation for autonomous mobile manipulation," *Autonomous Robot*, vol. 28, no. 1, pp. 113–130, 2010.

[39] A. Hornung and M. Bennewitz, "Adaptive level-of-detail planning for efficient humanoid navigation," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2012, pp. 997–1002.

[40] J. Yang, P. Dymond, and M. Jenkin, "Exploiting hierarchical probabilistic motion planning for robot reachable workspace estimation," in *Informatics in Control Automation and Robotics, LNEE85*. Springer-Verlag, 2011, pp. 229–241.

[41] K. Gochev, A. Safonova, and M. Likhachev, "Planning with adaptive dimensionality for mobile manipulation," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2012, pp. 2944–2951.

[42] J. Vannoy and J. Xiao, "Real-time adaptive motion planning (RAMP) of mobile manipulators in dynamic environments with unforeseen changes," *IEEE Transctions on Robotics*, vol. 24, no. 5, pp. 1199–1212, 2008.

[43] J. Gonzalez and A. Stentz, "Planning with uncertainty in position: An optimal and efficient planner," in *Proc.of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, Edmenton, Canada, May 2005, pp. 1327–1332.

[44] N. Roy and S. Thrun, "Coastal navigation with mobile robots," in *Advances in Neural Processing Systems 12*, 1999, pp. 1043–1049.

[45] P. Missiuro and N. Roy, "Adapting probabilistic roadmaps to handle uncertain maps," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2006, pp. 1261–1267.

[46] B. Burns and O. Brock, "Sampling-based motion planning with sensing uncertainty," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, 2007, pp. 3313–3318.

[47] A. Nakhaei and F. Lamiraux, "A framework for planning motions in stochastic maps," in *Proc.of the International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2008, pp. 1959–1964.

[48] L. Guibas, D. Hsu, H. Kurniawati, and E. Rehman, "Bounded uncertainty roadmaps for path planning," in *Workshop on Algorithmic Foundations of Robotics (WAFR)*, ser. Springer Tracts in Advanced Robotics, vol. 57. Springer, 2008, pp. 199–215.

[49] D. Hsu, J. C. Latombe, and H. Kurniawati, "On the probabilistic foundations of probabilistic roadmap planning," *International Journal of Robotics Research (IJRR)*, vol. 25, no. 7, pp. 627–643, 2006.

[50] R. A. Knepper and M. T. Mason, "Real-time informed path sampling for motion planning search," *International Journal of Robotics Research (IJRR)*, vol. 31, no. 11, pp. 1231–1250, 2012.

[51] C. Stachniss, G. Grisetti, and W. Burgard, "Information gain-based exploration using rao-blackwellized particle filters," in *Proc.of the Robotics: Science and Systems (RSS)*, 2005, pp. 65–72.

[52] S. M. LaValle, *Planning Algorithms.* Cambridge, UK: Cambridge University Press, 2006.

[53] H. Choset, K. M. Lynch, S. Hutchinson, G. A. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations.* Cambridge, MA: MIT Press, 2005.

[54] S. Karaman and E. Frazzoli, "Sampling-based algorithms for optimal motion planning," *International Journal of Robotics Research (IJRR)*, vol. 30, no. 7, pp. 846–894, 2011.

[55] L. Torabi, "Integrated view and path planning for a fully autonomous mobile-manipulator system for 3d object modeling," Ph.D. thesis, Simon Fraser University, 2011.

[56] P. Lehner, A. Sieverling, and O. Brock, "Incremental, sensor-based motion generation for mobile manipulators in unknown, dynamic environments," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 4761–4767.

[57] C. Dornhege and A. Kleiner, "A frontier-void-based approach for autonomous exploration in 3d," in *Proc.of the IEEE International Symposium on Safety, Security, and Rescue Robotics*, November 2011, pp. 351–356.

[58] L. Torabi and K. Gupta, "An autonomous six-dof eye-in-hand system for in-situ 3d object modeling," *International Journal of Robotics Research (IJRR)*, vol. 31, no. 1, pp. 82–100, January 2012.

[59] L. Torabi and K. Gupta, "An autonomous 9-dof mobile-manipulator system for in situ 3d object modeling," in *video Proc.of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, Portugal, 2012, pp. 4540–4541.

[60] S. Shen, N. Michael, and V. Kumar, "Autonomous indoor 3d exploration with a micro-aerial vehicle," in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, 2012, pp. 9–15.

[61] V. Pilania and K. Gupta, "A hierarchical and adaptive mobile manipulator planner," in *Proc. of the IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Madrid, Spain, November 2014, pp. 45–51.

[62] V. Pilania and K. Gupta, "A hierarchical and adaptive mobile manipulator planner with base pose uncertainty," *Autonomous Robots*, vol. 39, no. 1, pp. 65–85, June 2015.

[63] V. Pilania and K. Gupta, "A localization aware sampling strategy for motion planning under uncertainty," in *Proc.of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[64] V. Pilania and K. Gupta, "Localization aware sampling and connection strategies for incremental motion planning under uncertainty," *Autonomous Robots*, vol. 41, no. 1, pp. 111–132, 2017.

[65] V. Pilania and K. Gupta, “Mobile manipulator planning under uncertainty in unknown environments,” *International Journal of Robotics Research (IJRR)*, vol. 37, no. 2-3, pp. 316–339, 2018.

[66] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, “Monte carlo localization: Efficient position estimation for mobile robots,” in *Proc.of the National Conference on Artificial Intelligence*, 1999, pp. 343–349.

[67] S. Kullback, *Information Theory and Statistics.* New York: Wiley, 1959.

[68] N. Roy and S. Thrun, “Online self-calibration for mobile robots,” in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, 1999, pp. 2292–2297.

[69] T. N. Y. Jr. and C. R. Shelton, “Simultaneous learning of motion and sensor model parameters for mobile robots,” in *Proc.of the IEEE International Conference on Robotics and Automation (ICRA)*, 2008, pp. 2091–2097.

[70] H. Choset, K. M. Lynch, S. Hutchinson, G. A. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of Robot Motion: Theory, Algorithms, and Implementations.* Cambridge, MA: MIT Press, 2005, ch. 7, pp. 242–246.

Appendix A

Mobile Base Belief Estimation using BRM Approach

This background information is included primarily for completeness and is largely taken from [26]. We use extended Kalman filter (EKF) to estimate the state of mobile base, in which the state distribution is assumed to be Gaussian. The next state s_t and observation z_t are given by the following equations,

$$s_t = g(s_{t-1}, u_t, w_t), \qquad w_t \sim N(0, W_t) \tag{A.1}$$

$$z_t = h(s_t, q_t), \qquad q_t \sim N(0, Q_t) \tag{A.2}$$

where u_t is a control action, w_t and q_t are random, unobservable noise variables. The EKF computes the state distribution at time t in two steps: a process step and a measurement step. The process step follows as

$$\bar{\mu_t} = g(\mu_{t-1}, u_t), \qquad \bar{\Sigma_t} = G_t \Sigma_{t-1} G_t^T + V_t W_t V_t^T \tag{A.3}$$

where G_t is the Jacobian of g with respect to s and V_t is the Jacobian of g with respect to w. For convenience, we denote $R_t = V_t W_t V_t^T$. Similarly, the measurement step follows as:

$$\mu_t = \bar{\mu_t} + K_t(H_t \bar{\mu_t} - z_t), \qquad \Sigma_t = (I - K_t H_t)\bar{\Sigma_t} \tag{A.4}$$

where H_t is the Jacobian of h with respect to s and K_t is known as the Kalman gain, given by

$$K_t = \bar{\Sigma_t} H_t^T (H_t \bar{\Sigma_t} H_t^T + Q_t)^{-1} \tag{A.5}$$

we denote $M_t = H_t^T Q_t^{-1} H_t$.

A.1 Belief Updating as a One-Step Operation

Briefly, *one* descriptor matrix $S_{1:T}$ to compose the filter updates for T time steps (a sequence of controls and measurements) along an edge between two nodes i and j can be calculated using the operator Redheffer star product (denoted with a '$\star$') as,

$$S_{1:T} = \begin{bmatrix} G_{1:T} & R_{1:T} \\ -M_{1:T} & G_{1:T}^T \end{bmatrix} = S_1 \star S_2 \star ... \star S_T. \tag{A.6}$$

where S_t is given by the star product of control update scattering matrix S_t^C and measurement update scattering matrix S_t^M

$$S_t^C = \begin{bmatrix} G & R \\ 0 & G^T \end{bmatrix}_t \qquad S_t^M = \begin{bmatrix} I & 0 \\ -M & I \end{bmatrix}_t \tag{A.7}$$

$$S_t = S_t^C \star S_t^M = \begin{bmatrix} G & R \\ -M & G^T \end{bmatrix}_t \tag{A.8}$$

We can then compute posterior covariance Σ_T from initial condition Σ_0 using

$$\begin{bmatrix} \cdot & \Sigma_T \\ \cdot & \cdot \end{bmatrix} = \begin{bmatrix} I & \Sigma_0 \\ 0 & I \end{bmatrix} \star S_{1:T} \tag{A.9}$$

(The matrix element $\cdot$ are irrelevant to the final solution for the covariance).

A.2 Motion Model and Sensor Model

Below we present the linearized version of motion and sensor models for use in EKF. Note that, for readability, we omit time index subscripts; however, all matrices derived are time-varying quantities.

We use the following non-linear probabilistic motion model with the assumption that the drive and turn commands are independent [68],

$$g_x = x + D\ cos(\theta + R) \tag{A.10}$$

$$g_y = y + D\ sin(\theta + R) \tag{A.11}$$

$$g_\theta = (\theta + R)\ \ mod\ 2\pi \tag{A.12}$$

where g_x, g_y and g_θ are the components of g corresponding to each state variable, and the control variable u_t is given by $u_t = [D\ R]^T$ where D and R denote the robot's translation and rotation, respectively.

In the EKF, the state transition matrix G is the Jacobian of the motion model with respect to the state, and is computed by linearizing the state transition function g about the mean state μ.

$$G = \begin{bmatrix} 1 & 0 & -D\ sin(\mu_\theta + R) \\ 0 & 1 & D\ cos(\mu_\theta + R) \\ 0 & 0 & 1 \end{bmatrix} \tag{A.13}$$

The linearized process noise in state space is computed as $R = VWV^T$ where W is the covariance matrix of the noise in control space

$$W = \begin{bmatrix} \sigma_D^2 & 0 \\ 0 & \sigma_R^2 \end{bmatrix} \tag{A.14}$$

and V is the motion noise matrix mapped from control to state space, computed as the Jacobian of the motion model with respect to the control space components

$$V = \begin{bmatrix} cos(\mu_\theta + R) & -D\ sin(\mu_\theta + R) \\ sin(\mu_\theta + R) & D\ cos(\mu_\theta + R) \\ 0 & 1 \end{bmatrix} \tag{A.15}$$

Now we explain the sensor model. The sensor model for 2D range sensor used in our experiments is given by $z^i = h^i(x) + N^i(0; Q)$, superscript i denotes i^{th} ray in the range scan

$$r^i = \sqrt{(x - x_{r^i})^2 + (y - y_{r^i})^2} + N^i(0; Q) \tag{A.16}$$

where (x, y, θ) is the robot pose, x_{r^i}, y_{r^i} is the obstacle location where i^{th} ray hit the object, r^i is the range of i^{th} ray. The linearized transformation from measurement space to state space is computed as the measurement Jacobian H^i, computed as the partial derivatives of the measurement function $h^i(x)$ with respect to each component of the state. The measurement Jacobian at time t is given by

$$H_t^i = \begin{bmatrix} \frac{(x - x_{r^i})}{\sqrt{(x - x_{r^i})^2 + (y - y_{r^i})^2}} & \frac{(y - y_{r^i})}{\sqrt{(x - x_{r^i})^2 + (y - y_{r^i})^2}} & 0 \end{bmatrix} \tag{A.17}$$

Note that $(x - x_{r^i}) = r^i cos(\theta_m)$ and $(y - y_{r^i}) = r^i sin(\theta_m)$, where $\theta_m = atan2(y - y_{r^i}, x - x_{r^i})$ is the angle of measurement relative to the robot pose. Therefore, Equation A.17 becomes

$$H_t^i = \begin{bmatrix} cos(\theta_m) & sin(\theta_m) & 0 \end{bmatrix} \tag{A.18}$$

The measurement noise covariance Q at time t for i^{th} sensor ray is

$$Q_t^i = \begin{bmatrix} \sigma_{r^i}^2 \end{bmatrix} \tag{A.19}$$

Note that in Chapter **??**, where beacons are used to demonstrate our LAS and LAC strategies, we use a different sensor model as mentioned in [26]. Also for learning of motion model and sensor model parameters, we refer the reader to [68] and [69].

Appendix B

Probabilistic Completeness Proof for RRBT-LAS

In this section, we provide a formal proof that a planner with our localization aware sampling strategy is probabilistically complete for DistTH less than or equal to half of the inscribed radius of the robot.

The worst case situation that leads to probabilistic completeness issues with our approach is using RangeModel 1 where the sensors (beacons in our case) have limited range. In that case the heuristic used in our sampling strategy will limit the samples to only one (within ball of radius DistTH) for regions with low uncertainty reduction. This is where the completeness issue arises. If the value of DistTH is large then the planner that uses our sampling strategy may not be able to find a path. We show that if we keep DistTH below half of the inscribed radius of the robot (a reasonable assumption) then if there exists a collision-free path, a planner that uses our sampling strategy will also find one. For the proof we assume that the entire path passes through regions with low uncertainty reduction (a worst case scenario for our sampling strategy). Also note that our proof builds along the lines of [70], therefore, we follow most of their notations.

Suppose $q_s,\ q_g \in C_{free}$ (free region of C-space) are two robot configurations that can be connected by a path in C_{free}. RRBT-LAS is considered to be probabilistically complete, if for any given (q_s, q_g)

$$\lim_{n\to\infty} Pr[(q_s, q_g)FAILURE] = 0 \tag{B.1}$$

where $Pr[(q_s, q_g)FAILURE]$ denotes the probability that RRBT-LAS fails to answer the query (q_s, q_g) after a roadmap in C_{free} with n samples has been constructed. The outline of the probabilistic completeness proof is as follows: First we assume that a path π from q_s to q_g exists. We then tile the path with a set of carefully chosen balls such that generating a sample in each ball ensures that these samples can be connected with appropriate collision-free edges and hence a collision-free path, $\hat{\pi}$ between q_s and q_g will be found by RRBT-LAS and the probability of generating such samples approaches 1 with increasing n.

Assume a path π (of length L) from q_s to q_g exists in d dimensional C-space. The clearance of π, denoted $\rho = clr(\pi)$, is the farthest distance away from the path at which a given point

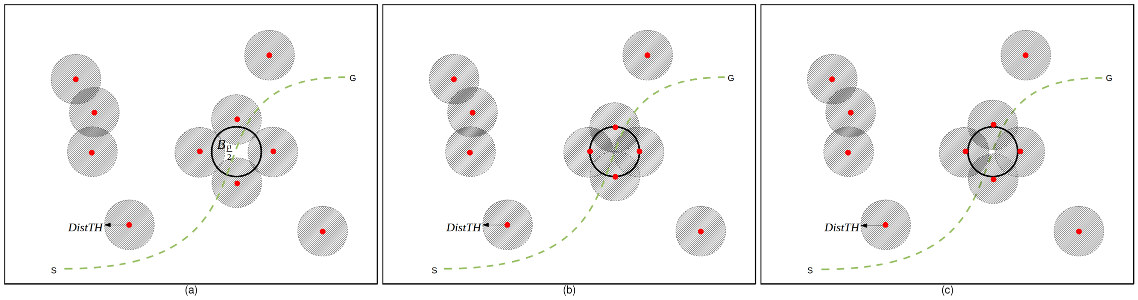

Figure B.1: [Case $DistTH \leq \frac{\rho}{2}$] - black colour (bold) circle denotes one of the balls $B_{\frac{\rho}{2}}(q_i)$ that is used to tile a path, red colour dots represent randomly placed samples, and hatched region (of radius DistTH=$\frac{\rho}{2}$) around each sample denotes the restricted region where samples can not be placed according to heuristic used in localization aware sampling. This figure shows the situation (excluding b) where none of the samples has yet been placed inside the black ball. (a) neighbouring samples around $B_{\frac{\rho}{2}}(\cdot)$ restrict some region (hatched areas inside the black ball) inside the black ball where samples can not be placed, (b) neighbouring samples totally covered $B_{\frac{\rho}{2}}(\cdot)$ but in that case samples lie on the periphery (closed set), (c) samples are placed at a distance d such that $\frac{\rho}{2} < d < \frac{\rho}{2} + \epsilon$, even in this worst case scenario the probability of generating a sample in $B_{\frac{\rho}{2}}(\cdot)$ is greater than 0 (see text for explanation).

can be guaranteed to be collision-free. Note that $\rho \geq 2r$, where r is the inscribed radius of the robot. The measure μ denotes the volume of a region of space, e.g, $\mu(B_\epsilon(x))$ measures the volume of an open ball $B_\epsilon(x)$ of radius ϵ centered at x. If $A \subset C_{free}$ is a measurable subset and x is a random point chosen from C_{free}, then

$$Pr(x \in A) = \frac{\mu(A)}{\mu(C_{free})} \tag{B.2}$$

We now tile the path π with balls each of radius $\frac{\rho}{2}$. Let $m = \lceil \frac{2L}{\rho} \rceil$ and observe that there are m points (centers of balls) on the path such that $dist(q_i, q_{i+1}) < \frac{\rho}{2}$, where $dist$ is a Euclidean metric on $\mathbb{R}^d$. Let $y_i \in B_{\rho/2}(q_i)$ and $y_{i+1} \in B_{\rho/2}(q_{i+1})$. Then the line segment $\overline{y_i y_{i+1}}$ must lie inside C_{free} since both endpoints lie in the ball $B_\rho(q_i)$. An illustration of this basic fact is given in Figure 7.17 of [70]. Let $V \subset C_{free}$ be a set of n configurations generated by our localization aware sampling strategy. If there is a subset of configurations $\{y_1, ..., y_m\} \subset V$ such that $y_i \in B_{\rho/2}(q_i)$, then each ball will get a sample and a path from q_s to q_g will be found. Let $I_1, ..., I_m$ be a set of indicator variables such that each I_i witnesses the event that there is a $y \in V$ and $y \in B_{\rho/2}(q_i)$. It follows that RRBT-LAS succeeds in answering the query (q_s, q_g) if $I_i = 1$ for all $1 \leq i \leq m$. If at least one of the indicator variables is 0 then RRBT-LAS would fail. Therefore, the probability of failure (Equation B.1) then can be written as

$$Pr[(q_s, q_g) FAILURE] \leq Pr\left(\bigvee_{i=1}^{m} I_i = 0\right) \tag{B.3}$$

$$\leq \sum_{i=1}^{m} Pr[I_i = 0] \tag{B.4}$$

where the last inequality follows from the union bound. We now mainly focus on the computation of $Pr[I_i = 0]$ for i^{th} ball, i.e., after placing n samples by our localization aware sampling strategy what is the probability that none of these samples lie in a ball $B_{\rho/2}(q_i)$.

For RangeModel 1, in regions outside the sensor range where there is no sensor information, hence no uncertainty reduction, our localization aware sampling strategy does not allow another sample within the vicinity (DistTH) of an already placed sample point (see Fig B.1). Therefore, the probability of failure to generate a second sample in a ball $B_{\rho/2}(q_i)$ depends on where the first sample was placed and so on. Let $I_i^1, ..., I_i^n$ be a set of indicator variables for the i^{th} ball such that each I_i^k, for all $1 \leq k \leq n$, witnesses the event that the k^{th} sample does not lie in ball $B_{\rho/2}(q_i)$. These events are dependent on each other. Below we provide the expressions to compute $Pr[I_i^k = 0]$ that will lead us to the computation of $Pr[I_i = 0]$. For the first two samples, the probability of failure to generate a sample inside ball $B_{\rho/2}(q_i)$ can be written as

$$Pr[I_i^1 = 0] = \left\{ 1 - \frac{\mu(B_{\rho/2}(q_i))}{\mu(C_{free})} \right\} \tag{B.5}$$

$$Pr[I_i^2 = 0] = \int Pr(I_i^2 = 0 \mid x^1) Pr(x^1) dx^1 \tag{B.6}$$

In above expression x^1 denotes the position of first sample. Above expression is just the marginalization over the position of first sample. Similarly, the expression for the third sample is

$$Pr[I_i^3 = 0] = \iint Pr(I_i^3 = 0 \mid x^1, x^2) Pr(x^2 \mid x^1) Pr(x^1) dx^2 dx^1 \tag{B.7}$$

and for n^{th} sample the expression (for $Pr[I_i^n = 0]$) is

$$= \int \cdots \int Pr(I_i^n = 0 \mid x^1, ..., x^{n-1}), ..., Pr(x^2 \mid x^1) Pr(x^1) dx^{n-1} dx^{n-2}, ..., dx^2 dx^1 \tag{B.8}$$

Clearly, parameters $\frac{\rho}{2}$ (radius of ball B) and DistTH (restricted region around a sample) are embedded in above expressions. Using Equations B.5-B.8, Equation B.4 can now be written as

$$Pr[(q_s, q_g) FAILURE] \leq \left\lceil \frac{2L}{\rho} \right\rceil \left(\prod_{k=1}^{n} Pr[I_i^k = 0] \right) \tag{B.9}$$

Note that $Pr(I_i^n = 1 \mid x^1, ..., x^{n-1})$ denotes the probability of generating the n^{th} sample inside ball $B_{\rho/2}(q_i)$ given that $n-1$ samples have been placed. This is nothing but the ratio of volume of white region inside the black ball (after excluding the hatched region) and total volume of white region (with reference to Fig B.1). In general, this can be written as

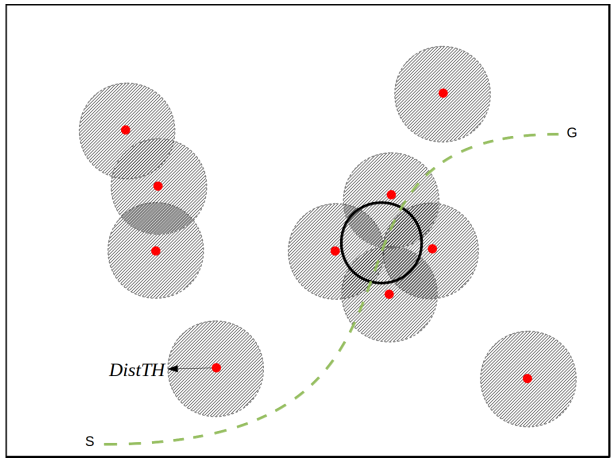

Figure B.2: [Case $DistTH > \frac{\rho}{2}$] - This figure shows that if $DistTH > \frac{\rho}{2}$ then the restricted regions of neighbouring samples may completely block the ball $B_{\rho/2}(q_i)$ and will prevent generation of a sample inside it. This will lead to the failure of a planner that uses our localization aware sampling strategy.

$$Pr(I_i^n = 1 \mid x^1, ..., x^{n-1}) = \frac{\mu({}^B C_{free}^R)}{\mu(C_{free}^R)} \tag{B.10}$$

where C_{free}^R denotes the C_{free} left after excluding the restricted regions around already placed samples and ${}^B C_{free}^R$ denotes the same but inside ball $B_{\rho/2}(q_i)$. This ratio approaches one as more and more samples are placed outside the black ball. That implies that $Pr(I_i^n = 0 \mid x^1, ..., x^{n-1})$ approaches zero. The expression in Equation B.8 is one of the product terms in RHS of inequality B.9. Convergence of $Pr[I_i^n = 0]$ (to 0) will lead to the convergence of RHS of inequality B.9. Therefore, $Pr[(q_s, q_g)FAILURE]$ converges to 0 as the number of samples increases, hence showing the completeness of RRBT-LAS. The same completeness can not be guaranteed for $DistTH > \frac{\rho}{2}$ (see Fig B.2).

Appendix C

Index to Multimedia Resources

Table C.1: Table of Multimedia Extensions

Extension	Type	Description
1	Video	HAMP demonstration corresponding to scenario B (simulation)
2	Video	HAMP demonstration corresponding to scenario C (simulation)
3	Video	HAMP demonstration corresponding to scenario D (simulation)
4	Video	HAMP demonstration corresponding to scenario E (simulation)
5	Video	HAMP-U demonstration on SFU mobile manipulator
6	Video	HAMP-BAU demonstration using autonomous system for mobile pick-and-place task in unknown environment (simulation)
7	Video	HAMP-BAU-TC demonstration using autonomous system for mobile pick-and-place task in known environment (simulation)
8	Video	HAMP-BAU demonstration using autonomous system for mobile pick-and-place task in unknown environment (real experiment on SFU Mobile Manipulator)
9	Video	HAMP-BAU-TC demonstration using autonomous system for mobile pick-and-place task in known environment (real experiment on SFU Mobile Manipulator)