PROBABILISTIC TOPOLOGICAL MAPS

A Dissertation
Presented to
The Academic Faculty

by

Ananth Ranganathan

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
College of Computing

Georgia Institute of Technology
March 2008

PROBABILISTIC TOPOLOGICAL MAPS

Approved by:

Frank Dellaert, Advisor
College of Computing
Georgia Institute of Technology

Tucker Balch
College of Computing
Georgia Institute of Technology

Henrik I. Christensen
College of Computing
Georgia Institute of Technology

Benjamin J. Kuipers
Dept. of Computer Sciences
Univ. of Texas at Austin

James M. Rehg
College of Computing
Georgia Institute of Technology

Date Approved: February 29, 2008

for Amma and Appa

अर्जुन उवाच ।
संन्यासं कर्मणां कृष्ण पुनर्योगं च शंससि ।
यच्छ्रेय एतयोरेकं तन्मे ब्रूहि सुनिश्चितम् ॥ ५-१ ॥

श्रीभगवानुवाच ।
यत्साङ्ख्यैः प्राप्यते स्थानं तद्योगैरपि गम्यते ।
एकं साङ्ख्यं च योगं च यः पश्यति स पश्यति ॥ ५-५ ॥

Arjuna :

> Krishna, first you tell me to give up practice and apply myself to theoretical study, but then again you exhort me to pursue practical application. Pray, tell me, which of these two paths is the better one?

Krishna :

> That which can be achieved through analytical study can also be achieved through practical application. He who sees theory and practice as leading to the same goal, sees things as they truly are.

Bhagavad Gita, Ch. 5, Verses 1,5

ACKNOWLEDGEMENTS

This thesis is as much the result of my advisor, Frank Dellaert's efforts as it is of mine. I have learnt a huge number of things from Frank in the last five years, and in addition, this process has been quite enjoyable. The ways in which Frank dug out interesting research questions from projects, which I was convinced were merely drudgery, was quite remarkable. His insistence on high coding standards, lucid writing, and eye-catching presentations will hold me in good stead for the rest of my life.

My peer group at Georgia Tech has made lab life bearable, and for a significant part of the time, outright enjoyable. Thanks to the Mobile Robot lab people (Endo, Alan, Patrick, and Zsolt), with whom I spent my first two years at Tech, for providing many deep political and philosophical discussions, and for all the fun times outside school. Many thanks to my fellow lab rats of the BORG lab, in particular to Michael for being a perfectionist and for a steady stream of German chocolates, to Sangmin for helping me develop appreciation for art and music, to Grant for many critical discussions on movies and books, and to Mingxuan for bringing cheer and ebullience wherever she goes. A big thank you to George, my roommate of three years, for interesting times spent together.

My gang of Indian friends have been the mainstay of my social life in Atlanta and have conjured a realistic simulation of home for me here. I am very grateful to (in order of appearance) Gaurav, Tarun, Rahul, Anar, Vivek, Sonal, Nirmeet, Jagpreet, Alina, and Himani for the love, affection, and good food that they provided me with. My fondest memories of Atlanta will be of the times I spent with them.

Finally, I must thank my parents for good genes, unquestioning support, deserved punishments, undeserved rewards, and a thousand other things. As a token of my appreciation for all this and much more, this dissertation is dedicated to them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Topological maps are light-weight, graphical representations of environments that are scalable and amenable to symbolic manipulation. Thus, they are well-suited for basic robot navigation applications, and also provide a representational basis for the procedural and semantic information needed for higher-level robotic tasks. However, their widespread use has been impeded in part by the lack of reliable, general purpose algorithms for their construction.

In this dissertation, I present a probabilistic framework for the construction of topological maps that addresses topological ambiguity, is failure-aware, computationally efficient, and can incorporate information from various sensing modalities. The framework addresses the two major problems of topological mapping, namely topological ambiguity and landmark detection.

The underlying idea behind overcoming topological ambiguity is that the computation of the Bayesian posterior distribution over the space of topologies is an effective means of quantifying this ambiguity, caused due to perceptual aliasing and environment variability. Since the space of topologies is combinatorial, the posterior on it cannot be computed exactly. Instead, I introduce the concept of Probabilistic Topological Maps (PTMs), a sample-based representation that approximates the posterior distribution over topologies given the available sensor measurements. Sampling algorithms for the efficient computation of PTMs are described.

The PTM framework can be used with a wide variety of landmark detection schemes under mild assumptions. As part of the evaluation, I describe a novel landmark detection technique that makes use of the notion of "surprise" in measurements that the robot obtains, the underlying assumption being that landmarks are places in the environment that

generate surprising measurements. The computation of surprise in a Bayesian framework is described and applied to various sensing modalities for the computation of PTMs.

The PTM framework is the first instance of a probabilistic technique for topological mapping that is systematic and comprehensive. It is especially relevant for future robotic applications which will need a sparse representation capable of accomodating higher level semantic knowledge. Results from experiments in real environments demonstrate that the framework can accomodate diverse sensors such as camera rigs and laser scanners in addition to odometry. Finally, results are presented using various landmark detection schemes besides the surprise-based one.

CHAPTER

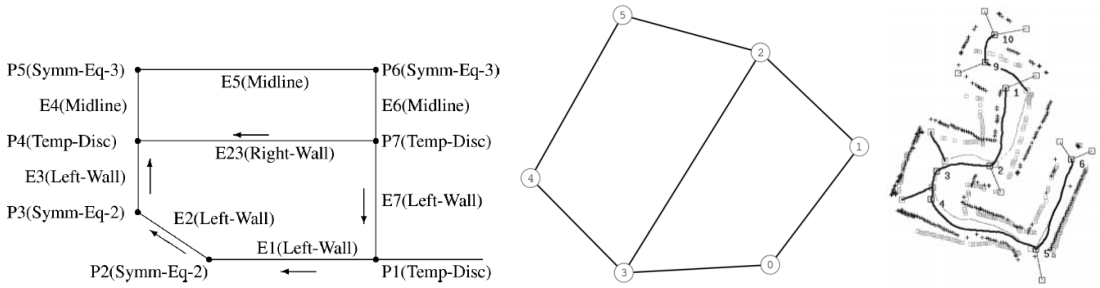
INTRODUCTION

Having a map of the environment is almost an essential pre-requisite for robots to perform any task requiring mobility. As robots enter everyday life via tasks as diverse as vacuum cleaning, nursing, and military transport, the requirement for maps that can enable mobility is obvious. Tasks such as surveying a disaster site during a search and rescue operation also require a map so that the location of possible victims or hazardous areas can be communicated.

Robot mapping is the problem wherein a robot is required to map out an environment through exploration. This problem is also widely known as Simultaneous Localization and Mapping (SLAM) since the robot has to be aware of its location in the environment (i.e be localized) before it can map it. SLAM is a hard problem since the information available to the robot is purely local and is obtained through noisy sensors. Moreover, the error in the robot's estimated position, when computed using these local, noisy measurements, is cumulative and increases without bound over time.

0.1 Topological Mapping

Topological maps are viable and useful representations of the environment for robotic tasks. Topological mapping is the subset of robot mapping, in which the map is a graph-based representation of the environment where certain easily distinguishable places in the environment, labeled as landmarks, are designed as nodes. The edges are deemed to represent navigable connections. In addition, the edges of the topological graph may be annotated with information relating to navigating the corresponding regions in the environment. Various examples of topological maps in the literature are given in Figure 1. Topological maps are intuitive in that they represent distinctive places in the environment prominently, and



(a) (b) (c)

Figure 1: Examples of topological maps in the literature (a) a topological map with control annotations along the edges [50] (b) topological map with metric landmark locations upto a scale and a rotation [78] (c) a Generalized Voronoi Graph (GVG) [13].

(a) (b)

Figure 2: Illustration of two common forms of metric maps (a) an occupancy grid [95] (b) a feature-based map [77]

may also contain procedural information on navigating between these places. This has been shown to be similar to how people mentally perceive their local environment [58].

The primary competitor to topological mapping is the metric mapping paradigm. Metric maps preserve physical distances and require a global frame of reference and are arguably the more popular map representation. Metric maps can in turn be classified into two main types as shown in Figure 2. Grids maps discretize the environment into grids, each of which is marked as navigable or non-navigable. On the other hand, feature maps identify and maintain the locations of certain distinct features in the environment.

While metric maps are conceptually simple and also easy to use for navigation tasks,

they also have a number of associated problems, which can be resolved to a large extent using topological maps -

- Scalability

  Metric maps quickly become unwieldy for large environments because the representation is not light-weight, i.e. it increases in density significantly with the size of the environment. In contrast, the rate at which the complexity of a topological map increases with the size of the environment is usually much less than that of a metric map. This is illustrated intuitively in Figure 3.

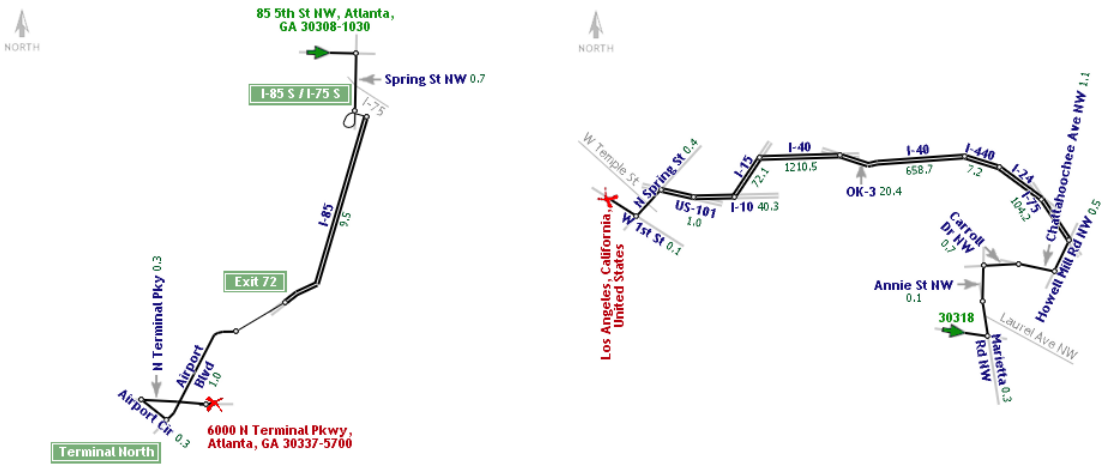
- Global inaccuracy

  Metric map construction is dependent on incremental addition of noisy measurements so that the accumulating error makes the maps progressively globally inaccurate. Special purpose loop-closing methods have to be incorporated to obtain valid global maps. On the other hand, constructing a topological map involves answering the question, "Have I been here before?", which sidesteps the problem of error accumulation.

- Lack of higher-level knowledge

  Most robotic tasks beyond simple navigation require some form of semantic knowledge. Adding semantic information to a metric map in a manner amenable to symbolic or high-level processing is harder due to the dense underlying representation. Since a topological graph is a symbolic, abstracted view of the environment, it supports, more easily than metric maps, higher level concepts such as objects and semantic labeling.

Topological maps are not widely used mainly due to the lack of a general mathematical theory for their construction. Inspite of the advantages listed above, topological maps have



(a) (b)

Figure 3: Two topological maps obtained from MSN Maps' LineDrive® feature that demonstrate the sparsity of the representation and its scalability (a) a map from midtown Atlanta to the airport (b) a map from midtwon Atlanta to Los Angeles, CA.

not gained widespread use because of a major procedural advantage that metric mapping enjoys -

- Metric map-building has been cast into a mathematical formalism which promotes understanding and for which systematic solutions are available. No such general mathematical formulation exists for topological mapping.

In their seminal paper, Smith and Cheeseman [87] demonstrated how the Kalman filter [44] could be used to solve the metric mapping problem. Since then almost all metric mapping algorithms are based on this theoretical framework, and more sophisticated techniques such as the Extended Kalman Filter (EKF) [88][54][11][18][22][90][89], Extended Information Filter (EIF) [97][25], Particle filtering [68][67][34][33], and smoothing [57][17][43, 77] have been introduced to overcome various shortcomings in the original formulation. Even for the pieces that do not currently have good solutions, such as the correspondence problem in feature maps, the mathematical theory is well understood. This has resulted in a surge in metric mapping research and its deployment in physical robots in a large number of varied domains.

(a) (b)

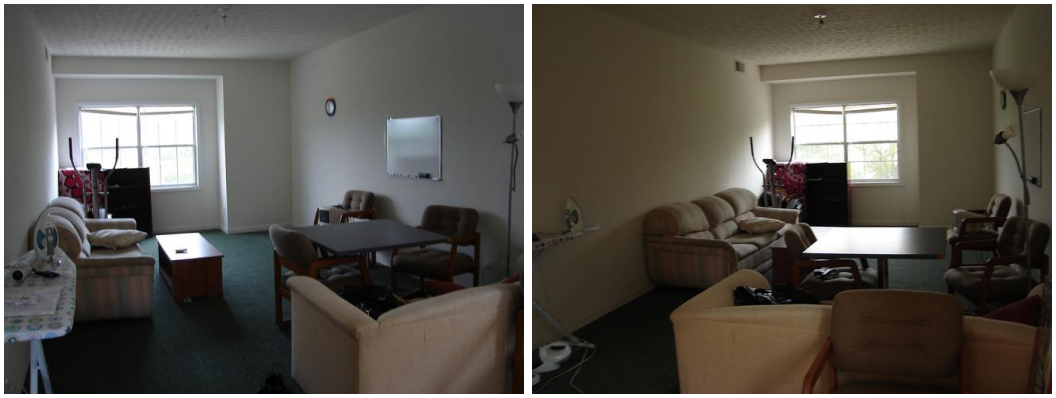
Figure 4: Topological ambiguity can arise due to perceptual aliasing where different landmarks look the same due to repetitive structure in the environment, the nature of the sensors, or due to noise.

Mathematical formulations of topological mapping are hard to come by due to complications introduced by the need to go from the continuous space of sensor measurements to a discrete, symbolic graph representation. This involves addressing problems such as data association, landmark detection, overcoming topological ambiguity, and being cognizant to failure. Of these, overcoming topological ambiguity when possible and being failure-aware when not, is especially important.

0.1.1 Ambiguity in Topological Mapping

Overcoming topological ambiguity is crucial to successful topological mapping. As the robot moves in the environment, it visits a sequence of landmarks, so that building a topological map reduces to determining whether each landmark in the sequence is a new one or one that has been visited by the robot (possibly multiple times) previously. In other words, the number of distinct landmarks and the number of times and the order in which the robot visited each landmark has to be determined. However, determining this is problematic as two or more places in the environment may have similar appearance. Even in the case where the true appearance of a pair of landmarks is dissimilar, perceptual processes on the robot may confuse their identities.

Failure to label landmarks appropriately creates ambiguity in the topology because the



(a) (b)

Figure 5: Image variability, where the same landmark may look different due to change in illumination and viewpoint, also gives rise to topological ambiguity.

number of distinct nodes and loops in the graph become uncertain. Two situations may be envisaged under which topological ambiguity occurs -

1. ***Perceptual Aliasing***

   Two or more distinct landmarks are *truly* similar in appearance, i.e the environment itself is ambiguous, or the landmarks are dissimilar but appear to be the same to the robot's sensors. This is illustrated in Figure 4.

2. ***Perceptual Variability***

   A single landmark visited two or more times appears distinct each time in the robot's sensory stream. This may occur due to viewpoint or illumination effects among other causes. This is illustrated in Figure 5.

Unless strong assumptions are made on the environment and the sensing model, for example the presence of properly located landmarks that are pairwise distinguishable, any kind of approach to map-building is prone to ambiguity and must deal with it. Additionally, a robust topological mapping system has to deal with both the above ambiguities in practice.

Topological ambiguities are also important because they appear in the domain of metric mapping, wherein the map consists of features and landmarks laid out in precise geometric fashion within a global frame of reference. The well-researched problem of loop closing in

metric maps is topological in nature since it deals with perceptual aliasing on a large scale. Hence, a solution to topological ambiguity is also applicable to metric mapping.

Existing mapping algorithms do not address topological ambiguity and the other attendant problems in topological mapping in a systematic manner, nor are all these issues addressed simultaneously by any single method.

0.2 Thesis Statement and Claims

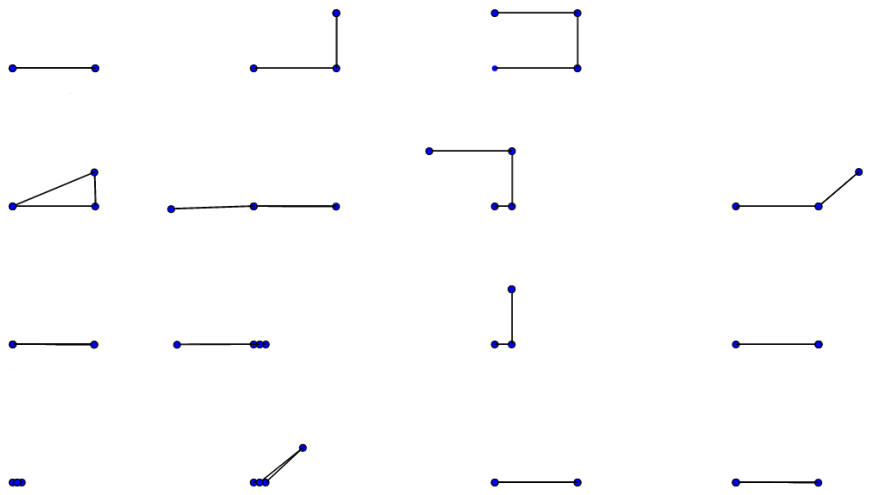
This dissertation deals with topological mapping. I present a new topological mapping framework called Probabilistic Topological Maps (PTMs) that addresses all of the above-mentioned issues in topological mapping to various extents. The contributions made by PTMs are codified in my thesis statement :

> Probabilistic Topological Maps provide a systematic framework for topological mapping that overcomes topological ambiguity when it is possible and is cognizant to failure when it is not. Further, PTMs are practical, efficient, compatible with various landmark detection schemes, and generalizable to diverse sensing modalities.

The thesis can be split into five claims that I will defend in this dissertation.

1. PTMs overcome topological ambiguity when possible and are cognizant to failure when not
2. PTMs are practical to compute
3. PTMs can be computed efficiently
4. PTMs can be used with various landmark detection schemes
5. PTMs can accomodate diverse sensing modalities and sensor models

In the rest of this chapter, I explain the terms in the thesis statement and outline the arguments backing these claims.



(a)

Figure 6: The space of topologies for the case when four landmarks are observed by a robot. There are 15 possible topologies in this case.

0.3 Probabilistic Topological Maps

A PTM is defined to be the posterior over the space of topologies, by which we mean the collection of all possible topologies for a given number of landmarks. This is illustrated in Figure 6. The PTM is thus a collection of topological maps with their associated probabilities. The reason for defining a PTM in this manner, and how this definition satisfies the claims stated above, is outlined in the following subsections.

0.3.1 A Probabilistic Solution to Topological Mapping

Probabilistic Topological Maps (PTMs) overcome topological ambiguity and hence, provide a systematic solution to the topological mapping problem. The definition of a PTM arises from the intuitive observation that the right way to overcome topological ambiguities is to evaluate every possible topology and assign a 'correctness' score to each based on its agreement with sensor measurements. In the absence of any ambiguity, only one topology will be correct while, when the the environment is highly aliased, multiple topologies may have high 'correctness' scores.

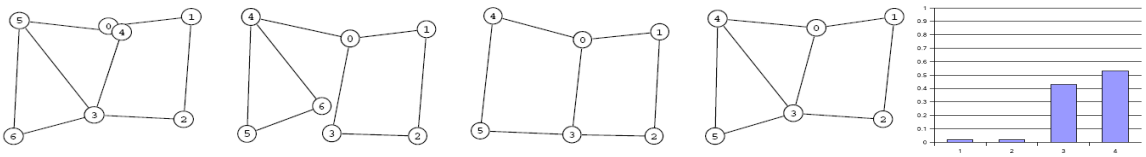


Figure 7: An example of a PTM showing the four most probable topologies in increasing order from left to right. The histogram of probability masses is shown on the right. In this case, the ground truth topology is the third one from the left which is not the maximum likelihood topology. The almost equal probabilities of the last two topologies, however, gives an indication of this error.

A theoretically sound way of assigning 'correctness' scores is to define a probability distribution over all possible topologies. When sensor measurements are considered, this distribution over the space of topologies is nothing but the Bayesian posterior. In accordance with the discussion above, the probability mass of the posterior will be overwhelmingly placed on a single topology in the absence of ambiguity, while the distribution will spread out with increasing ambiguity.

The PTM acknowledges the fact that in many cases it is impossible to distinguish between topologies of the environment based on simply the sensor measurements obtained by the robot. In such cases, the use of a maximum-likelihood or greedy mapping approach is bound to fail as it is forced to select a single topology. In contrast, the probabilities associated with the maps in the PTM provide an accurate estimate of the confidence with which the individual maps may be used. The PTM for an example case where the mostly likely map is an incorrect one is shown in Figure 7. The use of a maximum-likelihood technique in this case would simply yield a wrong topology. With the complete PTM, however, we have more information than can help us make an informed decision as to the correctness of the topology and its further use.

PTMs are cognizant to failure and can indicate failure in cases where ambiguity cannot be overcome. PTMs give an estimate of the uncertainty of the result of the algorithm. If a number of topologies have the same probability, this implies high ambiguity that the algorithm is currently unable to deal with. This can be resolved by providing more information

in the form of measurements or selecting fewer landmarks. This process could, in theory, be iterated until the PTM has a single topology with high probability mass, which indicates that the algorithm is certain about the correctness of the result. This enables the use of PTMs for tasks such as probabilistic planning using an ensemble of maps. One way to do this is to convert the PTM into an MDP where the transition probabilities of edges are the sum of the probabilities of the topologies in the PTM containing that edge. Probabilistic planning on MDPs is described, for example, in [3].

0.3.2 Practical Computation of PTMs

PTMs can be computed in a practical manner. Even though the number of possible topologies increases hyper-exponentially with the number of landmarks, the space of topologies can be leveraged so that the posterior, which is the PTM, can be computed approximately in a tractable manner. A naive algorithm that exhaustively computes the probability for each possible topology will be hopelessly intractable. The crucial observation here is that while the space of topologies may be enormous, most of the topologies in this space are irrelevant as they are a complete mismatch with respect to the sensor measurements. In other words, they will have a posterior probability of zero, and so, need not be evaluated. Moreover, the topologies that have a significant non-zero probability will be similar and will be 'close' together in the space of topologies. This fact, that the topologies with non-zero posterior probability are few and local, is illustrated in Figure 8.

On the basis of the above structure of the posterior over topologies, I will demonstrate that an effective method for approximately computing the PTM is to use sampling. Methods such as Markov Chain Monte Carlo (MCMC), where only topologies with non-zero probabilities are evaluated, are ideal for our purposes. MCMC works by running a Markov chain through the state space, where the chain visits the topologies in the space in proportion to their posterior probability. Each topology that the chain visits is a sample from the posterior distribution, and the number of visits is an approximation to the topology's

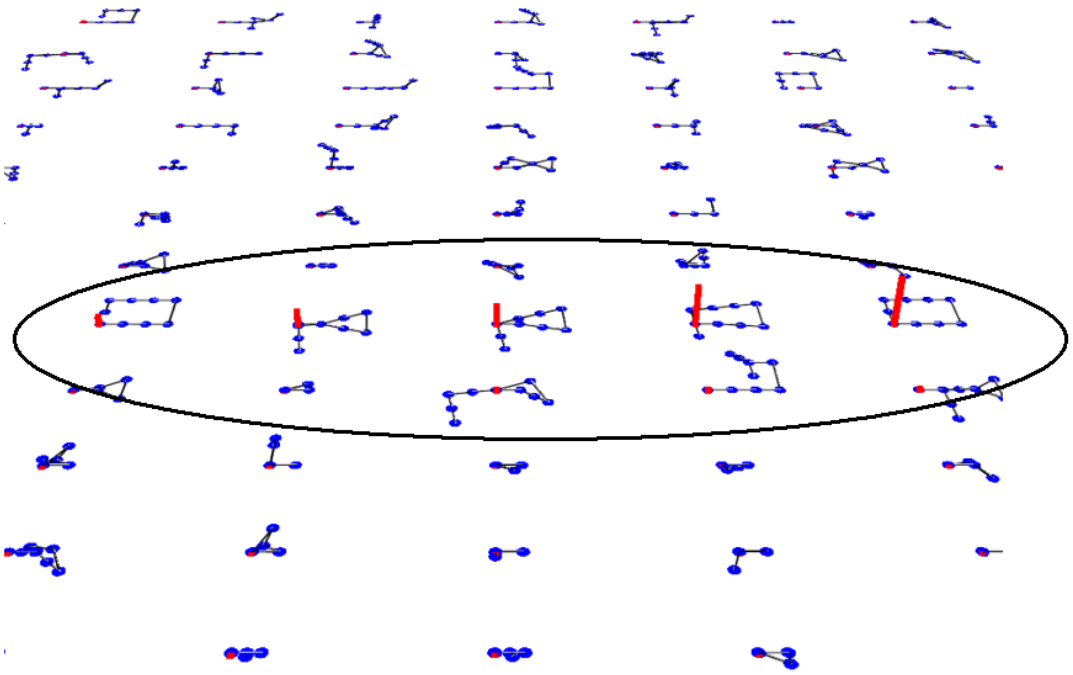
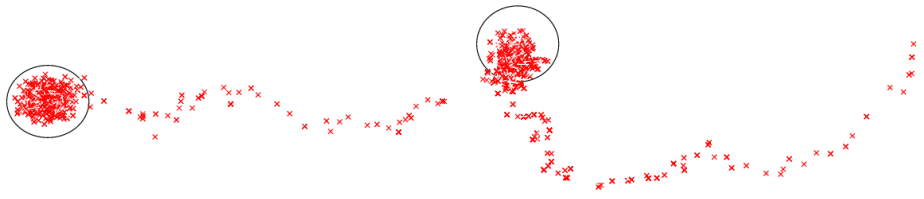


Figure 8: Even though the space of topologies is combinatorial, topologies with non-negligible probabilities are relatively few and localized.

posterior probability. Hence, the PTM becomes a sample-based approximation to the true posterior over the space of topologies.

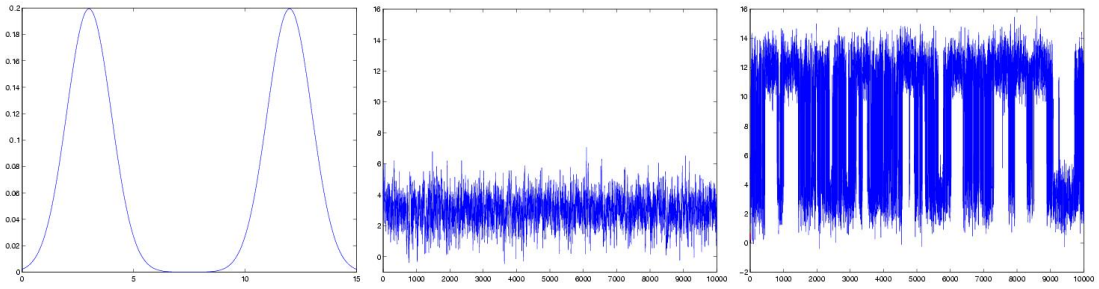
0.3.3 Efficient and Online Algorithms for Computing PTMs

The computation of PTMs can be made efficient and online. The vanilla MCMC algorithm, while generally applicable, may be inefficient in a number of circumstances. Inefficiencies



(a) (b)

Figure 9: Illustration of convergence of MCMC (a) a good proposal helps a chain started far away from the mode of the probability distribution to converge quickly (b) a poor proposal distribution takes longer to converge. Samples are shown in red while the isothermal curve for the mode of target distribution is shown in black.



(a) (b) (c)

Figure 10: Mixing in MCMC essentially means the ease with which the chain moves between the various high probability regions of the state space (a) a 1-D bimodal probability distribution that needs to be sampled from (b) a poorly mixing chain remains in a single mode for a long time. The figure shows the sample number along the x-axis and the sample values along the y-axis (c) a fast mixing chain moves between the modes quickly and easily

occur due to two important characteristics of the chain - convergence and mixing.

Convergence speed is the initial time taken by the Markov chain to start generating samples from the distribution of interest. Convergence can be sped-up using a smart proposal distribution, which determines how the Markov chain moves in the environment. An illustration of this is given in Figure 9. I will show that the use of data-driven proposals, i.e. proposal distributions that take into account the measurements, and not just the properties of the topology, improve convergence markedly.

Mixing is related to how quickly the chain can move through the complete space and is illustrated in Figure 10. Mixing time can become large, especially in cases where the distribution has two or more modes separated by a significant distance as for the one dimensional continuous space in the Figure. I will show that the use of techniques such as Simulated Tempering and Proposal chaining help the sampling algorithm overcome slow mixing in almost all cases.

Finally, MCMC is a batch algorithm, i.e. all the measurements have to be available when inference is done, and the addition of a new measurement requires starting the inference afresh. An incremental algorithm, in this case, is one that allows the efficient computation of a new PTM when a landmark is added to an existing PTM. I will present a particle filtering algorithm for computing PTMs that is incremental and hence, online.

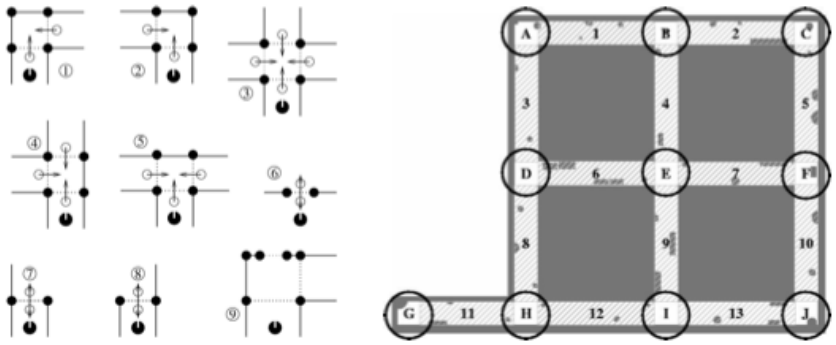


Figure 11: Gateways in the environment, especially man-made environments, correspond to landmarks in the topological graph. Figure taken from [82]

A particle filter maintains the PTM as a set of weighted samples which is amenable to efficient updation.

0.3.4 Incorporating Automatic Landmark Detection

PTMs can be used with diverse landmark detectors under mild assumptions. While the bulk of this dissertation discusses techniques for resolving topological ambiguity using the ideal landmark detector, i.e. where landmarks are selected by hand, PTMs are also evaluated using landmark detectors that span the spectrum of landmark detection, from the simplest case of placing landmarks at equi-distant intervals to the use of local low-level characteristics.

Landmarks are special places in the environment that anchor the topological graph. Intuitively, these correspond to decision points such as corridor junctions or entrances to rooms in indoor environments, where a navigating robot has to decide which of the possible routes to pursue. Examples of such gateways in the environments are illustrated in Figure 11. However, since the appearance of decision points varies widely, it is quite difficult to decide if the current location is a decision point based on just the current sensor measurements. Instead, a common approach consists of detecting changes in the environment as observed through the sensors. For instance, the area swept by a laser range scan might suddenly increase when a robot moves from a hallway into a junction, or suddenly decrease as

it exits a room through a door. Hence, local optima of certain sensor-derived quantities can serve as landmark detectors. That this strategy is hardly optimal is obvious. Consequently, more sophisticated approaches that take advantage of various spatial structures, such as Voronoi diagrams, exist; for example [6].

In general, PTMs can work with any landmark detection scheme as long as the detector fires at most of the actual points of interest in the environment, i.e. has few false negatives, though it may yield many more false positives. However, the larger the number of false positives, the stronger the measurements need to be, since the possibility of ambiguity increases with an increasing density of landmarks.

As a contribution to the field of landmark detection, and to demonstrate the working of the PTM framework with a novel detector, I present the notion of "surprise" for landmark detection. The change in the environment marking a landmark location can be captured in a general manner using this notion. Surprise is defined as the change in the current model of model when a new measurement is used to update it. If the change in the model is large, the measurement is said to be surprising. Alternately, the measurement is surprising if it is not sufficiently explained by the current model, thus requiring a large change in the model during the posterior update. Computing the change in the model and deciding when it is large enough can be done in a systematic manner. Subsequently, I postulate that surprising places are also landmarks. I will show that computation of surprise can be done in a sensor-independent manner by defining appropriate sensor models that abstract the sensor characteristics.

0.3.5 General Applicability of PTMs

PTMs are general in the sense that they can accommodate various sensors and sensing models. A topological mapping algorithm that is reliant on one sensor cannot claim generality since robotic systems routinely use multiple sensors of varied nature, examples being

odometers, laser range scanners and cameras. While sampling techniques make PTM computation practical and efficient, the use of a Bayesian framework automatically overcomes the challenge of generalizing to varied sensing modalities. This is true since, under the common assumption of conditional independence for the measurements, each additional sensor used to compute the PTMs only requires the definition of a measurement model. Even when conditional independence is violated, the provision of an appropriate joint likelihood makes inference of PTMs feasible. Thus, the PTM framework and computational algorithms are themselves rendered independent of the sensor type and do not make use their low level characteristics.

In this dissertation, I will make use of odometry, laser range scans, and panoramic images as measurements. Further, multiple measurement models for some of these sensing modalities will be proposed, thus demonstrating the wide applicability of PTMs.

0.4 Existing Approaches to Topological Mapping

PTMs advance the state of the art by addressing topological ambiguity in a systematic manner and being capable of accommodating diverse sensing and landmark detection schemes. While a large body of work on topological mapping exists in the literature, dealing with topological ambiguities has so far proved hard for modern robotic systems, mainly due to the lack of principled approaches for dealing with uncertainty in the discrete topological domain.

Historically, topological mapping is descended from the theory of cognitive maps proposed as a means of spatial representation in cognitive science [99]. The reasoning behind topological maps is based on navigation in animals and humans. Researchers in Cognitive science have , through cognitive simulation[1], gathered a number of pieces of evidence that suggest the use of representations similar to topological maps in people. These studies have

[1]Cognitive simulation is the process of testing cognitive models by comparing their simulated results with actual human behavior.

shown that in addition to landmarks and other special markers in environments, procedural information regarding navigation between two specific nodes is also used. Psychological studies have also confirmed these findings [104].

Kuipers and his group have been the pioneers in bringing the cognitive map view of topological maps into robotics [48][49]. An early instance of a working mapping system based on topological representation in provided in [52], which was subsequently extended to encompass a complete ontology for representing various abstractions in the environment [50].

Existing methods can be roughly categorized in the following manner -

- **Maximum-Likelihood Techniques**

  Most existing techniques approach the mapping problem in a maximum-likelihood framework with the aim of finding the topology that minimizes some error function. However, in the presence of aliasing, the most likely topology can frequently be wrong. Additionally, the error function to be optimized may have local minima which also results in an incorrect map. The pioneering work in this regard is by Shatkay and Kaelbling [85] that uses the Baum-Welch algorithm, a variant of the EM algorithm used in the context of HMMs, to solve the aliasing problem for topological mapping. Other examples of HMM-based work include [42][32] and [4] where a second order HMM is used to model the environment. A similar but slightly more sophisticated approach was given by Simmons and Koenig [86], in which the environment is modeled using a POMDP that updates belief states based on observations received by the robot. A variation from the maximum-likelihood methods is the topological mapping system given by Goedeme et. al. [30] that uses image clustering to define regions of space as nodes in the topology. Loop closing and correspondence are done using Dempster-Shafer decision theory, but again the decision is binding once taken. Finally, Lisien et al. [55] describe a method that combines locally estimated feature-based maps with a global topological map. Data association

for the local maps is performed using a simple heuristic wherein each measurement is associated with the existing landmark having the minimum distance to the measured location. Kuipers and Beeson [51] apply a clustering algorithm to the measurements to identify distinctive places, thus providing a maximum-likelihood solution to resolving ambiguity.

- **Sensor-specific Techniques**

  Many existing algorithms use low-level characteristics specific to particular sensing modalities such as obstacle distances from laser scanners to characterize landmarks. These methods cannot be retargeted to other sensors. An instance is Valgren and Duckett [103] perform topological mapping using an omnidirectional camera and model places using SIFT histograms. Ambiguity is solved using maximum likelihood matching of SIFT features, done by computing an affinity matrix of the images, and thus involves binding decisions at each step. Spectral clustering using the affinity matrix is also performed by Newman et. al. [72], albeit for the loop closing problem in the context of metric maps. Dedeoglu et al. [16] provide a mapping technique that uses specific features of the environment such as open doors and orthogonal walls, and identifies them using low-level characteristics of laser scans. Dudek and Jugessur [21] use Fourier transforms of feature patches detected using attention operators for recognizing landmarks and overcoming ambiguity.

- **Active control Techniques**

  A common way of overcoming perceptual aliasing involves exploration by the robot until a distinct landmark is observed that localizes the robot. Examples of this approach include Choset's Generalized Voronoi Graphs [13] and Kuipers' Spatial Semantic Hierarchy [52]. Other approaches that involve behavior-based control for exploration-based topological mapping are also fairly common. Mataric [60] uses

boundary-following and goal-directed navigation behaviors in combination with qualitative landmark identification to find a topological map of the environment. A complete behavior-based learning system based on the Spatial Semantic Hierarchy that learns at many levels starting from low-level sensori-motor control to topological and metric maps is described in [74]. Yamauchi et al. [105][106] use a reactive controller in conjunction with an Adaptive Place Network that detects and identifies special places in the environment. These locations are subsequently placed in a network denoting spatial adjacency. While the use of control is a valid approach, it can be wasteful in terms of time and energy. This work, in contrast, attempts to extract the maximum information possible from available data, though it is also general enough to incorporate an active localization approach if needed.

- **Multiple Hypothesis Tracking Techniques**

  Though approaches exist that track multiple topological hypotheses when encountering ambiguity, these are limited in the sense that the whole space of hypotheses is not explored due to its combinatorial nature. For instance, Thrun et al. [96] use the EM algorithm to solve the correspondence problem while building a topological map. The computed correspondence is subsequently used in constructing a metric map. By contrast, Thrun [94] first computes a metric map using value iteration and uses thresholding and Voronoi diagrams to extract the topology from this map. Another recent approach gives an algorithm to build a tree of all possible topological maps that conform to the measurements, but in a non-probabilistic manner [81][79]. Dudek. et. al. [20] have also given a technique that maintains multiple hypotheses regarding the topological structure of the environment in the form of an exploration tree. Most instances of previous work extant in the literature that incorporate uncertainty in topological map representations do not deal with general topological maps, but with the use of Markov decision processes to learn a policy that the robot follows to navigate the environment. An approach that is closer to our ideal in the sense

of maintaining a multi-hypothesis space over correspondences, is given by Tomatis et al. [100] and also uses POMDPs to solve the correspondence problem. However, while in their case a multi-hypothesis space is maintained, it is used only to detect the points where the probability mass splits into two. Also, like a lot of others, this work uses specific qualities of the indoor environment such as doors and corridor junctions, and hence is not generally applicable to any environment. Similarly, Tapus [91] proposes the use of POMDPs for disambiguation. The distinguishing features of this work is however, the use of "fingerprints of places" that incorporate various different features such as edges, lines, and color histograms, and help in resolving ambiguity to a significant extent. Work by Modayil et. al. [66] generates an ensemble of topological maps and uses them to construct a global metric map. However, they do not provide a probabilistic ordering to their ensemble of maps as the posterior on topologies constructed by our algorithm does.

The drawbacks of existing methods are addressed by PTMs as outlined in the previous section and as will be demonstrated in the rest of this dissertation.

0.5 Organization

The rest of the dissertation is organized with the intention of defending the claims stated in Section 0.2 in order.

Chapter 1 explains the Probabilistic Topological Mapping (PTM) framework and provides theoretical proof of the ability of PTMs to cope with topological ambiguities and be cognizant to failure. The equivalence between topologies and set partitions is presented, which is the basis for the computation of the sample-based approximation to the true posterior. Prior distributions over the space of topologies are also presented.

Chapter 2 provides details on the powerful Markov Chain Monte Carlo (MCMC) sampling algorithm for computing PTMs. The use of MCMC makes the PTM framework's use practical in robotic applications. Topological maps obtained through the use of this

algorithm are presented as results.

The vanilla MCMC algorithm is inefficient in many respects as it converges slowly. Also it cannot be used in an incremental fashion. Chapter 3 presents improvements to the basic MCMC algorithm that include proposal distributions that incorporate measurements, and multiple chain MCMC methods that converge rapidly. These methods make PTM computation efficient. A particle filtering algorithm is described that makes PTM computation incremental and online. Timing results demonstrate the improvements in efficiency and topological maps of physical environments provides evidence for correctness of these methods.

A complete topological mapping framework needs to address landmark detection in addition to resolving topological ambiguity. In Chapter 4, we present the results of evaluating PTMs with landmark detection schemes of varying sophistication. The novel surprise-based landmark detection scheme is also presented and evaluated both as a stand-alone landmark detector, and in conjunction with PTMs. This provides evidence for the wide applicability of the PTM framework.

Finally, Chapter 5 recaps the use of various sensors and sensor models in the results of the previous chapters. The variety of environments in which the PTM algorithms have been validated is also highlighted. This emphasizes the generality of the PTM framework and the algorithms therein.

CHAPTER I

A PROBABILISTIC SOLUTION TO TOPOLOGICAL MAPPING

This chapter presents the core idea of the thesis, which is a probabilistic framework to deal with the three kinds of topological ambiguities discussed in the Section 0.1.1. The aim is to provide a robust solution that fails gracefully even when the environment itself is highly aliased, a situation where the correct topology is impossible to obtain.

The basis of the solution is to quantify the uncertainty in the measurements and in the environment by defining a probability distribution over the space of topological maps. Given the available sensor measurements, this distribution is nothing but the Bayesian posterior distribution on the space of all possible topological maps.

The intuitive reason for computing the posterior is to solve the aliasing problem for topologies in a systematic manner. The set of all possible correspondences between sensor measurements and physical landmarks is exactly the set of all possible topologies. By inferring the posterior on this set, whereby each topology is assigned a probability, it is possible to locate the more probable topologies without committing to a specific correspondence greedily at any point in time, thus providing the most general solution to the aliasing problem. Even in pathological environments, where almost all current algorithms fail, this technique provides a quantification of uncertainty by pegging a probability of correctness to each topology.

However, inference in the space of topologies requires us to understand the structure of this space. Topological graphs are combinatorial objects that are not inherently amenable to computation. The key idea used to gain leverage over the problem is the equivalence between topologies and set partitions. As the combinatorial properties of set partitions are well understood in the literature, this enables us to manipulate topologies while performing

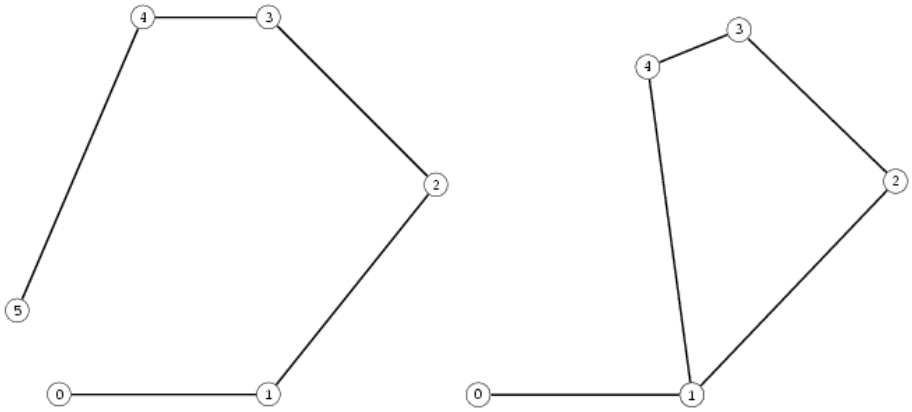


Figure 12: Two topologies with 6 observations each corresponding to set partitions (a) with six landmarks $(\{0\},\{1\},\{2\},\{3\},\{4\},\{5\})$ and (b) with five landmarks $(\{0\},\{1,5\},\{2\},\{3\},\{4\})$ where the second and sixth measurement are from the same landmark.

probabilistic inference.

1.1 Topologies as Set Partitions

Consider a scenario where a robot moves around an environment and visits six locations that are deemed to be landmarks. It is required that the robot identify the topology of the environment from these six observations. Consider two specific scenarios (note that these are not the only two possible) - one in which each of the landmarks is unique and the second in which the second and the last measurements come from the same landmark. We can illustrate these two scenarios as shown in Figure 12. It can be seen that the measurements corresponding to the same landmark can be grouped into a set, and this grouping then defines a set partition on the set of measurements. Each set partition, in turn, is equivalent to a topology of the environment.

The set partition corresponding to the topology can also be viewed as a label sequence as shown in Figure 13. Each set in the partition corresponds to a distinct label, which in turn, corresponds to a unique landmark. This in made explicit in Figure 14, that shows some of the possible topologies when five landmarks have been observed by the robot. The two extreme cases occur when all the landmarks are unique or all of them are the same (i.e

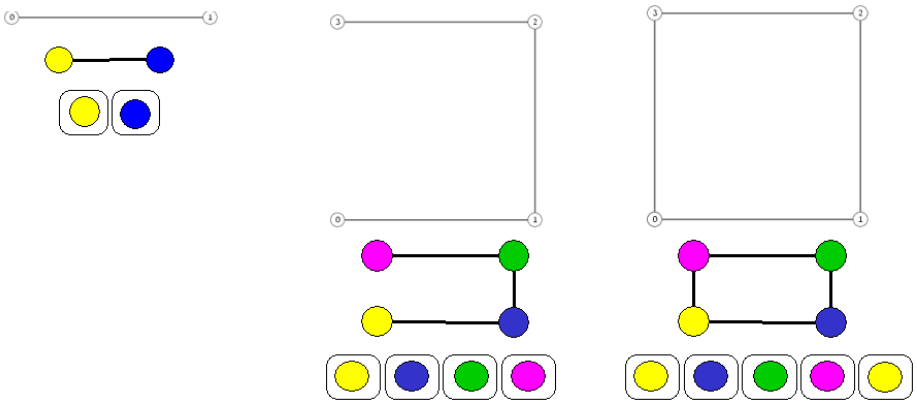


Figure 13: An example of topologies as label sequences (bottom), with each label colored differently. Each label sequence corresponds to a set partition.

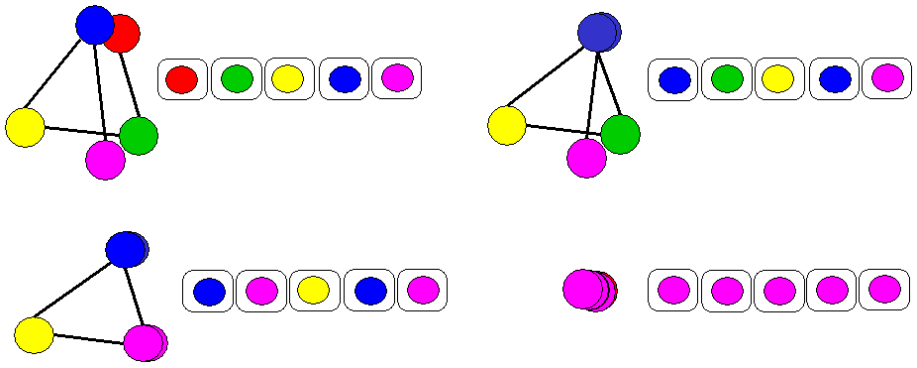


Figure 14: Some possible topologies for the case when five landmarks are observed by the robot. The topology on the top left occurs when each measurement corresponds to a unique landmark and the bottom right one corresponds to the case when all of them correspond to the same landmark.

the robot hasn't moved). The corresponding label sequences are also shown in the figure. It can be seen that a topology is nothing but the assignment of measurements to sets in the partition, resulting in the above mentioned isomorphism between topologies, set partitions, and label sequences.

To state the intuitions acquired above in a formal manner, we begin our consideration by assuming that the robot observes N "special places" or landmarks during a run, not all of them necessarily distinct. The number of distinct landmarks in the environment, which is unknown, is denoted by M. For the N element measurement set $Z = \{Z_i | 1 \leq i \leq N\}$, a partition T can be represented as $T = \{S_j \mid j \in [1,M]\}$, where each S_j is a set of measurements such that $S_{j1} \cap S_{j2} = \phi \ \forall j1, j2 \in [1,M],\ j1 \neq j2$, $\bigcup_{j=1}^{M} S_j = Z$, and $M \leq N$ is the number of sets in the partition. In the context of topological mapping, all members of the set S_j represent landmark observations of the jth landmark.

1.2 A General Framework for Inferring PTMs

The aim of inference in the space of topologies is to obtain the posterior probability distribution on topologies $P(T|Z)$, given a set of measurements Z. In this section, we describe the general theory for evaluating the posterior at any given topology.

Using Bayes law on the posterior $P(T|Z)$, we obtain

$$P(T|Z) \propto P(Z|T)P(T) \tag{1}$$

where $P(T)$ is a prior on topologies and $P(Z|T)$ is the observation likelihood.

Since the topology T is a set partition on measurements, it can be represented as $T = \{s_1, s_2, \cdots, s_n\}$ where s_i are the index sets of measurements in the partition corresponding to the topology. Each set is associated with a distinct place in the topology. While evaluation of the specific sensor models will be dealt with later, the general technique for computing the measurement likelihoods can be stated now. For measurements that are non-sequential, we assume that the measurements corresponding to each set are conditionally independent

of each other, giving rise to a product partition model [35]

$$P(Z|T) = \prod_{s \in T} P(Z_s|T) \tag{2}$$

Note that this formulation does not model the sequence in which measurements are obtained, but considers all measurements arising from a physical landmark to be exchangeable.

Since all the measurements in a set Z_s arise from the same physical place, we can model these measurements as being generated by the same underlying "cause", parametrized by a distribution with parameter θ. However, since we are not interested in computing the "cause" parameter θ, it is marginalized over to yield the final general formula for measurement likelihood computation

$$P(Z|T) = \prod_{s \in T} \int_{\theta_s} P(Z_s|\theta_s) P(\theta_s) \tag{3}$$

where $P(\theta_s)$ is a prior. Concrete implementations of this computation are provided for the case of laser range scans and various forms of appearance measurements in the following chapters.

Partitioning the measurements according to location does not work for odometry since sequential processing is crucial here. The likelihood of odometry measurements can be computed by marginalizing over the landmark locations visited by the robot, since each odometry is simply a measurement on the distance between these landmarks

$$P(O|T) = \int_X P(O|X,T) P(X|T) \tag{4}$$

where X is the vector of M landmark locations and $P(X|T)$ is a prior on landmark locations that has to be defined. This definition and computation of the likelihood is demonstrated in subsequent chapters.

According to (1), we need to define a prior distribution on topologies $P(T)$ to compute the posterior. Various models for computing the prior on topologies are discussed in the following sections.

1.3 Urn Model Priors Over Topologies

The prior on topologies $P(T)$, required to evaluate (1), assigns a probability to topology T based on the number of distinct landmarks in T and the total number of measurements. The prior plays an important role in our context since it provides a distribution on the number of distinct landmarks in a sequence of measurements obtained by the robot.

In this dissertation, urn models [40] are used as priors over topologies. The generic urn model consists of one or more urns in which balls of different colors are added or removed according to a fixed set of stochastic rules. Assumptions regarding the problem at hand can be encoded in the urn model by appropriately defining these rules.

In the following sections, three prior distributions based on different assumptions are described. These three distributions cover almost all the scenarios faced by a robot exploring and mapping an unknown environment. However, prior distributions with different assumptions may easily be defined using urn models with varied rules.

1.3.1 The Classical Occupancy Distribution

The first prior is obtained through the use of the Classical Occupancy Distribution [40]. This prior is useful when an estimate of the number of landmark locations in the environment is available. For example, in an indoor mapping scenario, the number of rooms and corridors in the building can be estimated from its size. In such cases, we would like to use the knowledge about the number of landmarks to affect the probability mass assigned to each topology. The prior defined below does precisely this.

Consider a scenario where the total number of landmarks in the environment is known to be L. Let the number of landmarks observed by the robot (i.e the number of measurements) be N. We would like to compute the probability of a topology containing M distinct landmarks, assuming that the probability of all such topologies is the same.

To derive the expression for the prior, we note that the setup can be converted into an urn-ball model by considering landmarks to be urns and measurements to be balls, yielding

L urns and N balls. We now show that the urn-ball model yields a prior over set partitions, which is also a prior over topologies due to the isomorphism between topologies and set partitions.

A set partition on the measurements is created by randomly adding the balls to the urns, where it is assumed that a ball is equally likely to land in any urn (i.e. there is a uniform distribution on the urns). The distribution on the number of occupied urns, after adding all the N balls randomly to the urns, is given by the Classical Occupancy Distribution [40] as

$$P(M) = \binom{L}{M} L^{-N} M! \left\{ \begin{array}{c} N \\ M \end{array} \right\} \tag{5}$$

where $\left\{ \begin{array}{c} N \\ M \end{array} \right\}$ is the Stirling number of the second kind, defined as the number of ways a set of size N can be partitioned into M sets.

The number of occupied urns after adding all the balls corresponds to the number of distinct landmarks in the topology, while the specific allocation of balls to urns (called an allocation vector) corresponds to the topology itself. Also, (5) assigns an equal probability to all ball allocations with the same number of occupied urns. Hence, we can interpret (5) as

$$\begin{aligned} P(M) \propto\ & P(\text{allocation vector with } M \text{ occupied urns}) \times \\ & \text{No. of allocation vectors with } M \text{ occupied urns} \end{aligned} \tag{6}$$

The number of allocation vectors with M occupied urns is equal to the number of partitions of the set of balls into M subsets. This is precisely the Stirling number of the second kind $\left\{ \begin{array}{c} N \\ M \end{array} \right\}$. Combining this observation with (5) and (6) yields

$$P(\text{allocation with } M \text{ occupied urns}) \propto \binom{L}{M} L^{-N} M!$$

As mentioned previously, the probability of an allocation vector corresponds to the probability of a topology. Hence, the prior probability of a topology T with M landmarks is

$$P(T|L) = k\frac{L^{-N} \times L!}{(L-M)!} \tag{7}$$

where k is a normalization constant. Specifying a different distribution on the allocation of balls to urns, rather than the uniform distribution assumed above, yields different priors on topologies.

The prior (7) has not yet been completely specified since it is contingent on knowing the number of landmarks in the environment, L, exactly. This, however, is clearly not the case. Hence, we assume a Poisson prior on L, giving

$$P(L|\lambda) = \frac{\lambda^L e^{-\lambda}}{L!}$$

and subsequently, marginalize over L to get the actual prior on topologies

$$\begin{aligned} P(T) &= \sum_L P(T|L)P(L|\lambda) \\ &\propto e^{-\lambda} \sum_{L=M}^{\infty} \frac{L^{-N} \times \lambda^L}{(L-M)!} \end{aligned} \tag{8}$$

where λ is the Poisson parameter, which is an estimate of the number of landmarks in the environment rather than the exact number itself, and the summation replaces the integral as the Poisson distribution is discrete. In practice, the prior on L is a truncated Poisson distribution since the summation in (8) is only evaluated for a finite number of terms.

1.3.2 The Dirichlet Process Prior

We can also define priors when an estimate of the size of the environment is unavailable. Here we may make two reasonable assumptions -

1. the probability of visiting a new landmark goes down with time
2. the probability of visiting a landmark is proportional to the number of times it has been visited in the past

The first assumption is valid in cases where the robot has to explore and map completely an environment of reasonable size. The second assumption is especially appropriate for indoor environments where a central corridor or hallway may have to be traversed repeatedly.

The two assumptions above are encoded by the well known Dirichlet Process (DP) model [26], which is usually used as a prior for infinite dimensional functional space, written as

$$G \sim \mathcal{DP}(\alpha G_0) \tag{9}$$

Here G is a function that is sampled from a Dirichlet Process with a prior base distribution G_0 and a concentration parameter α that determines how similar the samples from the Dirichlet Process are to G_0. Intuitively, α can be viewed as a variance parameter in functional space. Details of the Dirichlet Process are provided in Appendix A. Here we give a brief overview of its use as a prior in topological mapping.

For computational purposes, where working with a functional space is impossible, the Dirichlet Process is transformed into an urn model, wherein the number of urns may be infinite, and as before, we associate measurements with balls and landmarks with urns. This is done by marginalizing out G from the hierarchical model for measurements shown below

$$\begin{aligned} G &\sim \mathcal{DP}(\alpha G_0) \\ z_i &\sim G \end{aligned}$$

Marginalizing out G results in the Polya urn model [8], which can be understood as follows. For every new ball, we pick an already occupied urn with probability proportional to the number of the balls in that urn and a new urn with probability proportional to a constant parameter. This urn model in illustrated in Figure 15.

Let the topology T consist of the sequence of landmark observation $s_1, s_2, \ldots, s_n$. The probability of the nth landmark observation, conditioned on the previous observations, is

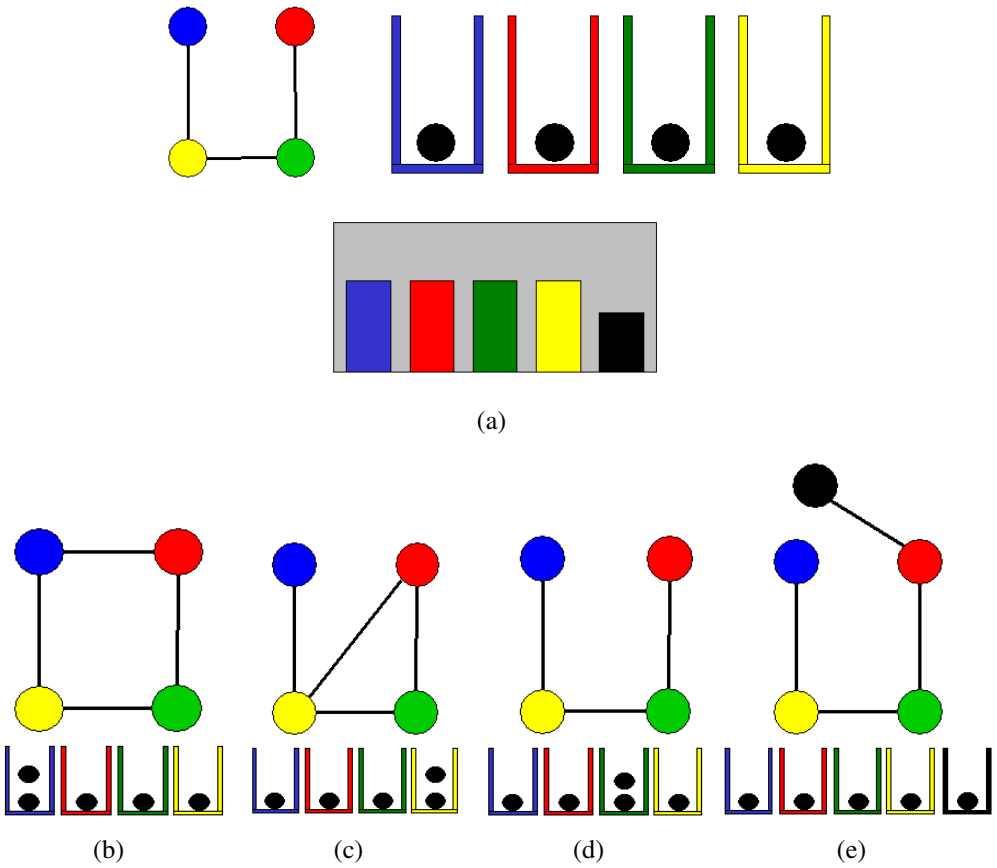


Figure 15: Illustration of the Polya Urn model as a prior on topologies. (a) An example topology with 4 distinct landmarks can be converted to an urn-ball model by considering the measurements as balls and the physical landmarks as urns. In this case, since each landmark has been visited once, we have one measurement per landmark and hence, one ball per urn in the urn model. This yields a discrete prior distribution (shown at the bottom) on the landmark that will be visited next. In the case of the Polya Urn model, this probability is proportional to number of times each landmark has been visited, i.e. the number of balls in each of the urns. The probability of visiting a new landmark is governed by a parameter, and is shown here in black. (b)-(e) show the topologies resulting by sampling the possibilities from the discrete prior. (b) the blue landmark is selected resulting in the topology shown and its corresponding urn model. (c) the yellow landmark is selected (d) the green landmark is selected which does not change the topology but does change the urn model (e) a new landmark (in black) is selected resulting in an urn model with an extra urn.

given by the Polya urn model as

$$P(s_n = j \mid s_{1:n-1}) \quad = \quad \begin{cases} \frac{1}{\alpha+n-1} \sum_{i=1}^{n-1} \delta(s_i = j) & \exists k \leq n, \quad \text{s.t. } s_k = j \\ \frac{\alpha}{\alpha+n-1} & \text{otherwise} \end{cases} \tag{10}$$

where α is a parameter of the Dirichlet process that encodes the probability of new landmarks at each step, and therefore indirectly, the total number of unique landmarks at any given point in time. Note that the probability of a new landmark goes down inversely with n.

The expression for the prior probability of the topology T can be obtained in closed form from (74) in a straight-forward manner

$$P(T) = \frac{\alpha^m}{\prod_{j=1}^{n} (\alpha + j - 1)} \prod_{s \in T} (|s| - 1)! \tag{11}$$

where m is the number of unique landmarks in the topology T. Note that (11) is in the form of a product partition model in the manner of (2) since

$$P(T) \quad = k \quad \prod_{s \in T} P(s)$$

Since both the measurement model and the prior in the Bayes equation (1) are product partition models, the posterior has the same form. Further, inference using the Dirichlet process prior for infinite mixture models is well-known and can be extended to the case of topological mapping. These favorable computational properties and the inherently reasonable assumptions make the use of Dirichlet process prior more desirable than the alternatives. However, alternative priors may have to be used when the Dirichlet process assumptions are clearly violated.

1.3.3 The Yule-Simon-Zipf Model

In many scenarios, the robot is required to explore a small portion of a vast environment. This is the likely case in search and rescue operations. In such environments, the assumption of made by the Dirichlet process with regard to the likelihood of new landmarks does

not hold. Instead of the probability of visiting new landmarks decaying with time, it remains constant, since the size of the environment is essentially infinite as far as the robot. Also, the probability of re-visiting a landmark is independent of the number of times it was visited previously. This is also in contrast to the Dirichlet process.

An urn model with these assumptions was recently proposed in [14] as the Yule-Zipf-Simon model and independently discovered by us [76]. Let the topology T consist of the sequence of landmark observation $s_1, s_{2,}, \ldots, s_n$ as above. The probability of the nth landmark observation, conditioned on the previous observations, is given by the Yule-Zipf-Simon model [14] as

$$P(s_n = k | s_{1:n-1}) = \begin{cases} (1-u)\frac{1}{z(n)} & 0 < i < z(n),\ n > 1 \\ u & i = z(n),\ n > 1 \\ 1 & n = 1 \end{cases} \quad (12)$$

where u is the constant probability of seeing new landmarks, and $z(n)$ is the number of unique landmarks that have been visited so far. The joint probability of the complete sequence of landmarks (i.e. the topology) cannot be evaluated in closed form using (12) [14], unlike for example, the Polya urn. However, given a specific topology, its probability can be evaluated. Since the algorithms that we will describe only require the evaluation of the probability of a given topology, the unavailability of an analytical expression for the joint prior on the topology is not a huge concern.

1.4 Intractability of Computing the Posterior over Topologies

The basis for computing the posterior over topologies is the Bayes equation for the posterior (1), using the appropriate measurement models and prior over topologies. A naive computation of the posterior would require evaluating the posterior for each and every topology in the space of topologies given some number of measurements. As is shown below, the number of topologies grows exceedingly large with the number of measurements, and hence, this straight-forward strategy has to be given up in favor of more sophisticated approaches.

The tractability of computing the posterior probability of every topology depends on the size of the space of topologies. From Section 1.1, we know the equivalence of topologies of set partitions. This equivalence can be used to compute the size of the space of topologies.

Consider partitions of n measurements, where the set partition contains k sets. These set partitions can be separated into two classes. The nth measurement can either be a separate set and be added to a pre-existing set of $n-1$ measurements and $k-1$ sets, or it can be added to one of the sets in a pre-existing partition of $n-1$ sets and k sets. This reasoning is expressed concisely by the recursive formula

$$\left\{ \begin{array}{c} n \\ k \end{array} \right\} = \left\{ \begin{array}{c} n-1 \\ k-1 \end{array} \right\} + k \left\{ \begin{array}{c} n-1 \\ k \end{array} \right\}$$

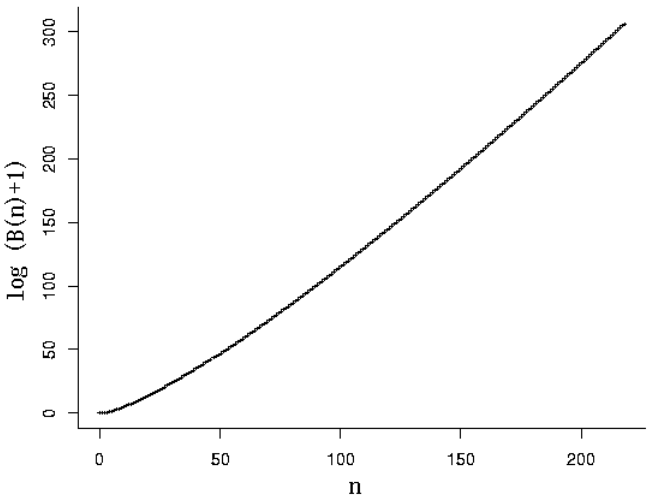
where $\left\{ \begin{array}{c} n \\ m \end{array} \right\}$ is the Stirling number of the second kind that gives the number of ways in which n measurements can be partitioned into k non-empty sets. Note that the above recursive formula can be used to compute Stirling numbers, since $\left\{ \begin{array}{c} n \\ n \end{array} \right\} = \left\{ \begin{array}{c} n \\ 1 \end{array} \right\} = 1$.

The number of set partitions is now simply given by the sum of Stirling numbers over the number of sets

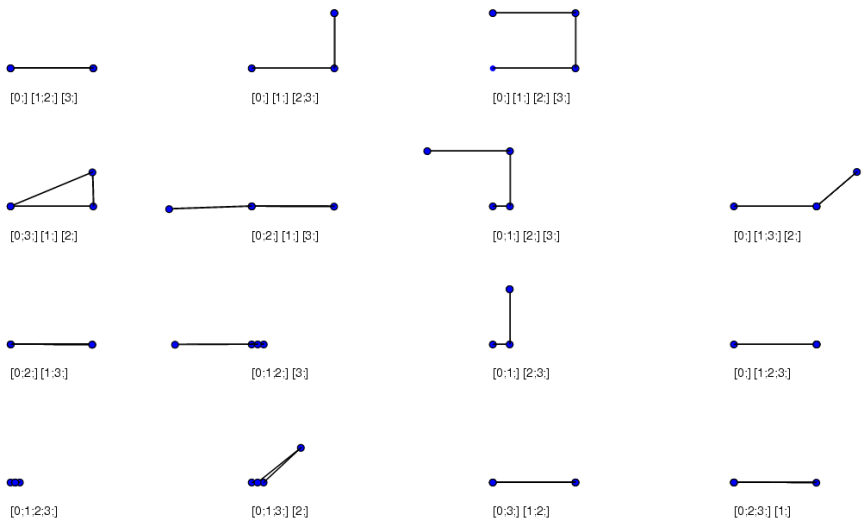
$$B_n = \sum_{k=1}^{n} \left\{ \begin{array}{c} n \\ k \end{array} \right\}$$

where B_n is called the Bell number [73]. The number of possible topologies for a given set of measurements is thus the Bell number corresponding to the number of measurements.

The Bell number grows hyper-exponentially with the number of elements in the set n. This can be seen from the fact that the asymptotic formula for the Bell numbers, known as de Bruijn's formula, involves the factorial on n, which in turn grows at the rate of $O(n^n) > O(e^n)$. $B_3 = 5$ and $B_7 = 877$ but $B_{20} = 51724158235372$. This growth is illustrated in Figure 16. From this it is clear that enumerating all possible topologies and evaluating the posterior distribution for each is hopelessly intractable.



(a)



(b)

Figure 16: (a) The number of possible topologies for a given number of landmarks is called the Bell number and grows at a rate faster than the exponential. The Bell number is plotted on a log scale here. (b) There are 15 possible topologies for the case of four measurements. The set partitions corresponding to the topologies are given below each topology.

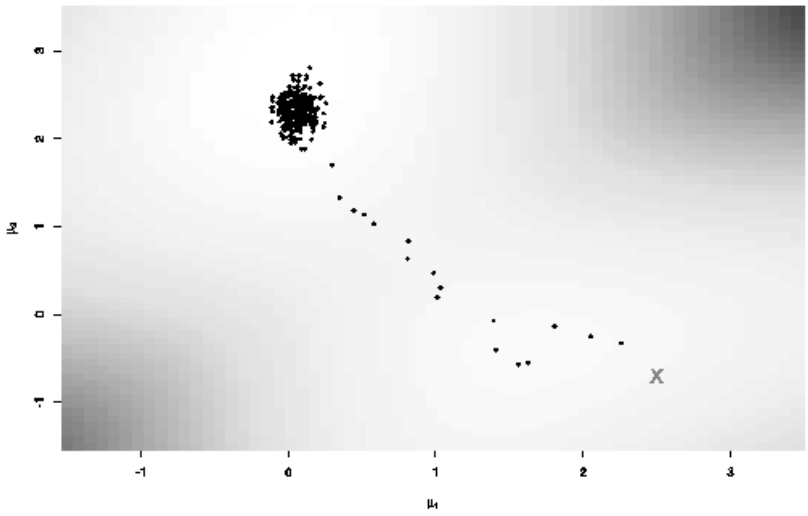


Figure 17: Illustration of a sampling algorithm in a space with a probability distribution on it. The distribution is shown using probability contours while the states evaluated by the algorithm are shown as black points. High probability regions are explored and evaluated preferentially. Most of the low probability regions are not evaluated.

1.5 Sampling for Computing the Posterior

Even though the space of topologies is huge, most of the topologies in this space have zero or negligible probability under the posterior. This is because most topologies are wildly inconsistent with the measurements obtained. Any computation that evaluates these topologies is wasted. Ideally, we would like an algorithm for computing the posterior that evaluates only the topologies that have a significant probability mass in the posterior. On the other hand, topological mapping involves finding exactly these topologies.

Efficient computation of posterior is still possible under two fairly general assumptions

-

1. Locality

   We assume that the highly probable topologies are surrounded by other topologies which are also probable, and hence, regions of high probability exist in the space of topologies. This assumption encodes the intuitive observation that topologies that look similar have similar probabilities. While there may be multiple regions of high

probability, once we reach such a region, a large portion of the probability mass of the posterior can be computed easily.

2. Sparseness

   We assume that the available measurements are sufficiently discriminative so that only a few topologies have significant probability mass. Clearly, if the measurements are insufficient or are highly ambiguous, a large number of topologies will satisfy them, and searching through the space of topologies to find all of them will be intractable. However, we still require that a topological mapping detect such a condition as failure so that more measurements or better sensors can be provided to overcome the problem of ambiguity.

The thesis of this dissertation in part is that under the conditions above, sampling algorithms provide an efficient mechanism for computing the posterior.

Sampling algorithms work by "moving around" randomly in the space of interest, searching for highly probable states. Once regions of high probability are found, computation is focused on the states in these regions while most of the remaining space is not even explored. The result of the algorithm is a histogram-based representation of the posterior, where a probability is associated with each of states that have been explored. The unexplored states implicitly have a negligible probability.

The following chapters discuss the design and application of sampling algorithms for computing the posterior over topologies as a sampled-based approximation.

CHAPTER II

PRACTICAL COMPUTATION OF PTMS

While we have proposed models for the measurements and the prior on topologies, we have not yet provided algorithms for performing inference using these models. As seen in Section 1.4, enumeration of the topological space is impossible but also unnecessary since most of the topologies are wildly inconsistent with the measurements, and consequently have negligible probability. We can also exploit the locality inherent in the posterior over topologies wherein most of the topologies with significant probability mass are clustered into small regions of the space.

Sampling methods that reconstruct a target probability distribution by "visiting" points in the space according to their probability are ideal for computing distributions with these properties. A probability density over the space of topologies can be approximated by drawing a sample of possible maps from the posterior distribution. Using the samples, it is possible to construct a histogram on the support of this sample set., which provides an approximation to the posterior itself.

The sampling-based approximation to the posterior on topologies will hereon be referred to as a Probabilistic Topological Map (PTM). A PTM is thus an ensemble of maps, each of which has a probability mass associated with it. We would like that the set of maps

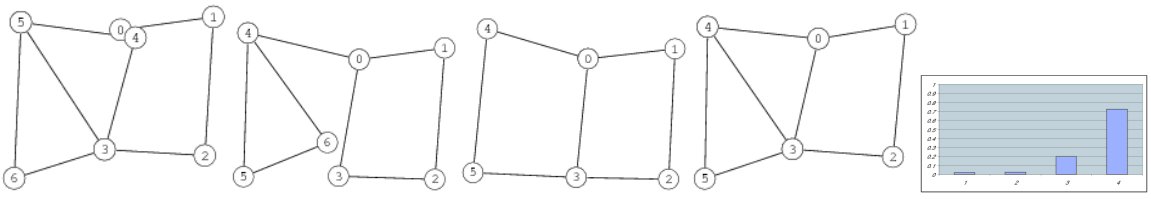


Figure 18: An example of a PTM giving the most probable topologies in the posterior distribution obtained using MCMC sampling. The histogram gives the probability of each topology.

in the PTM captures a large percentage of the probability mass of the true posterior so that the PTM is a good approximation to it. Figure 18 depicts an example of a PTM in the form of a histogram.

This chapter describes the use of Markov Chain Monte Carlo (MCMC) sampling for computing PTMs. MCMC is a batch technique that can be applied when all the measurements are available and operates by running a Markov chain over the space of interest. Extending the chain to a new topology is the main step in the sampling algorithm and requires the design of a proposal distribution and the calculation of an acceptance ratio. These details are discussed in the following sections.

2.1 Markov Chain Monte Carlo for Inferring PTMs

All MCMC methods work by running a Markov chain over the state space with the property that the chain ultimately converges to the target distribution of interest. Once the chain has converged, subsequent states visited by the chain are considered to be samples from the target distribution. The Markov chain itself is generated using a proposal distribution that is used to propose the next state in the chain, a move in state space, possibly by conditioning on the current state. The Metropolis-Hastings algorithm provides a technique whereby the Markov chain can converge to the target distribution using any arbitrary proposal distribution, the only important restriction being that the chain be capable of reaching all the states in the state space.

The pseudo-code to generate a sequence of samples from the posterior distribution $P(T|Z)$ over topologies T using the Metropolis-Hastings algorithm is shown in Algorithm 1 (adapted from [29]). In this case the state space is the space of all set partitions, where each set partition represents a different topology of the environment. Intuitively, the algorithm samples from the desired probability distribution $P(T|Z)$ by rejecting a fraction of the moves generated by a proposal distribution $Q(T_t';T_t)$, where T_t is the current state

Algorithm 1 The Metropolis-Hastings algorithm

1. Start with a valid initial topology T_t, then iterate once for each desired sample
2. Propose a new topology $T_t^{'}$ using the *proposal distribution* $Q(T_t';T_t)$
3. Calculate the *acceptance ratio*

$$a = \frac{P(T_t'|Z^t)}{P(T_t|Z^t)} \frac{Q(T_t;T_t')}{Q(T_t';T_t)} \tag{13}$$

   where Z^t is the set of measurements observed up to and including time t.

4. With probability $p = \min(1,a)$, accept T_t' and set $T_t \leftarrow T_t'$. If rejected we keep the state unchanged (i.e. return T_t as a sample).

and T_t' is the proposed state. The fraction of moves rejected is governed by the acceptance ratio a given by (13), which is where most of the computation takes place. The acceptance ratio enforces the condition of time-reversibility on the Markov chain, i.e. the condition $P(T_t'|Z)Q(T_t;T_t') = P(T_t|Z^t)Q(T_t';T_t)$, which is required for convergence by the Metropolis-Hastings algorithm. Computing the acceptance ratio, and hence, sampling using MCMC, requires the design of a proposal density and evaluation of the target density, the details of which are discussed below.

We use a simple split-merge proposal distribution that operates by proposing one of two moves, a split or a merge with equal probability at each step. Given that the current sample topology has M distinct landmarks, the next sample is obtained by splitting a set, to obtain a topology with $M+1$ landmarks, or merging two sets, to obtain a topology with $M-1$ landmarks. The proposal is illustrated in Figure 19 for a trivial environment. If the chosen move is not possible, the current topology is re-proposed. An example of an impossible move is a merge move on a topology containing only one landmark.

The *merge move* merges two randomly selected sets in the partition to produce a new partition with one less set than before. The probability of a merge is simply $1/N_M$ where N_M is the number of possible merges and is equal to the binomial coefficient $\binom{M}{2}$, $(M > 1)$.

The *split move* splits a randomly selected set in the partition to produce a new partition

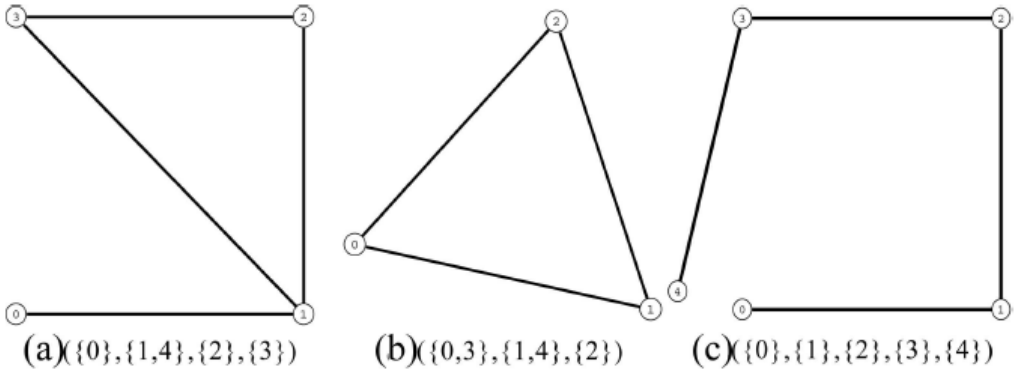


Figure 19: Illustration of the proposal - Given a topology (a) corresponding to the set partition with N=5, M=4, the proposal distribution can (b) perform a merge step to propose a topology with a smaller number of landmarks corresponding to a set partition with N=5, M=3 or (c) perform a split step to propose a topology with a greater number of landmarks corresponding to a set partition with N=M=5 or re-propose the same topology.

with one more set than before. To calculate the probability of a split move, let N_S be the number of non-singleton sets in the partition. Clearly, N_S is the number of sets in the partition that can be split. Out of these N_S sets, we pick a random set R to split. The number of possible ways to split R into two subsets is given by $\left\{ \begin{array}{c} |R| \\ 2 \end{array} \right\}$, where $\left\{ \begin{array}{c} n \\ m \end{array} \right\}$ denotes, as before, the Stirling number of the second kind that gives the number of possible ways to split a set of size n into m subsets. Combining the probability of selecting R and the probability of splitting it, we obtain the probability of the split move as $p_{\text{split}} = \left(N_S \left\{ \begin{array}{c} |R| \\ 2 \end{array} \right\} \right)^{-1}$.

The proposal distribution is summarized in pseudo-code format in Algorithm 2, where Q is the proposal distribution and $r = \frac{q(T' \to T)}{q(T \to T')}$ is the proposal ratio, a part of the acceptance ratio in Algorithm 1. Note that this proposal does not incorporate any domain knowledge, but uses only the combinatorial properties of set partitions to propose random moves.

In addition to proposing new moves in the space of topologies, we also need to evaluate the posterior probability $P(T|Z)$. This is done as described in Section 1.2. We evaluate the posterior distribution, which is also the MCMC target distribution, using Bayes rule (1). It

Algorithm 2 The Proposal Distribution

1. Select a merge or a split with probability 0.5
2. **Merge move:**
   - if T contains only one set, re-propose $T' = T$, hence $r = 1$
   - otherwise select two sets at random, say R and S
     (a) $T' = (T - \{R\} - \{S\}) \cup \{R \cup S\}$ and $Q(T \to T') = \frac{1}{N_M}$
     (b) $Q(T' \to T)$ is obtained from the reverse case 3(b), hence $r = N_M \left(N_S \left\{ \begin{array}{c} |R \bigcup S| \\ 2 \end{array} \right\} \right)^{-1}$, where N_S is the number of possible splits in T'
3. **Split move:**
   - if T contains only singleton sets, re-propose $T' = T$, hence $r = 1$
   - otherwise select a non-singleton set U at random from T and split it into two sets R and S.
     (a) $T' = (T - \{U\}) \cup \{R, S\}$ and $Q(T \to T') = \left(N_S \left\{ \begin{array}{c} |U| \\ 2 \end{array} \right\} \right)^{-1}$
     (b) $Q(T' \to T)$ is obtained from the reverse case 2(b), hence $r = N_M^{-1} N_S \left\{ \begin{array}{c} |U| \\ 2 \end{array} \right\}$, where N_M is the number of possible merges in T'

is important to note that we do not need to calculate the normalization constant in (1) since the Metropolis-Hastings algorithm requires only a ratio of the target distribution evaluated at two points, wherein the normalization constant cancels out.

Before presenting results using the MCMC sampling algorithm, we describe the computation of odometry and appearance measurement likelihoods as these are prerequisites for evaluating the target distribution over topologies.

2.2 Evaluating Odometry Likelihood

As explained in Section 1.2, it is not possible to evaluate odometry likelihood given the topology alone. Instead, an intermediate quantity has to be introduced that enables the transition from the metric measurement to the topological graph.

The intermediate quantity that enables us to bridge the gap between metric measurements in the form of odometry and topological representation is the locations of landmarks. However, since the landmark locations are not required when inferring topologies, the set of landmark locations X is integrated over resulting in the marginal distribution $P(O|T)$, as was shown in (4):

$$P(O|T) = \int_X P(O|X,T)P(X|T) \tag{14}$$

Under the assumption, common in robotics literature, that landmark locations and odometry measurements have the 2D form $X = \{l_i = (x_i, y_i) | 1 \leq i \leq N\}$ and $O = \{o_k = (x_k, y_k, \theta_k) | 1 \leq k \leq N-1\}$ respectively. $P(O|X,T)$ is the odometry measurement model given the landmark locations, and $P(X|T)$ is a prior over landmark locations. Evaluation of the likelihood using (14) requires the specification of a prior distribution $P(X|T)$ over landmark locations in the environment and the measurement model $P(O|X,T)$.

2.3 Prior Over Landmarks

We use a simple prior on landmarks that encodes our assumption that landmarks do not exist close together in the environment. If the topology T places two distinct landmarks l_{i1}

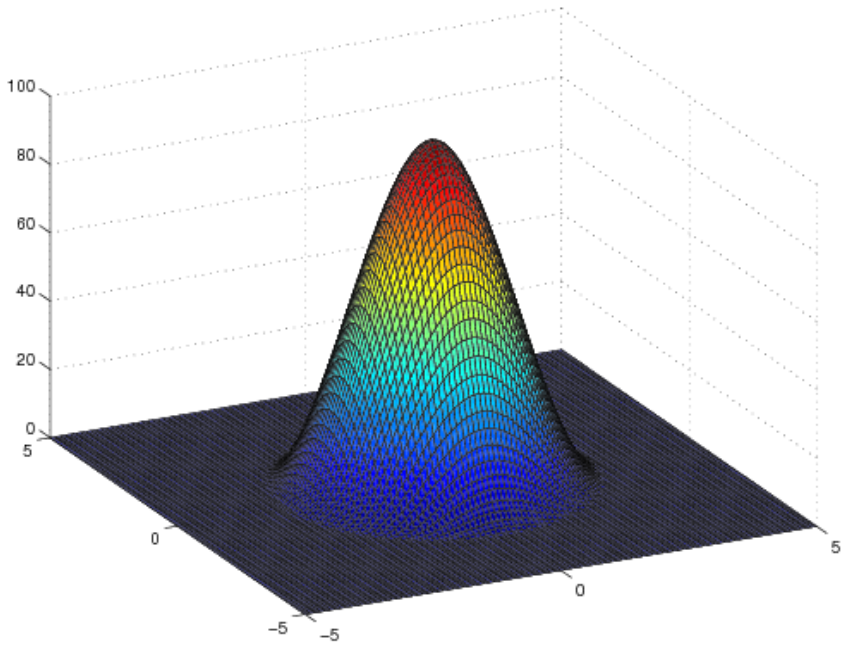


Figure 20: Cubic penalty function (in this case, with a threshold distance of 3 meters) used in the prior over landmark density

and l_{i2} within a distance d of each other, the negative log likelihood corresponding to the two landmarks is given by the penalty function

$$L(l_{i1}, l_{i2}; T) = L(l_{i2}, l_{i1}; T) \quad = \quad \begin{cases} f(d) & d < D \\ 0 & d \geq D \end{cases} \tag{15}$$

where d is the Euclidean distance between l_{i1} and l_{i2}, D is a threshold value, called the "penalty radius", and we define $f(d)$ to be a cubic function as shown in Figure 20. The cubic function is defined using two parameters - the penalty radius D at which the function becomes zero, and the maximum value of the function at the origin. The total probability $P(X|T)$ of landmark locations X given topology T is then calculated as

$$-\log P(X|T) = \sum_{\substack{1 \leq i1 < i2 \leq N \\ l_{i1} \notin S(l_{i2})}} L(l_{i1}, l_{i2}) \tag{16}$$

where $S(l_{i2})$ denotes the set containing l_{i2}.

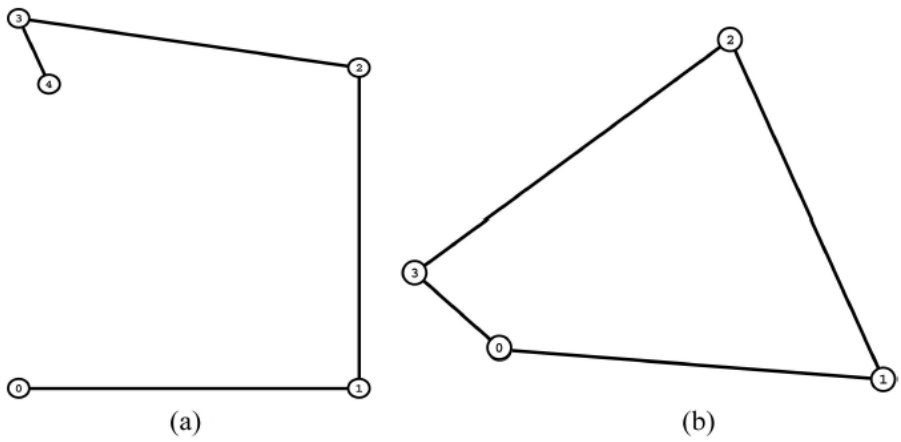


Figure 21: Illustration of optimization of the odometry likelihood. The observed odometry in (a) is transformed to the one in (b) because the topology used in this case, $(\{0,4\},\{1\},\{2\},\{3\})$, tries to place the first and last landmarks at the same physical location.

2.4 Numerical Computation of Odometry Likelihood

Evaluation of the odometry likelihood is performed using (14). The odometry likelihood function $P(O|X,T)$ in (14) encodes the deviation between the measured odometry and the odometry predicted by the topology and the landmark locations. Intuitively, the topology T constrains some measurements as being from the same location even though the odometry may put these locations far apart. The likelihood function accounts for the two types of errors: those from distorting the odometry and those from not conforming to the topology T. Hence, the log-likelihood for the odometry can be written as

$$-\log P(O|X,T) = \left(\frac{\|X - X_O\|}{\sigma_O}\right)^2 + \sum_{S\in T}\sum_{i1,i2\in S}\left(\frac{l_{i1} - l_{i2}}{\sigma_T}\right)^2 \tag{17}$$

where S is a set in the partition corresponding to T, σ_O and σ_T are standard deviations explained below, and X_o is the set of landmark locations obtained from the odometry measurements. The first term on the right hand side of (17) corresponds to the error from the odometry distortion while the second term corresponds to the topology constraints. The standard deviations for the odometry and topology constraints, σ_O and σ_T respectively, encode the amount of error that we are willing to tolerate in each of these quantities.

A simple example illustrating the constraints is given in Figure 21. In this example, the topology constrains X_o and X_4 (the first and last landmarks) to the same location causing a distortion in the odometry. This results in the topology and landmark locations in Figure 21(b).

In some cases, it may be possible to evaluate the integral in (14) analytically using the functional forms of the log-likelihood given in (17), and (16). If closed form evaluation is not possible, it may still be possible to use an analytical approximation technique such as Laplace's method [98] to evaluate (14).

However, in general, it is not possible to use any form of analytical evaluation to compute (14). Instead, we employ a Monte Carlo approximation, using importance sampling [27] to approximate the integrand $P(O|X,T)P(X|T)$. Importance sampling works by generating samples from a proposal distribution that is easy to sample from. Each sample is then weighted by the ratio of the target distribution to the proposal distribution evaluated at the sample location. The Monte Carlo approximation is subsequently performed by summing the weighted samples. The primary condition on the proposal distribution is that it should be non-zero at all locations where the target distribution is non-zero. In addition, importance sampling is efficient if the proposal distribution is a close approximation to the target distribution.

In our case, the importance sampling proposal distribution is obtained from the odometry log-likelihood (17). This function is a lower bound on the log of the integrand, $\log\left(P(O|X,T)P(X|T)\right)$, since the prior term given by (16) is never negative. Consequently, (17) can be used to obtain a valid importance sampling distribution. We employ Laplace's method to obtain a multivariate Gaussian distribution from $-\log P(O|X,T)$, which is used as the proposal distribution. This is achieved by computing the maximum likelihood path $X^\star$ through a non-linear optimization of $-\log P(O|X,T)$, and creating a local Gaussian approximation $Q(X|O,T)$ around $X^\star$

$$X^\star = \underset{X}{\operatorname{argmax}} \ \left(-\log P(O|X,T)\right)$$

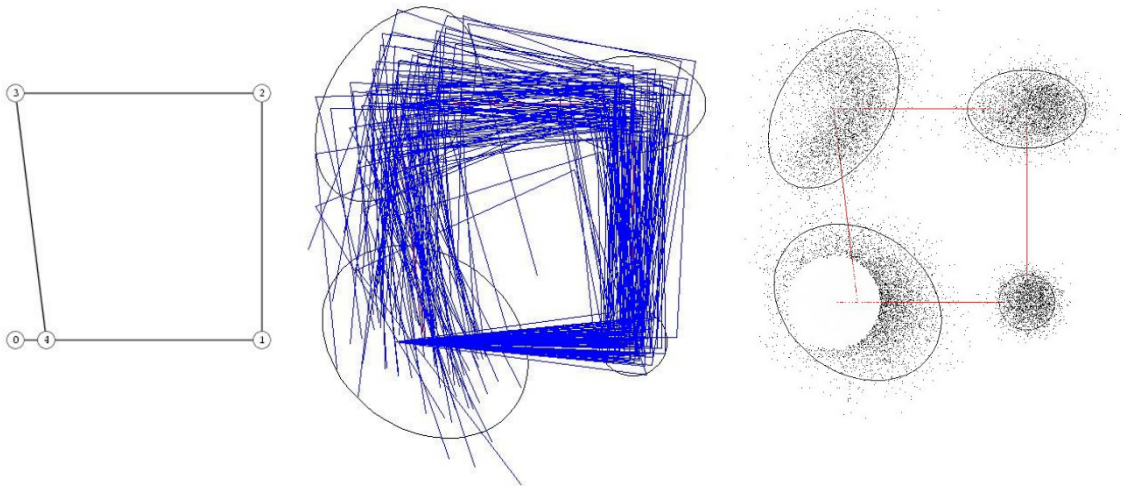


Figure 22: Illustration of the proposal distribution for importance sampling. The left figure shows an example topology where two distinct nodes are close together. The proposal distribution is a Gaussian around a topology whose node locations are obtained using an optimization. Samples from this Gaussian are shown in the middle figure. The importance weights of these samples are shown in the right figure, where darker dots represent larger weights. Note that due to the landmark prior, samples that place nodes 0 and 4 close together get very low weights. This can be seen in the circular region around node 0 where all samples get low weights.

$$Q(X \mid O,T) = \frac{1}{\sqrt{|2\pi\Sigma|}} e^{-\frac{1}{2}(X-X^\star)^T \Sigma^{-1}(X-X^\star)}$$

where Σ is the covariance matrix relating to the curvature of $-\log P(O|X,T)$ around X^*. Details of the Laplace approximation are given in Appendix B. The distribution $Q(X|O,T)$ is then used as the proposal distribution for the importance sampler. This proposal distribution is illustrated in Figure 22.

In practice, we use the Levenberg-Marquardt algorithm in conjunction with a sparse QR solver to perform the optimization described above. The Levenberg-Marquardt algorithm requires the derivative of the objective function that is being minimized, in this case the function $\psi(X) = -\log P(O|X,T)$ in (17). To compute the (sparse) Jacobian H given by $H = \frac{\partial \psi(X)}{\partial X}$, we use an automatic differentiation (AD) framework. Automatic differentiation (AD) is a technique for augmenting computer programs with derivative computations. It exploits the fact that by applying the chain rule of differential calculus repeatedly to elementary operations, derivatives of arbitrary order can be computed automatically and accurately to working precision. See [31] for more details.

The odometry likelihood given by (14) is now evaluated using the Monte Carlo approximation

$$\int_X P(O|X,T)P(X|T) \approx \frac{1}{N}\sum_{i=1}^{N} \frac{P(O|X^{(i)},T)P(X^{(i)}|T)}{Q(X^{(i)}|O,T)} \tag{18}$$

where the $X^{(i)}$ are samples obtained from the Gaussian proposal distribution $Q(X|O,T)$ and N is the number of samples.

2.5 Appearance Modeling Using Fourier Signatures

In this section, we present a model for appearance measurements derived from a specific camera setup. What we have termed as appearance measurements can come in a wide variety of forms, with equally varied models. The appearance model presented here may be considered as an example that demonstrates how various forms of appearance measurements can be incorporated into our topological mapping framework.

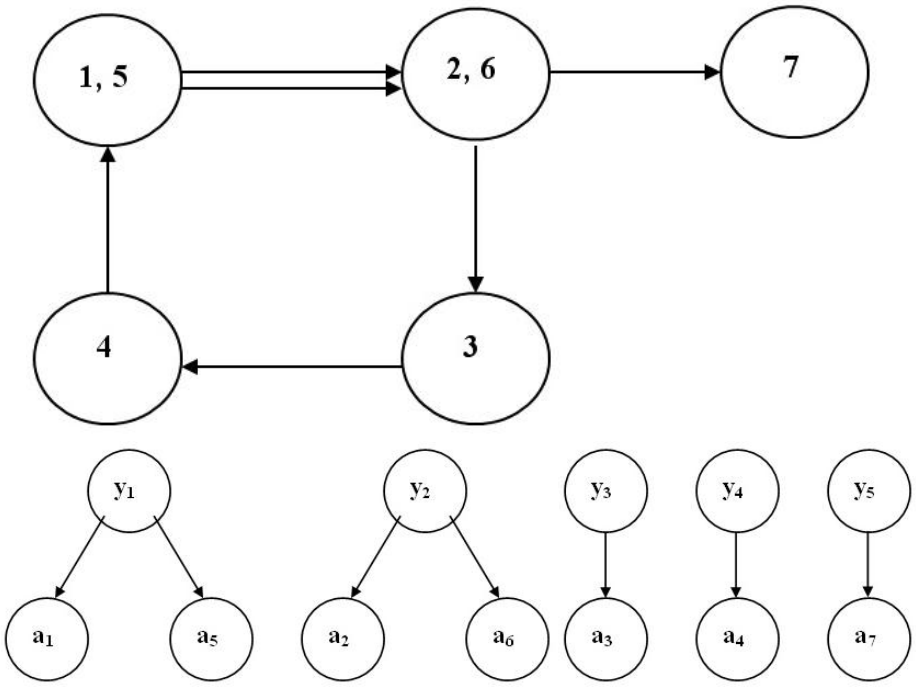


Figure 23: The Bayesian network (b) that encodes the independence assumptions for the appearance measurements in the topology (a) given the true appearance $Y = \{y_1, \ldots, y_5\}$ at all the landmark locations. The measurements corresponding to different landmarks are independent.

In the general case, estimation of the appearance likelihood $P(A \mid T)$, where $A = \{a_i | 1 \leq i \leq N\}$ is the set of abstracted appearance measurements, is performed via the product partition model formulation given by (2) and (3). This done by introducing the hidden parameter $Y = \{y_s | s \in T\}$, which denotes the "*true appearance*" corresponding to each physical landmark in the topology. As we do not need to compute Y when inferring topologies, we marginalize over it so that

$$P(A \mid T) = \int_Y P(A \mid Y, T) P(Y \mid T) \tag{19}$$

where $P(A|Y,T)$ is the measurement model and $P(Y \mid T)$ is the prior on the appearance. We assume that the appearance of a landmark is independent of all other landmarks, so that each y_s is independent of all other $y_{s'}$. The prior $P(Y \mid T)$ can thus be factored into a product of priors on the individual landmark appearances y_s.

$$P(Y \mid T) = \prod_{s \in T} P(y_s) \tag{20}$$

The topology T introduces a partition on the set of appearance measurements by determining which "true appearance" y_s each measurement a_i actually measures, i.e the partition encodes the correspondence between the set A and the set Y. Also, given Y, the likelihood of the appearance can be factored into a product of likelihoods of the individual appearance instances. This is illustrated using an example topology in Figure 23, where the Bayesian network encodes the independence assumptions in the appearance measurements. Hence, denoting a set in the partition as s, we rewrite $P(A \mid Y,T)$ as -

$$P(A \mid Y,T) = \prod_{s \in T} \prod_{a_i \in s} P(a_i \mid y_s) \tag{21}$$

where the dependence on T is subsumed in the partition. Combining Equations (19), (20) and (21), we get the expression for the appearance likelihood as

$$P(A \mid T) = \prod_{s \in T} \int_{y_s} P(y_s) \prod_{a_i \in s} P(a_i \mid y_s) \tag{22}$$

In the above equation, $P(y_s)$ is a prior on appearance in the environment, and $P(a_i \mid y_s)$ is the appearance measurement model. Evaluation of the appearance likelihood requires the specification of these two quantities.

We now instantiate the appearance model presented in the previous section using Fourier signatures [36][62] of panoramic images as measurements. The panoramic images are obtained from a camera rig of eight cameras mounted on a robot as shown in Figure 24. An example panoramic image is shown in Figure 25. Fourier signatures, which have previously been used in the context of memory-based navigation [62] and localization using omni-directional vision [63], are a low-dimensional representation of images using Fourier coefficients. They allow easy matching of images to determine correspondence. Further, due to the periodicity of panoramic images, Fourier signatures are rotation-invariant. This property is of prime importance when determining correspondence since the robot may be moving in different directions when the images are obtained.

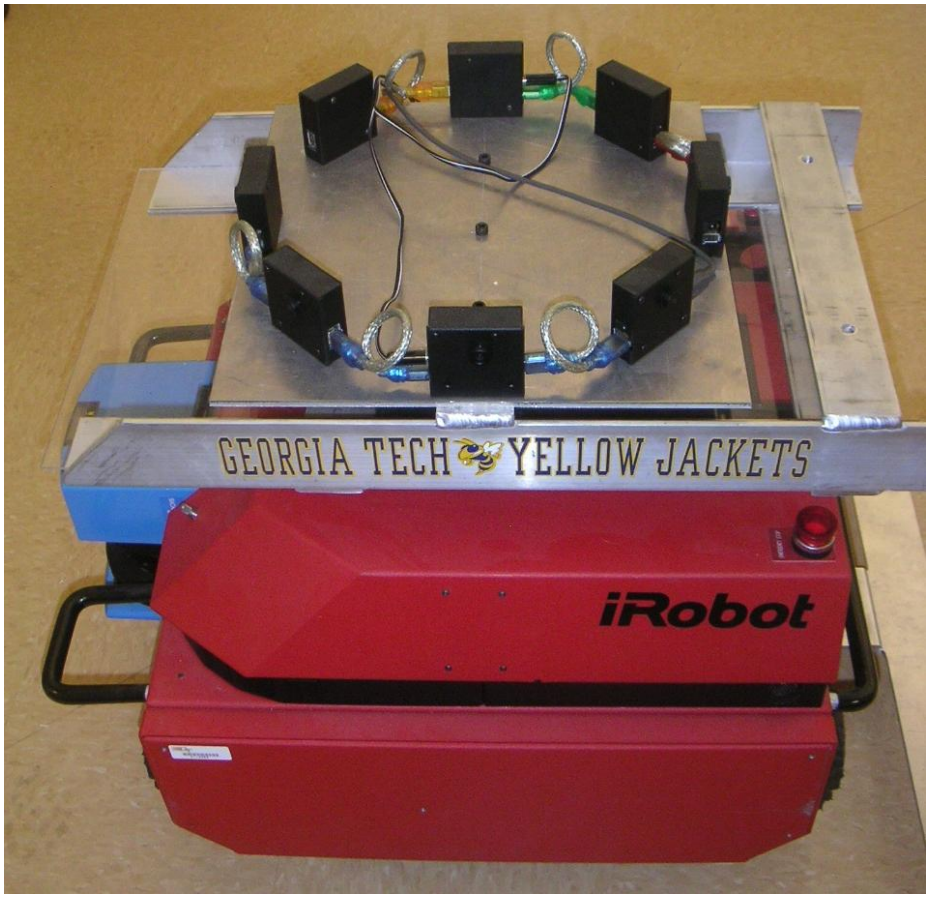


Figure 24: The camera rig mounted on the robot used to obtain panoramic images

Fourier signatures are computed by calculating the 1-D Fourier transform of each row of the panoramic image and storing only the few coefficients corresponding to the lower spatial frequencies [62]. While more popular dimensionality reduction techniques such as PCA [41] exist, the drawback of such systems is the need to further preprocess the measurement images in order to obtain rotational invariance. In contrast, the magnitudes of Fourier coefficients in a Fourier signature are invariant to in-plane rotation since panoramic images

Figure 25: A panoramic image obtained from the robot camera rig

are periodic. Hence, a Fourier signature yields a low-dimensional, rotation-invariant representation of the image. We use images obtained from an eight-camera rig mounted on a robot to produce panoramic images. The eight images obtained at each point in time are stitched together automatically to form a 360^o view of the environment.

In our case, Fourier signatures are calculated using a modification of the procedure given in [62]. Firstly, a single row image obtained by averaging the rows of the input image is calculated and subsequently, the one-dimensional Fourier transform of this image is performed. This gives us the Fourier signature of the image. It is to be noted that Fourier signatures do not comprise a robust source of measurements, since the measurements contain many false positives, in the sense that images from distinct physical locations often yield similar Fourier signatures. This is due to perceptual aliasing and the extreme compression of the Fourier signature. However, they have the advantage of being simple to compute and model. Moreover, in conjunction with odometry, they still produce good results as will be demonstrated.

Evaluation of the appearance likelihood is performed using (22). However, in this case, each appearance measurement a_i is a Fourier signature vector given as $a_i = \{a_{i1}, a_{i2}, \ldots, a_{iK}\}$, where a_{ik} is the kth Fourier component in the Fourier signature. We assume a similar vector form for the hidden appearance variables y_s, so that $y_s = \{y_{s1}, y_{s2}, \ldots, y_{sK}\}$. We can then write (22) as

$$P(A|T) = \prod_{s \in T} \int_{y_s} P(y_{s1}, \ldots, y_{sK}) \times \prod_{a_i \in s} P(a_{i1}, \ldots, a_{iK} \mid y_{s1}, \ldots, y_{sK}) \tag{23}$$

The various frequency components of the Fourier signature are assumed to be independent conditioned on the corresponding appearance variable, and can be factored, as can be the prior over the hidden appearance variables. Consequently, we modify (23) to get the expression for the appearance likelihood as

$$P(A \mid T) = \prod_{s \in T} \prod_{k=1}^{K} \int_{y_s} P(y_{sk}) \prod_{a_i \in s} P(a_{ik} \mid y_{sk}) \tag{24}$$

We assume the measurement noise in the Fourier signatures to be Gaussian distributed so that the model for appearance instance a_{ik}, belonging to the set s, is also a Gaussian centered around the "true appearance" y_{sk} with variance σ_{sk}^2. Since we do not know either of these parameters, we further model them hierarchically. Conjugate priors are placed on σ_{sk}^2 and y_{sk} : the prior on σ_{sk}^2 being an inverse gamma distribution while the prior on y_{sk} is taken to be a Gaussian distribution with mean μ and variance $\frac{\sigma_{sk}^2}{\kappa}$ [27]. This particular choice of priors also allows the integration in (24) to be performed analytically. The appearance model can then be summarized as

$$\begin{aligned} a_{ik} &\sim \mathcal{N}(y_{sk}, \sigma_{sk}^2) \quad \text{where } a_i \in s \ , \ s \in T \\ y_{sk} &\sim \mathcal{N}(\mu, \frac{\sigma_{sk}^2}{\kappa}) \\ \sigma_{sk}^2 &\sim IG(\alpha_k, \beta_k) \end{aligned} \tag{25}$$

where IG denotes the inverse gamma distribution. Note that while the value of κ is generally chosen so that the prior on y_{sk} is vague, we usually have some extra "world knowledge" that can be used to set the values of the hyper-parameters α_k and β_k. For example, if we expect the value of the Fourier signature to vary by only a small amount in the neighborhood of a given location, the prior on σ_{sk}^2 should reflect this knowledge by being peaked about a specific value.

The generative model for Fourier signature measurements specified by (25) is now used to compute the appearance likelihood given by (24). In addition to integrating over y_{sk}, we also integrate over the variance σ_{sk}^2 as we are not interested in its value. It follows that

$$\begin{aligned} P(A \mid T) = \prod_{s \in T} \prod_{k=1}^{K} \int_{\sigma_{sk}^2} IG(\alpha_k, \beta_k) \times \\ \int_{y_{sk}} \mathcal{N}(\mu, \frac{\sigma_{sk}^2}{\kappa}) \left(\mathcal{N}(y_{sk}, \sigma_{sk}^2)\right)^{|s|} \end{aligned} \tag{26}$$

To compute the expression for the likelihood, consider the integral above which is the probability of a set in the topology taking into account only one frequency component

$$P(s) = \int_{\sigma_{sk}^2, y_{sk}} IG(\alpha_k, \beta_k) \mathcal{N}(\mu, \frac{\sigma_{sk}^2}{\kappa}) \left(\mathcal{N}(y_{sk}, \sigma_{sk}^2)\right)^{|s|}$$

Plugging in the functional forms of the distributions, we get

$$P(s) = K_s \int_{\sigma_{sk}^2} \left(\sigma_{sk}^2\right)^{-A_s} e^{-\frac{\beta}{\sigma_{sk}^2}} \int_{y_{sk}} e^{-\frac{1}{2\sigma_{sk}^2} B_{sk}}$$

where

$$\begin{aligned} K_s &= \frac{\beta^\alpha}{\Gamma(\alpha)} \frac{\kappa^{\frac{1}{2}}}{(2\pi)^{\frac{|s|+1}{2}}} \\ A_s &= \alpha + \frac{|s|}{2} + \frac{3}{2} \\ B_{sk} &= \kappa(y_{sk} - \mu)^2 + \sum_{a_i \in s} (a_{ik} - y_{sk})^2 \end{aligned}$$

Performing the inner integration, we get

$$P(s) = K' \int_{\sigma_{sk}^2} \left(\sigma_{sk}^2\right)^{-\gamma_s} e^{-\frac{1}{\sigma_{sk}^2}\left(\beta + \frac{1}{2}\Phi_{sk}\right)} \tag{27}$$

where

$$\begin{aligned} K' &= \frac{1}{(2\pi)^{\frac{|s|}{2}}} \frac{\beta^\alpha}{\Gamma(\alpha)} \left(\frac{\kappa}{\kappa + |s|}\right)^{\frac{1}{2}} \\ \Phi_{sk} &= \kappa(\mu^\star - \mu)^2 + \sum_{a_i \in s} (a_{ik} - \mu_{sk}^\star)^2 \\ \mu_{sk}^\star &= \frac{\kappa\mu + \sum_{a_i \in s} a_{ik}}{\kappa + |s|} \\ \gamma_s &= \alpha + \frac{|s|}{2} + 1 \end{aligned}$$

We now provide here a useful definition of the Gamma function

$$\int_0^\infty e^{-\alpha t} t^\gamma dt = \frac{\Gamma(\gamma+1)}{\alpha^{(\gamma+1)}}$$

using which (27) can be integrated (note that t corresponds to σ_{sk}^{-2}) to yield

$$P(s) = K' \frac{\Gamma(\gamma_s + 1)}{\left\{\beta + \frac{1}{2}\Phi_{sk}\right\}^{(\gamma_s+1)}}$$

whence we get the expression for the appearance likelihood as

$$P(A|T) \propto \prod_{s \in T} C_s^K \prod_{k=1}^{K} \Gamma(\gamma_{sk} + 1) \left(\beta + \frac{1}{2}\Phi_{sk}\right)^{-(\gamma_s+1)} \tag{28}$$

where

$$C_j \;=\; (\kappa + |s|)^{-\frac{1}{2}}$$

Φ, γ, and $\mu^\star$ are as above, and constants that do not affect the likelihood ratio have been omitted.

The appearance model presented above is not specific to Fourier signatures. Indeed, it is a general purpose clustering model that assumes that the data to be clustered are distributed as a mixture of Gaussians, where the number of mixture components is determined by the topology.

2.6 Results

We now present results obtained from our implementation of the MCMC sampling algorithm for computing PTMs. All experiments were performed using an ATRV-Mini mounted with an eight-camera rig. The landmarks in the experiments were selected manually. In all cases, we initialized the sampler with the partition that assigned each measurement to its own set.

The first set of experiments were conducted using odometry measurements alone. Nine landmark locations were observed during a run of approximately 15 meters. The raw odometry obtained from the robot, labeled with the landmark locations, and the ground-truth topology are shown in Figure 26. Considering only the odometry measurements leads to a posterior given by

$$P(T|O) \;\propto\; P(O|T)P(O)$$

where the odometry likelihood was evaluated as explained in Section 2.2. The penalty radius parameter used in the landmark prior was set to 2.5 meters for this experiment. The Dirichlet process prior presented in Section 1.3.2 was used in these and all other experiments presented in this chapter.

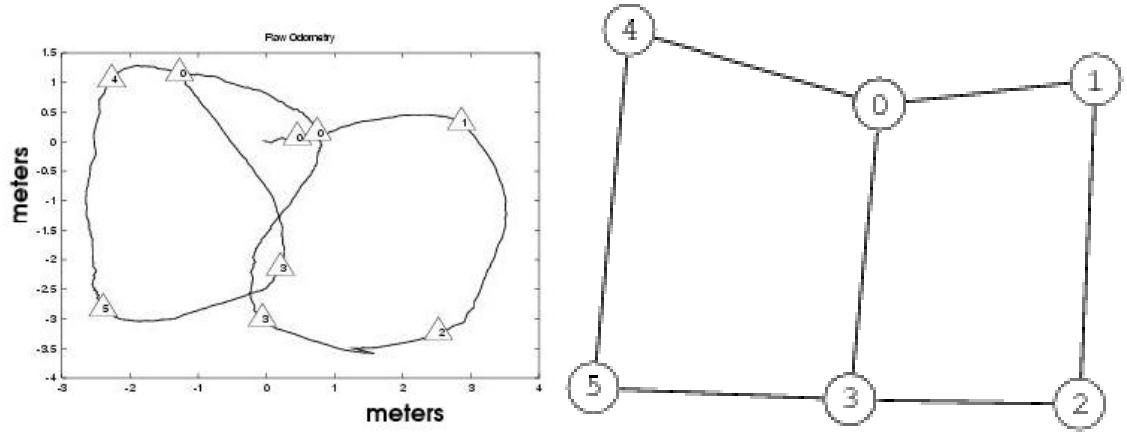


Figure 26: (a) Raw odometry (in meters) and (b) Ground truth topology from the first experiment involving 9 observations

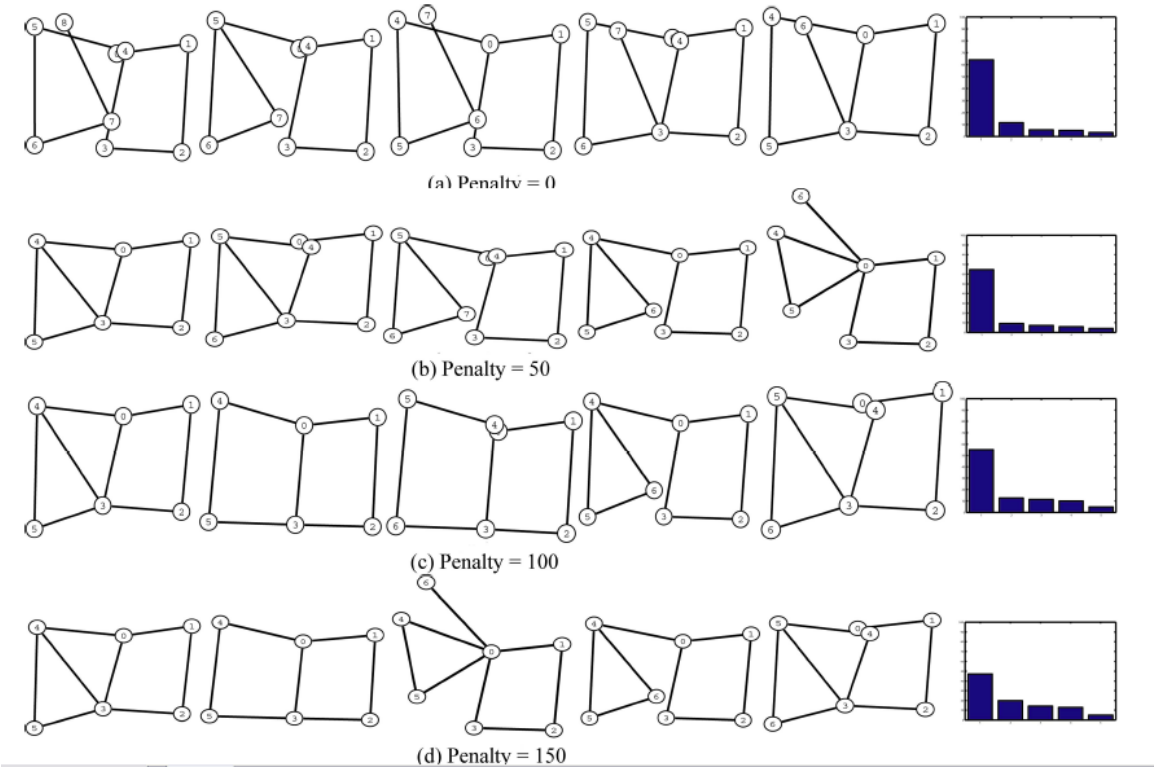


Figure 27: Change in probability mass with maximum penalty of the five most probable topologies in the histogrammed posterior. The histogram at the end of each row gives the probability values for each topology in the row.

Figure 27 shows the evolution of the MCMC sampler for different values of the penalty radius parameter. The penalty term facilitates merging of nodes arising from the same physical landmark. Without any penalty, the system has no incentive to move toward a topology with fewer number of nodes as this increases the odometry error. Table 27(a) illustrates this case. Here, the maximum penalty value is zero, and hence, the topology that is closest to the raw odometry data and also has the maximum possible nodes gets the maximum probability mass. For the remaining cases with maximum penalties equal to 50, 100, and 150 respectively, the most likely solution has fewer nodes, though it is still the one closest to the odometry. Since odometry is the only type of measurements that we have, and these measurements have a large error, it is a perfectly valid result that the ground truth topology is less likely. However, the ground truth topology is still the second-most likely topology for maximum penalty values 100 and 150. This is because as the penalty is increased the effect of odometry diminishes. However, a very large penalty swamps odometry data and makes absurd topologies more likely.

The second experiment demonstrates the usefulness of appearance in disambiguating noisy odometry measurements. Appearance is incorporated using the likelihood computation described in Section 2.5. The posterior over topologies is computed assuming conditional independence of appearance and odometry given the topology, i.e.

$$P(T|O,A) \quad \propto \quad P(O|T)P(A|T)P(T)$$

The experiment involving appearance measurements was conducted in an indoor office environment in the CRB building at Georgia Tech where the robot traveled along the corridors in a run of approximately 200 meters and observed nine landmarks. A floorplan of the experimental area is shown in Figure 28. The landmark locations obtained using odometry are shown in Figure 29. As in the first experiment, the five most likely topologies from the target distribution were obtained using only odometry measurements. A penalty radius of 20 meters and a maximum penalty of 100 were used to obtain the topologies, which are shown in Figure 30. As before, the ground truth topology receives only a small probability

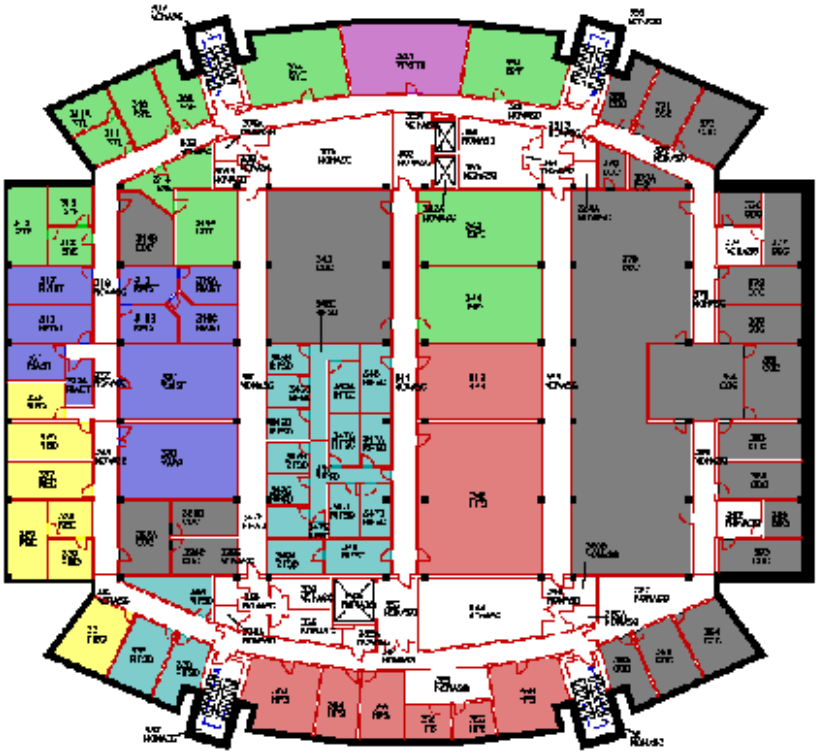


Figure 28: Floorplan of experimental area for CRB dataset

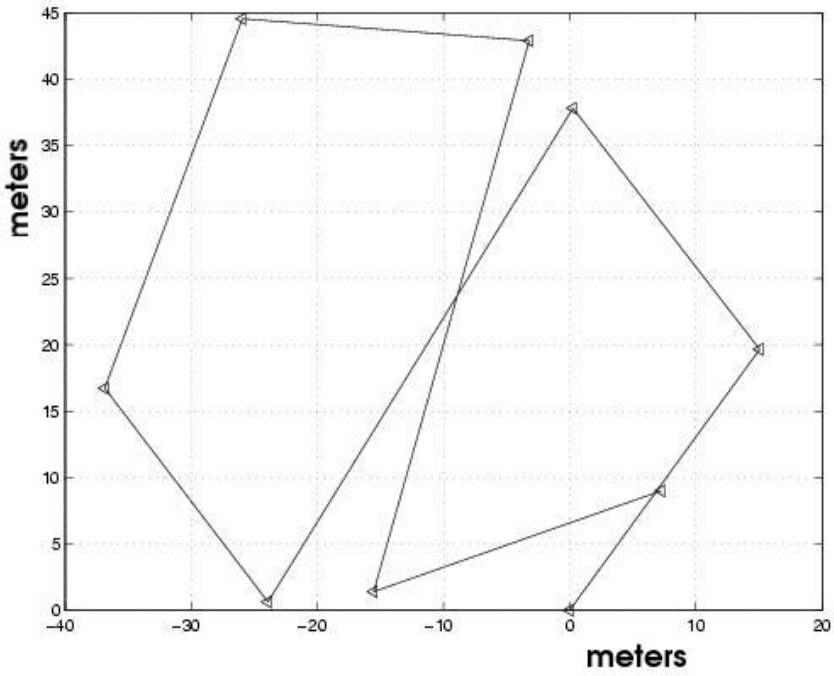


Figure 29: Landmark locations (in meters) plotted using odometry for the CRB dataset

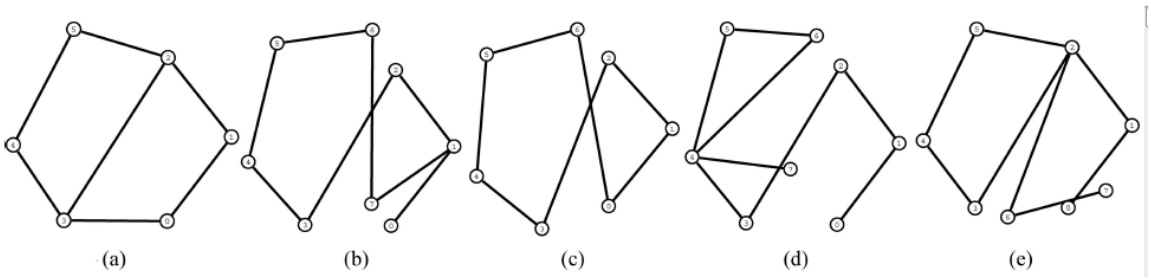


Figure 30: The topologies with highest posterior probability mass for the second experiment using only odometry (a) an incorrect topology receives 91% of the probability mass while the ground truth topology (b) receives 6%, (c), (d) and (e) receive 0.9%, 0.8% and 0.7% respectively.

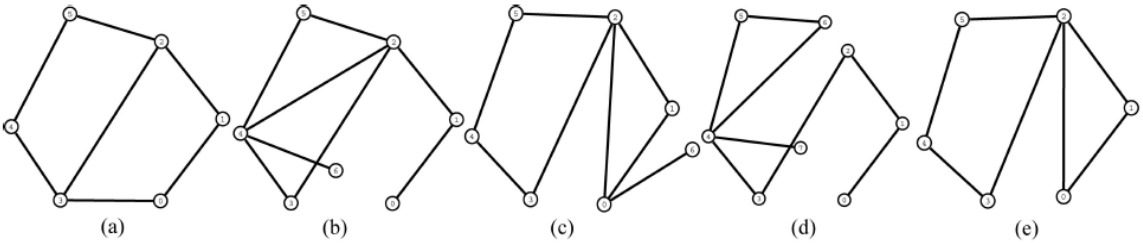


Figure 31: Topologies with highest posterior probability mass for the second experiment (CRB dataset) using odometry *and* appearance (a) The ground truth topology receives 94% of the probability mass while (b), (c), (d) and (e) receive 3.2%, 1.2%, 0.3% and 0.3% of the probability mass respectively.

due to noisy odometry.

We now repeat the experiment, but this time also using the appearance measurements, i.e. the Fourier signatures of the panoramic images obtained from the landmark locations, in addition to the odometry. The first five frequencies of the Fourier signatures were used for this purpose. The values of the variance hyper-parameters in the appearance model were set so that the prior over the variance is centered at 500 with a variance of 50. The five most likely topologies in the resulting probability histogram are shown in Figure 31. Since the environment is only minimally perceptually aliased and appearance measurements provide strong information, the ground truth topology now gets the majority of the probability mass. This experiment illustrates how additional information can easily be incorporated into the PTM framework to yield a much stronger result, due to its Bayesian nature.

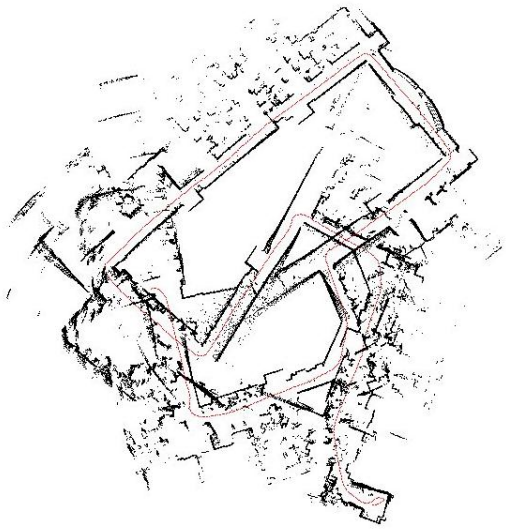


Figure 32: Odometry of the robot plotted with the laser measurements for the TSRB experiment.

The third experiment was conducted over an entire floor of the TSRB building at Georgia Tech and was complex in the sense that the robot run contained two loops, a bigger loop enclosing a smaller loop. Twelve landmarks were observed by the robot during the run, shown overlaid on a floorplan of the experimental area in Figure 33. The odometry of the robot with the laser plotted on top is shown in Figure 32. A penalty radius of 3.5 meters and a maximum penalty value of 100 were used in this experiment. Using only the odometry measurements, the ground truth topology did appear in the five most topologies in the PTM, but received a low probability mass. These results are given in Figure 34.

When appearance is also included, the results shift dramatically even though there is significant perceptual aliasing in this environment. Only two topologies appear in the PTM with the ground truth receiving almost all the probability mass. This experiment illustrates the fact that even when none of the measurement streams are highly reliable, their combination can produce good results in the sense that the PTM computed by our approach is sharply peaked and concentrated on very few topologies.

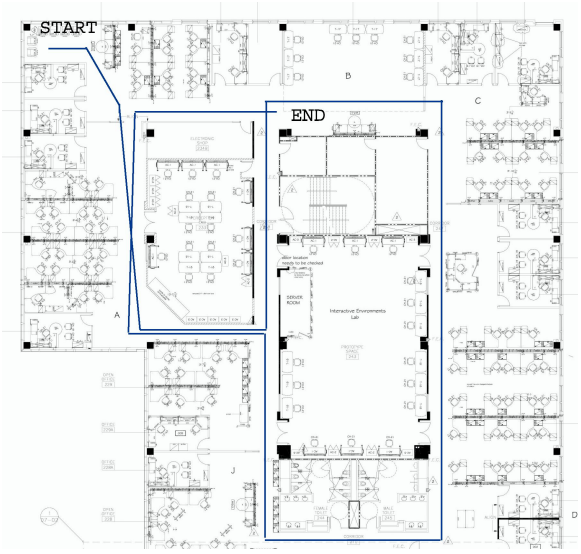


Figure 33: Floor plan with approximate robot path overlaid for the TSRB experiment.

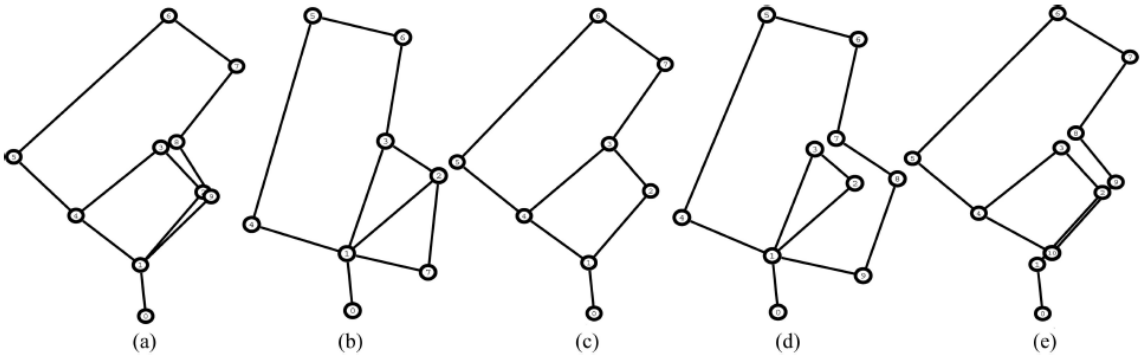


Figure 34: Topologies with highest posterior probability mass for the TSRB experiment using only odometry. (a) receives 43% of the probability mass while (b), (c), (d) and (e) receive 14%, 7.3%, 3.9% and 2.8% of the probability mass respectively. The ground truth topology is (c).

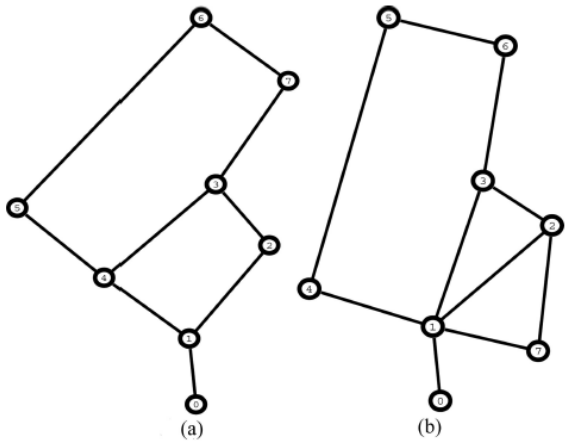


Figure 35: The two topologies constituting the PTM when both odometry and appearance measurements are used. The ground truth topology on the left receives 99.5% of the probability mass.

CHAPTER III

EFFICIENT AND ONLINE ALGORITHMS FOR COMPUTING PTMS

The MCMC algorithm described in the previous chapter maybe inefficient in many cases. These inefficiencies arise due to slow mixing and delayed convergence, two well-known problems with Markov chain methods. Further, MCMC is not an incremental algorithm, and hence, the addition of a new measurement to the inference requires starting from scratch, which is wasteful. In this chapter, I first describe variants of MCMC that overcome these problems and result in efficient algorithms for computing PTMs. Subsequently, I will present a particle filtering algorithm that enables incremental, online inference for PTMs. A vanilla particle filtering algorithm is however, also not completely efficient, as it often encounters the problem of particle degeneracy. I will describe the use of data-driven proposals to overcome this problem.

3.1 Convergence of Mixing in MCMC Methods

Markov Chain Monte Carlo (MCMC) techniques are evaluated for correctness in terms of two criteria - convergence and recurrence. An MCMC sampler is convergent if after a bounded amount of time, the Markov chain produces samples from the target distribution. On the other hand, a Markov chain is recurrent if every state in the state space that has a non-zero probability according to the target distribution is guaranteed to be visited within a finite time regardless of the initial state of the chain[1].

In our case, convergence of the sampler is guaranteed through use of the Metropolis-Hastings, which has been proven to be convergent due to the way the acceptance ratio is

[1]Technically, this is Harris recurrence, where the probability of visiting every state is 1. Simple recurrence requires only that the probability of visiting every state is non-zero.

computed [29]. However, recurrence is dependent on the design of the proposal distribution. The split-merge proposal distribution described in Section 2.1 produces a recurrent Markov chain since theoretically, it can reach any state in the space of topologies starting from any other state.

While correctness ensures that the sampler will eventually produce the correct samples, it does not say anything about efficiency. The sampler may take thousands of samples to converge to the correct distribution or the time between visits to high probability states may be large. Efficiency concerns regarding an MCMC sampler are closely related to correctness criteria -

- Fast Convergence

  The initial period when the sampler is converging to the target distribution is called the "burn-in". While it is not possible to compute the burn-in, it should be as small as possible, since the computations in the "burn-in" period are essentially wasted.

- Fast Mixing

  Mixing time is a characteristic property of a Markov chain that measures the time taken to move between high-probability regions in the state space. A Markov chain may be recurrent but slow mixing so that a large number of samples are needed to get an accurate sample-based representation of the target distribution.

Convergence depends to a large extent on the proposal distribution. If more samples from high probability regions are proposed often, fewer samples are rejected in the Metropolis-Hastings algorithm, and convergence is faster. In essence, the closer the proposal is to the target distribution, the faster the convergence. Mixing time is a reflection of the multi-modality of the target distribution in a discrete space. A large number of modes connected by low probability regions will make the sampler mix slowly since crossing over between the modes will take a long time.

The following sections in this chapter provides modifications on the original MCMC

sampler to enable fast convergence and rapid mixing. Convergence is sped up using smart data-driven proposal distributions that propose samples after looking at measurements, and hence, are able to provide more samples that are accepted into the Markov chain. Fast mixing is enabled through the use of simulated tempering, wherein multiple Markov chains with successively looser bounds on the target distribution are extended in the state space. Experiments at the end of the chapter demonstrate the improvements in convergence and mixing, and hence in efficiency, through the use of these techniques.

In the next section, a data-driven proposal distribution that uses odometry is described. While we only use the odometry-based proposal, proposals based on other measurements are also possible and useful if a particular measurement stream is superior to others. In this vein, an appearance-base proposal distribution is presented in Appendix C. Following this, we present the simulated tempering algorithm for fast mixing, and validate the algorithms with experiments and results.

3.2 Data-driven Proposals

The simplest proposal distribution that incorporates measurements is the target distribution itself. However, since the very use of MCMC implies that the target is extremely hard to sample from, this is of no use to us. Intuitively, we would like to incorporate those aspects of the data in the proposal that make it similar to the target distribution while also not being too computationally expensive. This increases the proportion of proposed samples that are accepted, while at the same time speeding up convergence.

The split-merge proposal distribution described in Section 2.1 does not take into account any domain knowledge, and hence, converges slowly. We now describe a proposal distribution that uses domain knowledge in the form of expected landmark locations, and leads to faster convergence of the Markov chain, thus making the PTM algorithm more efficient. A more general aspect of this proposal is that it demonstrates a means to include

pose information into any MCMC proposal that deals with the space of all possible clusterings. This is true since the space of topologies is exactly the same as that of all possible clusterings of available measurements.

Data-driven proposals have previously been used various fields - for example in Computer Vision for image segmentation [101], and in Statistics to analyze mixture models [39]. In general, data-driven proposals cause a significant speed-up in the sampling algorithm in cases where the state space being considered is enormous. In such cases, a normal proposal would provide a number of samples that are from regions of low probability and hence get rejected, wasting the computation involved in their generation. A proposal that utilizes the data, on the other hand, directs the proposed samples towards regions of higher probability, thus increasing the MCMC acceptance ratio and reducing the number of cases where the proposed sample is rejected.

3.3 An Odometry-based Proposal

Consider a topology $T = \{S_j | j \in [1, M]\}$, where the S_j are sets in a set partition of the measurements as before. If the Markov chain is currently in the state T, the task of the proposal distribution is to propose a new topology T' from T. With reference to the calculation of the Metropolis-Hastings acceptance ratio in (13) the probability of the step from T to T' as well as the reverse step has to be computed.

The basic steps of the proposal consist of the merge and the split moves as in Section 2.1. A *merge* move occurs when two of the sets in the set partition corresponding to T are merged to yield T'. Analogously, a *split* move occurs when a set in T is split into two. The number of ways in which a merge move can occur is given as $N_M = \binom{M}{2}$, $M > 1$, and the number of possible ways to split R into two subsets is given by $\left\{ \begin{matrix} |R| \\ 2 \end{matrix} \right\}$. Here $\left\{ \begin{matrix} n \\ m \end{matrix} \right\}$ denotes the Stirling number of the second kind as before, which gives the total number of ways a split move can occur on T as $N_S = N_{St} \left\{ \begin{matrix} |R| \\ 2 \end{matrix} \right\}$.

Algorithm 3 Data-driven Proposal Distribution using Odometry

1. Select a merge or a split with probability $\left\{\frac{N_M}{N_M+N_S}, \frac{N_S}{N_M+N_S}\right\}$
2. **Merge move:**
   (a) If T contains only one set, re-propose $T' = T$, hence $r = 1$
   (b) Obtain a discrete distribution on all merges in T as follows. For each pair of sets R and S in T
      i. Let D be the distance between the locations corresponding to R and S obtained by optimizing the odometry wrt T
      ii. Compute the probability of proposing the new topology $T'_{RS} = (T - \{R\} - \{S\}) \cup \{R \cup S\}$ as $Q(T \rightarrow T'_{RS}) = \frac{\exp\left(-\frac{D^2}{\sigma^2}\right)}{N_M+N_S}$
   (c) From the discrete distribution on merges computed above, sample a merge move. Let the new topology proposed be T'.
   (d) Probability of the reverse move $Q(T' \rightarrow T)$ is obtained from the reverse case 3(c), hence $r = \frac{N_M+N_S}{N'_M+N'_S} \exp\left(\frac{D^2}{\sigma^2}\right)$, where N'_M and N'_s are the number of merge and split moves possible from T'
3. **Split move:**
   (a) If T contains only singleton sets, re-propose $T' = T$, hence $r = 1$
   (b) Otherwise select a non-singleton set U at random from T and split into two sets R and S.
      i. Let D be the distance between the locations corresponding to R and S obtained by optimizing the odometry wrt the new topology $T' = (T - \{U\}) \cup \{R, S\}$
      ii. Compute the probability of proposing the new topology $Q(T \rightarrow T') = \frac{1}{N_M+N_S}$
      iii. Probability of the reverse move $Q(T' \rightarrow T)$ is obtained from the reverse case 2(b), hence $r = \frac{N_M+N_S}{N'_M+N'_S} \exp\left(-\frac{D^2}{\sigma^2}\right)$, where N'_M and N'_s are as defined in 2(b)

The data-driven proposal distribution, which computes the proposal ratio r used in the calculation of the acceptance ratio in (13), is given in Algorithm 3. The proposal begins by picking a split or merge move according to the proportion of the number of ways these moves are possible.

If a merge move is chosen, a discrete distribution of the probabilities of all possible merges is compiled. This is an $O(M^2)$ computation. Knowledge of the odometry measurements is introduced in computing the probability of a each possible merge. Intuitively, measurements that are taken when the robot pose is almost the same have a higher probability of being from the same landmark, and should have a higher probability of being merged. The landmark locations corresponding to the sets to be merged are obtained from the optimal robot trajectory, which in turn is obtained by optimizing the odometry under the constraints required by topology T as described above in Section 2.4. The topology T requires certain landmark measurements to correspond to the same physical landmark, i.e to occur at the same physical location. However, enforcing this constraint causes the trajectory of the robot to diverge from the odometry measurements. The optimal trajectory minimizes the total error due to divergence from the odometry measurements and not enforcing the constraints dictated by the topology T. The probability of the merge step is then obtained using the distance D between the landmarks that we are proposing to merge as $\exp\left(-\frac{D^2}{\sigma^2}\right)$, where σ^2 is a variance that encodes our belief in the distance between landmarks, or equivalently, the scale of the environment.

Once the distribution on merges is available, it is sampled to obtain a merge. This guarantees that sets corresponding to landmarks that are close together are merged preferentially.

The split step uses odometry data only for computing the reverse merge move. While, in theory, we could use the landmark locations to preferentially select splits that would result in a large separation, this is computationally expensive since the number of possible splits is much large than the number of possible merges. Also, since the initial state in sampling

Algorithm 4 Proposal Chaining

1. Select a chaining number C between 1 and C_{max} uniformly at random.
2. Initialize the forward and reverse move probabilities $p_f = p_r = \frac{1}{C_{max}}$.
3. For 1 to C do
   (a) Propose a sample according to a split-merge proposal (data-driven or general). Let the forward and reverse move probabilities be f and r respectively.
   (b) $p_f = p_f * f$ and $p_r = p_r * r$
4. Propose the last sample from step 3 with the move probabilities as p_f and p_r

is usually a topology where each measurement corresponds to a landmark, merges happen much more frequently than splits, so that proposing the right merge moves results in a quantum leap in efficiency. The split step is now simple since the probability of the split itself is just the inverse of total possible moves from T.

3.4 Proposal Chaining

While data-driven proposals speed up convergence by preferentially proposing samples that have a higher probability of acceptance, the design of our proposal distribution is still not ideal. In particular, the use of the split-merge proposal as the underlying basis of all the proposal algorithms results in the drawback that each successive state in the Markov chain can differ from its predecessor by only a single split or merge.

Consider the case where most of the neighbors of a topology, where neighbors are topologies that differ from each other by a single split or merge, have a low posterior probability. In this case, even the use of data-driven proposals will not help convergence since the chain cannot "leap over" the low probability neighborhood.

Proposal chaining offers a solution to the above problem by, as its name suggests, chaining multiple proposals to obtain topologies that can be significantly different from

the current one. At each link in the proposal chain, the forward and reverse proposal probabilities are multiplied so that the computation of the acceptance ratio (13) in the Metropolis-Hastings algorithm is correct. A synopsis of the chaining algorithm is given in Algorithm 4.

The number of times which the basic proposal has to be chained is decided randomly. Note that simply setting the chaining number to a constant does not help convergence. For example, if we were to chain two proposals all the time, the neighbors of a topology will only rarely be reached by the Markov chain. In our experiments, the maximum chaining number C_{max} is set to 3 and the number of chaining steps per proposal is randomly chosen between 1 and C_{max}. A very large C_{max} makes the chain jump around erratically and increases convergence time since the high probability regions in topological space are compact.

3.5 Simulated Tempering for Fast Mixing

MCMC methods work by locally extending a Markov chain through the state space to obtain samples from a distribution. The "speed" with which the Markov chain moves around in the space is called the mixing rate of the sampler. Naturally, fast mixing leads to a good representation of the posterior in the sample set, and in addition, fewer samples need to be computed overall.

One prime reason for slow mixing, especially if the proposal distribution is good, is the multi-model nature of the target function. This is illustrated by Figure 36. In such a function, once the chain reaches a modal location, it cannot easily escape it since all proposals to the lower probability regions surrounding the mode will be overwhelmingly rejected. Hence, even after thousands of samples. the sample-based posterior will give the wrong impression that the function is unimodal.

A number of solutions exist to this problem of slow mixing. One simple solution is to periodically reinitialize the Markov chain to a random state. The disadvantage of this

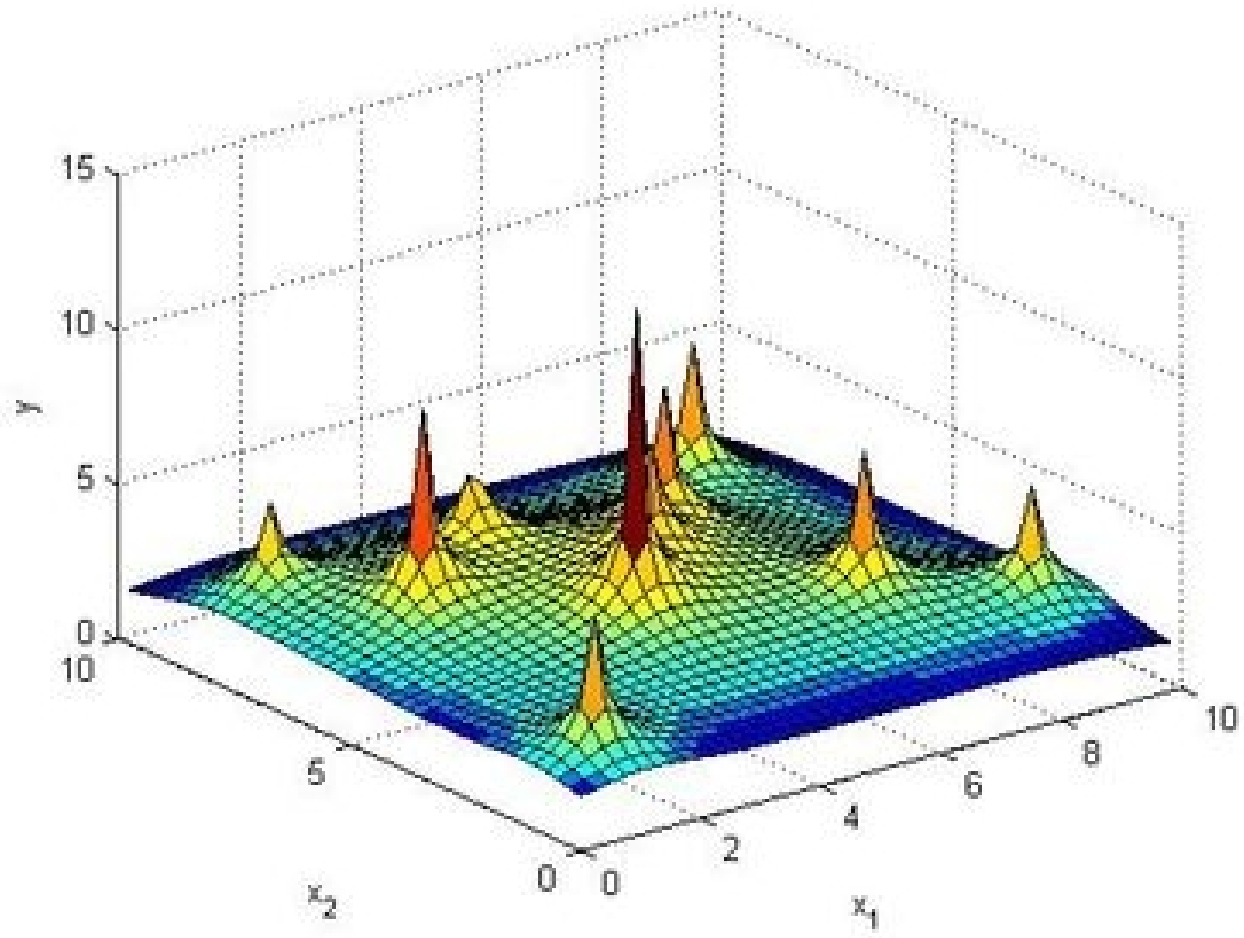


Figure 36: A highly-peaked multi-model function. The Markov chain may spend a huge amount of time in a single mode and mix very slowly if proper care is not taken.

strategy is that a burn-in time has to be allowed after each re-initialization, which may again take a long time. Another heuristic in continuous spaces is to use a proposal distribution with a large variance to enable the chain to jump across the probability troughs between the modes. However, increasing the variance too much will cause the chain to rapidly jump about spending proportionally little time in the modes. Also the results are usually highly sensitive to variance value leading to a lack of robustness. Furthermore, we are interested in the discrete space of set partitions where the notion of variance is ill-defined in any case.

We use the simulated tempering algorithm to enable fast mixing. Simulated Tempering is coupled Monte Carlo technique in which multiple Markov chains are run through the state space. The basic property of these chains is that only one of them samples from the original target distribution. The rest of chains sample from successively relaxed versions of the target distribution, in the sense that these distributions do not have the large variation between the modes and the troughs that characterizes the original target distribution.

While many variations are possible in the basic simulated tempering technique, we use the Metropolis-coupled MCMC (MC-cubed) algorithm of Geyer [28]. The MC-cubed algorithm works by running N coupled Markov chains in parallel with the first chain having the target distribution of interest, i.e. the posterior over topologies$P(T|Z)$, and the other chains having "heated" target distributions $P(T|Z)^{\beta}$ where $\beta = \frac{1}{1+(i-1)t}$ for the ith chain and t is a constant temperature increment. The peaks in the heated target distributions get increasingly smoothed out so that these chains mix more rapidly.

The samples from the heated chains are not useful as output. To enable the original Markov chain to mix rapidly, the states in the chains are exchanged after each step. The heated chains still produce states from high probability regions but also mix more rapidly so that this scheme can be considered as a "smart" version of the proposal with increased variance. Only samples from the original chain are considered for output. Considering the time it would take for a single chain to converge, this is still advantageous for reasonable N.

Algorithm 5 The Metropolis Coupled MCMC Algorithm

1. Let T_i be the current state of the ith chain - total number of chains being N.
2. For all chains $i \in (1,2,\ldots,n)$ do
   - Propose a new value for T_i using the proposal distribution and acceptance ratio

$$a_i = \min\left(1, \frac{q(T_i^{'} \rightarrow T_i)}{q(T_i \rightarrow T_i^{'})}\left(\frac{P(T_i^{'}|Z)}{P(T_i|Z)}\right)^{\beta_i}\right)$$

$$\beta_i = \frac{1}{1+(i-1)t}$$

3. After all chains have advanced one cycle, for each consecutive pair of chains i and $i-1$ (starting with n), swap the states of the chains with probability

$$r = \min\left(1, \frac{P(T_i|Z)^{\beta_{i-1}}}{P(T_{i-1}|Z)^{\beta_{i-1}}}\frac{P(T_{i-1}|Z)^{\beta_i}}{P(T_i|Z)^{\beta_i}}\right)$$

4. Goto step 2

The MC-cubed algorithm we use is given in Algorithm 5. In step 2, all the chains are extended using the same proposal distribution. In step 3, consecutive chains are swapped according to a swapping ratio r. This ratio ensures that the convergence properties of the Metropolis-Hastings algorithm remain applicable to the MC-cubed algorithm. An illustration of the algorithm using a toy problem of sampling from a bimodal target distribution is shown in Figure 37.

3.6 Results

The advantage due to the data-driven proposal was quantified by repeating the TSRB experiment from Chapter 2, this time using the new proposal distribution. As before, an ATRV-Mini mounted with the eight-camera rig was used, and both odometry and appearance measurements were incorporated. The PTM obtained was the same as in Figure 35 and is given in Figure 38 again. This proves that the MCMC sampling algorithm with the data-driven proposal produces the same results as before ans hence, is correct.

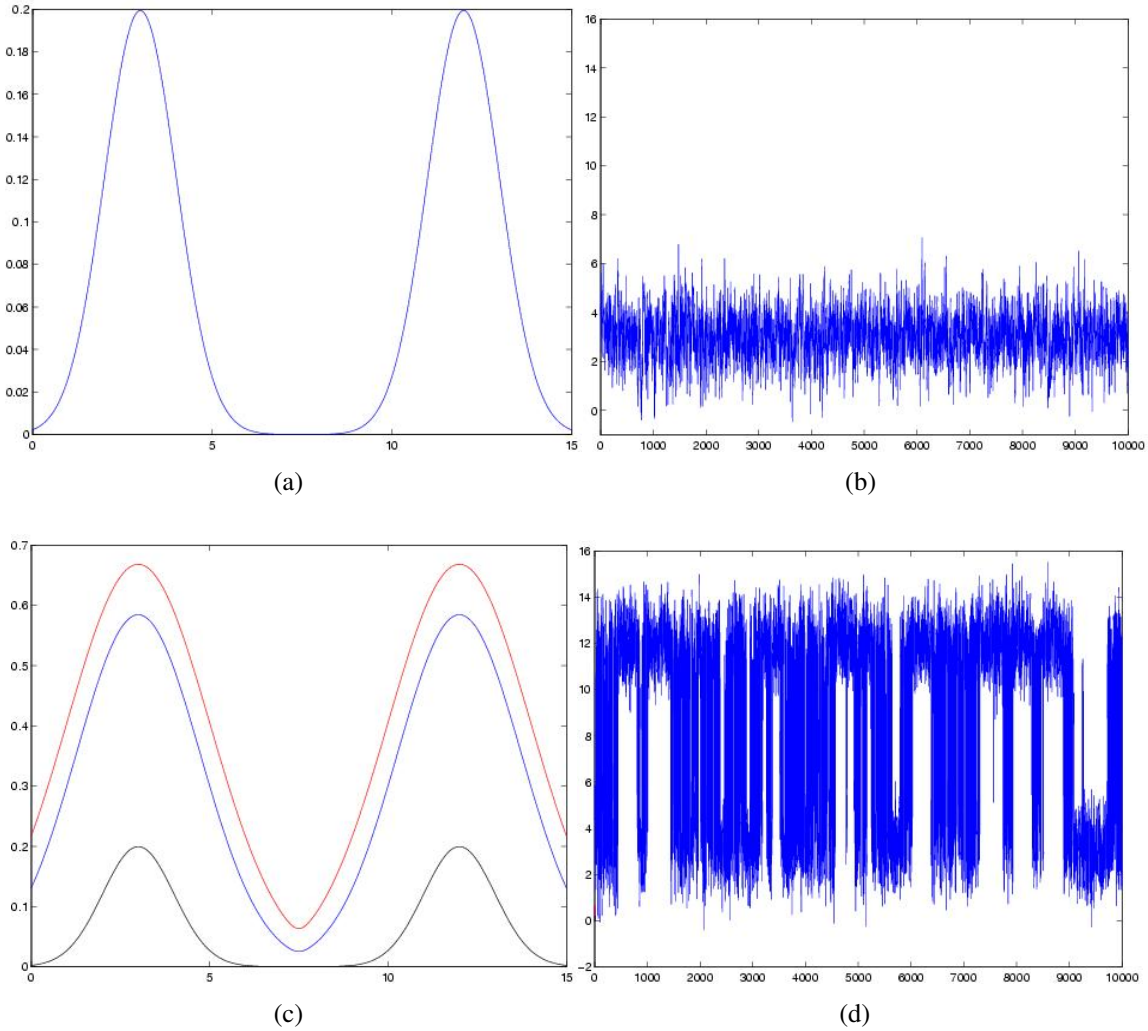


Figure 37: Illustration of Simulated Tempering. (a) The target distribution from which samples are to be obtained. Note that the two modes of the distribution are connected by a trough of extremely low probability that would normally not yield samples.
(b) Sampling using the Metropolis-Hastings algorithm yields a Markov chain that does not mix well. In this case, the chain stays in one mode of the distribution even after 10000 samples. The figure shows number of samples along the x-axis and the sample values along the y-axis. The sample trace is confined to the first mode. (c) The target distributions used for the MC-cubed algorithm with 3 chains and a temperature increment between chains of $t = 10$. The distribution in black is the original distribution of (a) while the other two distributions are the heated distributions at $t = 10$ (blue) and $t = 20$ (red). Note that heating makes the connection between the modes more likely. (d) The sample trace of the MC-cubed algorithm shows that samples mix evenly and rapidly between the two modes. The axes in this figure are the same as in (b).

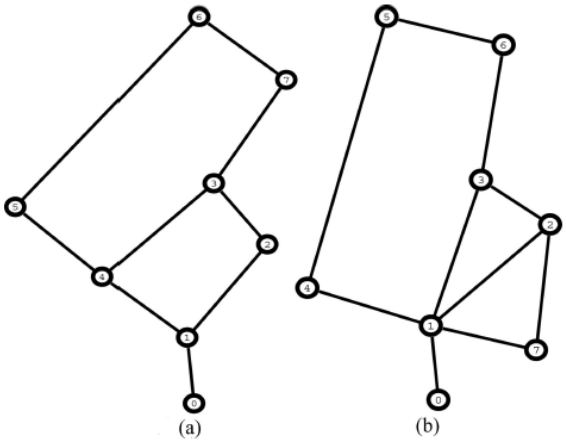


Figure 38: The two topologies constituting the PTM when both odometry and appearance measurements are used. The ground truth topology on the left receives 99.5% of the probability mass.

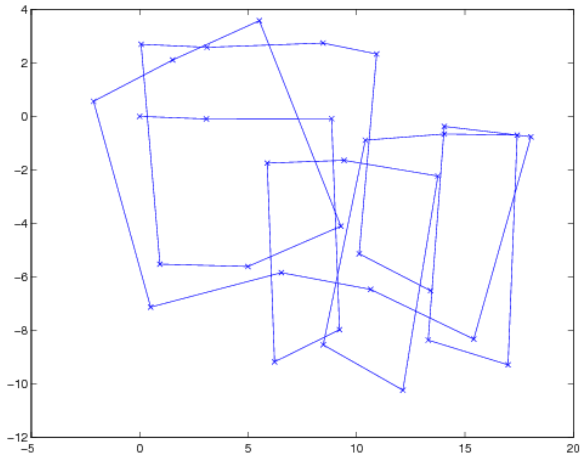
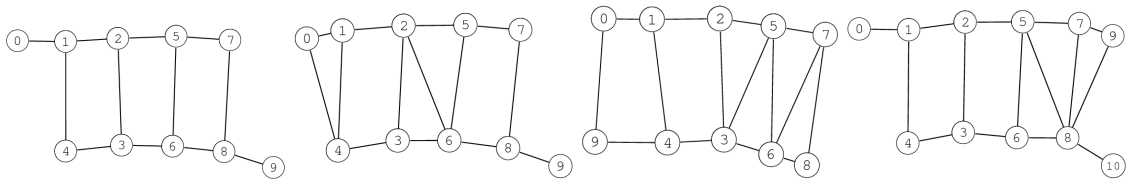


Figure 39: Landmark locations obtained from simulated odometry.

To demonstrate the scalability of the algorithm using the new proposal, we conducted a second experiment in simulation in an environment where the robot was made to traverse a number of loops. A total of 33 landmarks were observed by the robot in the run. The landmark locations obtained from odometry generated during the simulated run are shown in Figure 39. No appearance measurements were provided. The four most likely topologies in the PTM are shown in Figure 40.

The time to convergence was measured in both the experiments by running the algorithms multiple times with the number of samples successively doubled. When doubling



(a) (b) (c) (d)

Figure 40: Topologies with highest posterior probability mass for the simulation experiment. (a) the ground truth topology receives **71%** of the probability mass while (b), (c), and (d) receive 9.1%, 8.2%, and 6% of the probability mass respectively. The ground truth topology is (a).

	Data-driven proposal	*General proposal*
1^{st} *experiment*	156 seconds	~11 minutes
2^{nd} *experiment*	411 seconds	> 1 hour

Figure 41: Running times for computing the PTM using the two proposals in both the experiments. The data-driven proposal speeds up the algorithm by at least a factor of five.

the number of samples did not change the resulting PTM, the algorithm was declared to have converged and the time for the next to last run was noted. By this performance metric, using the data-driven proposal speeds up the PTM algorithm by a factor of six (Table 41) over both the experiments as compared to the general split-merge proposal of Section 2.1. For the simulated experiment, the convergent runtime of the original algorithm is unacceptable for almost all robot scenarios.

We tested mixing using the MC-cubed algorithm using Scaled Regeneration Quantile (SRQ) plots [71]. An SRQ plot displays visit times to a particular state plotted against the visit count, with both axes scaled to unity. The state chosen is usually a high probability one so that a number of visits can be observed. If the total length of the run is long enough, the plot should be close to a straight line through the origin with unit slope. In essence, this states that if the chain is well mixed, the visit times should not be dependent on their location within a run. Significant changes from unit slope, especially horizontal or vertical segments, signal poor mixing, i.e. the presence of tours from the state under observation that take much longer than others.

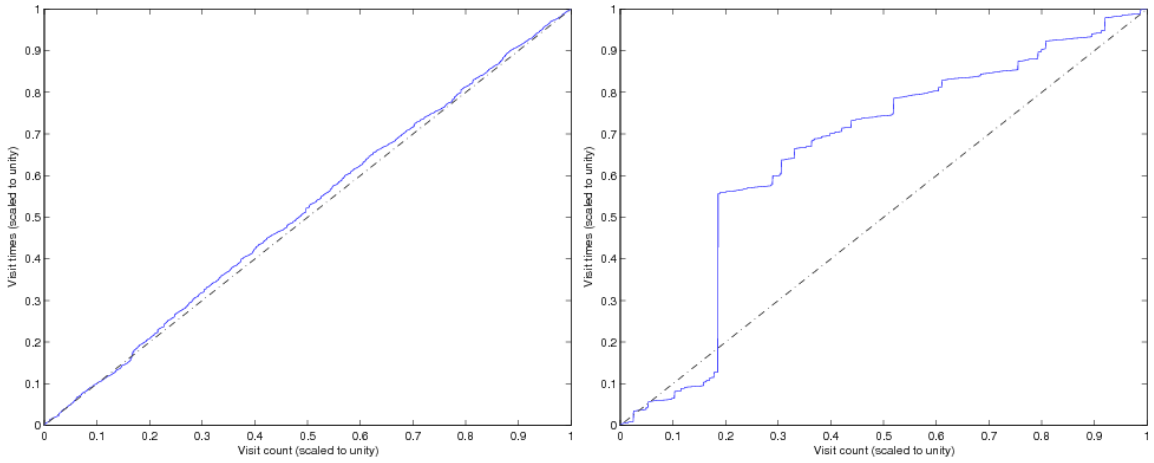


Figure 42: SRQ plots for (a) MC-cubed algorithm (b) single chain MCMC obtained using 15000 samples. The chain produces stable estimates if there are no significant deviations from unit slope.

Figure 42 shows the SRQ plots for the first experiment. It can be seen that the MC-cubed algorithm has an SRQ plot that closely follows the unit slope line, thus providing evidence for good mixing. On the other hand, the original algorithm with the general proposal mixes poorly.

The improvement in run time efficiency when using simulated tempering is quantified in Figure 43. The computation times shown are for plain MCMC, MCMC with the data-driven proposal, and the MC-cubed algorithm with the data-driven proposal. Topologies with 4 nodes (a simple square topology), 9 nodes, 12 and 33 nodes (experiments above) were considered for the comparison. As can be seen, the MC-cubed algorithm greatly improves the scalability of our technique. Run times shown are times to convergences obtained via a "sample doubling" scheme as before.

3.7 Particle Filters for Topological Mapping

This section introduces a sequential importance sampling (SIS) algorithm for topological mapping. While the MCMC algorithm performs well, especially using the data-driven

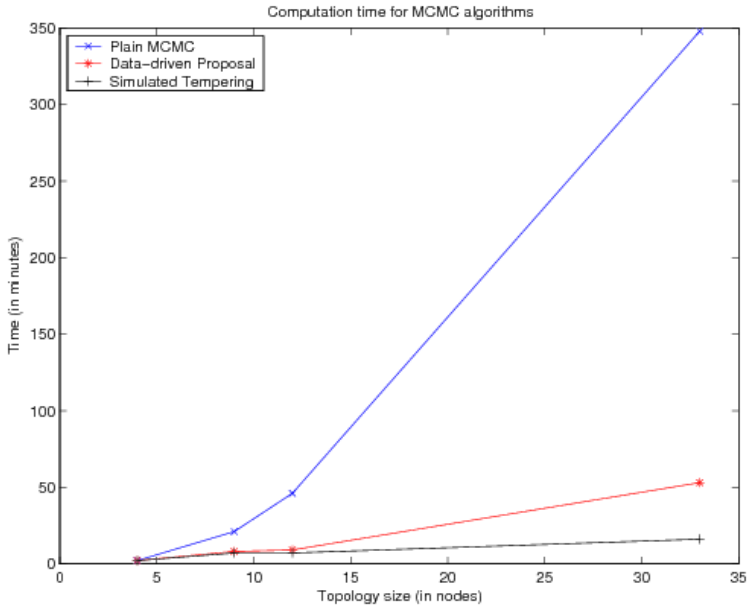


Figure 43: Computation times (rounded to the nearest minute) for the various MCMC algorithms for computing PTMs.

proposals and the MC-cubed algorithm, MCMC is not an incremental algorithm. Measurements cannot be added one at a time to the algorithm efficiently since each new measurement requires the computation of the PTM from scratch. This batch operation makes the MCMC algorithm unsuitable for online, real-time computation.

Particle filtering, as SIS is also known, is incremental since as it updates the posterior from the previous step by incorporating a measurement to obtain a new posterior. It is a filtering algorithm since the current estimate of the posterior depends only on the newest measurement and previous estimate. It is also a particle-based algorithm since the representation of the posterior at each step is through a set of weighted samples, called particles.

Applying particle filtering to topological mapping requires significant changes to the basic algorithm. The primary difference that in addition to maintaining the PTM, we also need to maintain the landmark location estimates which are needed for evaluating the odometry and, as we will see, laser measurement likelihoods. A straight-forward solution would be to maintain a joint posterior on topologies and landmark locations and have

the particle filter compute the posterior over this joint space. This presents a number of problems. Firstly, the joint space is too large to sample from efficiently. Secondly, the space is a part discrete and part continuous, and presents challenges for algorithm design.

We use a Rao-Blackwellized particle filter (RBPF) [68][70] to overcome the dual problems posed by the large discrete-continuous space of topologies and landmark locations. The RBPF samples only from the discrete topological portion of the space while maintaining the posterior over the continuous landmark locations in analytical form. The analytical posterior is updated at each step using a measurement and is subsequently used in the computation of the discrete posterior over topologies. This enables the RBPF to focus all its samples on representing the space of topologies, thus increasing efficiency.

We first provide a brief overview of the basic SIS algorithm and its modification to obtain an RBPF. This is followed by the exposition of the topological mapping algorithm which includes appearance and laser range scan measurements in addition to odometry. Subsequently, a data-driven proposal to enable fast convergence is provided. We conclude the chapter with results obtained through robot experiments.

3.8 Sequential Importance Sampling

SIS is applicable to problems where a distribution is estimated using filtering. A filtering approach assumes that a first-order Markov assumption holds and that measurements are conditionally independent given the state from which each measurement was obtained, i.e

$$\begin{aligned} x_t|x_{1:t-1} &\sim p_t(x_t|x_{t-1}) \\ p\left(z_{1:t}|x_{1:t}\right) &= \prod_{i=1}^{t} p_i(z_i|x_i) \end{aligned}$$

where x_t is the state at time t and $z_{1:t}$ are all the measurements since $t = 1$. Given the above formulation, a recursive filtering equation can be derived. First, we apply Bayes law to the joint posterior on states at time t

$$p(x_{1:t}|z_{1:t}) = \frac{p(z_t|x_t)p(x_{1:t}|z_{1:t-1})}{p(z_t|z_{1:t-1})} \tag{29}$$

where $p(z_t|x_t)$ is the measurement model, and the conditional independence of measurements has been used. Further we use the Markov property on states to obtain the desired recursive formula

$$p(x_{1:t}|z_{1:t}) \quad \propto \quad \frac{p(z_t|x_t)p(x_t|x_{1:t-1})}{p(z_t|z_{1:t-1})} p(x_{1:t-1}|z_{1:t-1}) \tag{30}$$

where $p(x_t|x_{t-1})$ is called the motion model since it "moves" the previous state x_{t-1} to the current state x_t.

The Bayes filter (30) can be solved in closed form for certain function forms of the distributions involved. Most famously, if all the distributions are Gaussian, this yields the Kalman filter. More often, one or more distributions may have a form that precludes analytical solution. The denominator involving the normalizing constant may also be uncomputable in closed form.

SIS is a non-parametric filtering technique that is useful when a parametric filtering solution is not possible. It is non-parametric as the representation employed is a set of weighted samples that approximate the filtering distribution $p(x_t|z_{1:t})$. This set of weighted samples is updated using the Bayes filter equation at each time step.

The underlying sampling technique used in SIS is importance sampling. Given a distribution which is hard to sample from, importance sampling makes use of an alternate distribution, called the proposal distribution, that is easier to sample from. Each sample from the proposal distribution is annotated with an importance weight. The weighted samples can be used to represent the distribution of interest and also to compute expectations and other statistical quantities.

Consider the distribution $p(x)$, which is the distribution of interest but is hard to sample from. We pick a proposal distribution $g(x)$ that is easy to sample from and generate samples $x^{(i)} \sim g(x)$ and associate weights with each of the samples given as $w^{(i)} = \frac{p(x^{(i)})}{g(x^{(i)})}$. The distribution $p(x)$ can now be equivalently represented using the sample set as $p(x) \approx$

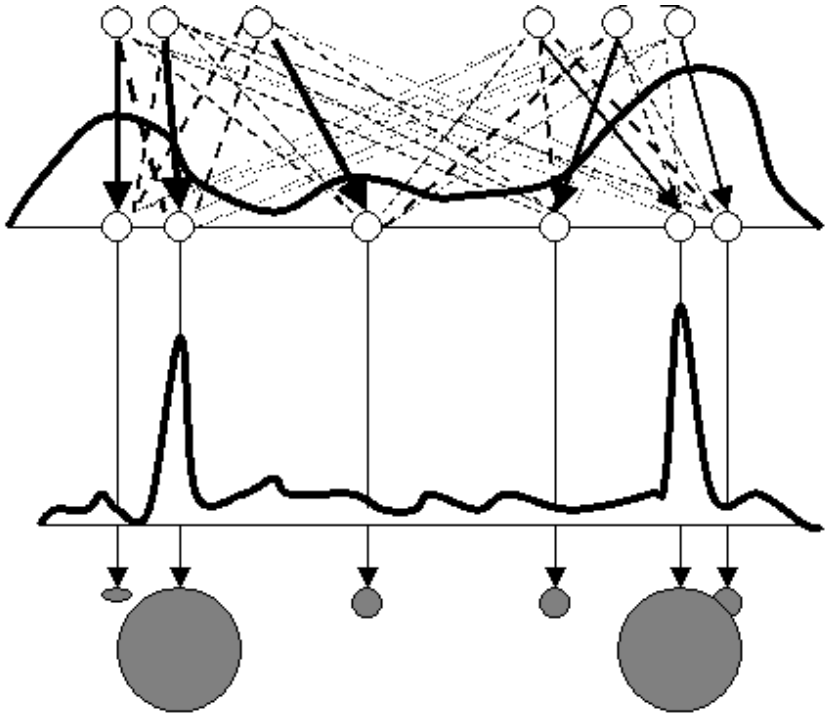


Figure 44: Importance sampling is performed through the use of a proposal distribution which is easy to sample from. Samples from the proposal distribution (top) are weighted by the target distribution (middle) to get samples with weights (bottom) which are the ratio of the target distribution and proposal distribution evaluated at the sample locations. Image obtained from http://www.lateral.hu/LSNIPS_html/HS_SIR.gif

$\left\{x^{(i)}, w^{(i)}\right\}_{i=1:n}$. An expectation involving $p(x)$ can be computed as

$$\int f(x)p(x)dx \quad = \quad \sum_i f(x^{(i)})w^{(i)} \tag{31}$$

Importance sampling is illustrated in Figure 44.

The choice of proposal distribution determines the correctness and efficiency of importance sampling. For correctness, the support of $g(x)$ should be a superset of the support of $p(x)$. Sampling will be progressively more efficient as the $g(x)$ and $p(x)$ coincide since few samples with very small weights will be computed in this case. The difference between the two distributions can be quantitatively characterized as the area bounded by them in 2D and the volume bounded by them in higher dimensional spaces. Due to this reason, a very good proposal distribution in 2D space may perform very poorly in 10D space. Consequently, the use of importance sampling in high dimensional spaces requires a huge number of samples unless very specialized proposal distributions are available.

SIS uses importance sampling to represent the filtering distribution as a set of weighted particles. The posterior at time $t-1$ is used to compute the posterior at time t by first using a proposal distribution to provide samples with x_t. Additionally, the weights of samples at time $t-1$ are updated. Given a proposal distribution $\pi\left(x_t|x^{(i)}_{1:t-1}, z_{1:t}\right)$, the SIS algorithm samples from π to obtain a new set of samples, which are weighted using the Bayes filter (30). The complete SIS algorithm is given in Algorithm 6.

The proposal distribution can also be the same as the motion model $p\left(x^{(i)}_t|x^{(i)}_{t-1}\right)$. In this case, the importance weight simplifies to include only the measurement likelihood.

$$w^{(i)}_t \quad = \quad w^{(i)}_{t-1} p\left(z_t|x^{(i)}_t\right) \tag{32}$$

The resampling step in Algorithm 6 is used to avoid the sample degeneracy problem where one sample has normalized weight close to unity while all other sample have negligible weights. The sampling degeneracy problem occurs because, in the absence of resampling, low probability samples that have found their way into the sample set have no way of being removed from the set. Hence, the algorithm performs wasteful computation

Algorithm 6 The Sequential Importance Sampling Algorithm with n samples

For times $t = 1, 2, \ldots$ do

- For $i = 1, 2, \ldots, n$, sample $x_t^{(i)} = \pi\left(x_t | x_{1:t-1}^{(i)}, z_{1:t}\right)$ and $x_{1:t} \stackrel{\Delta}{=} \left(x_{0:k-1}^{(i)}, x_k^{(i)}\right)$
- For $i = 1, 2, \ldots, n$, evaluate the importance weights up to a normalizing constant:

$$w_t^{(i)} = w_{t-1}^{(i)} \frac{p\left(z_t | x_t^{(i)}\right) p\left(x_t^{(i)} | x_{t-1}^{(i)}\right)}{\pi\left(x_t^{(i)} | x_{1:t-1}^{(i)}, z_{1:t-1}\right)}$$

- For $i = 1, 2, \ldots, n$, normalize the importance weights

$$\tilde{w_t}^{(i)} = \frac{w_t^{(i)}}{\sum_{j=1}^{n} w_t^{(i)}}$$

- Resampling:
  - For $i = 1, 2, \ldots, n$, sample indices $k(i)$ distributed according to the discrete distribution $p\left(k(i) = l\right) = \tilde{w_t}^{(l)}$ for $l = 1, \ldots, n$
  - For $i = 1, 2, \ldots, n$, $w_t^{(i)} = \frac{1}{n}$

to compute a zero weight for all these samples. In addition, the estimated posterior is also incorrect since it effectively contains only one sample. By resampling, we populate the sample set with highly probable samples at each step so that the final posterior is a good approximation. Resampling can also be viewed as taking advantage of the property of locality of high probability regions discussed in Section 1.5, as all the samples are made to focus on precisely these regions.

3.9 Rao-Blackwellized Particle Filters[2]

Recall from Section 2.2 that the computation of odometry likelihood requires the location of landmarks. Since the addition of a new measurement may result in a new landmark in the topology, these also have to be computed incrementally. In a particle filter, this requires the computation of the joint posterior on topologies and landmark locations which is a hybrid discrete-continuous space. Computation of a distribution on such a hybrid space is done using Rao-Blackwellized Particle Filters, that are explained in this section.

In a Rao-Blackwellized particle filter (RBPF), part of the state is treated analytically [70] in order to improve the accuracy of the filter. A particle filter is expected to do badly in high-dimensional state spaces because it relies on importance sampling as its main approximate inference algorithm. However, if some part of the state can be treated analytically, the dimensionality of the sampled part is reduced. This is possible due to the reduced variance of the Monte Carlo approximation, a consequence of the Rao-Blackwell theorem [10].

The basis for the RBPF is a different Monte Carlo approximation for the posterior $P(x_t|z_{1:t})$. Assume the state x_t is partitioned into l_t and a_t, where l_t is the discrete part represented using samples. The posterior $P(x_t|z_{1:t}) = P(l_t, a_t|z_{1:t})$ over the state x_t is approximated by a set of *hybrid particles*

$$S_t = \left\{ l_t^{(j)}, w_t^{(j)}, \alpha_t^{(j)}(a_t) \right\}_{j=1}^{n}$$

[2] ***This section has been adapted from an unpublished note by Frank Dellaert with the same title.***

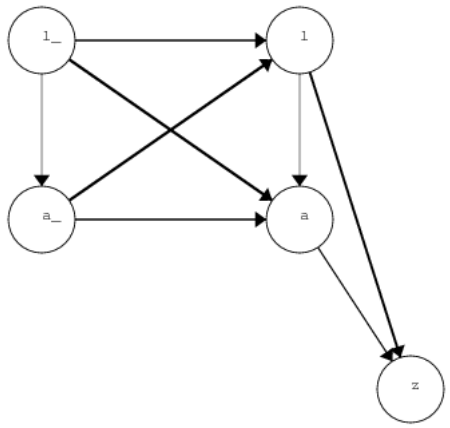


Figure 45: Dynamic Bayes Network for a general RB-filter, where the variables l will be approximated using a sample, but the belief over the variables a will be represented analytically.

each with its own conditional distribution $\alpha_t^{(j)}(a_t)$ over a_t [70]. This yields the following approximation to the posterior $P(l_t, a_t|z_{1:t})$:

$$p(l_t, a_t|z_{1:t}) \approx \left\{ w_t^{(j)} \delta(l_t, l_t^{(j)}) \alpha_t^{(j)}(a_t) \right\} \tag{33}$$

Formally, $\alpha_t^{(j)}(a_t)$ is defined as the distribution on a_t conditioned on particle j and the measurements $z_{1:t}$:

$$\alpha_t^{(j)}(a_t) \stackrel{\Delta}{=} P(a_t|l_t^{(j)}, z_{1:t}) \tag{34}$$

We assume an independence and causality relation hip as given by the Dynamic Bayes Network (DBN) in Figure 45, which implies the following factorization:

$$p(l_{t-1}, a_{t-1}, l_t, a_t, z_t|z_{1:t-1}) = p(z_t|l_t, a_t) p(a_t|l_{t-1}, a_{t-1}, l_t) p(l_t|l_{t-1}, a_{t-1}) p(a_{t-1}|l_{t-1}, z_{1:t-1}) p(l_{t-1}|z_{1:t-1})$$

Note that the above DBN is completely general: the only assumptions are the usual Markov assumptions.

The basic scheme for the RBPF is identical to that of the particle filter. Limiting our attention to l_t, the Bayes filter that recursively computes $p(l_t|z_{1:t})$ is given by:

$$p(l_t|z_{1:t}) = k_t p(z_t|l_t, z_{1:t-1}) \int_{l_{t-1}} p(l_t|l_{t-1}, z_{1:t-1}) p(l_{t-1}|z_{1:t-1}) \tag{35}$$

Algorithm 7 The Rao-Blackwellized Particle Filtering Algorithm

For times $t = 1, 2, \ldots$ do

- Choose the sample (mixture component) index $i \sim w_{t-1}^{(i)}$
- Sample from the chosen motion model for l (mixture component i):

$$l_t^{(j)} \sim p(l_t | l_{t-1}^{(i)}, z_{1:t-1})$$

- Calculate the importance weight:

$$w_t^{(j)} = p(z_t | l_t^{(j)}, z_{1:t-1}) \tag{37}$$

- Normalize the importance weights

$$\tilde{w_t}^{(i)} = \frac{w_t^{(i)}}{\sum_{j=1}^{n} w_t^{(i)}}$$

- Resampling:
  - Sample indices $k(i)$ distributed according to the discrete distribution $p(k(i) = l) = \tilde{w_t}^{(l)}$ for $l = 1, \ldots, n$
  - $w_t^{(i)} = \frac{1}{n}$

Since l_t is not the entire state, we cannot make the regular independence assumptions, i.e. $p(z_t | l_t, z_{1:t-1}) \neq p(z_t | l_t)$ and $p(l_t | l_{t-1}, z_{1:t-1}) \neq p(l_t | l_{t-1})$. This is because both the motion model and the measurement model may depend on the hidden part a of the state.

Substituting the Monte Carlo approximation consisting of weighted samples for the marginal $p(l_{t-1} | z_{1:t-1})$ above, we obtain an approximate Bayes filter on l:

$$p(l_t | z_{1:t}) \approx \hat{p}(l_t | z_{1:t}) \stackrel{\Delta}{=} k_t p(z_t | l_t, z_{1:t-1}) \sum_i w_{t-1}^{(i)} p(l_t | l_{t-1}^{(i)}, z_{1:t-1}) \tag{36}$$

The RBPF does importance sampling in the usual way, using the empirical predictive density $\hat{p}(l_t | z_{1:t-1}) \stackrel{\Delta}{=} \sum_i w_{t-1}^{(i)} p(l_t | l_{t-1}^{(i)}, z_{1:t-1})$ as the proposal density $Q(x_t)$. The RBPF algorithm is summarized in Algorithm 7.

Symbol	*Meaning*
n	Total number of landmarks observed
m	Number of distinct landmarks observed
o^n	The $n-1$ odometry measurements
s^n	Range scan measurements around the n landmarks
a^n	Appearance measurements form the n landmarks
z^n	Combined set of measurements $z^n = \left\{a^n, s^n, o^{n-1}\right\}$
L^n	Topology T^n represented as a label sequence
X^n	Landmark locations for the topology L^n
$\alpha_n(X^n)$	Analytic distribution on the landmark locations

Table 1: Notation used in the explanation of the algorithm

3.10 Topological Mapping using Rao-Blackwellized Particle Filters

We now describe the algorithm to construct Probabilistic Topological Maps (PTMs) using RBPFs. An intuitive description of the algorithm is possible using the equivalence of topologies with label sequences. To reiterate from Section 1.1, if we associate a label with each landmark, we can also represent the topology T by a label sequence $L_n = L_{1:n}$, where L_i is the label of the ith landmark. Further the number of unique labels in this sequence is equal to the number of sets in the set partition corresponding to the topology T, i.e. m. The posterior on topologies that we seek can then be written as $P(L_n|Z)$. A summary of all the notation used in the exposition of the algorithm is given in Table 1.

We use laser range scanners in the mapping framework to provide measurements in addition to odometry and appearance as before. Range scans are used to construct local map patches around landmark locations that the robot visits. These map patches are subsequently matched using scan matching techniques to provide a likelihood of their being from the same physical location. This gives us a sensor model for laser scans obtained at the landmark locations.

The posterior on topologies that we seek is represented as $p(L^n|a^n, s^n, o^n)$. Applying Bayes law on the required posterior to obtain the measurement likelihood and prior, we get

$$p(L^n|a^n, s^n, o^n) \quad \propto \quad p(L^n|z^{n-1}) p(a_n, s_n, o_n|L^n, z^{n-1}) \tag{38}$$

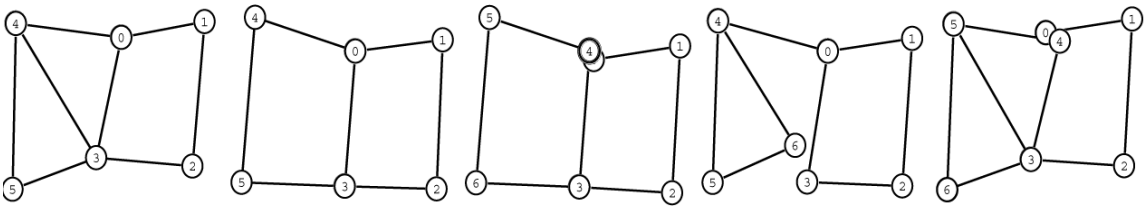


Figure 46: Example of a set of samples from the space of topologies for an environment. Each sample is associated with a weight in the particle filter.

where the measurements up to the $(n-1)$th landmark have been represented as $z^{n-1} = \left\{a^{n-1}, s^{n-1}, o^{n-1}\right\}$ and the likelihood of the measurements from the nth observed landmark is $p(a_n, s_n, o_n | L^n, z^{n-1})$. The prior $p(L^n|z^{n-1})$ can be further factorized to give an incremental prior on the label in the current (nth) time step -

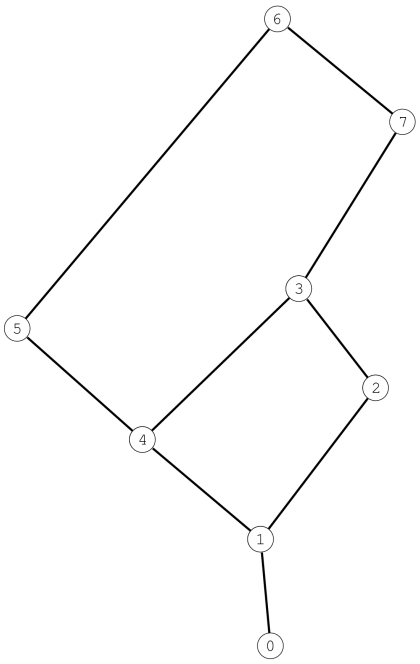
$$p(L^n|z^{n-1}) \quad = \quad p(L_n|L^{n-1}, z^{n-1}) p(L^{n-1}|z^{n-1}) \tag{39}$$

where $p(L_n|L^{n-1}, z^{n-1})$ is the prior (proposal) distribution for the label on the nth observed landmark and $p(L^{n-1}|z^{n-1})$ is the posterior from the previous step containing $n-1$ measurements. The prior gives a distribution on which of the distinct landmarks we are likely to see next, including the possibility of the next landmark being a previously unvisited one. It can be seen that equations (38) and (39) together give a recursive formulation for the posterior on topologies that is amenable for performing particle filtering. An illustration of a set of samples from the particle filter is given in Figure 46.

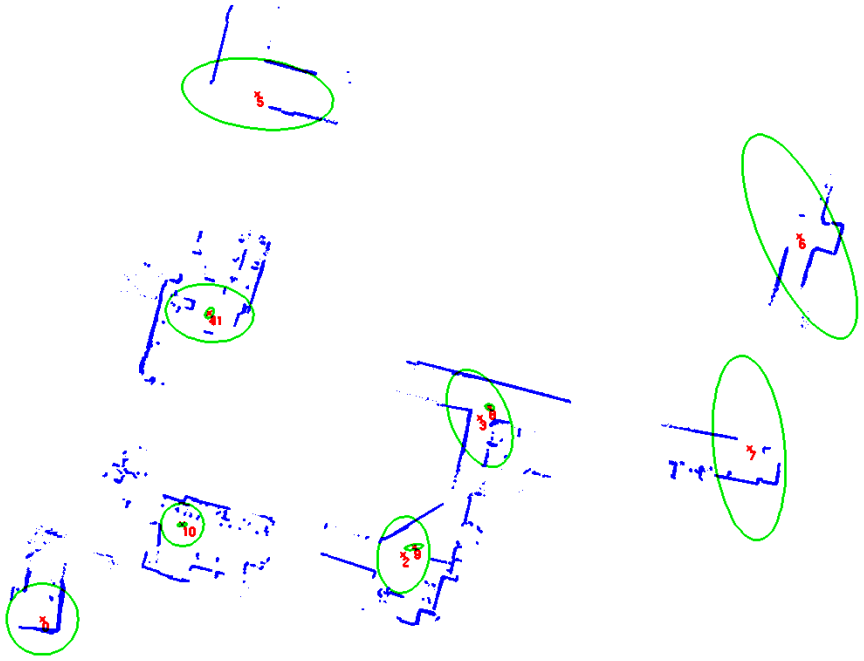
Simultaneously, the conditional posterior on landmark locations given the topology is also recursively computed

$$p(X^n|z^n, L^n) \quad \propto \quad p(s_n, o_n|X^n, L^n) p(X_n|X_{n-1}, z^{n-1}, L^n) p(X^{n-1}|z^{n-1}, L^{n-1}) \tag{40}$$

where Bayes law has been used and also the assumption that the landmark locations do not depend on the appearance measurements since appearance does not provide metric information. $p(X_n|X_{n-1}, z^{n-1})$ is a prior on the nth landmark and $p(X^{n-1}|z^{n-1}, L^{n-1})$ is the posterior from the previous step containing $n-1$ measurements.



(a)



(b)

Figure 47: A sample from the RBPF that contains (a) a topology and (b) an analytical distribution on the landmark locations in the form of a Gaussian. The red points in (b) are the mean landmark locations while the green ellipses denote marginal covariances.

As algorithm is based on an RBPF, we represent the posterior by a set of hybrid weighted particles containing a topology sample and an analytical function for the landmark location posterior (40).

$$S_n \quad = \quad \left\{ L^{n,(i)}, w_n^{(i)}, \alpha_n \left(X^n \right) \right\}_{i=1}^{N} \tag{41}$$

where $w_n^{(i)}$ is the weight on the ith particle and $\alpha_n^{(i)}(X^n) \stackrel{\Delta}{=} p(X^n | L^{n,(i)}, s^n, o^n)$ is the analytic form of the landmark location posterior. An example of a joint sample from the RBPF is shown in Figure 47.

The two components required to perform filtering are the proposal distribution and a method for computing the importance weights. These are explained in the following sections.

3.10.1 The Proposal Distribution

We use the predictive prior distribution on the current landmark label $p(L_n | L^{n-1}, z^{n-1})$, given in (39), as our proposal distribution. Using the sample notation of (41), the proposal distribution can be written as

$$L_n^{(i)} \quad \sim \quad p\left(L_n | L^{n-1,(i)}, z^{n-1} \right) \tag{42}$$

This is a discrete probability distribution on a vector of size $p+1$, where p is the number of distinct landmarks observed up to the $(n-1)$th step. Though any of the priors defined in Section 1.3 can be use for this purpose, we will assume the use of the Dirichlet Process prior for ease of exposition as this results in elegant expressions for various quantities of interest.

3.10.2 Importance Weight Computation

Using the definition of the importance sampling weights in the case where the proposal distribution is the same as the prior, we see from (32) that the expression for the importance

weights is the same as the measurement likelihood

$$w_n^{(i)} = \frac{\text{Target distribution}}{\text{Proposal distribution}} w_{n-1}^{(i)} \tag{43}$$

$$\propto p(a_n, s_n, o_{n-1} | L^{n,(i)}, z^{n-1}) w_{n-1}^{(i)} \tag{44}$$

where we have used the target distribution from (1) and proposal from (39).

The appearance measurement is conditionally independent of the scan and odometry measurement given the topology so that measurement likelihood can be written as

$$p(a_n, s_n, o_{n-1} | L^{n,(i)}, z^{n-1}) = p(a_n | L^{n,(i)}, z^{n-1}) p(s_n, o_{n-1} | L^{n,(i)}, z^{n-1}) \tag{45}$$

Evaluation of the appearance likelihood is described first followed by the scan and odometry likelihoods.

3.10.3 Appearance Likelihood Evaluation

The appearance likelihood is conditionally independent of scan and odometry measurements. Further, the nth measurement depends only on the label of the landmark observed at $t = n$. Hence, the expression for the appearance likelihood can be simplified to $p(a_n | L_n, a^{n-1})$ where we have dropped the sample index for simplicity.

The likelihood is evaluated by marginalizing over the "true appearance" parameter corresponding to the physical landmark denoted by the label L_n, as was explained in Section 2.5. Two situations now arise. If the label L_n corresponds to a previously unseen landmark, the prior distribution on true appearance is taken to be the Dirichlet Process prior function. On the other hand, if L_n denotes a landmark that is being revisited, all the measurements from the previous visits to the landmark are used to estimate the prior.

Formally, let the number of distinct landmarks at time $t = n-1$ be r. Then the appearance likelihood is evaluated as

$$p(a_n | L_n, a^{n-1}) = \begin{cases} \int_y p(a_n | y_n) G_0(y_n) & \text{if } L_n = r+1 \\ \int_y p(a_n | y_n) p(y_n | a_{L_n}^{n-1}) & \text{otherwise} \end{cases} \tag{46}$$

where G_0 is the prior measure on the Dirichlet Process prior as in (9) in Section 1.3.2, and $a_{L_n}^{n-1}$ is the set of appearance measurements corresponding to landmark label L_n.

The distributions involved are all Gaussian, as in Section 2.5, and hence (46) can be evaluated in closed form.

3.10.4 Odometry and Laser Scan Likelihood Evaluation

Odometry and scan likelihoods are evaluated by introducing the landmark locations and marginalizing over them. This is necessary since these measurements are metric in nature and cannot be evaluated without knowledge of the landmark locations. Upon performing the marginalization, we obtain (using the notation of (41))

$$\begin{aligned} p(s_n, o_n | L^{n,(i)}, z^{n-1}) = \quad & \int_{X^n} p(s_n, o_n | L^{n,(i)}, X^n) \\ & p(X^n | L^{n,(i)}, z^{n-1}) \end{aligned} \tag{47}$$

where X^n is the vector of landmark locations of length n and we have used the chain rule in the integrand. The prior on landmark locations $p(X^n | L^{n,(i)}, z^{n-1})$ can be further factorized into a predictive prior on the location of the current (nth) landmark and the posterior on locations from the previous step

$$\begin{aligned} p(X^n | L^{n,(i)}, z^{n-1}) = \quad & p(X_n | L^{n,(i)}, X^{n-1}, z^{n-1}) \\ & p(X^{n-1} | L^{n-1,(i)}, z^{n-1}) \end{aligned} \tag{48}$$

where X_n is the location of the nth landmark and $p(X^{n-1} | L^{n-1,(i)}, z^{n-1})$ is the posterior on landmark locations from the previous step. Notice that combining (47) and (48) gives us the posterior on landmark location (40) up to a normalization constant. Hence, evaluating the odometry and scan likelihoods simply involves integrating the unnormalized posterior on landmark locations.

To compute the posterior over landmark locations, integrating which we obtain the importance weights, first consider the measurement likelihood given the landmark locations

Figure 48: Scan measurements, obtained by concatenating scans from around landmark locations, used by the RBPF algorithm.

$p(s_n, o_n | L^{n,(i)}, X^n, z^{n-1})$. Assuming the independence of the scan and odometry measurements given the landmark measurements, we obtain

$$\begin{aligned} p(s_n, o_n | L^{n,(i)}, X^n, z^{n-1}) = \quad & p(s_n | L^{n,(i)}, X^n, z^{n-1}) \\ & p(o_n | L^{n,(i)}, X^n, z^{n-1}) \end{aligned} \tag{49}$$

The scan likelihood $p(s_n | L^{n,(i)}, X^n, z^{n-1})$ is obtained by performing scan matching between the map patches from the landmark locations. The map patches are obtained, in turn, by simply concatenating laser scans from a local area around the landmark as the robot moves through it (Figure 48). We use the scheme of Chen and Medioni [12], which involves point-to-plane matching, to perform scan matching. The odometry likelihood $p(o_n | L^{n,(i)}, X^n, z^{n-1})$ is evaluated simply through the use of an odometry model.

The prior on the landmark location $p(X_n | L^{n,(i)}, X^{n-1}, z^{n-1})$ encodes the notion that distinct landmarks do not usually occur close together in the environment. We use the same

Algorithm 8 The RBPF algorithm for inferring PTMs

1. Randomly select a particle $L^{n-1,(i)}$ from the previous time step according to the weights $w_{n-1}^{(i)}$.
2. Propose a new topology sample using the proposal distribution $p\left(L_n^{(j)}|L^{n-1,(i)}\right)$ in (42)
3. Calculate the Gaussian posterior density on landmark locations $\alpha_n^{(j)}(X^n)$ using Bayes law as in (40) and the Laplace approximation.
4. Calculate the importance weights $w_n^{(j)}$ from (45). The appearance likelihood is calculated using (46), and the odometry and scan likelihoods as the integral over the unnormalized $\alpha_n^{(j)}(X^n)$ in (47).

prior on landmark locations as given in Section 2.3. Topologies which place distinct landmarks close together in location are penalized by this prior.

As the odometry model is assumed to be Gaussian and the result of the scan matching operation is also a Gaussian distribution, all the distributions involved in the computation of the landmark location posterior (40) are Gaussian except for the landmark prior. This makes the posterior non-Gaussian.

The computation is kept recursive by projecting the non-Gaussian posterior onto a Gaussian posterior using the Laplace approximation. This involves replacing the true posterior by a Gaussian centered around the *maximum aposteriori* value of landmark locations. In practice, this step is performed by linearizing around the most likely landmark location, which is found through an optimization as in Section 2.4. The covariance at the MAP location is estimated through the Hessian matrix obtained from the optimization algorithm. Details of the Laplace approximation are given in Appendix B.

The weight computation of (47) can now be performed in closed form as it involves integrating a Gaussian distribution, albeit unnormalized.

We now have all the components to perform the inference using the RBPF. A summary of the algorithm is provided in Algorithm 8.

3.11 Data Driven Proposals for Particle Filters

The problem of proposing topologies with low probability, which led to slow convergence in the case of the MCMC algorithm, manifests itself in a particle filter as the "particle degeneracy" or "lack of diversity" problem [19], wherein only one sample has a significantly non-zero weight. The reason for this failure is that many samples fall into regions of low probability and die out during the filtering process. This results not only in the failure to converge to the correct posterior but also in wasted computation, since the algorithm is evaluating the weights of samples that will be lost in any case.

A data-driven proposal overcomes this problem by proposing more samples from regions of high probability so that samples and computation are not wasted. Note that the proposal distribution in (42) does not make use of the current measurement. We rectify the situation in this section by presenting a proposal distribution that uses the odometry to provide more likely samples.

The key idea behind the data-driven proposal is that the odometry likelihood can be incorporated into the proposal distribution while only the appearance and scan likelihoods are used to compute the importance weights. The measurement likelihood in (38) is thus split into two parts. This split also entails a two-step process for updating the analytic posterior on landmark locations since this posterior needs to be updated using both the odometry and scan measurements.

In the following exposition, we do not consider appearance measurements since the appearance likelihood is evaluated in exactly the same manner and is unaffected by the data-driven proposal.

Starting with the posterior on topologies, we obtain using Bayes Law the likelihood and prior

$$\begin{aligned} p(L^n|s^n,o^n) &\propto p(L^n|z^{n-1})p(s_n,o_n|L^n,z^{n-1}) \\ &= p(L^n|z^{n-1})p(o_n|L^n,z^{n-1})p(s_n|L^n,z^{n-1},o_n) \end{aligned} \tag{50}$$

where the likelihood is factored into two terms using the chain rule. The prior term can in turn be written using Bayes law as the product of the odometry likelihood and a prior on the current label.

$$p(L^n|z^{n-1}, o_{n-1}) \quad \propto \quad p(L_n|L^{n-1}, z^{n-1})p(L^{n-1}|z^{n-1})p(o_{n-1}|L^n, z^{n-1}) \tag{51}$$

The proposal distribution is taken to be the right hand side of (51), which can be written using the sample representation of (41) as

$$L_n^{(i)} \quad \sim \quad p\left(L_n|L^{n-1,(i)}, z^{n-1}\right) p(o_n|L^{n,(i)}, z^{n-1}) \tag{52}$$

The form of the predictive label distribution $p(L^n|z^{n-1})$ is the Dirichlet process prior as before. However, the odometry likelihood in (51) is evaluated by marginalization over the landmark locations

$$p(o_n|L^n, z^{n-1}) \quad = \quad \int_{X^n} p(o_n|L^{n,(i)}, X^n, z^{n-1})p(X^n|L^{n,(i)}, z^{n-1}) \tag{53}$$

where the same landmark prior and odometry model are used as in Section 3.10.4. Note that the prior in (53) can be evaluated using the posterior on the landmark locations from the previous by use of the chain rule as in (48).

One drawback of this proposal distribution is the need to perform m optimizations to compute it. These optimizations are required since the integral in (53), evaluated by linearizing around the optimum, needs to be computed for all the possible label values for L_n (except for the case when L_n is a new landmark), which are m in total. However, performing these extra optimizations once per filtering step is still preferable to evaluating the importance weight for all the particles that do not survive when a vanilla proposal is used.

From the target (50) and proposal (51) distributions and the definition of the importance weights (43), we get the expression for the importance weights in this case as

$$w_n^{(i)} \quad \propto \quad p(s_n|L^{n,(i)}, z^{n-1}, o_n)w_{n-1}^{(i-1)}$$

This is evaluated by marginalization over landmark locations

$$p(s_n|L^{n,(i)}, z^{n-1}, o_n) \quad \propto \quad \int_{X^n} p(s_n|L^{n,(i)}, X^n, z^{n-1}, o_n)p(X^n|L^{n,(i)}, z^{n-1}, o_n)$$

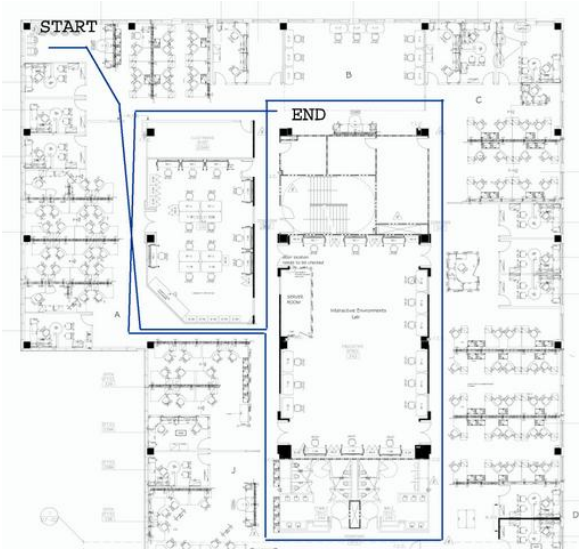


Figure 49: Schematic of robot path overlaid on a floorplan of the environment for the first experiment.

where the scan likelihood is evaluated using scan matching exactly as in Section 3.10.4, since it is independent of the odometry given the landmark locations. The location prior $p(X^n|L^{n,(i)}, z^{n-1}, o_{n-1})$ is the same as the integrand of (53) up to a normalizing constant and the linearized Gaussian approximation found therein is used again here.

3.12 Results

The same datasets used in the experiments in Section 3.6 were used to validate the particle filtering algorithm. Particle filtering was performed using 50 samples and the data-driven proposal was used in all the experiments. A value of 3.0 was used for the Dirichlet prior parameter α. The landmark location prior was used with a value of 10 meters for the penalty radius and 15 for the maximum penalty value. For a description of these parameters and their effect on the inferred posterior, see Section 2.6.

The first experiment was conducted in the TSRB building using an ATRV-mini robot in an indoor setting. A map of the experiment area along with the robot path, which is approximately 100 meters long and passes through twelve landmark locations, is shown in Figure 49. The map patches obtained by concatenating scans around the landmark locations

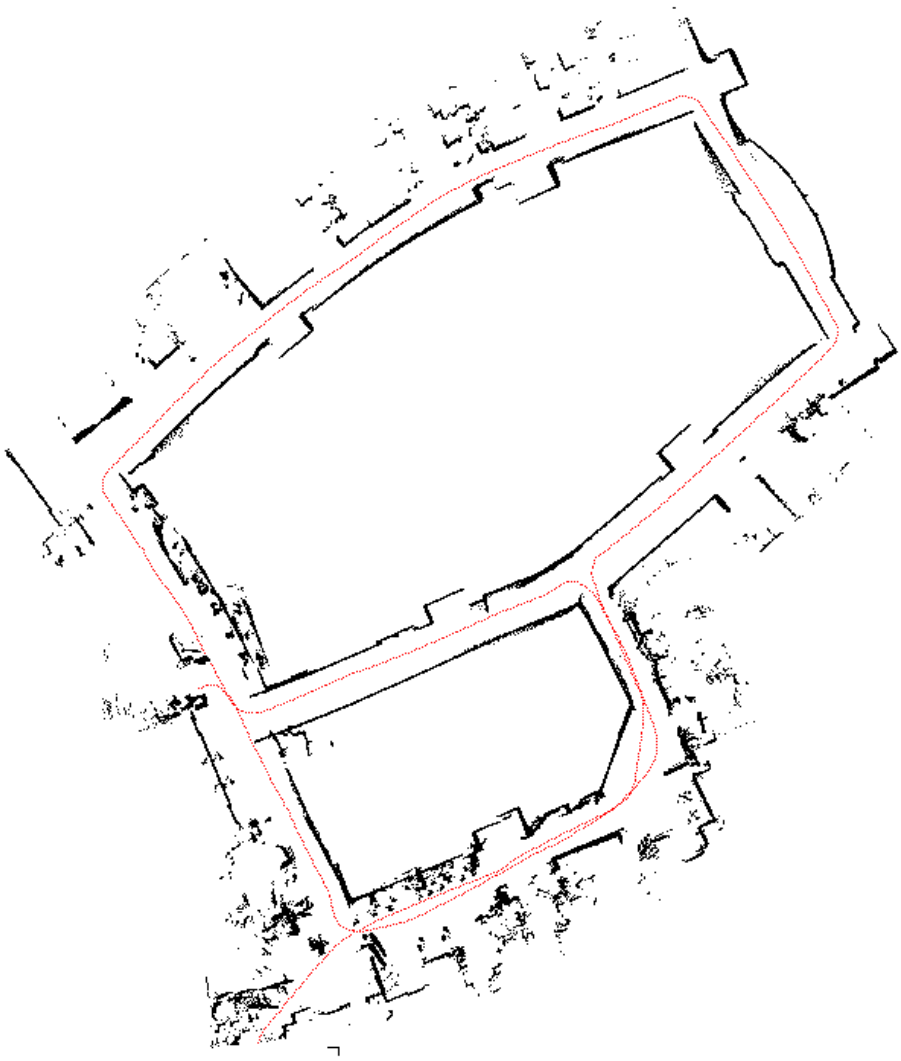


Figure 50: Global metric map obtained using topological constraints and landmark locations for the first experiment. The robot path is in red.

are shown in Figure 48.

The result of the filtering using the RBPF algorithm is a joint distribution on topologies and landmark locations. The maximum likelihood sample is shown in Figure 47. The distribution on the landmark locations is displayed in the figure through the marginal covariance ellipses along with the local map patches aligned using scan matching. The corresponding topology, shown in Figure 47(a), is also the ground truth topology and obtains 94% of the probability mass in the posterior. The topology constraints and the inferred landmark locations can be used to produce a global metric map using the global optimization technique

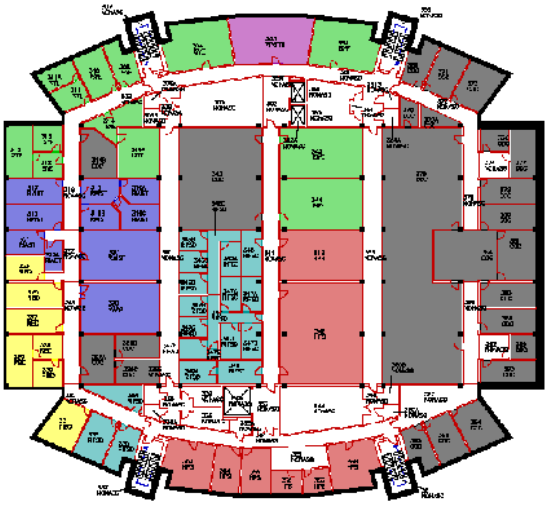


Figure 51: Floorplan of experimental area for second experiment.

of Lu and Milios [57]. The resultant metric map is given in Figure 50. It can be seen that this simple post-processing step produces a globally consistent metric map.

A second experiment was performed using the CRB dataset in a larger environment (about 60 meters across) to confirm our findings. A floorplan of the test area is shown in Figure 51. The RBPF algorithm computes the PTM that gives the ground-truth topology in Figure 53, 82% of the probability mass. The probability mass on the ground truth is lower in this case since there is perceptual aliasing around the corners of the building that scan matching is unable to resolve completely. The maximum-likelihood sample with the distribution on landmark locations is shown in Figure 54. The metric map obtained from the Lu-Milios step is given in Figure 52.

3.13 Tradeoffs in Particle Filtering vis-a-vis MCMC

From the above discussion, it may seem that the particle filtering algorithm has all the advantages of the MCMC algorithm in its functioning while not having its major disadvantage, i.e. the need for batch processing. However, as is usual in such cases, the incremental nature of particle filtering introduces certain inefficiencies.

The use of a poor proposal distribution in the MCMC algorithm leads in the worst

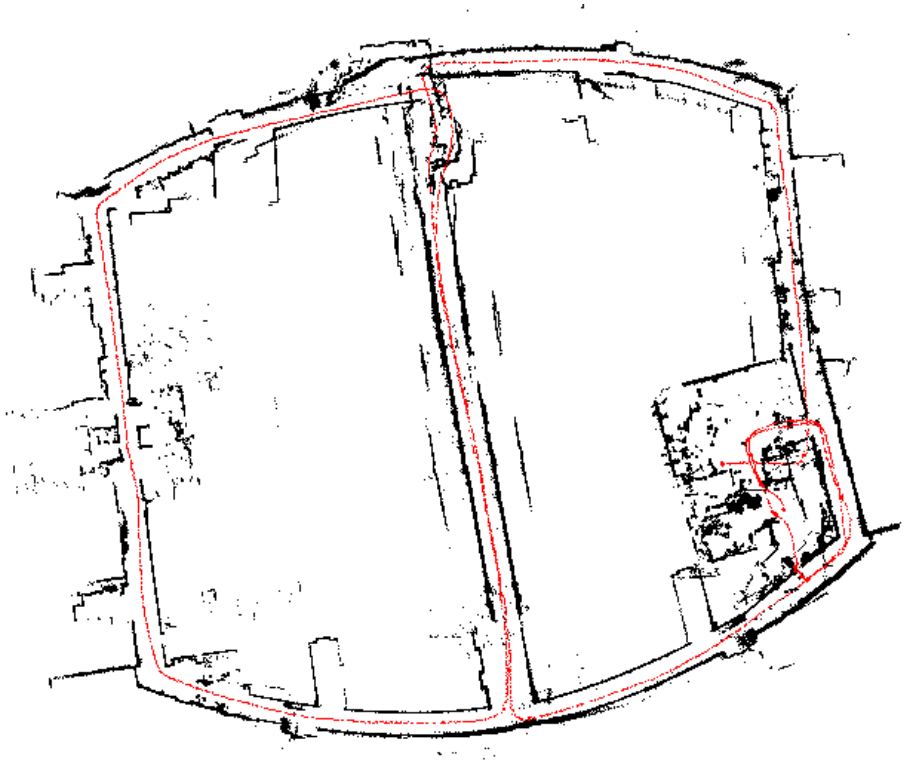


Figure 52: Metric map obtained using topological constraints for second experiment. The robot path is in red.

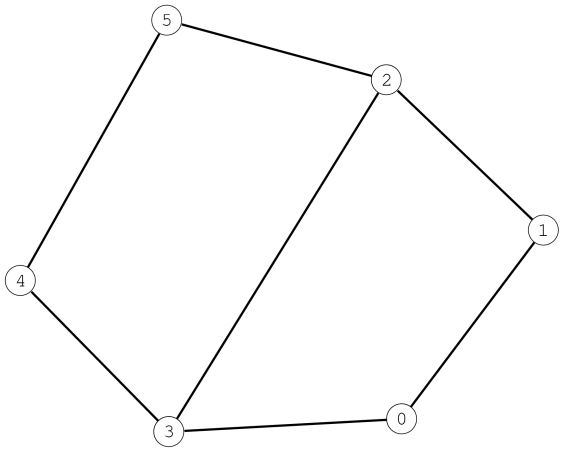


Figure 53: Ground truth topology for second experiment on the CRB dataset. This receives 89% of the probability mass in the PTM.

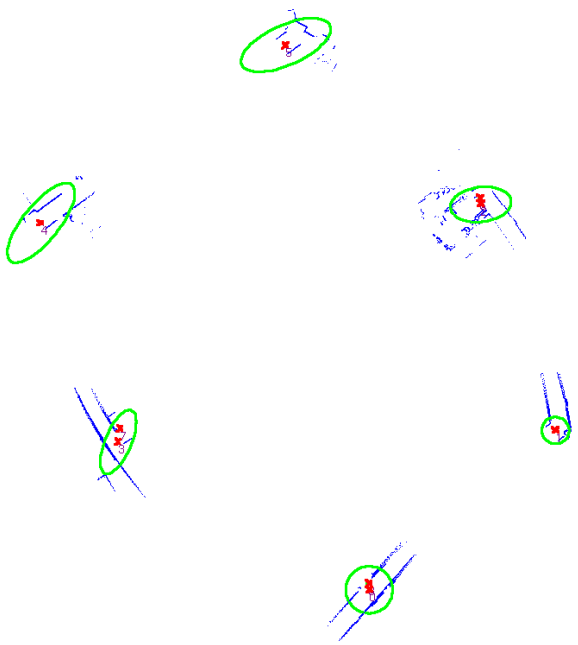


Figure 54: Maximum likelihood sample from the RBPF for second experiment. The red points are the mean landmark locations while the green ellipses denote marginal covariances.

case to an extremely slow converging sampler. In the particle filtering case, by contrast, it can lead to an incorrect result since the number of samples used is pre-determined and unchanged in the algorithm. Thus, the algorithm will terminate even if the PTM is incorrect, a condition that can be detected in MCMC using tests such as the SRQ plot described in Section 3.6. Also, in many instances, if the sample with the intermediate ground-truth topology is lost due to lack of diversity in the particle set, the algorithm cannot recover.

In general, it is best to start the particle filtering algorithm with a large number of samples, say in the hundreds, and subsequently, repeat the experiment with larger and smaller numbers of samples to confirm convergence. This is, however, a heuristic test at best since pathological cases where the algorithm produces a stable but incorrect estimate will go undetected.

In essence, the difference between the MCMC and particle filtering algorithms is in the availability of "future information" for any landmark in the topology. The MCMC algorithm can be opportunistic about selecting landmarks to merge or split so that easy

decisions that are highly probable can be made first regardless of the temporal ordering of the landmarks involved. For example, the first and last landmarks observed by the robot may be merged in the first proposal. These decisions may in turn lead to a bootstrap effect where other moves in the space of topologies become easier.

Particle filtering, on the other hand, makes the implicit assumption that the temporal ordering of landmarks provides also the best ordering for split/merge moves in topological space. The hope here is that the current set of probable topologies will turn out to be similar to the corresponding future set obtained by adding a single landmark. This is a reasonable assumption in almost all cases, especially in man-made environments. In cases where this assumption is violated, particle filtering performs poorly.

CHAPTER IV

INCORPORATING AUTOMATIC LANDMARK DETECTION

In evaluating the various PTM algorithms and their flavors, we have so far assumed the availability of hand-labeled landmarks. This translates to a perfect landmark detector that detects only decision points, i.e. corridor junctions, turns, entrances, etc. In practice, however, such a landmark detector algorithm does not exist. In this chapter, we discard the unreasonably strong assumption that is inherent in a perfect landmark detector, and evaluate the PTM algorithm with various practical landmark detectors. The introduction of landmark detection techniques also makes the PTM framework a complete topological mapping system.

Perfect landmark detection is not yet possible in the state of the art since the concept of landmark itself is vaguely defined. Human definitions of landmarks often include complex characteristics such as relative location of places, identity of contained objects and their locations, and other high-level descriptions that are far beyond the capabilities of any process that has only local, low-level measurements available to it. While a perfect algorithm is currently unachievable, algorithms that detect most landmarks in the environment while also containing false positives are quite feasible.

It is the aim of this chapter to demonstrate the working of the PTM algorithms on the spectrum of landmark detection techniques. This spectrum is illustrated in Figure 55, and consists of simple, "blind" landmark detectors that do not even consider measurements on one end. In contrast, the other end marks the perfect landmark detector as exemplified by the hand labeled landmarks of previous chapters. This could also be seen as a landmark detector that has perfect knowledge about semantic structure, relative importance of objects and structures in the environment, and so on. The intermediate region consists of algorithms

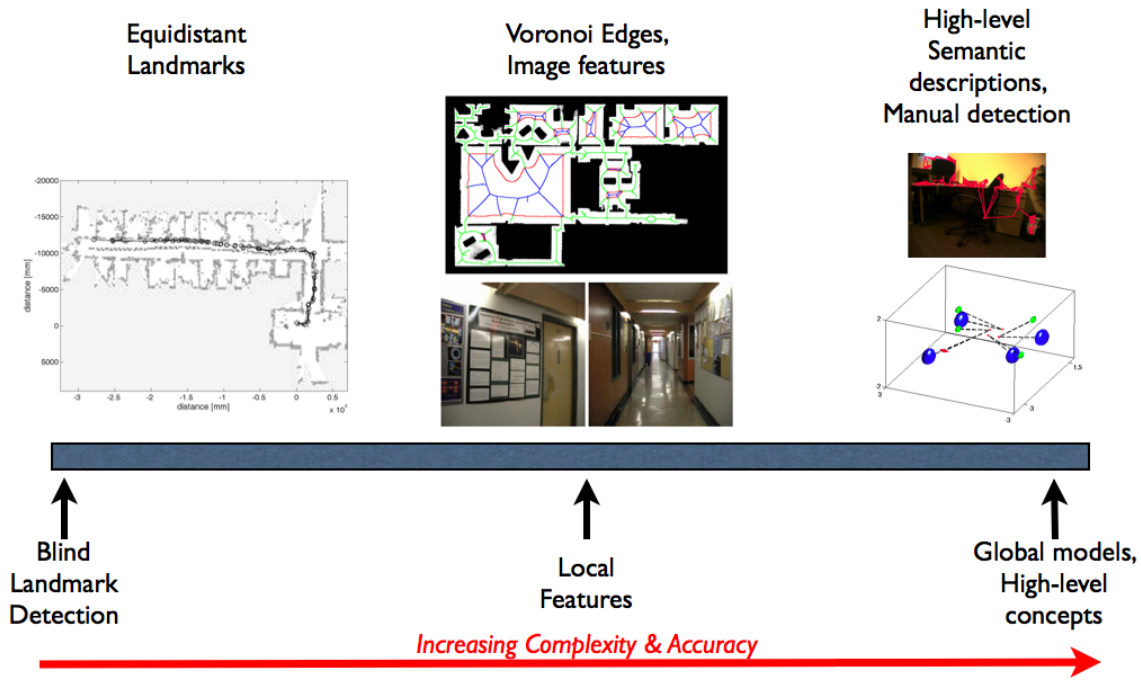


Figure 55: Spectrum of landmark detection techniques. The left end consists of techniques that do not even consider measurements from the environment, an example being placing landmarks at equidistant intervals. The right end consists of algorithms that have perfect, high-level descriptions of the environment including objects and other characteristics of interest. As we move from left to right, the complexity of the landmark detector increases while the number of false positives goes down.

that, to various extents, use local characteristics of the location derived from measurements.

A number of landmark detectors exist in the literature that take into account local information, and hence, populate the mid-portion of the landmark detection spectrum. While sophisticated techniques for using the local information exist, many techniques use the simple scheme of adding a node to the topology when the current measurement is significantly different from a temporally local previous measurement set [103][91]. This tends to create maps with densely packed nodes, thus destroying the sparsity of the graph. Hence, we would like a local landmark detector that detects all the points of interest while minimizing false positives.

We now provide a brief overview of prior work in landmark detection for topological mapping.

4.1 Prior Work in Landmark Detection

Simplistically, a landmark is simply a place that is strikingly distinct from the local surrounding environment, and this is exactly the definition that many mapping systems use in practice [52]. Another common definition of landmarks is using geometric nodal points in the environment such as intersection of Voronoi cells [13]. However, the use of such features may introduce a large number of landmarks in the map, thus destroying the sparse nature of the topological map. Beeson et. al. [6] have tried to overcome the problem of too many false positive landmarks by judiciously pruning the Voronoi graph so that spurious nodal points are not classified as landmarks. The Voronoi graph is computed from a local scrolling grid in this case.

Landmarks can be defined as "distinct" places using measures of distinctness that are sensor-specific, for instance Kortenkamp [45] uses range scans while Ramos et al. [75] use camera images. This leads to landmark detectors that use very specific features of the environment such as open doors and orthogonal walls, and moreover, are bound to a particular sensor [16]. Hence, most landmark detection methods avoid this route. Instead,

place modeling is used as an alternate means of landmark detection, the premise being that any place whose model is significantly different from models of already visited places is a landmark.

Kuipers and Beeson [51] present a bootstrap algorithm for place modeling and landmark detection that consists of two parts. First, images obtained from the robot are clustered using an unsupervised method (k-means) and a topology is learnt using techniques presented in [50] from this clustered image set. Subsequently, the topology is used to provide a data set for a better supervised learning (nearest neighbor) of the place models. The unsupervised learning thus bootstraps a better final place model. Also, the unsupervised algorithm increases aliasing through clustering while learning the topology removes this aliasing in the supervised learning phase.

Approaches for landmark detection and recognition based on features detected in images also abound. [84, 83] describes the use of SIFT features for this purpose, while [46] gives a method based on the combined use of SIFT features and image histograms to overcome illumination and viewpoint effects. Location fingerprints, described in [92], can be viewed as a sophisticated and general feature-based place modeling method as they incorporate a varied list of low-level features such as vertical edges, color patches from images, and corners from laser scans. The use of prominent image statistics instead of image features is also popular. For example, Principal Components Analysis (PCA) on omnidirectional images is advocated for place modeling by [47], while color histograms are proposed for this purpose by [102]. A more robust statistic is given in [21], which describes the use of PCA on sub-windows of an image using a Fourier basis for rotation invariance.

Recently, Ramos et. al. [75] have presented a Bayesian model for place recognition that is learnt in a supervised manner. The basic idea is to chop up training images from some special place into small patches and then reduce the dimensionality of these patches using the Isomap algorithm [93]. A mixture of Gaussians (with the appropriate conjugate hyperpriors) is fit to the reduced dimensionality patches (called essential features in the

paper) on the Isomap manifold. This is done using Variational Bayes EM [5] and the model thus learnt is the generative model of the place. During testing, the log likelihood of the patches in the testing image are computed for each model, and the maximum likelihood model is selected. This method, however, does not work well when featureless areas or occlusions are present.

4.2 Evaluating PTMs with Automatically Detected Landmarks

While landmark detection and topological ambiguity have been discussed independently thus far, there are, in fact, intimately connected. A powerful landmark detector that outputs few false detections, both false positive and negative, can reduce topological ambiguity significantly. This in turn makes the use of measurement streams having a low signal to noise ratio possible. Conversely, a poor landmark detector requires the use of strongly informative measurements to overcome ambiguity.

False negatives in landmark detection, i.e. true landmark locations where the detector does not fire, are much more serious than false positives. False positives most often only increase the number of nodes in the topology resulting in inefficient computation of the posterior over topological space. False negatives, however, destroy the topology of the environment since navigation using such a map is effectively impossible. Hence, a very simple but poor scheme such as placing landmarks at equi-distant intervals is preferable to a sophisticated scheme that may occasionally not detect a physical landmark.

Even if sophisticated statistics of the local scene are used, the fact remains that most landmark detection use fairly low level characteristics of the environment; characteristics that humans would not use to characterize the place of interest. People most often characterize spaces by high-level semantic descriptions involving objects, relative locations, and unique features. The characterization of landmarks at such a high semantic level is currently infeasible.

Results in the previous chapters have validated the performance of the PTM algorithms

with a perfect, manually constructed, landmark detector. We now provide details on incorporating landmark detectors from other regions of the landmark detection spectrum.

Firstly, the lower end of the spectrum is represented by the simplest possible landmark detection, i.e. one that places landmarks at equi-distant intervals. This is a "blind" scheme that does not consider measurements but detects most of the places of interest in the environment, though with a huge number of false positives.

We propose a new technique for landmark detection that incorporates the notion of surprise. Surprise quantifies the unexpectedness of the output, the premise being that unexpected places in the environment qualify as landmarks. We discuss the computation of surprise using Bayesian tools. This constitutes the mid-range in the landmark detection spectrum since it takes into account low-level, local characteristics of the location under consideration. A framework for computing surprise that is sensor-independent is presented and validated through application to laser range scanners and camera images respectively.

The use of weaker landmark detectors requires the availability of more informative measurements to offset the increased ambiguity. In particular, the appearance model presented in Section 2.5 using Fourier signatures of panoramic images provides relatively weak discriminative information that is insufficient when a large number of false positive landmarks are detected. Hence, we start by describing an appearance model that uses SIFT features detected in images, and provide a more discriminative measurement model for use with error-prone landmark detectors.

4.3 Appearance Modeling Using "Bag of Words" Models

Fourier signatures as appearance measurements are not sufficiently discriminative for the correct topology to be inferred across the spectrum of landmark detection algorithms. This is because Fourier signatures reduce a detailed panoramic image to merely a few numbers.

Recently, SIFT [56] descriptors of features have gained greatly in popularity due to their ability to convert many types of features into a standard, reproducible vector space.

Furthermore, SIFT descriptors of features detected on an image can be quantized and the quantized features can be considered to be the analogue of words in a document. Thus, the image becomes a sequence of "appearance words", making document analysis methods applicable to images. In keeping with the text processing community, methods that model images by converting SIFT features into appearance words are called "bag of words" models since they do not consider the sequence of the words themselves, but only their occurrence frequency.

We use images from the eight camera rig, as in Section 2.5, to model a place using features. Two types of features are detected on the images; the Harris Affine features by Mikolajczyk and Schmid [64], and the Maximally Stable Extremal Regions (MSER) by Matas et. al. [61]. The reason for two types of features is their complementary nature that ensures that both affine-invariant features and regions of intensity maxima are detected, thus ensuring a relatively dense representation of the images in feature space. All the features are subsequently transformed to a 128-dimensional vector space using SIFT descriptors.

Appearance words are obtained from the SIFT descriptors using vector quantization. The number of bins in the vector quantization, which corresponds to the number of words in a text document, is a parameter. Vector quantization is performed using the K-means algorithm, and is done as batch process over all the features detected across all the images. Each image is, subsequently, transformed into a histogram of word counts in each of the bins. Thus, the representation of an image in a bag-of-words model is a vector of word counts, which comprise a histogram.

4.4 The Multivariate Polya Model

The aim of using the SIFT histograms is to identify landmarks that are physically the same. This is done by clustering the histograms arising from the same landmark. However, since we have an appearance model that is conditioned on the topology, all that is needed is to

evaluate the clustering that is implied by the topology.

In keeping with the general appearance model described in Section 2.5, we model all the images arising from a landmark as having the same underlying "cause". Since the measurements are histograms of word counts, they are modeled using a multinomial distribution having its dimensions equal to the number of appearance words. Further the prior over the multinomial parameter is the conjugate Dirichlet distribution to aid in ease of computation

$$P(A|T) = \prod_{s\in T}\int_{\theta_s} P(\theta_s|\alpha_s)\prod_{a\in s}P(a|\theta_s) \tag{54}$$

where $\{a\}$ is the set of measurements indexed by set s, and $\theta_s = [\theta_{s1}, \theta_{s2}, \ldots, \theta_{sW}]$ and $\alpha_s = [\alpha_{s1}, \alpha_{s2}, \ldots, \alpha_{sW}]$ are the multinomial parameter and Dirichlet prior respectively. The number of distinct appearance words is denoted as W. Hence the distributions in the integrand above are

$$p(a|\theta_s) = \frac{n!}{n_1!n_2!\ldots n_W!}\theta_{s1}^{n_1}\theta_{s2}^{n_2}\ldots\theta_{sW}^{n_W} \tag{55}$$

$$p(\theta_s|\alpha_s) = \frac{\Gamma(\sum_{w=1}^{W}\alpha_{sw})}{\sum_{w=1}^{W}\Gamma(\alpha_{sw})}\theta_{s1}^{\alpha_{s1}-1}\theta_{s2}^{\alpha_{s2}-1}\ldots\theta_{sW}^{\alpha_{sW}-1} \tag{56}$$

The likelihood model in (54) with the multinomial and Dirichlet distributions defined in (55) and (56) is called the Multivariate Polya model, or equivalently in document modeling, the Dirichlet Compound Multinomial model [2]. The Multivariate Polya model can be shown to be a finite bin version of the Polya Urn model defined in (74). It models burstiness in the data, i.e. the empirical observation that if a word occurs once in a document, it is likely to occur many more times.

The integration in (54) can be performed in closed form since the Dirichlet process is the conjugate prior of the multinomial distribution. This yields the final form of the appearance likelihood as

$$P(A|T) = \prod_{s\in T}\frac{n_s!}{\prod_{w=1}^{W}n_{sw}}\frac{\Gamma(\alpha_s)}{\Gamma(n_s+\alpha_s)}\prod_{w=1}^{W}\frac{\Gamma(n_{sw}+\alpha_{sw})}{\Gamma(\alpha_{sw})} \tag{57}$$

where n_{sw} is the count of the wth appearance word across all the SIFT histograms in set s and $n_s = \sum_w n_{sw}$, $\alpha_s = \sum_w \alpha_{sw}$.

Given a collection of D images with features detected on them, the maximum likelihood value for α can be learned by using iterative gradient descent optimization. It can be shown that this leads to the following fixed point update [65]

$$\alpha_w^{new} = \alpha_w \frac{\sum_{d=1}^{D} \psi(n_{dw} + \alpha_w) - \psi(\alpha_w)}{\sum_{d=1}^{D} \psi(n_{dw} + \alpha) - \psi(\alpha)} \tag{58}$$

where $\alpha = \sum_w \alpha_w$ as before.

The appearance likelihood is evaluated by learning the α parameter for each set in the topology and using these values appropriately in (57). This appearance likelihood can be used in both the MCMC and Particle Filtering algorithms as explained in Chapters 2 and 3.

4.5 Landmarks at Equi-distant Intervals

We now present experiments involving landmarks placed at equi-distant intervals. This constitutes the simplest possible landmark detector as it does not take it into account any measurements.

The first experiment showcases the TSRB dataset. The SIFT-based appearance model described in Section 4.3 above was used to model images obtained from the camera rig illustrated in Figure 24. SIFT features detected from the landmark images were quantized into 1024 appearance words. A landmark was placed at every tenth image, i.e. approximately equally spaced in time. Hence, landmarks are closer spatially when the robot moves slowly and vice versa. A total of 35 landmarks were obtained using this scheme. The incremental particle filtering algorithm with the data-driven proposal was used to obtain the PTM. The data-driven proposal used is based on the appearance measurements and is described in Appendix C.

The PTM at various stages of the incremental inference is illustrated in Figure 56. The final PTM contains only one topology, which is also the ground-truth. The covariances marking the uncertainty in landmark positions are also shown for the final topology.

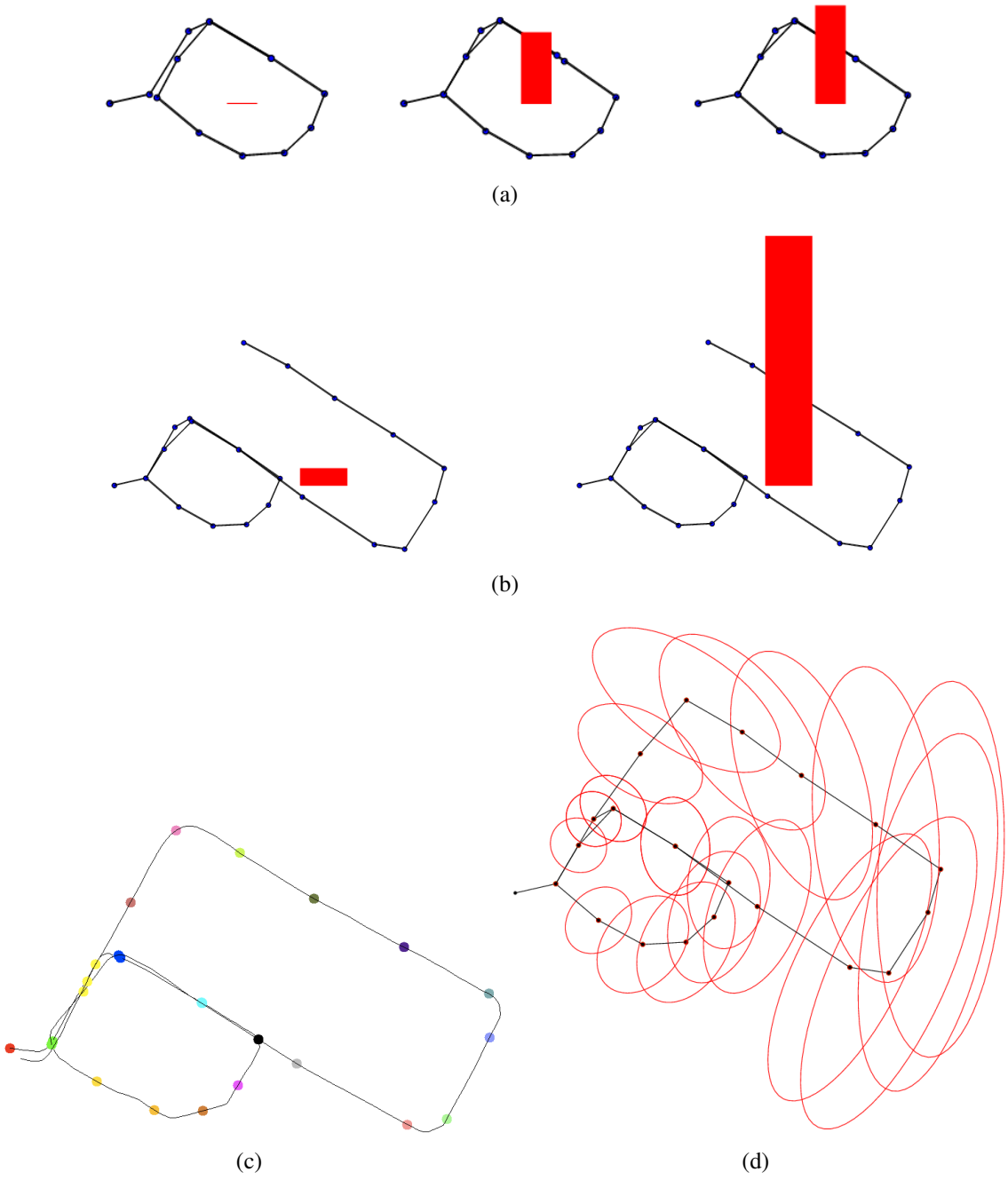


Figure 56: PTMs for the TSRB dataset with 35 landmarks placed equally in time computed using the incremental particle filtering algorithm (a) PTM after 13 landmarks (b) After 29 landmarks (c) Final PTM after 35 landmarks, which is also the groundtruth, with smoothed trajectory. Landmarks are shown as colored circles with nodes corresponding to the same physical landmark colored similarly (d) 5σ covariance ellipses for the landmark locations of the ground-truth topology.

The second experiment was performed using the well-known Intel dataset widely used the SLAM literature [34]. The dataset consists of odometry and laser measurements, and the laser measurement model as described in Section 3.10.4 was used to obtain the result. The MC-cubed variant of the MCMC algorithm, along with the data-driven proposal, was used. Landmarks were placed every 5 meters to obtain a total of 63 landmarks in the environment. The PTM contains 9 topologies, and the most likely topology, which is also the ground-truth, obtains approximately 72% of the probability mass. The PTM along with the most likely topology is given in Figure 57. The metric map obtained from [33] is also shown for reference.

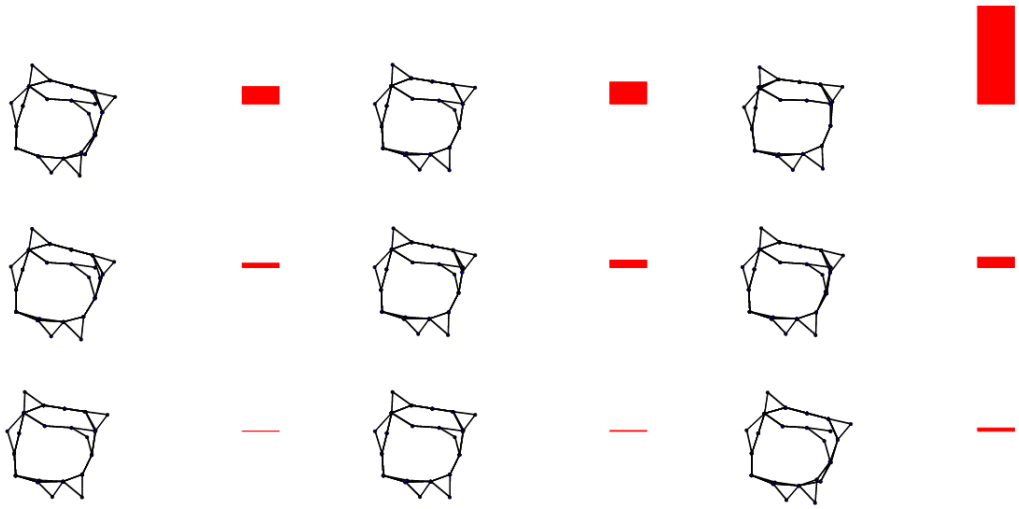
The smoothed trajectories for the above datasets were obtained using the optimization process described in Section 2.4 where it was used to compute the odometry likelihood, except that now all the odometry measurements are used instead of just the compounded odometry between landmarks. The figure also illustrates the locations of the landmarks. Nodes classified as being the same physical landmark share the same color.

4.6 Landmark Detection Through Bayesian Computation of Surprise

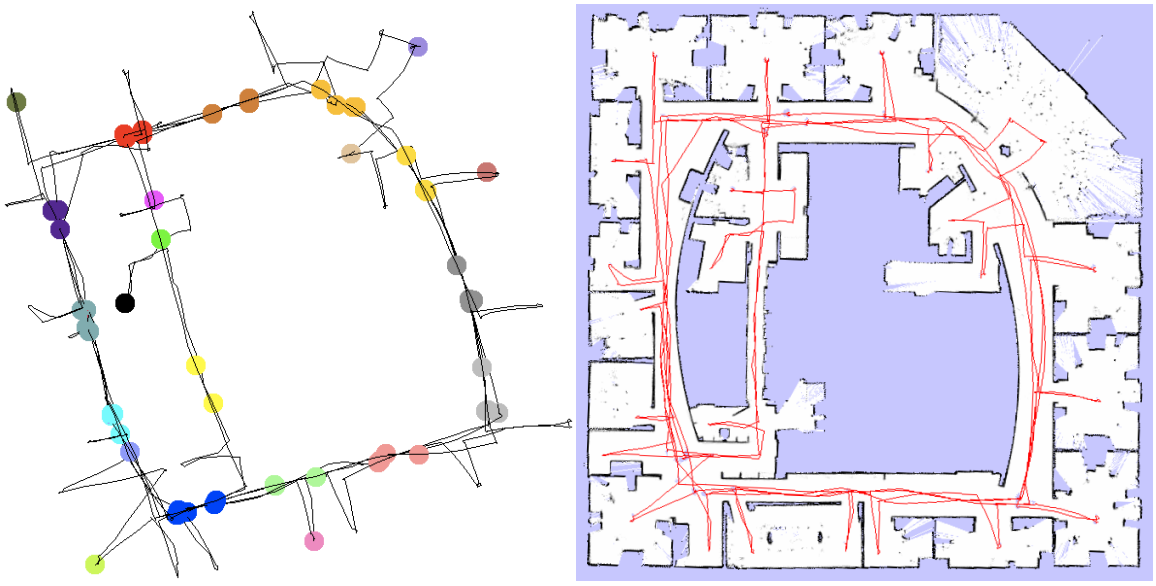
We now present a general purpose landmark detection scheme that can be applied to multiple sensing modalities. The scheme is based on the notion of "surprise", i.e. the unlikeliness of measurements according to the current model of the environment. Places that generate surprising measurements are classified as landmarks. By definition, surprise-based landmark detection is based on local information and hence, falls in the middle region of the landmark detection spectrum.

We base our surprise computation on the method proposed by Itti and Baldi [37, 38]. Consider the model at the current time as M and a prior distribution on the space of all possible models $P(M)$. Upon receiving a measurement z, the prior is updated to obtain a posterior on model space $P(M|z)$ using Bayes law

$$P(M|z) = \frac{P(z|M)P(M)}{P(z)}$$



(a)



(b)

Figure 57: (a) PTM for the Intel dataset obtained using the MC-cubed algorithm. 63 landmarks were placed in the environment at a distance of 5 meters from each other (b) Smoothed trajectory for the most likely topology. Landmarks are shown as colored circles with nodes corresponding to the same physical landmark colored similarly (c) Metric map of the Intel lab given for reference.

Surprise is now quantified as the change in the belief in the model upon observing the measurement. Clearly if the posterior is the same as the prior, there is zero surprise. This intuitive description of surprise can be made concrete by defining it as the KL-divergence between the prior and posterior distributions on model space, i.e.

$$S(z) = \int_M P(M) \log \frac{P(M)}{P(M|z)} \tag{59}$$

Note that the integral is over the space of all possible models. The computation of surprise using the above equation is inherently recursive as the posterior in one step becomes the prior for the subsequent step when the next measurement is obtained.

This definition of surprise is intuitive in the sense that if a measurement that is surprising at first is observed repeatedly, it loses its surprising nature. Such operation is required when we apply surprise to landmark detection as the landmark detector should fire when the robot moves into a new area but not after that.

4.7 SIFT Feature based Landmark Detection

We now apply the theory of surprise to the Multivariate Polya model discussed in Section 4.4. Consider the situation where the set of histogram measurements $A = \{a_i | 1 \leq i \leq n\}$ has been observed. The prior model for surprise computation is then simply the Multivariate Polya model learnt using A. If now a measurement z is observed, the posterior is the Multivariate Polya model learnt using the measurements $\{A, z\}$. Computation of surprise according to (59) can then be done as

$$S(z) = \int_{\alpha,a} P(a;\alpha) \log \frac{P(a;\alpha)}{Q(a;\alpha,z)}$$

where the posterior distribution has been denoted as Q. However, the integration over the α parameter is not possible in closed form. Hence we simply use the maximum aposteriori value of α as learned from the measurement sets in question. The integral over α is then replaced by the pdf value at the MAP location

$$S(z) = \int_a P(a;\alpha_{MAP}) \log \frac{P(a;\alpha_{MAP})}{Q(a;\alpha'_{MAP})} \tag{60}$$

where α_{MAP} is the value learned using A and α'_{MAP} is the value learned using $\{A, z\}$.

The computation of the KL divergence using (60) is still not possible in closed form due to the form of the Multivariate Polya model. We now briefly summarize the exponential family approximation to the Multivariate Polya model given by Elkan [23]. Using this approximation, the surprise can be computed in closed form.

4.7.1 Exponential Family Approximation

Consider the Multivariate Polya model from (57), repeated here for ease of exposition

$$P(x) = \frac{n!}{\prod_{w=1}^{W} n_w} \frac{\Gamma(\alpha)}{\Gamma(n+\alpha)} \prod_{w=1}^{W} \frac{\Gamma(n_w+\alpha_w)}{\Gamma(\alpha_w)}$$

Firstly, since many of the appearance words do not occur in a given histogram, we rewrite the above model so it can be computed using only words that do occur

$$P(x) = \frac{n!}{\prod_{w:n_w\geq 1} n_w} \frac{\Gamma(\alpha)}{\Gamma(n+\alpha)} \prod_{w:n_w\geq 1}^{W} \frac{\Gamma(n_w+\alpha_w)}{\Gamma(\alpha_w)}$$

Empirically, the learned values of $\alpha$is usually such that $\alpha_w \ll 1$ in most cases. For small α, the following approximation hold

$$\frac{\Gamma(x+\alpha)}{\Gamma(\alpha)} - \Gamma(x)\alpha = 0$$

so that we can substitute $\frac{\Gamma(x+\alpha)}{\Gamma(\alpha)}$ by $\Gamma(x)\alpha$. Also using the fact that $\Gamma(z) = (z-1)!$ yields the exponential family approximation to the Multivariate Polya model

$$q(x) = \frac{n!}{\prod_{w:n_w\geq 1} x_w} \frac{\Gamma(s)}{\Gamma(s+n)} \prod_{w:n_w\geq 1} \beta_w \tag{61}$$

where the parameters have been denoted as β following Elkan [23] to distinguish them from the exact model , and $s = \sum_w \beta_w$.

That the model specified by (61) is in the exponential family can be seen by writing it in the form

$$q(x) = \left(\prod_{w:n_w\geq 1} n_w^{-1} \right) n! \frac{\Gamma(s)}{\Gamma(s+n)} \exp\left(\sum_{w=1}^{W} I(n_w \geq 1) \log \beta_w \right)$$

and comparing it with the canonical exponential family model given as

$$p(x) = h(x)\exp\left\{\theta^T T(x) + A(\theta)\right\}$$

whence we get the canonical parameter of the exponential family approximation as $\theta = \log\beta$, the sufficient statistics as

$$T(x) = I(n_w \geq 1) \tag{62}$$

and the log normalization factor as

$$A(\theta) = -\log\frac{\Gamma(s+n)}{n!\Gamma(s)} \tag{63}$$

Given a collection of documents the maximum likelihood value of β can be learned in a similar manner to (58) using iterative fixed point equations as follows

$$s = \frac{\sum_w \sum_d I(n_{dw} \geq 1)}{\sum_d \psi(s+n_d) - |D|\psi(s)} \tag{64}$$

$$\beta_w = \frac{\sum_d I(n_{dw} \geq 1)}{\sum_d \psi(s+n_d) - |D|\psi(s)} \tag{65}$$

4.7.2 A Closed-form Expression for Surprise

Given the above discussion, we can now compute the KL-divergence between two exponential family Polya models using the expression for the model (61). This yields

$$KL(p||q) = \log\frac{\Gamma(s_q+n)}{\Gamma(s_p+n)} - \log\frac{\Gamma(s_q)}{\Gamma(s_p)} + \sum_{w=1}^{W}\log\frac{\beta_w^q}{\beta_w^p}\int q(x)I(n_w \geq 1)$$

where s_p, β_p and s_q, β_q are the parameters for the distributions p and q respectively.

A well-known property of exponential family distributions is now used, which states that the expected value of the sufficient statistic is equal to the derivative of the log normalization factor wrt the canonical parameter, i.e.

$$\int p(x)T(x)dx = \frac{dA(\theta)}{d\theta}$$

Using this property, and the log normalization function from (63), we get the KL-divergence as

$$KL = \log\frac{\Gamma(s_q+n)}{\Gamma(s_p+n)} - \log\frac{\Gamma(s_q)}{\Gamma(s_p)} - \big(\psi(s_q+n)-\psi(s_q)\big)\sum_{w=1}^{W}\beta_w^q \log\frac{\beta_w^q}{\beta_w^p} \quad (66)$$

Surprise, as defined for the Multivariate Polya model in (60), can be computed using the above equation. The parameter values are learned for the prior distribution using all the measurements observed up to the current time. The posterior parameter is learned similarly, but by also adding the current measurement to the dataset. The KL-divergence between these two distributions, which is the surprise, is computed using (66).

4.7.3 Landmark Detection

Landmark detection using surprise involves modeling the values of KL-divergence for the case where there is no surprise. Subsequently, any values outside the allowable range can be declared as surprising, resulting in a landmark being detected. While theoretically, the case of no surprise should result in a surprise value of zero, this is impossible in practise since the distribution always changes upon an update in a Bayesian framework.

As a first approximation to modeling the case of no surprise, consider the scenario where the robot observes a number of distinct of measurements, say n of them, but subsequently observes only a single measurement repeatedly. The evolution of KL-divergence on updating the model using this repetitious measurement can be determined empirically, and is shown in Figure 58 for different values of n. It can be seen that even if the robot observes exactly the same measurement repeatedly, the KL-divergence does not go to zero immediately but decays in an exponential manner.

However, the scenario where we consider exactly the same measurement repeatedly is also not realistic. Instead, we consider the measurement unsurprising if it does not conform to the current model learned using all the previous measurements. This is determined by computing the predictive KL-divergence at each step.

To compute the predictive KL-divergence, we generate a measurement from the current

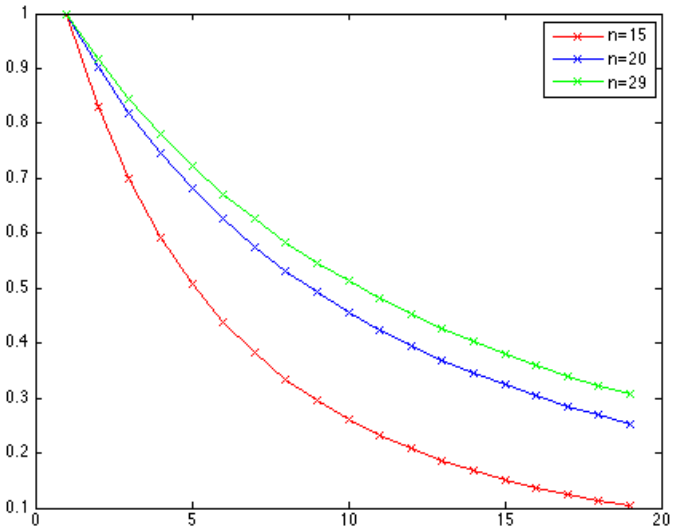


Figure 58: Evolution of KL-divergence for updates involving the same measurement observed repeatedly. n is the number of distinct measurements used to learn the initial model after which updates are done using the same measurement repeatedly. The x-axis shows the number of updates and the y-axis shows the normalized KL-divergence values.

model and use it update the model to get the posterior. The KL-divergence between this posterior and the original model is the predictive divergence. In practice, this update using a simulated measurement is done a number of times and the mean and variance of the predictive KL-divergence is computed. Doing this overcomes the problem that the model may generate an unlikely measurement occasionally.

Landmark detection is done by comparing the actual surprise (KL-divergence) and the predictive surprise (KL-divergence) computed as above. If the actual surprise is outside a certain confidence interval of the predicted mean and variance of the surprise, we declare it to be a landmark. Since KL-divergence is a continuous function, it decays gradually, resulting in multiple closely spaced landmark detections. Hence, only the maxima and minima of the actual KL-divergence that lie outside the confidence interval are considered as landmarks.

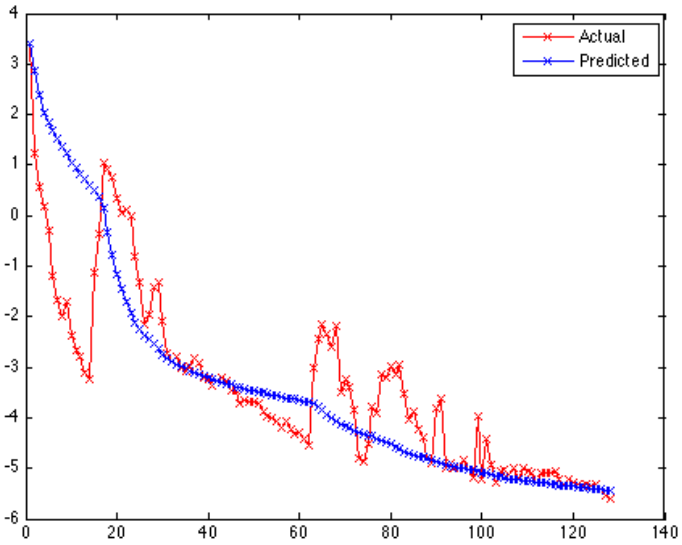
4.7.4 Results

The above landmark detection scheme was applied to the TSRB dataset where measurements were obtained from a camera rig. SIFT features were obtained and appearance words computed in exactly the same fashion as Section 4.3 with 1024 appearance words being computed using K-means clustering. The MC-cubed variant of the MCMC algorithm with an odometry-based data-driven proposal was used to compute the PTM in this experiment.

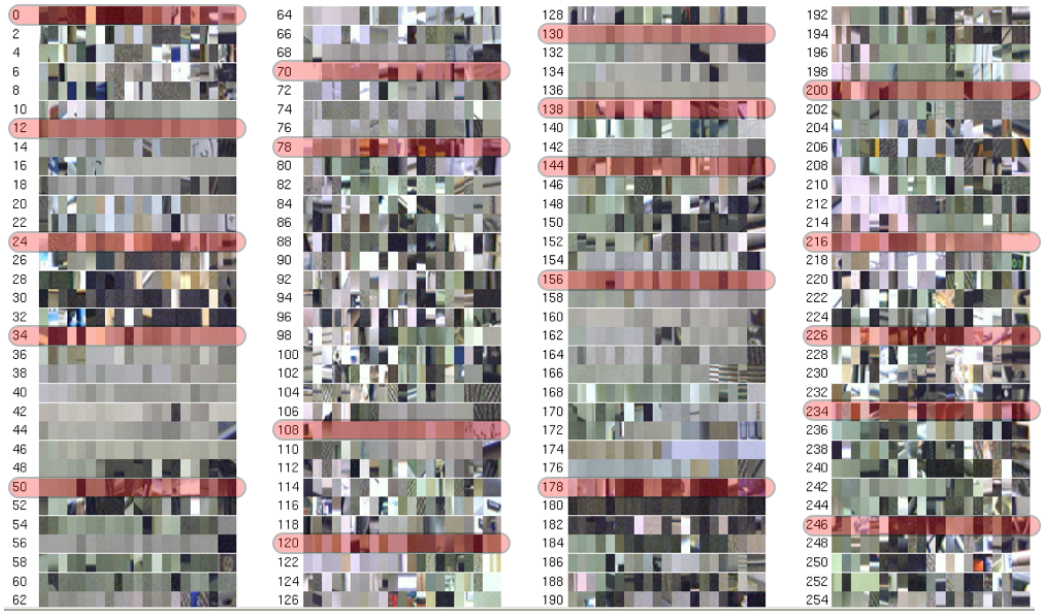
The predictive KL-divergence curve and the actual values are shown in Figure 59(a). The figure also shows the top twenty SIFT features from the appearance histogram for certain places. The PTM obtained using the landmarks detected contains only the ground truth topology and is shown in Figure 60 along with the smoothed trajectory. Colors of the nodes depict the set in the topology that they belong to, so that nodes classified as being the same place are colored similarly. Note that all the decision points are classified as landmarks, while a few false positives also exist. Mosaics of a few of these landmarks in Figure 61 show that they indeed correspond to locations that are qualitatively different from their surrounding areas.

4.8 Laser based Landmark Detection

We now provide a landmark detection scheme using laser range scans that is based on the computation of Bayesian surprise. Firstly, we convert the laser scans to a representation that can be used to model places. We choose a very simple representation and model a place by the area of the laser scan obtained there. The "model" for the place is thus a single scalar quantity. The area contained in a laser scan can be computed by triangulation followed by computation of the areas of the triangles which are summed up to obtain the desired area. Since in most cases, only a single laser is available, the robot has a forward facing view of the world. This implies that if the robot were to approach the same place from a different direction, the place models would not match. We get around this problem by building map patches incrementally around each place as the robot moves. These patches



(a)



(b)

Figure 59: (a) Actual and predictive KL-divergences for the TSRB dataset. The variances for the predictive divergences are so small that even 3σ curves are hard to view at this scale. (b) Top 20 SIFT features by histogram count for each location denoted by the measurement number. Only every second measurement is shown. The measurements corresponding to landmarks (i.e. where the landmark detector fires) are shown in red. It can be seen that these correspond to the start of sub-sequences of measurements that differ from the preceding measurements.

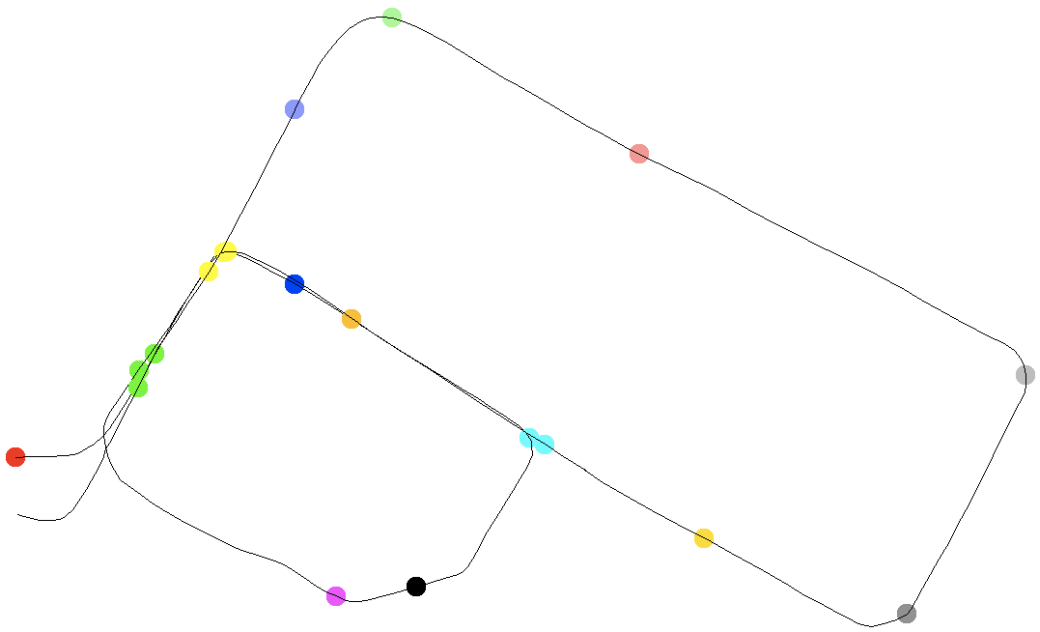


Figure 60: PTM containing single topology with all the probability mass, showing landmarks detected using Bayesian surprise computation. The smoothed trajectory is also shown. Nodes belonging to the same physical landmark are colored similarly.

give an omni-directional, orientation-independent model for places.

Since a place in a topology does not imply a precise metric location, the area measured by laser scans in the same place will differ slightly due to the robot not being in exactly the same location. This uncertainty is modeled using a Gaussian distribution, which is the parametric model distribution used for computing Bayesian surprise.

Given the above model, the computation of surprise is straight-forward. As before, we sample from the Gaussian model to obtain measurements and use these samples to compute the predicted KL-divergence and its variance. The actual KL-divergence is subsequently compared with the predicted value to obtain landmarks.

The actual KL-divergence between two Gaussian distributions, which is the Bayesian surprise in this case, is computed as follows

$$KL(p||q) = 0.5 \log \frac{\sigma_p^2}{\sigma_q^2} + \frac{\mu_q^2 + \mu_p^2 + \sigma_p^2 - 2\mu_p\mu_q}{2\sigma_p^2} - 0.5 \quad (67)$$

As before, since KL-divergence is a continuous function, it decays gradually so that

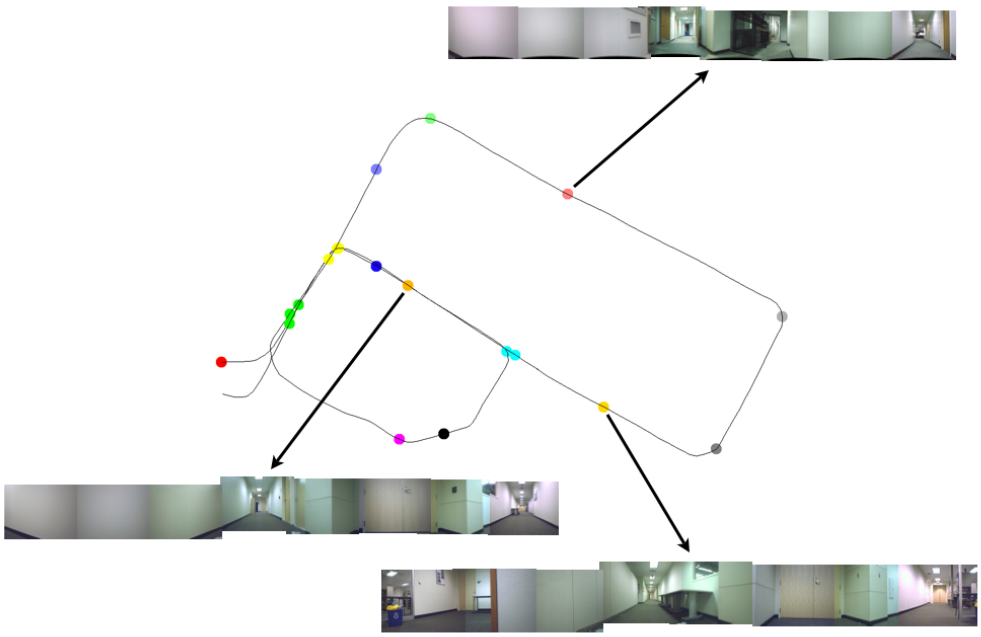


Figure 61: Smoothed trajectory for the ground truth topology with the rig panoramas corresponding to a few landmarks. This illustrates that many of the landmarks that seem to be false positives at first glance are, in fact, genuine landmarks due to the presence of doors and gateways, even though the trajectory does not indicate this.

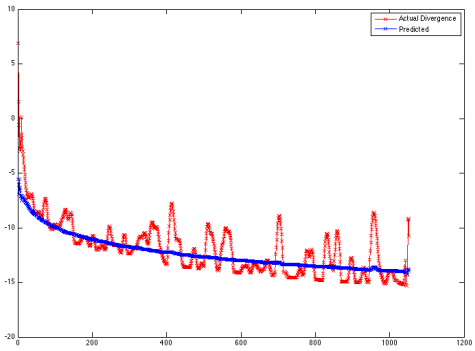


Figure 62: Actual and predicted KL-divergence (surprise) for the CRB dataset using laser measurements. 19 landmarks are detected in total.

only the maxima and minima of the actual KL-divergence that lie outside the confidence interval are considered as landmarks.

The laser-based Bayesian surprise computation was applied to the CRB dataset described in Section 3.12. The dataset contains a total of 2106 laser scans. The actual and predicted KL-divergence for each step are shown in Figure 62. 19 landmarks were detected in total. The particle filtering algorithm with data-driven proposal was used to compute the PTM in this and all the subsequent experiments in this section. The PTM obtained using these landmarks has the ground truth topology as the most likely one, receiving 64% of the probability mass, as shown in Figure 63. The smoothed trajectories corresponding to a few of the topologies in the PTM are also shown in Figure 63. Landmarks at the corners are detected when the laser sees around the corner for the first time, and hence, anticipate the actual corners slightly. The number of landmarks and their placement is almost perfect in this case.

We next apply the landmark detection scheme to the MIT Killian Court dataset [9] which is another widely used dataset in the SLAM community. The dataset consists of 1941 poses and corresponding laser scans. The ground-truth metric map with laser scans and robot trajectory is shown in Figure 64 for reference. A total of 61 landmarks were detected using laser-based surprise and the PTM obtained using these landmarks, which also

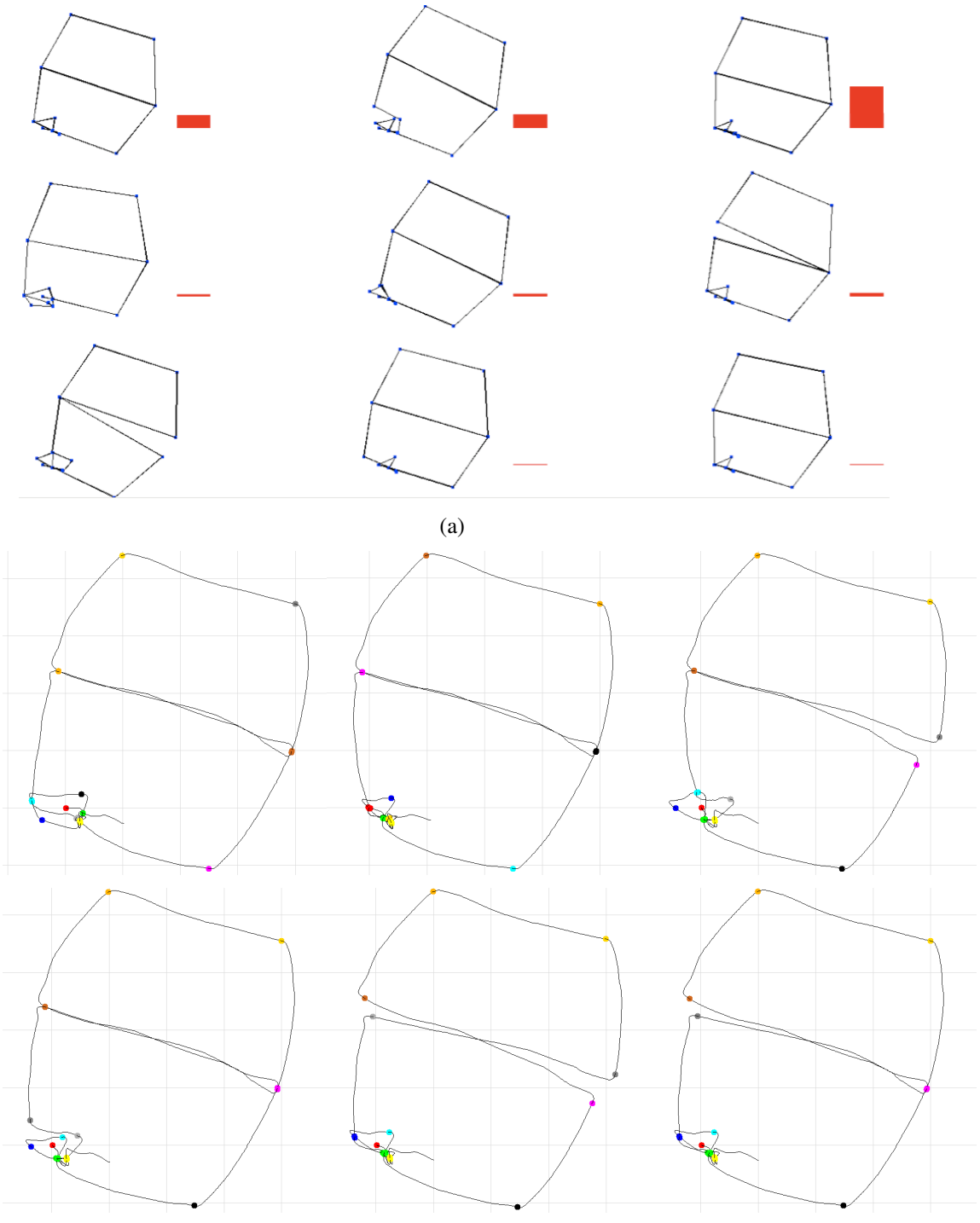


Figure 63: (a) PTM for the CRB dataset with automatic landmark detection using Bayesian surprise. The topology at the top right with the maximum probability is the ground truth. (b) The smoothed trajectories for some of these topologies (not in order of the PTM), where the first one (top left) corresponds to the ground truth topology. Nodes belonging to the same physical landmark are colored similarly.

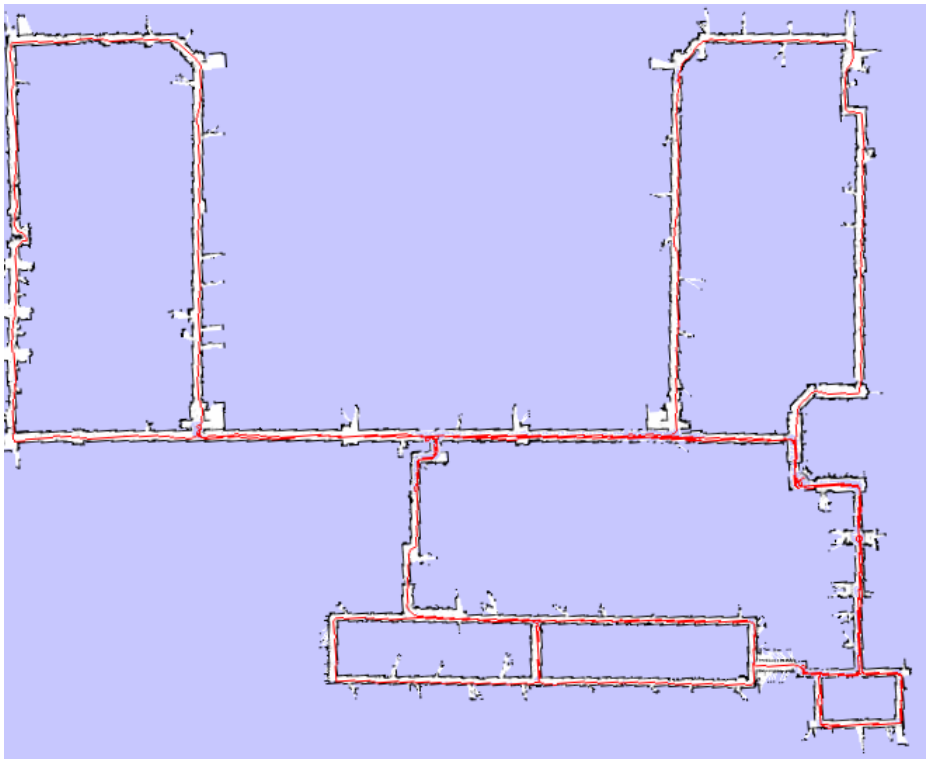
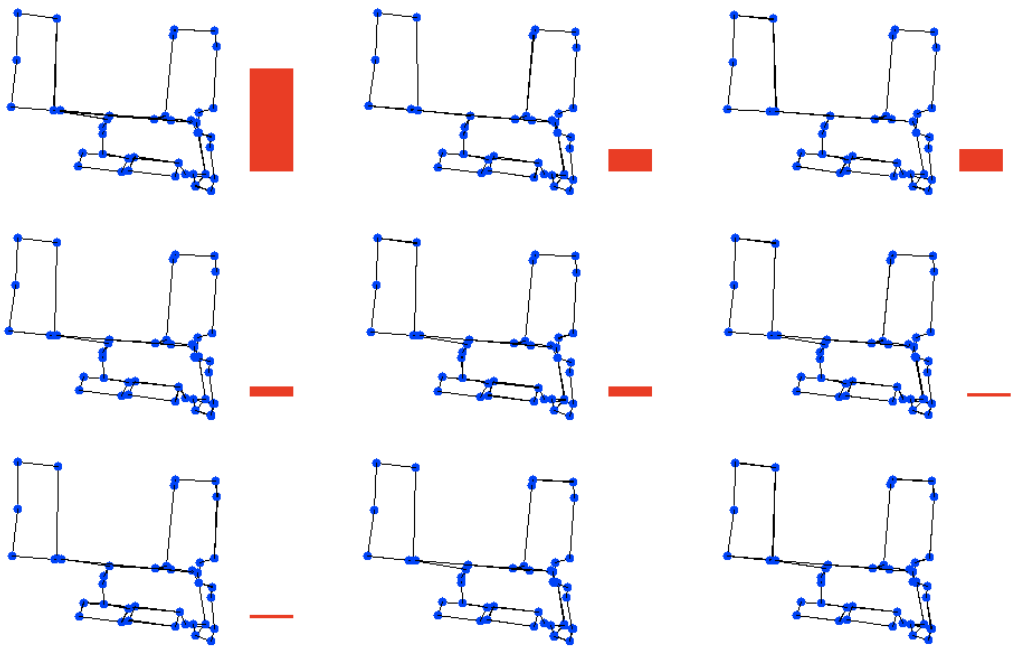


Figure 64: Metric map of Killian Court dataset obtained from [9].

contains the ground truth as the most likely topology, is shown in Figure 65. The ground truth receives 81% of the probability mass. Figure 65(b) gives the trajectory smoothed with the topological constraints and also the color-coded nodes as before. It can be seen that only a few false positives are found, and crucially, all the actual landmarks, i.e. the junctions and gateways, are accurately detected. The robot trajectory in this dataset spans an area of more than 200x200 meters and is considered challenging for metric mapping algorithms. It is however, a relatively easy sequence for performing topological mapping due to the wide separation between most landmarks, thus illustrating the advantage of a topological map over metric maps in this case.



(a)

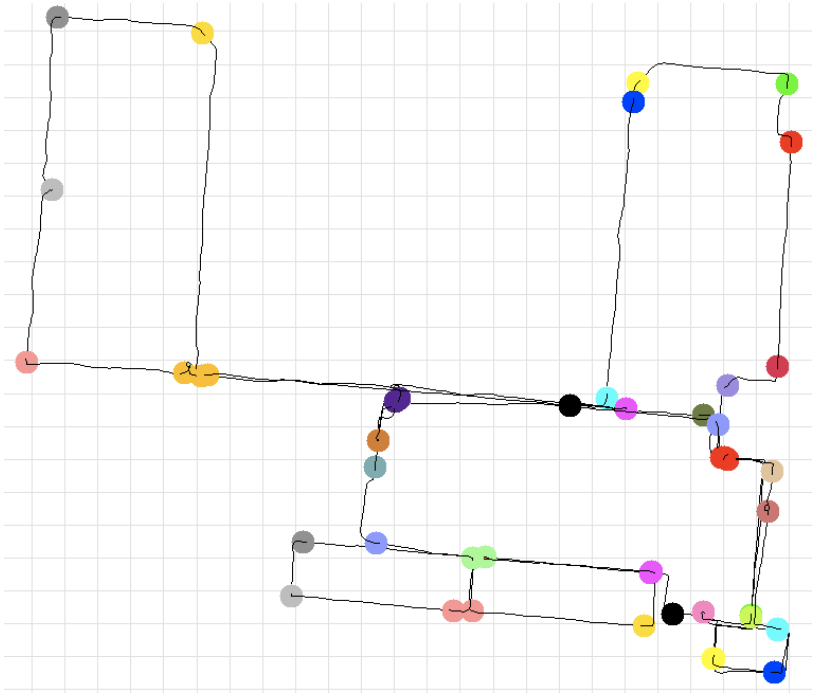


Figure 65: (a) PTM for the MIT Killian Court dataset with automatic landmark detection using Bayesian surprise. The topology at the top left with the maximum probability is the ground truth. (b) The smoothed trajectory corresponding to the ground truth topology.

CHAPTER V

GENERAL APPLICABILITY OF PTMS

The PTM framework does not make use of the low-level characteristics of any specific sensors. The use of a Bayesian framework abstracts the details of the individual sensors into the sensor models and provides sensor independence. All the algorithms described so far have also been independent of the hardware details. Even the algorithmic pieces that rely on specific sensor characteristics, such as the data-driven proposals, can be modularized and are exchangeable with other similar pieces based on different sensors.

A caveat here is that all the sensor models used so far have been omni-directional in nature. This is because each measurement is treated like a place model and is required to be orientation and direction invariant; hence necessitating an omni-directional view. However, that the sensor models are required to be omni-directional does not place a similar constraint on the sensors themselves, can be seen from the case of the laser and its sensor model. The laser range scanner used here has only a 180^o view but the sensor model overcomes this by creating omni-directional patches around the landmark locations.

In this chapter, I will provide evidence for the claim made in the thesis (Section 0.2) regarding the wide applicability and sensor-independence of the PTM framework. In truth, this claim has already been verified through the results presented in the previous chapters which involved various sensors and diverse environments. Hence, all that is left to do here is to recap these results and emphasize the diversity of scenarios to which PTMs are applicable. In the following sections, I summarize the results obtained on various datasets using different sensors and sensor models.

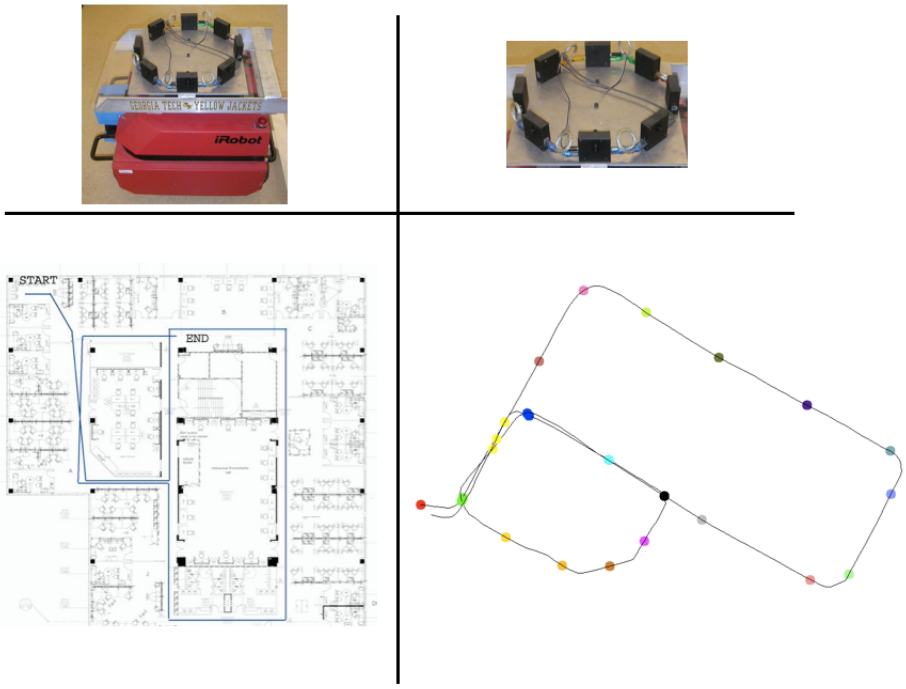


Figure 66: Figure showing the ATRV-mini robot used in the TSRB experiments, the eight camera rig used to obtain panoramic appearance measurements, the ground-truth robot trajectory on a floorplan of the building, and the most probable topological map from the PTM, which is the groundtruth computed using appearance and odometry. Landmarks were placed at equi-distant intervals.

5.1 TSRB Dataset with Appearance

PTMs were tested using the TSRB dataset with two different appearance models, both derived from a camera rig that provides panoramic images. The TSRB dataset has two loops, the smaller one enclosed in the larger one, and hence though small, is an effective dataset. In Chapter 2, a PTM was computed using Fourier signatures as appearance measurements, with a hierarchical generative model. Here, I demonstrated using the MCMC algorithm that when additional measurements are provided to the PTM framework, it results in more confident posteriors. This was shown by first computing the PTM using only odometry, and then comparing it to the one obtained using both odometry and appearance. Both these PTMs were obtained using the vanilla MCMC algorithm. These experiments were repeated using the data-driven proposal and MC-cubed algorithm in Chapter 3 to demonstrate the speed-up in PTM construction.

In Chapter 4, I computed the PTM for the same dataset, but with a more sophisticated appearance model involving SIFT features as measurements. Automatic landmark detection, both by placing landmarks at equi-distant intervals and through surprise-based detection, was included. The PTM was computed using the particle filtering algorithm with an appearance-based data-driven proposal. Since the appearance model is sophisticated and highly discriminative, a very confident posterior can be obtained without the use of odometry.

5.2 TSRB Dataset with Laser

The use of laser measurements for PTM computation was demonstrated using the particle filtering algorithm in the context of the TSRB dataset. While the laser measurements were not omni-directional, this was overcome by maintaining local patches as the robot moved. These omni-directional patches were then used as measurements. Once the measurement model is defined, using scan matching in this case, the PTM particle filtering algorithm can be used exactly as before without any changes. This demonstrates the versatility of the

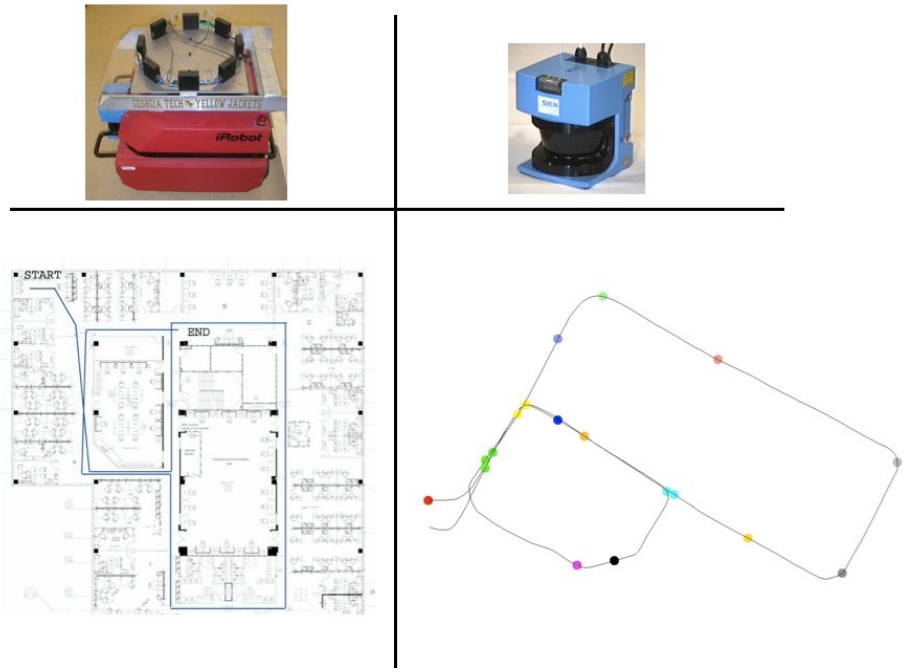


Figure 67: A quad chart showing the robot, sensor, ground-truth trajectory, and most likely topology from the PTM computed using laser scans for the TSRB experiment

framework.

5.3 CRB Dataset with Appearance and Laser

The CRB dataset was used to validate the PTM algorithms using both appearance and laser measurements. This dataset involves large loops but is relatively simple due to the small number of landmarks and low possibility of aliasing. However, the metric scale of the environment is much larger than the TSRB dataset, and so, provides evidence for the scale-invariance of the PTM algorithms.

Chapter 2 provides results of PTM construction, first with odometry alone, and subsequently, with odometry and appearance. This provides further evidence for the ease of incorporating additional measurements into the framework, and its effect in making the posterior more confident. Automatic surprise-based landmark detection, using laser patches constructed from scans, was included in the context of this environment in Chapter 4. Partly due to the simplicity of the environment, surprise-based detection is particularly

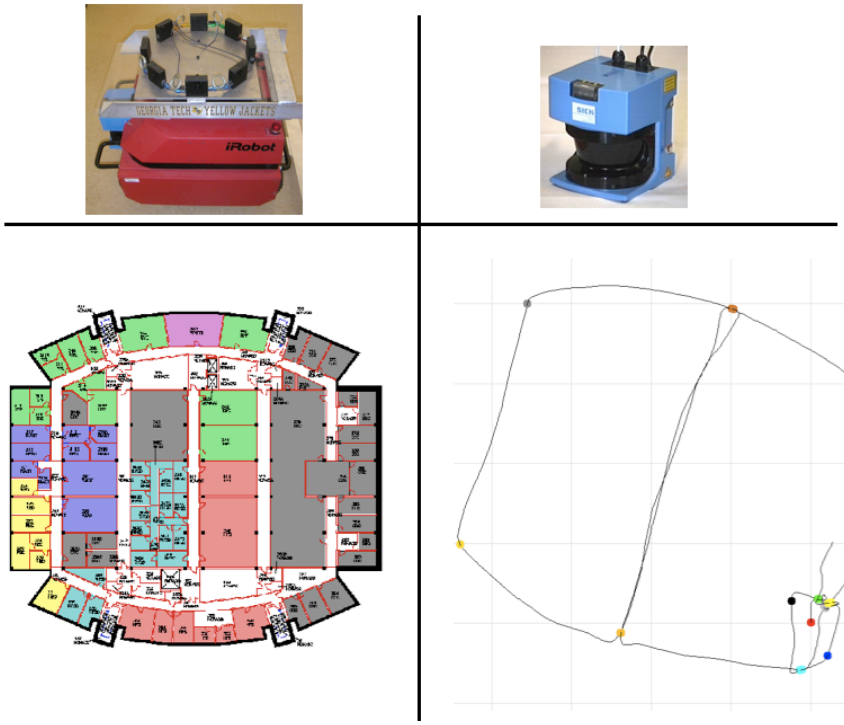


Figure 68: Quad chart showing the robot, sensor, floorplan of the building, and most likely topology from the PTM computed using laser scans for the CRB dataset. Landmarks were detected using Bayesian surprise.

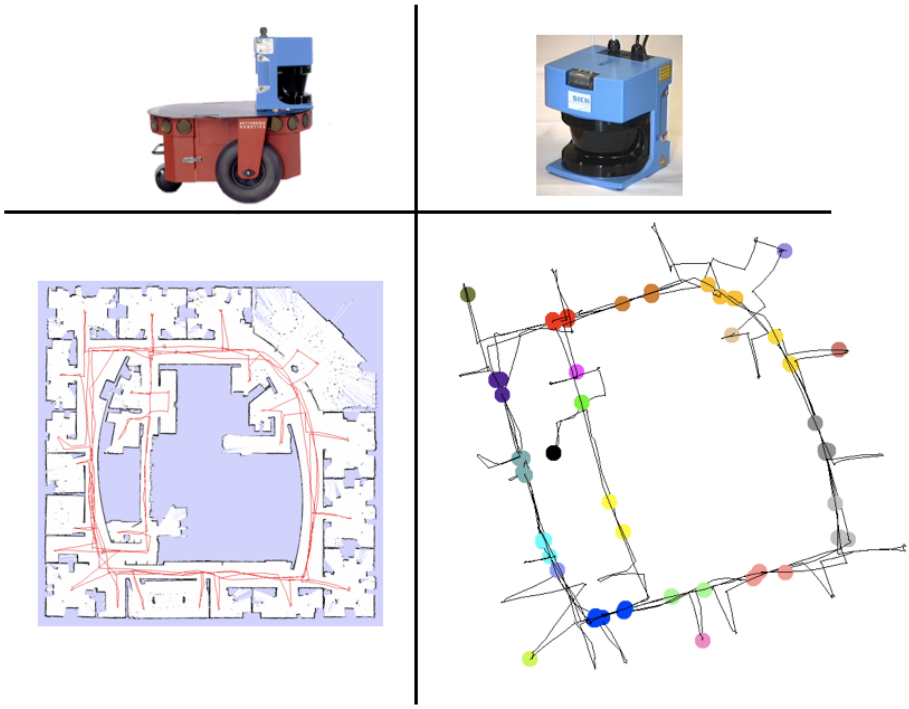


Figure 69: Quad chart showing the Pioneer 2 robot, sensor, metric map from [33], and most likely topology from the PTM for the Intel dataset. Landmarks were placed every 3 meters in the environment.

effective in this environment.

5.4 Intel Dataset with Laser

PTMs were also validated using well-known datasets from the SLAM community. The first of these was the Intel dataset, which was collected using a Pioneer 2 robot mounted with a laser scanner at the Intel laboratory in Seattle. The dataset consists of multiple loops around a building with excursions into a number of rooms along the perimeter of the building, and is considered challenging for metric mapping algorithms.

The PTM for this dataset was computed by placing landmarks at equi-distant intervals and using the scan matching model for laser patches. The particle filtering algorithm was used to perform incremental inference on the space of topologies with 63 landmarks. This demonstrates that PTMs scale to environments with a large number of landmarks.

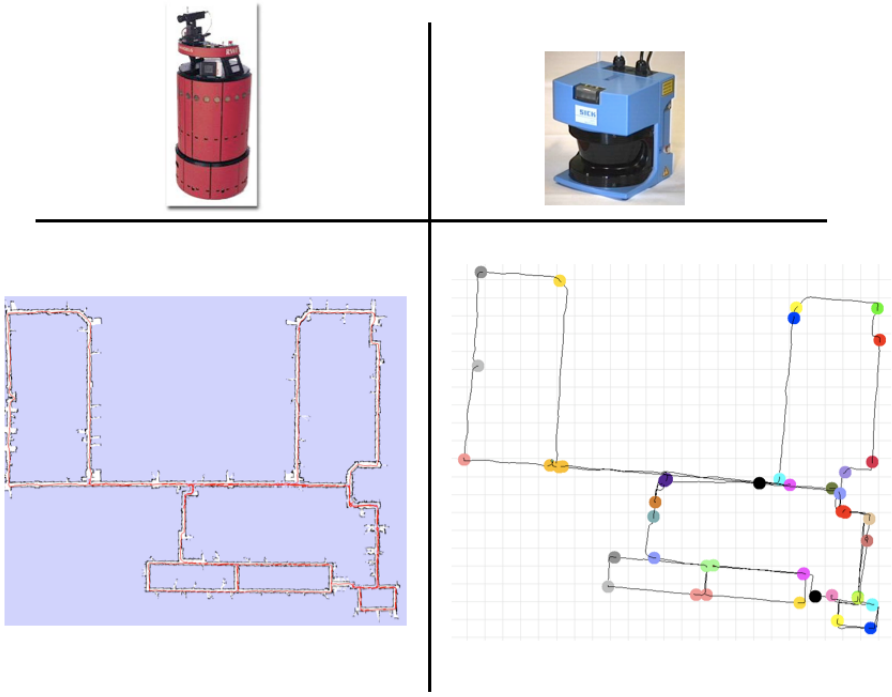


Figure 70: Quad chart showing the B21 robot, sensor, metric map from [9], and most likely topology from the PTM for the MIT Killian Court dataset. Landmarks were detected automatically using Bayesian surprise.

5.5 MIT Killian Court Dataset with Laser

Another well-known dataset on which PTMs were validated is the MIT Killian Court dataset with the "infinite corridor". This dataset is considered particularly challenging for metric mapping since odometry error accumulates the long corridor, leading to loop closing errors and an incorrect map. Odometry and laser range scans are the only measurements available in the dataset.

Automatic surprise-based landmark detection was performed to detect 61 landmarks in the environments. All the landmarks, and a few false positives, were detected in this case, which validates the landmark detection scheme. The MC-cubed algorithm with a data-driven proposal was used to compute the PTM. The ground-truth topology obtains the highest probability by far, attesting to the robustness of PTMs.

5.6 Discussion

I demonstrated the working of PTMs with three sensors - odometry, panoramic images, and laser range scans. I also showed that multiple sensor models may be used for the same sensor depending on the inference requirements. For instance, when odometry was fairly discriminative in the case when landmarks were detected manually, the weak appearance information provided by modeling Fourier signatures was deemed sufficient. However, when landmarks are present much more densely in the environment, for instance when they are placed at equi-distant intervals, a stronger and more discriminative sensor model was required, and provided in the form of the Multivariate Polya model. Hence, simpler more efficient models may be used in easier environments or when good landmark detection is present, ensuring greater flexibility and efficiency.

Some of the models presented here extend to other commonly used sensors that were not used in this dissertation. For example, images from omnidirectional cameras can be modeled by the Multivariate Polya model, exactly as described here, after SIFT features have been detected on them. Similarly, sonar and lidar sensors can be modeled similar to laser range scans.

All our test sets cover indoor environments, and the primary reason for this is the higher utility of topological maps in man-made environments where landmarks, regions, and gateways are clearly defined and can be extracted automatically with some measure of success. Landmark detection in outdoor environments is, in comparison, a vastly more difficult problem and hence, hinders the widespread use of topological maps in this domain. Provided landmarks can be detected with relative stability, the PTM framework can be applied to outdoor environments with very few changes.

CHAPTER VI

DISCUSSION

6.1 Thesis Restated

The thesis statement presented in Section 0.2 can now be restated with all the terms explained in detail, and all the claims made therein defended through experimental results:

> Probabilistic Topological Maps provide a systematic framework for topological mapping that overcomes topological ambiguity when it is possible and is cognizant to failure when it is not. Further, PTMs are practical, efficient, compatible with various landmark detection schemes, and generalizable to diverse sensing modalities.

6.2 Synopsis

I have demonstrated in this dissertation that Probabilistic Topological Maps (PTMs) solve the problem of topological ambiguity in a robust manner, and are also capable of incorporating various landmark detection schemes and sensing modalities in a seamless manner.

The mathematical techniques used in this dissertation mainly relate to Bayesian statistics and machine learning. The sample based PTM representation is computed using the MCMC and Particle filtering algorithms, which are used to perform inference in the combinatorial space of topologies. I showed that variants of these sampling algorithms, such as Simulated Tempering and Rao-Blackwellized Particle filters, can be used to vastly improve the runtime of the inference. Sensor models were defined using hierarchical Bayesian models where applicable with the marginalization of nuisance variables, as is appropriate in a fully Bayesian scheme. The starting point for the whole analysis is, however, the equivalence between topological maps and set partitions, followed by the use of the combinatorial

properties of set partitions for the design of smart proposal strategies.

Starting from a simple proposal strategy that can sample from the space of topologies using two simple steps, viz. split and merge, efficient data-driven proposals were constructed, and I have shown, through concrete algorithms for the case of odometry and SIFT-based appearance, how different measurement streams may be used as sources of the data incorporated in these proposals. I introduced novel prior distributions on the space of topologies that encode various assumptions about the robot's motion. The Dirichlet Process prior, which has in particular been widely used in experiments, can also model the measurements as arising from a potentially infinite mixture model, where each component of the mixture corresponds to a physical landmark.

While initially, the availability of a perfect, abstract landmark detector was assumed, this assumption was dropped in Chapter 4 where I validated the PTM framework with various landmark detectors, starting from a simple blind detector that places equidistant landmarks. That the use of the PTM framework cleanly decouples the problem of topological ambiguity from that of landmark detection can be clearly seen from the use of multiple landmark detectors without requiring any change in the PTM algorithms themselves.

A new scheme for landmark detection was proposed based on Bayesian computation of a quantity characterized as "surprise", which measures the effect of a new measurement on the current model for places: the greater the effect, the more "surprising" the measurement. A systematic method for quantizing the case where a measurement elicits no surprise was formulated and used to detect surprising measurements, and hence, landmarks. I presented results for validating the various techniques through experiments performed on robots with varying sensory equipment. Standard datasets used in the SLAM community for evaluation were also used to present the case for the correctness of the PTM algorithm and surprise-based landmark detection scheme. The results in Chapter 4 validate both the landmark detection scheme and the PTM algorithms used with this scheme.

6.3 Future work

A number of improvements are possible to the techniques presented in this dissertation. Firstly, I have not considered the inclusion of any domain-specific topological information into the prior on topologies. The form of domain-specific knowledge may include information such as the planarity of topological maps in a 2D environment [81]. In addition, man-made environments only very rarely have junctions of high-order, i.e. where more than 4-5 paths meet. The use of such information in the prior will prune the search space leading to more efficient inference. On the other hand, the space of topologies may no longer be connected, making more sophisticated and special-purpose proposal moves necessary. I have also side-stepped the issue of learning the parameters involved in the prior distributions, examples of which are the parameters in the landmark location prior defined in Section 2.3. These parameters may, in theory, be learned if a large database of actual topological maps with landmark locations is provided. However, even if such a dataset were available, the large variation in types of environments and landmark distributions would make a completely automatic determination of parameter values a difficult task.

Another omission in this dissertation is regarding the assumption that the landmark detector fires reliably at every actual landmark in the environment, while also yielding false positives. In practice, every landmark detector yields both false positives and false negatives; true landmarks that are not detected as such. The inference mechanisms presented in this dissertation do not account for the possibility of undetected landmarks between two detected ones. While false negatives in landmark detection can be accounted for, this increases the inference space enormously and requires strong assumptions regarding the performance of the landmark detector, i.e. how many and how often false negatives are produced. Without this knowledge inference is essentially impossible since the number of undetected landmarks between two detected ones may be as large as imaginable.

With regard to landmark detection, the surprise-based approach presented here maintains the same model for all time and updates it continually as measurements are received.

As the KL-divergence between models computed at consecutive steps decays over time due to the increasing number of measurements incorporated into the model, this necessitates the computation of a 'control curve' that measures the KL-divergence for the case of no surprise. The actual divergence is then compared against this to detect landmarks. However, an alternate method using change point detection that does not need a control curve is possible. In this scenario, we assume that change points in the environment, where the generative model for measurements changes, are landmarks. At each time step, inference is performed to detect if the current location is a change point, and the current model is discarded if this is the case. A systematic analysis of change-point detection requires that all possible locations of change-points be considered. This leads to a combinatorial explosion, requiring the use of approximations to maintain tractability. The method I have proposed here has the advantage of simplicity and efficiency over change-point detection. However, it is future work to compare the performance of the change-point detection scheme to the current technique.

Many challenges in topological mapping remain that have not been addressed in this dissertation. The current map representation only supports the task of robotic navigation. Hence, a major challenge is of incorporating higher-level concepts and constructs into the maps that can enable common tasks that need to be performed by all robots interacting with humans. A first step in this direction would be to annotate nodes in the topology with objects contained therein and their locations. However, the varieties of semantic information that robots may need and that can be accommodated in maps are endless, ranging from labeling spaces by human usage to detecting doors and openings[69].

The surprise-based landmark detection method has been shown to work in indoor environments. However, outdoor environments offer much less variation in measurements at landmarks, for instance, a tree used as a landmark looks similar in an image from 10 meters away as it does from another that is taken from a distance of 1 meter. Further, even in indoor environments, the local nature of the detector makes false positives a problem. A

good solution to these problems involves the use of higher-level concepts again. Interesting objects and other semantic labels detected in the environment can be used to determine whether a place is interesting enough to be declared a landmark. Further, a feedback loop is possible that seeks out certain semantic objects if the robot is in the vicinity of a landmark, thus creating a bootstrap effect between landmark detection and semantic mapping.

APPENDIX A

DIRICHLET PROCESS PRIORS AND MIXTURE MODELS

The Dirichlet distribution forms our first step toward understanding the DPM model. The Dirichlet distribution is a multi-parameter generalization of the Beta distribution and defines a distribution over distributions, i.e. the result of sampling a Dirichlet is a distribution on some discrete probability space. Let $\Theta = \{\theta_1, \theta_2, \ldots, \theta_n\}$ be a probability distribution on the discrete space $\mathcal{X} = \{\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_n\}$ s.t. $P(X = \mathcal{X}_i) = \theta_i$ where X is a random variable in the space $\mathcal{X}$. The Dirichlet distribution on Θ is given by the formula

$$P(\Theta \mid \alpha, M) = \frac{\Gamma(\alpha)}{\prod_{i=1}^{n} \Gamma(\alpha m_i)} \prod_{i=1}^{n} \theta_i^{\alpha m_i - 1} \tag{68}$$

where $M = \{m_1, m_2, \ldots, m_n\}$ is the *base measure* defined on $\mathcal{X}$ and is the mean value of Θ, and α is a precision parameter that says how concentrated the distribution is around M. Both Θ and M are normalized, i.e. sum to unity, since they are proper probability distributions. α can be regarded as the number of pseudo-measurements observed to obtain M, i.e. the number of events relating to the random variable X observed apriori. The greater the number of pseudo-measurements the more our confidence in M, and hence, the more the distribution is concentrated around M.

To make the above discussion concrete, consider the example of a 6-faced die. A Dirichlet distribution can be defined on the space of possible observations from the die, i.e. the space $\mathcal{X} = \{1, 2, 3, 4, 5, 6\}$. If we consider the die to be fair *apriori*, then M can be defined as $M = \{\frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}, \frac{1}{6}\}$ and we arbitrarily set $\alpha = 6$ (which can be understood as corresponding to the case of our having observed every outcome of the die once apriori). The Dirichlet distribution defined by these values of α and M can now be sampled to yield, for example, $\Theta = \{0.113767, 0.179602, 0.273959, 0.153161, 0.169832, 0.109679\}$.

Clearly, the distribution used in the above example is not the only one possible on die

observations. We could, for instance, consider a Dirichlet distribution on the space $\mathcal{X}' = \{\{1,2\},\{3,4\},\{5,6\}\}$, so that $M = \{m_1, m_2, m_3\}$ and $\Theta = \{\theta_1, \theta_2, \theta_3\}$ are vectors of length 3. Θ is then a distribution on the random variable X taking a value from one of the sets in $\mathcal{X}'$, i.e. $P(X \in \{1,2\}) = \theta_1$ and so on. More generally, for any partition of a discrete space $\mathcal{X}$ into n sets $\mathcal{X}' = \{\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_n\}$ s.t. $\mathcal{X}_i \cap \mathcal{X}_j = \Phi \quad \forall \mathcal{X}_1, \mathcal{X}_2 \in \mathcal{X}'$ and $\bigcup_{i=1}^{n} \mathcal{X}_i = \mathcal{X}$, we can define a Dirichlet distribution $Dir(\Theta; \alpha, M)$ on $\mathcal{X}'$, where $P(X \in \mathcal{X}_i) = \theta_i$ for $1 \leq i \leq n$. We now introduce new notation replacing θ_i by $\Theta(\mathcal{X}_i)$ (and, correspondingly, m_i by $M(\mathcal{X}_i)$, so that the Dirichlet distribution on $\mathcal{X}$ can be written as

$$\Theta(\mathcal{X}_1), \Theta(\mathcal{X}_2), \ldots, \Theta(\mathcal{X}_n) \sim Dir(\Theta; \alpha, M) \tag{69}$$

where $Dir(.)$ is the Dirichlet density function. The intuition behind (69) is important as it forms the definition of the Dirichlet process in continuous spaces.

A.1 Posterior update using the Multinomial distribution

Consider N observations $X_1, X_2, \ldots, X_N$ that are multinomially distributed according to Θ. If n_i is the number of times the event $\mathcal{X}_i$ is observed in the N observations, the posterior probability on Θ can be obtained simply using Bayes Law as follows

$$\begin{aligned} P(\Theta \mid \alpha, M, X_{1:N}) &= kP(X_{1:N} \mid \alpha, M, \Theta) P(\Theta \mid \alpha, M) \\ &= k \prod_{i=1}^{n} \theta_i^{n_i} \times \prod_{i=1}^{n} \theta_i^{\alpha m_i - 1} \\ &= k \prod_{i=1}^{n} \theta_i^{\alpha m_i + n_i - 1} \\ &= Dir(\Theta; \alpha^*, M^*) \end{aligned}$$

where k is a normalization constant and

$$\begin{aligned} \alpha^* &= \alpha + N \\ M^* &= \frac{\alpha M + N\hat{F}}{\alpha + N} \end{aligned} \tag{70}$$

where $\hat{F}$ is the empirical distribution (i.e, simply the proportion of occurrence) of the n events in the observations. The posterior is again a Dirichlet distribution with altered parameters and so the Dirichlet distribution is a conjugate prior to the Multinomial distribution.

Now consider the probability of the $(N+1)$th observation X_{N+1}, given all the previous observations and the Dirichlet distribution parameters, $P(X_{N+1} \mid X_{1:N}, \alpha, M)$. Specifically, we want to calculate the probability that X_{N+1} is the event $\mathcal{X}_j$ in the space $\mathcal{X}$, i.e. $P(X_{N+1} \in \mathcal{X}_j \mid X_{1:N}, \alpha, M)$. The calculation is performed by marginalizing over Θ

$$\begin{aligned} P(X_{N+1} \in \mathcal{X}_j \mid X_{1:N}, \alpha, M) &= \int_\Theta P(X_{N+1} \in \mathcal{X}_j \mid \Theta) P(\Theta | X_{1:N}, \alpha, M) \\ &= \int_\Theta \theta_j Dir(\Theta | \alpha^\star, M^\star) \\ &= E(\theta_j) \\ &= \frac{\alpha m_j^\star}{\sum_{i=1}^n \alpha m_i^\star} = m_j^\star \end{aligned}$$

where $\alpha^\star$ and $M^\star = \{m_1^\star, m_2^\star, \ldots, m_n^\star\}$ are as defined in (70) so that $m_j^\star = \frac{\alpha m_j + \sum_{i=1}^N \delta(X_i = \mathcal{X}_j)}{\alpha + N}$. Hence, we get

$$P(X_{N+1} \in \mathcal{X}_j \mid X_{1:N}, \alpha, M) = \frac{\alpha m_j + \sum_{i=1}^N \delta(X_i = \mathcal{X}_j)}{\alpha + N} \tag{71}$$

Note that the derivation above uses the property of the Dirichlet distribution that $E(\theta_j) = \frac{M(\mathcal{X}_j)}{M(\mathcal{X})}$.

A.2 The Dirichlet distribution through the Polya Urn Model

Many probability distributions can be obtained using urn models [40]. The urn model that corresponds to the Dirichlet distribution is the Polya Urn model.

Consider a bag with α balls of which initially αm_j are of color j, $1 \leq j \leq n$ (assuming for now that all the αm_js are integers). We draw balls at random from the bag and at each step, replace the ball that we drew by two balls of the same color. Then, if we denote

probability of the obtaining a ball of color j at the ith step $P(X_i = j)$, it is easy to obtain

$$\begin{aligned} P(X_1 = j) &= \frac{\alpha m_j}{\sum_{i=1}^{n} \alpha m_i} = m_j \\ P(X_2 = j \mid X_1) &= \frac{\alpha m_j + \delta(X_1 = j)}{\alpha + 1} \end{aligned}$$

and so on, till we get

$$P(X_{N+1} = j \mid X_{1:N}) = \frac{\alpha m_j + \sum_{i=1}^{N} \delta(X_i = j)}{\alpha + N} \tag{72}$$

which is the same as (71). Hence, a Polya urn process gives rise to the Dirichlet distribution in the discrete case. In fact, this is trivially true from the definition of the Polya Urn model.

A.3 The Dirichlet Process

The Dirichlet process is simply an extension of the Dirichlet distribution to continuous spaces. Referring back, we see that (69) implies the existence of a Dirichlet distribution on every partition of any (possibly continuous) space $\mathcal{X}$, since the partition is itself a discrete space. The Dirichlet Process $\mathcal{DP}(\Theta; \alpha, M)$ is the unique distribution over the space of all possible distributions on $\mathcal{X}$, such that the relation

$$\Theta(\mathcal{X}_1), \Theta(\mathcal{X}_2), \ldots, \Theta(\mathcal{X}_n) \sim Dir(\alpha, M) \tag{73}$$

holds for every natural number n and every n-partition $\{\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_n\}$ of $\mathcal{X}$ [26], where we denote the Dirichlet process as $\mathcal{DP}(.)$.

At first glance, it may seem that Θ is a continuous distribution since M is continuous. However, Blackwell [7] showed that Dirichlet Processes are discrete as they consist of countably infinite point probability masses. This gives rise to the important property that values observed from a Dirichlet process previously have a non-zero probability of occurring again.

All the properties of the Dirichlet distribution, including the equivalence with the Polya urn scheme, also hold for the Dirichlet process. Indeed, an alternate method for obtaining the Dirichlet process is to consider the limit as the number of colors in the Polya

urn scheme tends to a continuum [8]. This limit yields an important formula called the Blackwell-MacQueen formula that forms the basis of the majority of algorithms for performing inference over Dirichlet processes. The formula is analogous to (72) in continuous spaces, and is given as

$$P(X_{N+1} = j \mid X_{1:N}) = \begin{cases} \frac{1}{\alpha+N}\sum_{i=1}^{N}\delta(X_i = j) & \exists k \leq N, \quad \text{s.t. } X_k = j \\ \frac{\alpha}{\alpha+N}M(j) & X_k \neq j, \quad \forall 1 \leq k \leq N \end{cases} \tag{74}$$

A.4 The Dirichlet Process Mixture Model

Consider a mixture model of the form $y_i \sim \sum_{i=1}^{k}\pi_i f(y \mid \theta_i)$. Hence y is distributed as a mixture of distributions having the same parametric form f but differing in their parameters. Also let all the parameters θ_i be drawn from the same distribution G_0. This mixture model can be expressed hierarchically as follows-

$$\begin{aligned} y_i \mid c_i, \Theta &\sim f(y \mid \theta_{c_i}) \\ c \mid \pi_{1:k} &\sim Discrete(\pi_1, \pi_2, \ldots, \pi_k) \\ \theta_i &\sim G_0(\theta) \\ \pi_1, \pi_2, \ldots, \pi_K &\sim \mathcal{D}ir(\alpha, M) \end{aligned} \tag{75}$$

Here c_i are the indicators or labels that assign the measurements y_i to a parameter value θ_{c_i} and π_i are the mixture coefficients that are drawn from a Dirichlet distribution. Given the mixture coefficients, the indicator variables are distributed multinomially (an individual label is discretely distributed). It is to be noted that the latent indicator variables are used here only to simplify notation. If the number of components in the mixture is known a priori, the parameters for each component can be drawn from G_0 beforehand, and then the Dirichlet distribution would be on the probability of selection of these parameters i.e., the set $\{\theta_1, \theta_2, \ldots, \theta_k\}$.

Let us now consider the limit of this model as $k \to \infty$. It can be seen that in the limit,

the Dirichlet distribution becomes a Dirichlet process with base measure M. For each indicator c_i drawn conditioned on all the previous $(i-1)$ indicators from the Multinomial distribution, there is a corresponding θ_i that is drawn from G_0. In the limit $k \to \infty$, the labels lose their meaning as the space of possible labels becomes continuous. We can discard the use of labels in the model and let the parameters be drawn from a Dirichlet process with base measure G_0 instead.

Hence, the DPM model is

$$\begin{aligned} y_i \mid \theta_i &\sim f(y \mid \theta_i) \\ \theta_i \mid G &\sim G(\theta) \\ G &\sim \mathcal{DP}(\alpha G_0(\theta)) \end{aligned} \tag{76}$$

where $\mathcal{DP}(\alpha_0 G_0)$ is the Dirichlet Process with base measure G_0 and spread α, and G is a random distribution drawn from the DP.

The alternate way to express the above argument is as follows. Using (71), we obtain the marginal distribution of c_i given $c_{1:i-1}$ as

$$P(c_i = c \mid c_1, c_2, \ldots, c_{i-1}) = \frac{n_{i,c} + m_c}{m_c + i - 1} \tag{77}$$

where m_c is the prior expectation of c using the measure M, and $n_{i,c}$ is the number of occurrences of c in the first $i-1$ indicator variables. As $K \to \infty$, the prior expectation of any one specific label tends to zero (the probability of any point in a continuous distribution is zero) and hence, the limit of the above prior reaches the following

$$P(c_i = c \mid c_1, c_2, \ldots, c_{i-1}, \alpha, M) = \begin{cases} \frac{n_{i,c}}{\alpha+i-1} & \exists j < i, s.t.\ c_j = c \\ \frac{\alpha}{\alpha+i-1} & \forall j < i, c_j \neq c \end{cases} \tag{78}$$

Now from (76), it can be seen that the marginal probability of θ_i given $\theta_{1:i-1}$ is given by the Blackwell-MacQueen Polya Urn formula (74).

$$P(\theta_i = \theta \mid \theta_1, \theta_2, \ldots, \theta_{i-1}, \alpha, G_0) = \begin{cases} \frac{1}{\alpha+i-1} \sum_{j=0}^{i-1} \delta(\theta - \theta_j) & \exists j < i, s.t.\ \theta_j = \theta \\ \frac{\alpha}{\alpha+i-1} G_0 & \forall j < i, \theta_j \neq \theta \end{cases} \tag{79}$$

Due to the correspondence between equations (78) and (79), it can be seen that in the limit $k \rightarrow \infty$, the model (75) and (76) are the same.

A mechanical though unintuitive method for testing the applicability of the DPM to a problem is as follows. Any parametric model for the measurements y_i described hierarchically as

$$\begin{aligned} y_i \mid \theta_i &\sim f(y \mid \theta_i) \\ \theta_i \mid \psi &\sim h(\theta \mid \psi) \end{aligned} \tag{80}$$

can be replaced with a DPM model of the form (76) by removing the assumption of the parametric prior h at the second stage and instead replacing it with a general distribution G that has a Dirichlet process prior [26]

A.5 Sampling using a DPM

Escobar [24] first showed that MCMC techniques, specifically Gibbs sampling, could be brought to bear on posterior density estimation if the Blackwell-MacQueen Polya Urn formulation of the DP is used. Consider (79) again, but now, we condition on not only $\theta_{1:i-1}$ but on all θ indexed from 1 to n except i, where n is some whole number. We denote this vector by $\theta^{(i-)}$. (Note-: We can only do this because samples from the DP are exchangeable, meaning that the joint distribution of the variables does not depend on the order in which they are considered).

Our aim is to find the posterior on θ_i , given a data instance y_i. The posterior can be calculated using Bayes law as

$$P(\theta_i \mid \theta^{(i-)}, y_i) \propto P(y_i \mid \theta_i) P(\theta_i \mid \theta^{(i-)}) \tag{81}$$

where all the probabilities are implicitly conditioned on the Dirichlet process parameters.

The prior on θ_i is obtained from (79) as

$$P(\theta_i = \theta \mid \theta^{(i-)}) = \frac{\alpha}{\alpha+n-1} G_0(\theta) + \frac{1}{\alpha+n-1} \sum_{\substack{j=1 \\ j \neq i}}^{n} \delta(\theta - \theta_j) \tag{82}$$

while the likelihood of the data is simply $f(y_i;\theta_i)$ from (76). The posterior is thus

$$P(\theta_i \mid \theta^{(-i)}, y_i) = b\alpha G_0(\theta_i) f(y_i;\theta_i) + b \sum_{\substack{j=1 \\ j \neq i}}^{n} f(y_i;\theta_j)\delta(\theta - \theta_j) \tag{83}$$

$$b = \left(\alpha q_0 + \sum_{\substack{j=1 \\ j \neq i}}^{n} f(y_i;\theta_j) \right)^{-1}$$

$$q_0 = \int_\theta G_0(\theta) f(y_i \mid \theta) \tag{84}$$

where b is a normalizing constant, and $\delta(\theta_i - \theta_j)$ is a point probability mass at θ_j.

It can be observed that q_0 is actually the marginal distribution of y_i and hence, is the inverse of the normalizing term in (81). (83) is often written in terms of the posterior $h(\theta_i \mid y_i) = \frac{G_0(\theta_i) f(y_i;\theta_i)}{\int_\theta G_0(\theta) f(y_i|\theta)}$ as

$$P(\theta_i \mid \theta^{(-i)}, y_i) = b\alpha q_0 h(\theta_i \mid y_i) + b \sum_{\substack{j=1 \\ j \neq i}}^{n} f(y_i;\theta_j)\delta(\theta_i - \theta_j) \tag{85}$$

This can also be written in a form that demonstrates the mixture nature of the marginal posterior on θ_i and also gives a simple algorithm for sampling from $\theta_i \mid \theta^{(i-)}, y_i$

$$P(\theta_i \mid \theta^{(-i)}, y_i) = \begin{cases} \theta_j & \text{with probability} \quad b f(y_i;\theta_j) \\ \sim h(\theta \mid y_i) & \text{with probability} \quad b\alpha q_0 \end{cases} \tag{86}$$

A Gibbs sampling algorithm using (86) can be easily designed to perform sampling on the space of θs.

DPMs can be categorized as being conjugate models or non-conjugate models. In a conjugate model, the distributions f and G_0 are conjugate and hence, the integration in the calculation of q_0 can be performed analytically. If this is not the case, then the DPM is said to have a non-conjugate prior and inference becomes much harder. Only recently has a satisfactory solution to this problem been found [15, 59].

A.6 Bayesian Clustering using DPMs

Consider a situation where we have N measurements $Y = \{y_i | i \in [1,N]\}$ that are distributed as a mixture density $P(y_i) = \sum_{i=1}^{k} \pi_i f(y;\theta_i)$ where the θ are the parameters of the distribution f and the π are the mixing coefficients. The number of components in the mixture, k, is unknown. However, it is known that each measurement y_i is generated from only one of the components of the mixture, i.e. given a specific set of parameters θ_i^*, $y_i \mid \theta_i^* \sim f(y_i;\theta_i^*)$. The parameters θ are in turn modeled hierarchically as $\theta \sim h(\psi)$. The problem is to classify or cluster the measurements with regard to the mixture component that generated it (or to the mixture component that it "belongs" to). Hence, each mixture component is associated with a disjoint subset of the set of measurements and the mixture components give rise to a partition of the set of measurements.

The above problem is the general statement of Bayesian model-based clustering with exchangeable measurements and labels inside a cluster. It is model-based since we assume a parametrized distribution, or model, for each cluster. It is exchangeable since the joint likelihood of a cluster does not depend on any ordering of the measurements or the subset labels. For more details on clustering and partition models, the reader is referred to [35], [53], and references therein.

This clustering problem could be solved with Reversible Jump MCMC as it involves inferring a mixture density [80]. However, when using this technique (or many others), the

distribution h has to be specified, and the parameters θ and hyper-parameters ψ have to be inferred. The parameter estimation, in particular, adds significantly to the complexity of the problem. Non-parametric estimation overcomes this problem by eliminating the need for parameters. In addition, DPMs do not assume any particular parametric form for h , but instead replace it with a general distribution with a Dirichlet process prior as explained in the next section.

A.7 An Example

I will illustrate partition sampling using DPMs using the example of partitioning N 2D points $R = \{y_i | i \in [i,N]\}$ that are Gaussian distributed, i.e. $P(r) = \sum_{i=1}^{k} \pi_i \mathcal{N}(\mu_i, I_2)$, where I_2 is the 2x2 identity matrix. Each number in R is generated from the one of the components of the mixture and hence, each set in the partition corresponds to a particular 2D Gaussian distribution. The mean of the Gaussian distribution corresponding to a set in the partition can be seen as the "true" value which is represented by the (noisy) measurements that make up the set. The problem can also be viewed as that of finding the clustering distribution of R given that the elements in R are distributed as Gaussians (but with different parameters).

Comparing with (76), it can be observed that in this case f is a bivariate Gaussian distribution with unknown mean but a known, constant covariance matrix equal to I_2. The base measure G_0 is taken to be the standard Gaussian distribution $\mathcal{N}(0, I_2)$. We can then define our model to be the following

$$\begin{aligned} y_i \mid \mu_i &\sim \mathcal{N}(\mu_i, I_2) \\ \mu_i &\sim G(\mu) \\ G &\sim \mathcal{DP}(\alpha G_0(\mu)) \\ G_0 &= \mathcal{N}(0, I_2) \end{aligned} \tag{87}$$

Note that it is possible to extend the model to include parametrized distributions for the

case of unknown measurement covariance, α , and G_0. This is not done here to keep the exposition simple. An unknown measurement covariance can be handled by a Normal-Wishart prior model ([53] has the 2D case), while estimating the DP parameters is given in [24].

Performing the calculations using (83), we find

$$q_0 = \frac{1}{\pi}\exp -\frac{y_i^T y_i}{4}$$

and

$$h(\mu \mid y_i) = \mathcal{N}(\frac{1}{2}y_i, \frac{1}{2}I_2)$$

and hence, our Gibbs sampler becomes (from (86))

$$P(\mu_i \mid \mu^{(-i)}, y_i) = \begin{cases} \mu_j & \text{with probability proportional to } f(y_i;\mu_j) \\ \sim h(\mu \mid y_i) & \text{with probability proportional to } \alpha q_0 \end{cases}$$

We initialize the Gibbs sampler by consider each of the n input instances $y_{1:n}$ as being in its own set, i.e. $\mu_i^{(0)} = y_i$. Subsequently, the jth step of the Gibbs sampling is done in the following way

$$\begin{aligned}
&\text{Sample } \mu_1^{(j)} \text{from} \quad \mu_1 \mid \mu_2 = \mu_2^{(j-1)} \quad , \mu_3 = \mu_3^{(j-1)}, \ldots, \mu_n = \mu_n^{(j-1)} \\
&\text{Sample } \mu_2^{(j)} \text{from} \quad \mu_2 \mid \mu_1 = \mu_1^{(j)} \quad , \mu_3 = \mu_3^{(j-1)}, \ldots, \mu_n = \mu_n^{(j-1)} \\
&\qquad\qquad\qquad\qquad \vdots \\
&\text{Sample } \mu_n^{(j)} \text{from} \quad \mu_n \mid \mu_1 = \mu_1^{(j)} \quad , \mu_2 = \mu_2^{(j)}, \ldots, \mu_{n-1} = \mu_{n-1}^{(j)}
\end{aligned}$$

A sample from the Gibbs sampler, with the DP parameter α set to unity, is shown in Figure 71. Note that the various clusters at different scales and locations are discovered effectively. The centers of the clusters at the corners are slightly displaced towards the origin (center) due to the relatively tight base DPM distribution G_0 centered at the origin. "Loosening" up G_0 by increasing its covariance will remove this effect.

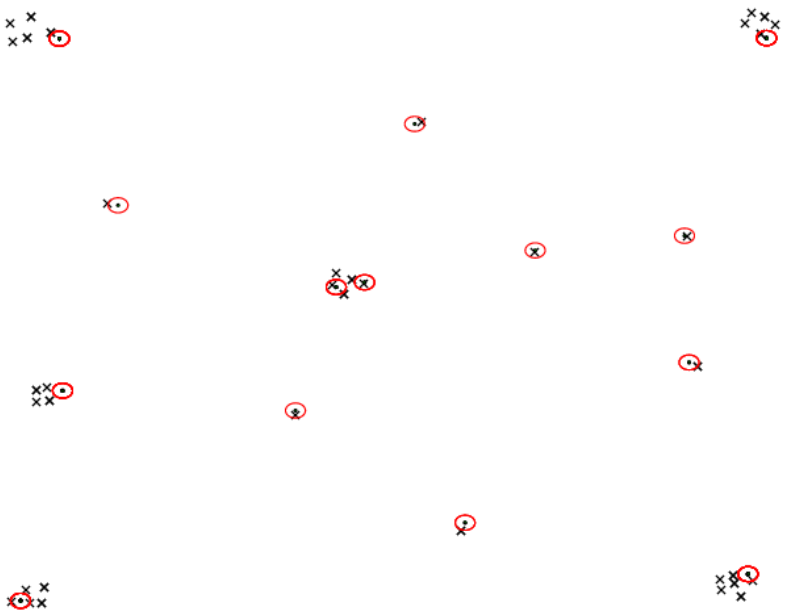


Figure 71: A sample from the DPM with a 2D Gaussian model prior, obtained using Gibbs sampling as described above. The crosses represent data points while the red circles centered on black dots represent the cluster covariances (fixed) and means.

APPENDIX B

THE LAPLACE APPROXIMATION USING LEVENBERG MARQUARDT OPTIMIZATION

In Bayesian analysis, a frequently encountered situation is the need to integrate over a complex distribution for which even the analytical form may not be available. The standard method in such cases is to employ Monte Carlo techniques to sample from the distribution and subsequently, replace the integral by the appropriate Monte Carlo sum. The Monte Carlo approximating becomes increasingly accurate with the number of samples.

However, sampling techniques are slow so that an analytical approximation that enables closed form integration is often a requirement. The Laplace approximation is one of the simplest ways to create such an analytical approximation. The approximation assumes that the distribution of interest $p(x)$ has a peak at $x = x^\star$ and replaces it by a Gaussian distribution $q(x)$ centered at $x^\star$.

We assume that $x \in \Re^n$ is the state vector with elements $(x_1, x_2, \ldots, x_n)$. Expanding the logarithm of the distribution $p(x)$ about $x^\star$ using a Taylor series, we get

$$\log p(x) \quad = \quad \log p(x^\star) - \frac{1}{2}(x - x^\star)^T A (x - x^\star) + \ldots$$

where the first order term has been omitted since the gradient is zero at the maximum. A is the matrix of second derivatives of $-\log p(x)$ at $x^\star$, i.e. the Hessian, defined as

$$A \stackrel{\Delta}{=} -\nabla^2 \log p(x) \quad \stackrel{\Delta}{=} \quad -\frac{\partial^2}{\partial^2 x_{ij}} \log p(x)_{\,x=x^\star}|$$

Truncating the Taylor series to the second term, we obtain the unnormalized Gaussian approximation to $p(x)$

$$q(x) \quad \propto \quad p(x^\star) \exp\left(-\frac{1}{2}(x - x^\star)^T A (x - x^\star) \right)$$

The integral $\int p(x)$ is now approximated by $\int q(x)$ and is equal to the normalization constant for a Gaussian, so that

$$\int p(x) \approx \int q(x) \quad = \quad p(x^\star)\sqrt{\left|\frac{2\pi}{A}\right|}$$

Note that the covariance of the approximate Gaussian distribution is given as $\Sigma = A^{-1}$.

In general, the maximum about which the Laplace approximation is done, is found by performing an optimization. We use the Levenberg-Marquardt (LM) optimization algorithm that, in addition to computing the optimum, also gives an estimate of the Hessian matrix. Obtaining the Hessian matrix is explained below.

B.1 Computing the Hessian Using the Levenberg-Marquardt Algorithm

The problem for which the LM algorithm provides a solution is called *Nonlinear Least Squares Minimization*. This implies that the function to be minimized is of the following special form :

$$f(x) = \frac{1}{2}\sum_{j=1}^{m} r_j^2(x)$$

where $x = (x_1, x_2, \ldots, x_n)$ is a vector, and each r_j is a function from $\Re^n$ to $\Re$. The r_j are referred to as a *residuals* and it is assumed that $m \geq n$.

To make matters easier, f is represented as a *residual vector* $r : \Re^n \rightarrow \Re^m$ defined by

$$r(x) = (r_1(x), r_2(x), \cdots, r_m(x))$$

Now, f can be rewritten as $f(x) = \frac{1}{2} \parallel r(x) \parallel^2$. The derivatives of f can be written using the Jacobian matrix J of r w.r.t x defined as $J(x) = \frac{\partial r_j}{\partial x_i}$, $1 \leq j \leq m$, $1 \leq i \leq n$.

Let us first consider the linear case where every r_i function is linear. Here, the Jacobian is constant and we can represent r as a hyperplane through space, so that f is given by the quadratic $f(x) = \frac{1}{2}\|Jx + r(0)\|^2$. We also get $\nabla f(x) = J^T(Jx + r)$ and $\nabla^2 f(x) = J^T J$. Solving for the minimum by setting $\nabla f(x) = 0$, we obtain $x_{min} = -(J^T J)^{-1} J^T r$, which is the solution to the set of *normal equations*.

Returning to the general, non-linear case, we have

$$\nabla f(x) = \sum_{j=1}^{m} r_j(x) \nabla r_j(x) = J(x)^T r(x) \tag{88}$$

$$\nabla^2 f(x) = J(x)^T J(x) + \sum_{j=1}^{m} r_j(x) \nabla^2 r_j(x) \tag{89}$$

The distinctive property of least-squares problems is that given the Jacobian matrix J, we can essentially get the Hessian ($\nabla^2 f(x)$) for free if it is possible to approximate the r_js by linear functions ($\nabla^2 r_j(x)$ are small) or the residuals ($r_j(x)$) themselves are small. The Hessian in this case simply becomes

$$\nabla^2 f(x) = J(x)^T J(x) \tag{90}$$

which is the same as for the linear case.

The common approximation used here is one of near-linearity of the r_is near the solution so that $\nabla^2 r_j(x)$ are small. It is also important to note that (90) is only valid if the residuals are small. Large residual problems cannot be solved using the quadratic approximation, and consequently, the quality of the Hessian approximation is poor in such cases.

APPENDIX C

AN APPEARANCE-BASED DATA-DRIVEN PROPOSAL DISTRIBUTION

The trade-off with data-driven proposal distributions is to propose the most likely samples at the least cost. Hence, data-driven proposals are usually chosen to utilize the most informative measurement stream. While proposals that incorporate measurements from more than one stream are possible, they are not usually preferred since the cost of each proposal increases dramatically in such cases.

In many cases, especially when using SIFT histograms to model places, appearance provides a strong measurement. We have provided an odometry-based proposal distribution in Algorithm 3. We now provide an appearance-based proposal for cases when appearance measurements are more informative than odometry.

An appearance model for SIFT histograms is presented in Section 4.3. This is a clustering model for histograms based on the Multivariate Polya distribution. While a straightforward proposal distribution would propose clusters that have a high probability according to this distribution, computing the distribution parameters for each of these proposals is expensive since it requires a fixed point iteration. Instead, we approximate the clustering using a simple metric distance between the histograms.

We use the Bhattacharya distance to perform metric clustering among the histograms. The Bhattacharya distance between two discrete normalized distributions p and q is given as

$$d_B(p,q) = \sum_x \sqrt{p(x)q(x)} \tag{91}$$

The Bhattacharya distance always lies in the interval $[0,1]$ and is a divergence measure

Algorithm 9 Scheme for computing inter- and intra-cluster distances for use in the proposal distribution.

1. **Computing the inter-cluster distance between sets R and S:**
   (a) For each of the normalized histograms $p \in R$ and $s \in S$, compute the pairwise Bhattacharya distance D_{ps}
   (b) Compute the average distance $D = \frac{1}{|R||S|} \sum_{p \in R, s \in S} D_{ps}$. This is the inter-cluster distance.
2. **Computing the intra-cluster distance for set S:**
   (a) For each pair of normalized histograms $p, q \in S$, compute the pairwise Bhattacharya distance D_{pq}
   (b) Compute the average distance $D = \frac{1}{\binom{|S|}{2}} \sum_{p,q \in S} D_{pq}$. This is the intra-cluster distance.

that signifies dissimilarity in the sense that it has a maximum value of unity if $p = q$. A geometric interpretation of this metric is as the cosine of the angle between two high dimensional vectors represented by the discrete distributions.

The proposal is based on split-merge operations as before. For the split move, a candidate set is selected from the partition and all the possible splits of the set are evaluated. The splits are sampled according to the probability of the clustering they create. The log probability of the clustering is computed as the difference between the total intra-cluster distance, which have to be maximized, and the total inter-cluster distance, which have to be minimized for a good clustering. All distances are computed using the Bhattacharya metric. The scheme for evaluating a clustering is made explicit in Algorithm 9.

The merge step is selected by evaluating all possible merges of the sets and sampling from the discrete distribution of the resulting clusterings. The proposal ratio is computed as in Section 3.10.1 with the appropriate probabilities.

The complete data-driven proposal based on appearance measurements in the form of SIFT measurements is given in Algorithm 10.

Algorithm 10 Data-driven Proposal Distribution using Appearance

1. Select a merge or a split with probability $\left\{\frac{N_M}{N_M+N_S}, \frac{N_S}{N_M+N_S}\right\}$
2. **Merge move:**
   (a) Obtain a discrete distribution on all merges in T as follows. For each pair of sets R and S in T
      i. Compute the inter-cluster distance D_1 between set $R \bigcup S$ and all the sets in $T - \{R\} - \{S\}$. Compute the intra-cluster distance D_2 of the set $R \bigcup S$.
      ii. Compute the probability of the merge as that of proposing the new topology $T'_{RS} = (T - \{R\} - \{S\}) \cup \{R \cup S\}$ as $Q(T \rightarrow T'_{RS}) = \frac{\exp(-D_1+D_2)}{N_M+N_S}$
   (b) From the discrete distribution on merges computed above, sample a merge move. Let the new topology proposed be T'.
   (c) Probability of the reverse move $Q(T' \rightarrow T)$ is obtained from the reverse case 3(c), hence $r = \frac{N_M+N_S}{N'_M+N'_S} \exp\left((D'_2 - D'_1) - (D_2 - D_1)\right)$, where N'_M and N'_s are the number of merge and split moves possible from T', and D'_1 and D'_2 are computed for T' from 3(a)
3. **Split move:**
   (a) Select a non-singleton set U at random from T and evaluate all splits of U into two sets R and S as follows.
      i. Compute the total inter-cluster distance between the set R and all the sets in $T - R$, and also the corresponding distance between S and $T - R - S$. Let the total inter-cluster distance be D_1. Compute the intra-cluster distance of R and S, and call the sum of these distances as D_2.
      ii. Compute the probability of the split as that of proposing the new topology $T' = (T - \{U\}) \cup \{R, S\}$ as $Q(T \rightarrow T') = \frac{\exp(-D_1+D_2)}{N_M+N_S}$
   (b) From the discrete distribution on splits computed above, sample a split move. Let the new topology proposed be T'.
   (c) Probability of the reverse move $Q(T' \rightarrow T)$ is obtained from the reverse case 2(b), hence $r = \frac{N_M+N_S}{N'_M+N'_S} \exp\left((D'_2 - D'_1) - (D_2 - D_1)\right)$, where N'_M and N'_s are the number of merges and splits possible in T' as defined in 2(b), and D'_1 and D'_2 are computed for T' from 2(a)

REFERENCES

[1] *Proc. 19th AAAI National Conference on AI*, (Edmonton, Alberta, Canada), 2002.

[2] "Modeling word burstiness using the dirichlet distribution," in *Intl. Conf. on Machine Learning (ICML)*, pp. 545–552, 2005.

[3] ATTIAS, H., "Planning by probabilistic inference," in *Proceedings of the 9th International Conference on Artificial Intelligence and Statistics*, 2003.

[4] AYCARD, O., CHARPILLET, F., FOHR, D., and MARI, J., "Place learning and recognition using hidden markov models," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pp. 1741–1746, 1997.

[5] BEAL, M., *Variational algorithms for approximate Bayesian inference*. PhD thesis, Gatsby Computational Neuroscience Unit, University College London, 2003.

[6] BEESON, P., JONG, N. K., and KUIPERS, B., "Towards autonomous topological place detection using the Extended Voronoi Graph," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2005.

[7] BLACKWELL, D., "Discreteness of Ferguson selections," *Annals of Statistics*, vol. 1, pp. 356–358, 1973.

[8] BLACKWELL, D. and MACQUEEN, J., "Ferguson distributions via polya urn schemes," *Annals of Statistics*, vol. 1, pp. 353–355, 1973.

[9] BOSSE, M., NEWMAN, P., LEONARD, J., and TELLER, S., "Simultaneous localization and map building in large-scale cyclic environments using the Atlas framework," *Intl. J. of Robotics Research*, vol. 23, pp. 1113–1139, December 2004.

[10] CASELLA, G. and ROBERT, C., "Rao-Blackwellisation of sampling schemes," *Biometrika*, vol. 83, pp. 81–94, March 1996.

[11] CASTELLANOS, J. and TARDOS, J., *Mobile Robot Localization and Map Building: A Multisensor Fusion Approach*. Boston, MA: Kluwer Academic Publishers, 2000.

[12] CHEN, Y. and MEDIONI, G., "Object modelling by registration of multiple range images," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pp. 2724–2729, 1991.

[13] CHOSET, H. and NAGATANI, K., "Topological simultaneous localization and mapping (SLAM): toward exact localization without explicit localization," *IEEE Trans. Robot. Automat.*, vol. 17, pp. 125 – 137, April 2001.

[14] CONSTANTINI, D., DONADIO, S., GARIBALDI, U., and VIARENGO, P., "Herding and clustering in economics: the Yule-Zipf-Simon model." Working Draft, 2004.

[15] DAMIEN, P., WAKEFIELD, J. C., and WALKER, S. G., "Gibbs sampling for Bayesian nonconjugate and hierarchical models using auxiliary variables," *Journal of the Royal Statistical Society Series B*, vol. 61, pp. 331–344, 1999.

[16] DEDEOGLU, G., MATARIC, M., and SUKHATME, G., "Incremental, online topological map building with a mobile robot," in *Proceedings of Mobile Robots*, 1999.

[17] DELLAERT, F., "Square Root SAM: Simultaneous location and mapping via square root information smoothing," in *Robotics: Science and Systems (RSS)*, 2005.

[18] DISSANAYAKE, G., DURRANT-WHYTE, H., and BAILEY, T., "A computationally efficient solution to the simultaneous localisation and map building (SLAM) problem." Working notes of ICRA'2000 Workshop W4: Mobile Robot Navigation and Mapping, April 2000.

[19] DOUCET, A., S.GODSILL, and ANDRIEU, C., "On sequential monte carlo sampling methods for Bayesian filtering," *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.

[20] DUDEK, G., HADJRES, S., and FREEDMAN, P., "Using local information in a non-local way for mapping graph-like worlds," in *IJCAI*, pp. 1639–1645, 1993.

[21] DUDEK, G. and JUGESSUR, D., "Robust place recognition using local appearance based methods," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pp. 1030–1035, 2000.

[22] DURRANT-WHYTE, H., MAJUNDER, S., THRUN, S., DE BATTISTA, M., and SCHEDING, S., "A Bayesian algorithm for simultaneous localization and map building," in *Proceedings of the 10th International Symposium of Robotics Research*, 2001.

[23] ELKAN, C., "Clustering documents with an exponential-family approximation of the dirichlet compound multinomial distribution," in *Intl. Conf. on Machine Learning (ICML)*, pp. 289–296, 2006.

[24] ESCOBAR, M. D., "Estimating the means of several normal populations by nonparametric estimation of the distribution of the means." Unpublished dissertation, Yale University, 1988.

[25] EUSTICE, R., SINGH, H., and LEONARD, J., "Exactly sparse delayed-state filters," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, (Barcelona, Spain), pp. 2428–2435, April 2005.

[26] FERGUSON, T. S., "A Bayesian analysis of some nonparametric problems," *Annals of Statistics*, vol. 1, pp. 209–230, 1973.

[27] GELMAN, A., CARLIN, J., STERN, H., and RUBIN, D., *Bayesian Data Analysis*. Chapman and Hall, 1995.

[28] GEYER, C. J. and THOMPSON, E. A., "Annealing Markov chain Monte Carlo with applications to ancestral inference," *Journal of the American Statistical Association*, vol. 90, pp. 909–920, 1995.

[29] GILKS, W., RICHARDSON, S., and SPIEGELHALTER, D., eds., *Markov chain Monte Carlo in practice*. Chapman and Hall, 1996.

[30] GOEDEME, T., NUTTIN, M., TUYTELAARS, T., and GOOL, L. V., "Omnidirectional vision based topological navigation," *International Journal of Computer Vision. Special Issue: Joint Issue of IJCV and IJRR on Vision and Robotics*, vol. 74, pp. 219–236, 2007.

[31] GRIEWANK, A., "On Automatic Differentiation," in *Mathematical Programming: Recent Developments and Applications* (IRI, M. and TANABE, K., eds.), pp. 83–108, Kluwer Academic Publishers, 1989.

[32] GUTIERREZ-OSUNA, R. and LUO, R. C., "Lola: Probabilistic navigation for topological maps," *AI Magazine*, vol. 17, no. 1, pp. 55–62, 1996.

[33] HÄHNEL, D., BURGARD, W., FOX, D., and THRUN, S., "A highly efficient FastSLAM algorithm for generating cyclic maps of large-scale environments from raw laser range measurements," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2003.

[34] HÄHNEL, D., BURGARD, W., WEGBREIT, B., and THRUN, S., "Towards lazy data association in SLAM," in *Proceedings of the 11th International Symposium of Robotics Research (ISRR'03)*, (Sienna, Italy), Springer, 2003.

[35] HARTIGAN, J. A., "Partition models," *Communications in Statistics, Part A - Theory and Methods*, vol. 19, pp. 2745–2756, 1990.

[36] ISHIGURO, H., NG, K. C., CAPELLA, R., and TRIVEDI, M. M., "Omnidirectional image-based modeling: three approaches to approximated plenoptic representations," *Machine Vision and Applications*, vol. 14, no. 2, pp. 94–102, 2003.

[37] ITTI, L. and BALDI, P., "A principled approach to detecting surprising events in video," in *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 631–637, 2005.

[38] ITTI, L. and BALDI, P., "Bayesian surprise attracts human attention," in *Advances in Neural Information Processing Systems (NIPS)*, (Cambridge, MA), pp. 1–8, MIT Press, 2006.

[39] JAIN, S. and NEAL, R., "A split-merge Markov chain Monte Carlo procedure for the dirichlet process mixture model," *Journal of Computational and Graphical Statistics*, vol. 13, pp. 158–182, March 2004.

[40] JOHNSON, N. L. and KOTZ, S., *Urn Models and their Applications*. John Wiley and Sons, 1977.

[41] JOLLIFFE, I. T., *Principal Component Analysis*. Springer, 1986.

[42] KAELBLING, L., CASSANDRA, A., and KURIEN, J., "Acting under uncertainty: Discrete Bayesian models for mobile-robot navigation," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 1996.

[43] KAESS, M., RANGANATHAN, A., and DELLAERT, F., "iSAM: Fast incremental smoothing and mapping with efficient data association," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, (Rome, Italy), pp. 1670–1677, April 2007.

[44] KALMAN, R. E., "A new approach to linear filtering and prediction problems," *Trans. ASME, Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960.

[45] KORTENKAMP, D. and WEYMOUTH, T., "Topological mapping for mobile robots using a combination of sonar and vision sensing," in *Proceedings of the Twelfth National Conference on Artificial Intelligence*, pp. 979–984, 1994.

[46] KOSECKA, J. and LI, F., "Vision based markov localization," in *ICRA*, 2004.

[47] KRÃ¶SE, B., VLASSIS, N., BUNSCHOTEN, R., and MOTOMURA, Y., "A probabilistic model for appearance-based robot localization," *Image and Vision Computing*, vol. 19, no. 6, pp. 381–391, 2001.

[48] KUIPERS, B., "The cognitive map: Could it have been any other way?," in *Spatial Orientation: Theory, Research, and Application* (JR., H. L. P. and ACREDOLO, L. P., eds.), New York: Plenum Press, 1983.

[49] KUIPERS, B., "Modeling spatial knowledge," in *Advances in Spatial Reasoning (Volume 2)* (CHEN, S., ed.), pp. 171–198, The University of Chicago Press, 1990.

[50] KUIPERS, B., "The Spatial Semantic Hierarchy," *Artificial Intelligence*, vol. 119, pp. 191–233, 2000.

[51] KUIPERS, B. and BEESON, P., "Bootstrap learning for place recognition," in *Proc. 19^{th} AAAI National Conference on AI* [1], pp. 174–180.

[52] KUIPERS, B. and BYUN, Y.-T., "A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations," *Journal of Robotics and Autonomous Systems*, vol. 8, pp. 47–63, 1991.

[53] LAU, J. W. and GREEN, P. J., "Bayesian model based clustering procedures." Under review, 2006.

[54] LEONARD, J. and DURRANT-WHYTE, H., "Simultaneous map building and localization for an autonomous mobile robot," in *IEEE Int. Workshop on Intelligent Robots and Systems*, pp. 1442–1447, 1991.

[55] LISIEN, B., MORALES, D., SILVER, D., KANTOR, G., REKLEITIS, I., and CHOSET, H., "Hierarchical simultaneous localization and mapping," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pp. 448–453, 2003.

[56] LOWE, D., "Distinctive image features from scale-invariant keypoints," *Intl. J. of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[57] LU, F. and MILIOS, E., "Globally consistent range scan alignment for environment mapping," *Autonomous Robots*, pp. 333–349, April 1997.

[58] LYNCH, K., *The Image of the City*. MIT Press, 1971.

[59] MACEACHERN, S. N. and MULLER, P., "Estimating mixture of dirichlet process models," *Journal of Computational and Graphical Statistics*, vol. 7, pp. 223–238, 1998.

[60] MATARIĆ, M. J., "A distributed model for mobile robot environment-learning and navigation," Master's thesis, MIT, Artificial Intelligence Laboratory, Cambridge, January 1990. Also available as MIT AI Lab Tech Report AITR1228.

[61] MATAS, J., CHUM, O., URBAN, M., and PAJDLA, T., "Robust wide baseline stereo from maximally stable extremal regions," in *British Machine Vision Conf. (BMVC)*, pp. 414–431, 2002.

[62] MENEGATTI, E., MAEDA, T., and ISHIGURO, H., "Image-based memory for robot navigation using properties of the omnidirectional images," *Journal of Robotics and Autonomous Systems*, vol. 47, no. 4, pp. 251–267, 2004.

[63] MENEGATTI, E., ZOCCARATO, M., PAGELLO, E., and ISHIGURO, H., "Image-based Monte-Carlo localisation with omnidirectional images," *Journal of Robotics and Autonomous Systems*, vol. 48, pp. 17–30, August 2004.

[64] MIKOLAJCZYK, K. and SCHMID, C., "An affine invariant interest point detector," in *Eur. Conf. on Computer Vision (ECCV)*, vol. 1, pp. 128–142, 2002.

[65] MINKA, T., "Estimating a dirichlet distribution." 2003.

[66] MODAYIL, J., BEESON, P., and KUIPERS, B., "Using the topological skeleton for scalable global metrical map-building," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2004.

[67] MONTEMERLO, M. and THRUN, S., "Simultaneous localization and mapping with unknown data association using FastSLAM," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2003.

[68] MONTEMERLO, M., THRUN, S., KOLLER, D., and WEGBREIT, B., "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proc. 19^{th} AAAI National Conference on AI* [1].

[69] MOZOS, O. M., JENSFELT, P., ZENDER, H., KRUIJFF, G.-J. M., and BURGARD, W., "From labels to semantics: An integrated system for conceptual spatial representations of indoor environments for mobile robots," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2007.

[70] MURPHY, K. and RUSSELL, S., "Rao-Blackwellised particle filtering for dynamic Bayesian networks," in *Sequential Monte Carlo Methods in Practice* (DOUCET, A., DE FREITAS, N., and GORDON, N., eds.), New York: Springer-Verlag, January 2001.

[71] MYKLAND, P., TIERNEY, L., and YU, B., "Regeneration in Markov chain samplers," *Journal of the American Statistical Association*, vol. 90, pp. 233–241, 1995.

[72] NEWMAN, P., COLE, D., and HO, K., "Outdoor SLAM using visual appearance and laser ranging," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, 2006.

[73] NIJENHUIS, A. and WILF, H., *Combinatorial Algorithms*. Academic Press, 2 ed., 1978.

[74] PIERCE, D. and KUIPERS, B., "Map learning with uninterpreted sensors and effectors," *Artificial Intelligence*, vol. 92, pp. 169–229, 1997.

[75] RAMOS, F., UPCROFT, B., KUMAR, S., and DURRANT-WHYTE, H., "A bayesian approach for place recognition," in *IJCAI Workshop on Reasoning with Uncertainty in Robotics (RUR-05)*, 2005.

[76] RANGANATHAN, A. and DELLAERT, F., "Probabilistic Topological Mapping for Mobile Robots using Urn Models," Tech. Rep. GIT-GVU-07-03, GVU, College of Computing, 2007.

[77] RANGANATHAN, A., KAESS, M., and DELLAERT, F., "Loopy SAM," in *Intl. Joint Conf. on Artificial Intelligence (IJCAI)*, (Hyderabad, India), pp. 2191–2196, 2007.

[78] RANGANATHAN, A., MENEGATTI, E., and DELLAERT, F., "Bayesian inference in the space of topological maps," *IEEE Trans. Robotics*, vol. 22, no. 1, pp. 92–107, 2006.

[79] REMOLINA, E. and KUIPERS, B., "Towards a general theory of topological maps," *Artificial Intelligence*, vol. 152, no. 1, pp. 47–104, 2004.

[80] RICHARDSON, S. and GREEN, P. J., "On Bayesian analysis of mixtures with an unknown number of components (with discussion)," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 59, pp. 731–792, 1997.

[81] SAVELLI, F. and KUIPERS, B., "Loop-closing and planarity in topological map-building," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2004.

[82] SCHROTER, D., WEBER, T., BEETZ, M., and RADIG, B., "Detection and classification of gateways for the acquisition of structured robot maps," in *Proceedings of 26th Pattern Recognition Symposium (DAGM)*, 2004.

[83] SE, S., LOWE, D., and LITTLE, J., "Local and global localization for mobile robots using visual landmarks," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2001.

[84] SE, S., LOWE, D., and LITTLE, J., "Global localization using distinctive visual features," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, pp. 226–231, 2002.

[85] SHATKAY, H. and KAELBLING, L., "Learning topological maps with weak local odometric information," in *Proceedings of IJCAI-97*, pp. 920–929, 1997.

[86] SIMMONS, R. and KOENIG, S., "Probabilistic robot navigation in partially observable environments," in *Proc. International Joint Conference on Artificial Intelligence*, pp. 1080 – 1087, 1995.

[87] SMITH, R. and CHEESEMAN, P., "On the representation and estimation of spatial uncertainty," *Intl. J. of Robotics Research*, vol. 5, no. 4, pp. 56–68, 1987.

[88] SMITH, R., SELF, M., and CHEESEMAN, P., "A stochastic map for uncertain spatial relationships," in *Int. Symp on Robotics Research*, 1987.

[89] SMITH, R., SELF, M., and CHEESEMAN, P., "Estimating uncertain spatial relationships in Robotics," in *Autonomous Robot Vehicles* (COX, I. and WILFONG, G., eds.), pp. 167–193, Springer-Verlag, 1990.

[90] SMITH, R., SELF, M., and CHEESEMAN, P., "A stochastic map for uncertain spatial relationships," in *Autonomous Mobile Robots: Perception, Mapping, and Navigation (Vol. 1)* (IYENGAR, S. S. and ELFES, A., eds.), pp. 323–330, Los Alamitos, CA: IEEE Computer Society Press, 1991.

[91] TAPUS, A., *Topological SLAM - Simultaneous Localization and Mapping with Fingerprints of Places*. PhD thesis, Swiss Federal Institute of Technology Lausanne (EPFL), 2005.

[92] TAPUS, A., TOMATIS, N., and SIEGWART, R., "Topological global localization and mapping with fingerprint and uncertainty," in *Proceedings of the International Symposium on Experimental Robotics*, 2004.

[93] TENENBAUM, J. B., DE SILVA, V., and LANGFORD, J. C., "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290 (5500), pp. 2319–2323, 2000.

[94] THRUN, S., "Learning metric-topological maps for indoor mobile robot navigation," *Artificial Intelligence*, vol. 99, no. 1, pp. 21–71, 1998.

[95] THRUN, S., FOX, D., and BURGARD, W., "A probabilistic approach to concurrent mapping and localization for mobile robots," *Machine learning*, vol. 31, pp. 29–53, 1998.

[96] THRUN, S., GUTMANN, S., FOX, D., BURGARD, W., and KUIPERS, B., "Integrating topological and metric maps for mobile robot navigation: A statistical approach," in *Proc.* 15^{th} *AAAI National Conference on AI*, pp. 989–995, 1998.

[97] THRUN, S., LIU, Y., KOLLER, D., NG, A., GHAHRAMANI, Z., and DURRANT-WHYTE, H., "Simultaneous localization and mapping with sparse extended information filters," *Intl. J. of Robotics Research*, vol. 23, no. 7-8, pp. 693–716, 2004.

[98] TIERNEY, L. and KADANE, J. B., "Accurate approximations for posterior moments and marginal distributions," *Journal of the American Statistical Association*, vol. 81, pp. 82–86, 1986.

[99] TOLMAN, E. C., "Cognitive Maps in Rats and Man," in *Behavior and Psychological Man*, University of California Press, 1951.

[100] TOMATIS, N., NOURBAKHSH, I., and SIEGWART, R., "Hybrid simultaneous localization and map building: Closing the loop with multi-hypotheses tracking," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, pp. 2749–2754, 2002.

[101] TU, Z. and ZHU, S., "Image segmentation by data-driven Markov chain Monte Carlo," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 24, no. 5, pp. 657–673, 2002.

[102] ULRICH, I. and NOURBAKHSH, I., "Appearance-based place recognition for topological localization," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, vol. 2, pp. 1023 – 1029, April 2000.

[103] VALGREN, C., LILIENTHAL, A., and DUCKETT, T., "Incremental topological mapping using omnidirectional vision," in *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2006.

[104] VAN VEEN, H., DISTLER, H., BRAUN, S., and BULTHOFF, H., "Navigating through a virtual city: Using virtual reality technology to study human action and perception," *Future Generation Computer Systems*, vol. 14, pp. 231–242, 1998.

[105] YAMAUCHI, B. and BEER, R., "Spatial learning for navigation in dynamic environments," *IEEE Transactions on Systems, Man, and Cybernetics-Part B, Special Issue on Learning Autonomous Robots*, vol. 26, pp. 496–505, 1996.

[106] YAMAUCHI, B. and LANGLEY, P., "Place recognition in dynamic environments," *Journal of Robotic Systems*, vol. 14, pp. 107–120, February 1997. Special Issue on Mobile Robots.