

Observable 2D SLAM and Evidential Occupancy Grids

by

Sindhu Radhakrishnan

A Thesis submitted to
the Faculty of Graduate Studies and Research
in partial fulfilment of
the requirements for the degree of
Master of Applied Science
in

System and Computer Engineering
Carleton University
Ottawa, Ontario, Canada
April

Copyright ©

2014 - Sindhu Radhakrishnan

Abstract

The two main challenges offered by Simultaneous Localization and Mapping (SLAM) are that of observability and extending state estimation to exploration. This thesis explores and uses solutions to render the SLAM problem observable, by proposing the Reconfigurable Extended Kalman Filter (EKF) that addresses imposing observability, maintaining observability and choice of observability constraints. Additionally, Bayesian theory and Dempster-Shafer theory of evidential reasoning are analyzed, and Occupancy grid based maps based on Dempster-Shafer theory of evidential reasoning are created and analyzed in large environment for their potential use in exploration and obstacle avoidance. Tackling both issues with different algorithms yield better solutions to the challenges offered by robotic exploration, and this is demonstrated through simulation results in representative environments.

Acknowledgments

To my amazing parents, who taught me the value of hard work, patience and perseverance.

I am very grateful to my advisor, Dr.V. Aitken for being a kind and patient teacher. Dr.L.Tabrizi, whose teaching left me in awe of control systems, has been and will continue to be an inspiring role model. I will always cherish how Rytis.V gently pushed me to work harder, learn better, and to see things from a new perspective. Each one of you had an unforgettable influence on my life and I'm very thankful for that.

Contents

Abstract	ii
Acknowledgments	i
List of Tables	v
List of Figures	vi
1 Advent of Mapping and Localization	1
1.1 Thesis Statement	4
1.2 SLAM: An Introduction	6
1.3 SLAM: A brief history	7
1.4 SLAM: Evolution of Approaches	8
1.5 Mapping and Exploration: An Introduction	11
1.6 Mapping and Exploration: A brief history	12
1.7 Mapping and Exploration: Evolution of Approaches	13
1.8 Thesis Overview	14
2 Theoretical Background	16
2.1 Problem Formulation	16
2.1.1 Vehicle Process Model	16
2.1.2 Landmark Process Model	20

2.1.3	Measurement Model	21
2.1.4	Jacobians of the Vehicle and Measurement Models	23
2.1.5	Summary of the SLAM problem	25
2.2	Extended Kalman Filter	26
2.2.1	Introduction to the discrete time Linear Kalman Filter	26
2.2.2	Introduction to the discrete time Extended Kalman Filter	30
2.3	Observability	32
2.3.1	Observability in linear and non-linear systems	33
2.3.2	Fisher’s Information and Cramer-Rao Lower Bounds	35
2.3.3	Requirements for observable 2D SLAM	38
2.3.4	Imposing Observability	40
2.4	Dempster-Shafer Evidential Theory and Occupancy Grids	44
2.4.1	Introduction to DS theory	45
2.4.2	Comparison with Bayesian theory	49
3	Observable EKF SLAM Simulations and Results	53
3.1	Simulation parameters	54
3.2	Overview of Simulation environments and Results	57
3.3	Case 1: Accuracy in Large Environments	61
3.4	Case 2: Observability Issues with Tether Estimation	67
3.5	Case 3: Observable Reconfigurable EKF SLAM	74
3.5.1	Maintaining Observability	74
3.5.2	Reconfigurable EKF: Impact of Entering and Departing Land- marks on the EKF	79
3.5.3	Analysis of Case 3	81
3.6	Case 4: Limitations of Anchor-Tether Re-selection	87

4 Dempster-Shafer Occupancy Grid Map Simulations and Results	97
4.1 Illustration of the Dempster-Shafer theory through examples	98
4.2 Case 5: Evidential Mapping of the Straight Ladder Environment . . .	114
4.3 Case 6: Evidential Mapping of the Circular Ladder Environment . . .	117
5 Conclusions and Future work	125
References	129

List of Tables

1	Straight Ladder Environment Spatial Description	59
2	Circular Ladder Environment Spatial Description	59
3	Automated Observable SLAM in a Straight Ladder Environment . . .	81
4	Case 3: Landmark 5 Event Chart	83
5	Automated Observable SLAM in a Circular Ladder Environment . . .	88
6	Case 4: Landmark 1 Event Chart	93
7	Emptiness Mass values at loops 1,2 and 5	104
8	Fullness Mass values at loops 1,2 and 5	105
9	Emptiness Belief values at loops 1,2 and 5	107
10	Fullness Belief values at loops 1,2 and 5	108
11	Emptiness Plausibility values at loops 1,2 and 5	111
12	Fullness Plausibility values at loops 1,2 and 5	112

List of Figures

1	Possible Sequence of Algorithms for Exploration	3
2	Relationship between World Centric and Vehicle Centric Reference Frames	17
3	Relationship between World Centric and Vehicle Centric Reference Frames	39
4	Impact of using only an Anchor to define F_w	41
5	Impact of using an Anchor and Tether to define F_w	42
6	Straight Ladder Environment	58
7	Circular Ladder Environment	58
8	Effect of Range and Bearing Sensor Errors on Cartesian Error	61
9	Case 1: Motion of Vehicle through the Straight Ladder Environment .	62
10	Case 1: State Error, Standard Deviation of Vehicle pose and Tether State Error	63
11	Case 1: State Error and Standard Deviation of Landmarks 3 and 4 .	64
12	Case 1: State Error and Standard Deviation of Landmarks 5 and 6 .	65
13	Case 1: State Error and Standard Deviation of Landmarks 7 and 8 .	66
14	Fisher's Information for varying Vehicle positions and Anchor-Tether orientations	69
15	Case 2: Motion of Vehicle through the Straight Ladder Environment .	71

16	Case 2: State Error, Standard Deviation of Vehicle pose and Tether State Error	72
17	Case 2: State Error and Standard Deviation of Landmarks 1 and 2 .	73
18	Case 2: State Error and Standard Deviation of Landmarks 5 and 6 .	73
19	Decision Sequence of Anchor-Tether re-selection	77
20	Case 3: State Error , Standard Deviation of Vehicle pose and Tether State Error	84
21	Case 3: State Error and Standard Deviation of Landmarks 3 and 4 .	85
22	Case 3: State Error and Standard Deviation of Landmarks 5 and 6 .	85
23	Case 3: Lifecycle of Landmark 5	86
24	Case 4: Motion of Vehicle through the Circular Ladder Environment	89
25	Case 4: State Error , Standard Deviation of Vehicle pose and Tether State Error	92
26	Case 4: Lifecycle of Landmark 1	94
27	Case 4: State Error and Standard Deviation of Landmarks 1 and 2 .	95
28	Case 4: State Error and Standard Deviation of Landmarks 3 and 4 .	95
29	Case 4: State Error and Standard Deviation of Landmarks 5 and 6 .	96
30	Case 4: State Error and Standard Deviation of Landmarks 7 and 8 .	96
31	Example of Environment used to illustrate Demspster-Shafer theory .	99
32	Emptiness belief map	113
33	Fullness belief map	113
34	Straight Ladder Environment	114
35	Case 5: Emptiness Belief Map Straight Ladder Environment	115
36	Case 5: Emptiness Plausibility Map Straight Ladder Environment . .	115
37	Case 5: Fullness Belief Map Straight Ladder Environment	116
38	Case 5: Fullness Plausibility Map Straight Ladder Environment . . .	116
39	Circular Ladder Environment	117

40	Case 6: Emptiness Belief Map Circular Ladder Environment	120
41	Case 6: Emptiness Plausibility Map Circular Ladder Environment	120
42	Case 6: Fullness Belief Map Circular Ladder Environment	121
43	Case 6: Fullness Plausibility Map Circular Ladder Environment	121
44	Comparison between Emptiness Belief and Plausibility Maps	123
45	Comparison between Fullness Belief and Plausibility Maps	124

Chapter 1

Advent of Mapping and Localization

Exploration using an unmanned vehicle, in simplest terms, is the act of traveling through an unknown area to learn about the environment. An unmanned vehicle equipped with sensors, when placed in an unfamiliar environment, has to accomplish Simultaneous Localization and Mapping (SLAM), which is an essential component of exploration. SLAM achieves two main goals- it creates a repository of valid information about the environment using measurements obtained (mapping), and using the created map, seeks the vehicle's position and heading (pose) within the map (localization). Thus, based on the estimated pose of the vehicle and the estimated positions of the landmarks, further evaluation of the map can be made to explore regions that have not been previously visited.

Since all problems of mapping correspond to sensory observations of the environment, the observations will most likely be features in the environment that are landmarks. Most state estimation algorithms that enable autonomous robots to perform localization and mapping consider landmarks as point features based on measurements in the form of range and bearing to landmarks relative to the vehicle pose. These vehicle centric measurements are used to estimate the pose of the vehicle and the position of landmarks relative to the world centric reference frame. While using point

features makes algorithms functionally efficient, it also introduces approximation errors in mapping and localization, since accuracy is lost in downsizing environmental landmarks to point features. These errors in turn affect the state estimation of the vehicle pose and the environmental map.

Thus if features in the environment can be modelled better, the increased accuracy in modelling in turn impacts accuracy in state estimation. Improved state estimation and mapping approaches involve state of the art sensors, complex environment modelling techniques and robust mapping algorithms. A significant amount of work has been done in computer vision that involves improved feature detection and 3 dimensional modelling of objects, however few authors like those of [1], [2], [3] and [4], have combined them with exploration or SLAM. Much work has been done in understanding the structure of objects using a variety of sensors inspired by vision and haptic as explored in [5], [6], [7] and [8], to create a realistic model of the environment.

Elfes of [9] demonstrates the effectiveness of using grid based maps produced from multiple sensors to merge into a single local map and combining multiple local maps into a global map that can be used for navigation. From an exploration point of view, Yamauchi of [10] uses occupancy grids to determine regions suitable for exploration by assessing occupancy values suggesting unexplored terrain. Elements of obstacle avoidance and navigation can also be integrated like authors of [9]. Mapping algorithms can be seen as either a source of information for or as a product of state estimation algorithms that could be used for exploration. A closer look at mapping algorithms reveals that a more descriptive approach of mapping for the purpose of SLAM can be explored by using evidential reasoning in occupancy grids.

Combining the various sensing, state estimation, mapping and path planning algorithms can lead to exploration. The main goal of this thesis is to establish the foundation for and verify two algorithms independently- Reconfigurable EKF for 2D Observable SLAM and Dempster-Shafer based Occupancy Grids, that can later be

used in conjunction with other techniques for exploration. The Reconfigurable EKF is proposed as a solution to the unobservability challenge that a 2D SLAM problem offers by imposing 3 observability constraints. Assuming a finite range sensor, the algorithm implements 2D Observable SLAM in 2 representative environments, and tests for accuracy in large environments and impact of landmarks in the environment entering and leaving the field of view of the sensor. Occupancy grids are analyzed for their usefulness in representing the 2D environment and Dempster-Shafer Evidential theory is used to populate the grids, which is chosen as an improvement over Bayesian theory.

Figure 1 shows one possible sequence of using multiple algorithms in exploration.

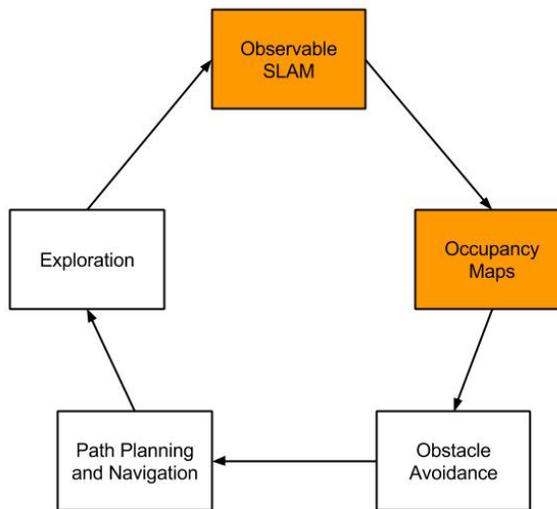


Figure 1: Possible Sequence of Algorithms for Exploration

Khan of [11] analyzed observability in 2D SLAM and Yang of [12] focused on the various mapping techniques, with the assumption that the localization problem was solved. Comparison of grid based and topological based maps were provided, with

analysis of evidential environment mapping in environments representative of indoor office -like layouts. This thesis has provided several modifications to the original works by both authors in the form of features added to the existing algorithms, testing and analysis in significantly larger environments with exploration as the end goal. The boxes highlighted in orange in Figure 1 are the parts that this thesis implements and analyzes independently. Since integrating both techniques depends on whether occupancy maps are used for the state estimation algorithm for Observable SLAM or vice versa, the integration has been left for future work after establishing an algorithmic sequence for exploration.

The rest of the chapter provides the thesis statement and literature review with respect to SLAM and mapping techniques.

1.1 Thesis Statement

Exploration by an unmanned vehicle in an unknown environment will require it to simultaneously map and localize itself, and without any previous information about its surroundings, the problem is no longer observable. To make the problem observable, observability constraints in the form of 2 coordinates of one special landmark and 1 coordinate of another special landmark is required. For efficient obstacle avoidance, it is imperative to create and maintain an information map that is richer in detail than a point landmark based map, as maintained by most Kalman Filtering and Particle Filtering methods.

The thesis focuses on two main goals:

- For a 2D SLAM problem in an environment with stationary landmarks, impose observability, maintain observability and reconfigure the EKF to handle loss of observability constraints, in the form of special landmarks departing the sensor's field of view.

- Produce an Evidential Occupancy grid map as an alternative to the point landmark map maintained by the EKF and analyze the usefulness of the Evidential Occupancy grid map for its effectiveness in representing the environment.

The algorithm proposed and used to perform 2D Observable SLAM is a Reconfigurable Extended Kalman Filter, and Dempster-Shafer theory of Evidential reasoning is used to populate the occupancy grid based map. The above mentioned goals have been addressed in the following manner.

1. Formulating and implementing an Observable 2D SLAM problem, in an environment with a vehicle equipped with a finite range sensor, as an improvement to [11], where the observable 2D SLAM problem is implemented without considering range limitations of sensors. Observability is enforced by imposing 3 observability constraints by holding constant the x and y coordinates of one special landmark, and the x or y coordinate of another landmark.
2. Modifications to the existing EKF to account for special and non-special landmarks that enter and depart the sensor's field of view are proposed and implemented.
3. Implications of using a finite range sensor on the observability constraints are considered, and solutions are proposed to re-impose observability constraints in cases where special landmarks exit the sensor's field of view. Evaluation tests are proposed and implemented to re-select special landmarks that are most suitable to be re-imposed as observability constraints.
4. The unobservability of the 2D SLAM problem is analyzed using Fisher's Information and simulations, for cases when 3 observability constraints are imposed, however, with the incorrect coordinate of the tether being imposed as the third

constraint. Solutions to overcome challenge have been proposed and implemented by tests that select the correct coordinate of the tether to be selected, based on the orientation of the special landmark pair.

5. The solutions proposed to implement items 2, 3 and 4 are integrated into the standard EKF algorithm, enabling it to automatically evaluate special landmarks, non special landmarks for re-imposing constraints and the orientation of the special landmark pair and accounting for entering and departing landmarks for every iteration. Since the EKF is continuously evolves and reconfigures itself, it is referred to as the Reconfigurable EKF.
6. Techniques used to represent likelihoods of occupancy and emptiness in grid based maps, namely Bayesian theory and Dempster-Shafer Evidential theory, are analyzed and compared. The effectiveness in combining existing map data with newly arriving sensor data is also analyzed in both techniques. Dempster-Shafer theory is chosen to implement Occupancy Grid maps in the same large environments that are used in the observable SLAM simulations, and an extensive analysis of the information represented is provided.

1.2 SLAM: An Introduction

When a mobile robot is faced with the task of moving through an unknown area, it has to accomplish two goals- firstly, to gain knowledge of the environment around itself (mapping), and secondly, to gain knowledge of its own position and orientation in the map (localization). After these goals have been achieved, further ideas of autonomy like navigation and exploration can be built on the pre-established concepts of localization and mapping. Considering the dependence of exploration on localization and mapping, a brief history to the concepts have been given below.

When approached simultaneously, the problems of localization and mapping are called SLAM (Simultaneous Localization and Mapping), that was coined from the former term of CML (Concurrent Mapping and Localization). Predicting a model of the mobile robot's surroundings and the robot's location in that environment used the idea of state estimation to find the solution- thus, the states that were estimated were that of the robot, and that of some distinguishable features/ landmarks in its visible surroundings. Using this technique, a map would then be built incrementally, as more sensor readings about the environment are obtained as a function of time. SLAM has been solved theoretically; it has also been implemented in a number of different domains from indoor robots to outdoor, underwater, and airborne systems, however, substantial issues remain in practically realizing more general SLAM solutions and notably in building and using perceptually rich maps as part of a SLAM algorithm [13].

1.3 SLAM: A brief history

Probabilistic SLAM came to existence when consistent mapping in SLAM was considered a problem because of computational and conceptual challenges. It was the breakthrough by Durrant-Whyte [14] that established a statistical way of characterizing landmarks and their relationships with geometric uncertainty. The authors of [14] also showed that there must exist a high degree of correlation amongst landmark estimates in the map, and that these correlations grow with successive observations, [13]. [15] showed that the error in estimates of the landmarks were necessarily correlated, because of the common error in estimated vehicle motion. The correlation of estimation error being related to error in vehicle motion is because measurements to landmarks are dependent on the vehicle's pose, which is in turn affected by estimation errors. This finding, therefore, implied that for a consistent solution to the complete

problem, the state to be estimated would involve the joint states of the vehicle pose as well as the landmark locations, which is then updated periodically. Computationally, the number of operations required for the entire process is a quadratic function of the number of landmarks observed, as for each updating iteration, the joint state and the covariance matrix will have to be updated.

Work was then seen to progress in visual and sonar based navigation, that predominantly used Kalman Filter (KF) algorithms. Areas of research included improving computational efficiency and data association/ correspondence problem/ loop closure. The 1999 International Symposium on Robotics Research (ISRR' 99) saw the convergence of KF based SLAM and probabilistic localization and mapping techniques, according to the authors of [13].

1.4 SLAM: Evolution of Approaches

Currently, there are multiple ways of approaching the SLAM problem, which can be classified into one of the following categories- state estimation algorithms that encompass most variants of Kalman Filtering algorithms, probabilistic methods as in [16] and [17], and vision based SLAM that utilize neural network based solutions as in [18]. The classification of approaches are a result of focusing on different areas of the issue for improvement - speed, computational efficiency, accuracy and robustness.

State estimation algorithms have evolved from the most basic filter- the Linear Kalman Filter (LKF), which is an unbiased minimum variance estimator for linear, Gaussian systems [19]. Most systems are modelled to be Gaussian, for approximation purposes and the LKF is then utilized as the state and variance estimator. Typically though, most systems are nonlinear and seldom accurately follow the Gaussian distribution. In cases like these, a nonlinear version of the Kalman Filter, the Extended Kalman Filter (EKF) is used. The EKF linearizes the estimate about the current

state or the current mean and covariance [20]. The Unscented Kalman Filter (UKF) uses the unscented transform to pick a minimal set of sample points around the mean so that the filters can avoid poor performance when the state transition and observation models are highly nonlinear [20]. There are many other variants of the Kalman Filter- Information form of the Kalman Filter or just the Information Filter [11] which attempts to simplify the estimation process computationally, Ensemble Kalman Filter (EnKF) [20] and the Compressed Extended Kalman Filter (CEKF) [11]. EKF is undoubtedly the most widely used nonlinear state estimation technique in mobile robotics, particularly for localization and mapping problems [20], thus spawning a wide range of variants of the EKF such as- Augmented EKF (AEKF) [21] and Interlaced Extended Kalman Filter [22] . For the purposes of this thesis, a Reconfigurable EKF is used for state estimation.

Algorithms like Monte Carlo Localization (MCL) [16], [17], Particle Filters and Fast SLAM are techniques that are predominantly based on a family of probabilistic methods- Markov Localization and sampling particles representing a distribution for Particle Filter based approaches. The advantage with techniques like these lie in the fact that the distributions are not restricted to being solely Gaussian, or unimodal. Particle filters are often an alternative to EKF or UKF with the advantage that, with sufficient samples, they approach the Bayesian optimal estimate, so they can be made more accurate than either the EKF or UKF [20] .

Fast SLAM has been a popular alternative to the EKF methods, in order to combat EKF's computational complexity. This complexity stems from the fact that the size of Kalman filters' covariance matrices grows quadratically with the number of landmarks, i.e. $\mathcal{O}(K^2)$ (K landmarks) and the entire covariance matrix has to be updated even if only a single landmark is seen, [23]. Fast SLAM attempts to optimize the estimation process by decoupling the robot pose and the landmark state estimation problems, and by conditioning the landmark estimation on the robot pose

estimate, and reduces computational complexity. It employs a particle filter for the localization process and a Kalman filter for each landmark being estimated, as done by authors of [23] and reduces computation complexity to $\mathcal{O}(nK)$ (n Kalman Filters per landmark) or vice versa, as done by authors of [24], which can be optimized even further. Further variants of Fast SLAM are Fast SLAM 1.0 [25], Fast SLAM 2.0 [26], EKF Fast SLAM, UKF Fast SLAM and UFast SLAM [27]. Most FastSLAM literature uses point landmarks, [28], [23], [29] and [25], however very few, like the authors of [24] attempt to produce grid based maps and/or use information from the grid based maps created, like authors of [30].

A more recent approach to SLAM is by using tools from computer vision, by using appearances for the basis of probabilistic methods as explored in [31] and [32], and neural network based approaches for classification of the environment into categories as explored in [18].

Observability of the SLAM problem has received little attention, and there have been very few authors who have focused on the unobservable nature of the SLAM problem. Hermann et al. [33] first defined the necessary rank condition for local weak observability, followed by Chen in [34] who focused on observability of linear discrete time systems. Nonlinear observability analysis was discussed by Nijmeijer in [35], Song in [36] and by Lee and Wijesoma in [37]. Authors of [38], [39], and [40] question the validity of convergence of the filter, when the system is unobservable. But pivotal results proving world centric SLAM is unobservable came in the form of contributions by authors of [37], [41] and [38]. More recently, authors Khan of [11] and Souici et al. of [42], have implemented solutions for observable SLAM.

1.5 Mapping and Exploration: An Introduction

Robot exploration often goes hand in hand with mapping, since if a priori maps were available, exploration in the context of spatial understanding, would not be needed. Thus we can define the problem of robotic mapping as that of acquiring a spatial model of a robot's environment [43]. Since mapping involves measurements acquired from sensory inputs, problems in mapping find their source from the area of sensor measurements. The key challenges in mapping can be attributed to five sources- measurement noise, high-dimensionality of entities being mapped, data association/correspondence, dynamic environments and choice of control/motion of robot during exploration, as summarized in [43].

For autonomous navigation and exploration, the vehicle requires knowledge about the environment, but since it is not usually available to the vehicle beforehand, the information has to be detected. While applications like exploration, obstacle avoidance or navigational motion are dependent on the environment itself, as the authors of [44] put it, there is always a question that demands if the environment needs representation, and if so, how many kinds there could be and what should be represented. Factors that affect the position of the robot are finely intertwined with perceiving the surroundings, and thus most mapping algorithms work on localizing the robot apart from the solving the mapping problem. Virtually all state of the art algorithms for robotic mapping are probabilistic in modelling vehicle motion, representing the environment and converting sensory information into usable map information [43].

An interesting part of mapping is to determine if the final map merely represents information from sensors or whether it also depicts the relationship amongst information in the map. Depending on the kind of map being used, deducing a relationship/interpreting mapping data will require a high degree of accuracy in data association.

This consists of identifying that an environment element perceived from different positions is actually the same [44]. Assuming that data association is solved correctly, the final step would be to fuse the information recently acquired to the map previously generated/updated, depending on the algorithm - the solution involves probabilistic techniques yet again. A synopsis of the foundation of the probabilistic nature of most mapping and exploration algorithms, and its history will follow in the next two sections.

1.6 Mapping and Exploration: A brief history

The authors of [44] take an interesting view on the early beginnings of the SLAM problem, with the mapping problem in particular, by mentioning sixteenth century cartographers had the same problem to solve, thus often resulting in skewed maps. The evolution of sensors has come a long way, and with it, came an increased number of solutions to tackle the mapping problem. The probabilistic nature of the issue still hasn't changed, however the different forms of measurements enable perception, interpretation and representation of the same environment with different information. A testimony to how various sources of sensing technology have been used is the vast variety in literature that deals with different mapping approaches - those employing Kalman filtering approaches and their modifications (although in conjunction with localization) [45], [46], [47], [48]; those focusing on Dempster's Expectation Maximization algorithms [49], [50], [51], [52] and computer vision based techniques [32], [18], [53], [54]. Mapping approaches are usually greedy, that is, they choose control by greedily maximizing information gain [43].

1.7 Mapping and Exploration: Evolution of Approaches

Considering the probabilistic nature of mapping and exploration, the first set of solutions proposed involved the classical Bayesian perspective in the form of Kalman filters [55]. The Bayes filter is the single dominating scheme to integrate temporal data [43] and has spawned multiple algorithms like those of dynamic Bayes Networks [56], hidden Markov Models [57] and partially observable Markov decision processes [58], [59]. The usage of Bayes filters requires that the state which is being estimated takes into account all uncertainties that may possibly influence its distribution, and in the case of robotic mapping, that includes the robot's pose and the map. Most authors in the mapping literature, for example Yang of [12] and [60] and Thrun of [43] have approached the mapping problem by assuming that the localization of the robot was solved. The main attraction with any Bayesian formulation is that it is recursive, and can be incremental, however the limitations came in the form of assumptions - that landmarks could be created as points and measurement information could be accurately associated to the landmarks. An improvement came in the form of the Lu/Milios algorithm [61] which combines the estimation phase of the Kalman Filter and the correspondence of nearby range measurements using maximum likelihood data association. While it has been successfully implemented by Gutmann in [62], the algorithm does not perform well, with large vehicle initialization errors.

A statistical alternative to the Kalman filtering techniques is the Expectation Maximization (EM) [43], [12], which is similar to the Kalman filter in that it has a prediction and correction phase, where the posterior over robot poses is calculated for a given map in the expectation phase, and the most likely map is determined given the pose expectations, in the maximization phase. EM algorithms are known to be

favoured over Kalman filters, because EM algorithms also tackle the data correspondence problem, when the robot pose is known. They do not however, fare well in large environments where initialization errors are huge, [43]. Additionally, EM (and incremental EM) is inferior to KF algorithms because they're offline and are subject to local maxima [43]. Yet another algorithm is the Particle Filter [63], [12], that has been used in context when the robot localization is assumed to be solved.

The other primary mapping techniques are more representative of the environment- grid based maps, topological maps, and object maps. Grid based maps depict information about the environment in terms of whether the region is occupied or not, without any special consideration to the physical features of landmarks. Topological maps recover a qualitative description of the environment, characterizing the relation of basic locations. Depending on the kind of sensor used, other maps like object maps and feature maps store visual information of the surroundings, and their storage of features can be made efficient by using tree structured Bayesian networks, [64]. Exploiting the efficiency of grid based maps by understanding that their information can be interpreted as more than binary values, authors of [65] have investigated frontier based exploration, authors of [24] who focus on exploration and loop closing using grid based maps, and some others, who focus on a more accurate representation of the environment specifically for exploration and obstacle avoidance, [66].

1.8 Thesis Overview

Chapter 1 provided an introduction to the SLAM and exploration & mapping challenges, along with their history and evolution of possible solutions. The rest of the thesis is organized as follows.

- Chapter 2 provides the theoretical background that formulates the 2D SLAM

problem, an introduction to the Extended Kalman Filter, requirements for observability in 2D SLAM and an introduction to Dempster-Shafer Theory of Evidential Reasoning.

- Chapter 3 provides an overview of parameters used in the simulation of the Reconfigurable EKF for SLAM in two different 2D environments. This is followed by results and analysis for four cases each of which demonstrates a different aspect of imposing and maintaining observability.
- Chapter 4 provides a detailed illustration of the theory Evidential Reasoning used to create Occupancy Grid maps in a simple environment. This is followed by results and analysis of the DS theory occupancy maps produced in two 2D environments.
- Chapter 5 provides a brief overview of the challenges offered by the 2D SLAM problem, the contributions of the thesis and possible areas for future work.

Chapter 2

Theoretical Background

This chapter focuses on formulating a 2 dimensional SLAM environment. It provides a detailed background of the Extended Kalman Filter (EKF), requirements for and tools used to assess observability, and Dempster-Shafer theory of Evidential reasoning used to create occupancy grid maps. Notations used in the rest of the thesis have also been defined in this chapter.

2.1 Problem Formulation

This section focuses on developing the system models consisting of the vehicle motion in terms of pose, landmark positions and vehicle centric measurement process models that are required to formulate the problem using the EKF. Initially developed in continuous time, the process models and the working of the EKF are then introduced in discrete time, for simulation.

2.1.1 Vehicle Process Model

A unicycle model for the vehicle has been considered because of its simplified non-linear motion model. Khan of [11], Huang et al. of [38] and Souici et al. of [42] have used the unicycle model to demonstrate SLAM. The unicycle model

is considered to consist of a single steering wheel with rear driven wheels, however the model does not include side slipping of the two active rear wheels. While the model includes non-linear kinematics, it does not completely model vehicle inertia or dynamics. However, for the purposes of simulation, the simplified motion model suffices to be used as part of SLAM experiments.

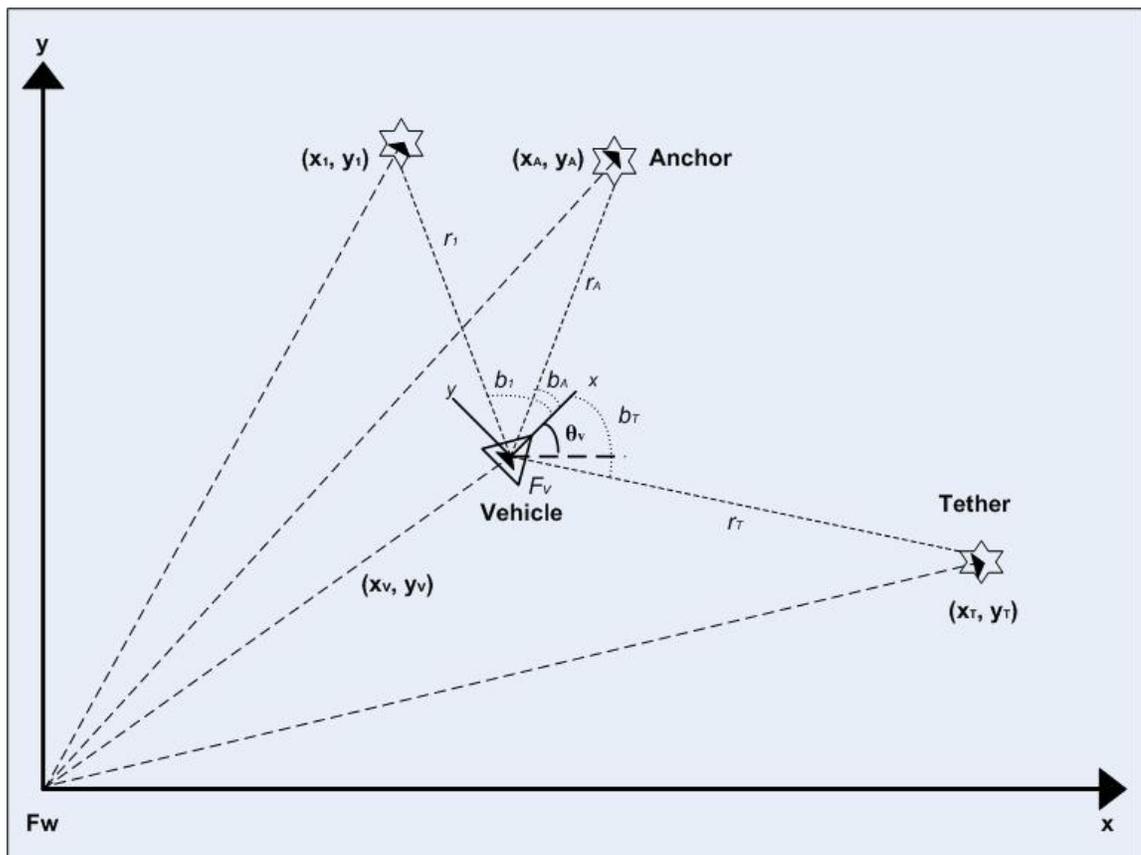


Figure 2: Relationship between World Centric and Vehicle Centric Reference Frames

Figure 2 represents the SLAM problem with the world centric reference frame denoted by F_w , the positions of all landmarks and the pose of the vehicle with respect to F_w , the vehicle centric reference frame denoted by F_v and the vehicle centric measurements in the form of range and bearing to all visible landmarks. All notations in bold in the figure indicate that they correspond to F_w .

Define the vehicle pose in the world centric reference frame as

$$\mathbf{x}_t^v = \begin{bmatrix} x_t^v \\ y_t^v \\ \theta_t^v \end{bmatrix}, \quad (2.1.1)$$

where x_t^v and y_t^v represent the vehicle's position in Cartesian coordinates in F_w , and θ_t^v represents the orientation or heading of the vehicle with respect to the horizontal axis of the world frame of reference. Superscript v and subscript t denote vehicle and continuous time, respectively. The vehicle's state space representation is given as follows

$$\dot{\mathbf{x}}_t^v = \begin{bmatrix} \dot{x}_t^v \\ \dot{y}_t^v \\ \dot{\theta}_t^v \end{bmatrix} = \begin{bmatrix} \cos \theta_t^v & 0 \\ \sin \theta_t^v & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_t \\ \omega_t \end{bmatrix}. \quad (2.1.2)$$

The system's external inputs are given as

$$\mathbf{u}_t = \begin{bmatrix} v_t \\ \omega_t \end{bmatrix}, \quad (2.1.3)$$

where v_t and ω_t are linear velocity and angular velocity respectively.

The state space representation can be expressed in discrete time, using Euler's discretization procedure. Provided the sample period T is sufficiently small, the following maps from continuous to discrete time

$$t = kT \quad (2.1.4)$$

where k is the timestep. The above expression can then be used to write the state of the system in discrete time

$$\dot{\mathbf{x}}_k \approx \frac{\mathbf{x}_{k+1} - \mathbf{x}_k}{T}. \quad (2.1.5)$$

Rearranging the terms in (2.1.5) gives the state of the system at timestep $k+1$ based on the state of the system at timestep k , thus creating a framework for an iterative process

$$\mathbf{x}_{k+1} = \mathbf{x}_k + T\dot{\mathbf{x}}_k. \quad (2.1.6)$$

Casting (2.1.2) into discrete time and using it in (2.1.6), results in

$$\mathbf{x}_{k+1}^v = \mathbf{f}^v(\mathbf{x}_k^v, \mathbf{u}_k) = \mathbf{I}_3 \mathbf{x}_k^v + T \begin{bmatrix} \cos\theta_t^v & 0 \\ \sin\theta_t^v & 0 \\ 0 & 1 \end{bmatrix} \mathbf{u}_k, \quad (2.1.7)$$

where \mathbf{f}^v is a non linear function that maps the current state of the system \mathbf{x}_k^v and the inputs of the system \mathbf{u}_k , to the state of the system at the next timestep. Equation (2.1.7), while representative of the ideal unicycle model, does not include modelling errors in the form of noise, which is necessary for the implementation of the EKF. Thus the complete vehicle model is given by

$$\mathbf{x}_{k+1}^v = \mathbf{f}^v(\mathbf{x}_k^v, \mathbf{u}_k) + \mathbf{q}_k^v, \quad (2.1.8)$$

where \mathbf{q}_k^v is noise sampled at timestep k from a zero mean, Gaussian distribution. Its covariance matrix is characterized by

$$\mathbf{Q}^v = \begin{bmatrix} \sigma_x^2 & 0 & 0 \\ 0 & \sigma_y^2 & 0 \\ 0 & 0 & \sigma_\theta^2 \end{bmatrix}. \quad (2.1.9)$$

2.1.2 Landmark Process Model

All modes of simulation in the thesis consider landmarks as stationary objects. The state of landmark i in the world centric reference frame is given by the coordinates of its position in Cartesian space such as

$$\mathbf{x}_t^i = \begin{bmatrix} x^i \\ y^i \end{bmatrix}. \quad (2.1.10)$$

Since the landmark is stationary, the process model is linear as the state of the landmark is independent of any external inputs or states of other elements in the system. Thus in discrete-time, the model of stationary landmark i , complete with modelling noise is given as follows

$$\mathbf{x}_{k+1}^i = \mathbf{f}^i(\mathbf{x}_k^i) + \mathbf{q}_k^i = \mathbf{I}_2(\mathbf{x}_k^i) + \mathbf{q}_k^i, \quad (2.1.11)$$

where \mathbf{I}_2 is a 2×2 identity matrix and \mathbf{q}_k^i is noise sampled at timestep k from a zero mean, Gaussian distribution. The constant covariance matrix of the state of the vehicle is characterized by

$$\mathbf{Q}^i = \begin{bmatrix} \sigma_{x^i}^2 & 0 \\ 0 & \sigma_{y^i}^2 \end{bmatrix}. \quad (2.1.12)$$

The environment consisting of multiple landmarks needs to have states of all visible landmarks at timestep k as part of the state vector representing the landmarks and this is accomplished by using state augmentation. Thus for landmarks $i = 1 : n$, the entire state representing landmarks is given by

$$\mathbf{x}_k^l = \begin{bmatrix} \mathbf{x}_k^{1T} & \dots & \mathbf{x}_k^{iT} & \dots & \mathbf{x}_k^{nT} \end{bmatrix}^T, \quad (2.1.13)$$

where superscript l denotes landmark, and T denotes transpose.

2.1.3 Measurement Model

Since the vehicle has no direct knowledge of its pose in the environment, it relies solely on its sensory measurements, which come in the form of range and bearing of landmarks with respect to the vehicle. The range and bearing are dependent on the vehicle's pose and landmark i 's position. Define

$$\Delta x_k^i = x_k^i - x_k^v \quad (2.1.14)$$

$$\Delta y_k^i = y_k^i - y_k^v. \quad (2.1.15)$$

The range to landmark i at timestep k is given by

$$r_k^i = \sqrt{\Delta x_k^{i2} + \Delta y_k^{i2}}. \quad (2.1.16)$$

The bearing to landmark i at timestep k is

$$b_k^i = \arctan\left(\frac{\Delta y_k^i}{\Delta x_k^i}\right) - \theta_k^v, \quad (2.1.17)$$

where \arctan is a 4-quadrant arc tangent function. The measurement model of landmark i at timestep k , complete with measurement noise is given as

$$\mathbf{z}_k^i = \mathbf{h}(\mathbf{x}_k^i, \mathbf{x}_k^v) + \mathbf{w}_k^i = \begin{bmatrix} r_k^i \\ b_k^i \end{bmatrix} + \mathbf{w}_k^i, \quad (2.1.18)$$

where \mathbf{w}_k^i is noise sampled at timestep k from a zero-mean, Gaussian distribution, whose constant covariance matrix is characterized by

$$\mathbf{R}^i = \begin{bmatrix} \sigma_r^2 & 0 \\ 0 & \sigma_b^2 \end{bmatrix}. \quad (2.1.19)$$

Procured measurements of all visible landmarks can be augmented into one measurement vector for timestep k . Thus, for landmarks from $i = 1 : n$ at timestep k , the measurement vector is given by

$$\mathbf{z}_k = \left[\mathbf{z}_k^{1T} \dots \mathbf{z}_k^{iT} \dots \mathbf{z}_k^{nT} \right]^T, \quad (2.1.20)$$

where superscript T stands for transpose.

The measurement model covariance for all visible landmarks $i = 1 : n$ for timestep k is then given by

$$\mathbf{R}_k = \text{blkdiag} \left[\mathbf{R}_k^1, \dots, \mathbf{R}_k^i, \dots, \mathbf{R}_k^n \right]. \quad (2.1.21)$$

2.1.4 Jacobians of the Vehicle and Measurement Models

The EKF uses the Jacobians of the vehicle and measurement models for time and measurement updates of states, which is discussed in detail in Section 2.2.

The vehicle process Jacobian is given by the derivative of the vehicle process model with respect to the vehicle's state at timestep k

$$\mathbf{F}_k^v = \frac{\partial \mathbf{f}^v(\mathbf{x}_k^v, \mathbf{u}_k)}{\partial \mathbf{x}_k^v} = \begin{bmatrix} 1 & 0 & -Tv_k \sin \theta_k^v \\ 0 & 1 & Tv_k \cos \theta_k^v \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.1.22)$$

The complete Jacobian for the dynamic model is given by

$$\mathbf{F}_k = \text{blkdiag}(\mathbf{F}_k^v, \mathbf{I}_{2n-3}), \quad (2.1.23)$$

where \mathbf{I}_{2n-3} is an identity matrix of dimension $2n - 3$ where n is the total number of visible landmarks. The measurement Jacobian is composed of two parts, one with respect to the vehicle and the other with respect to the landmarks. The measurement Jacobian of landmark i at timestep k with respect to the vehicle's pose at timestep k is given by

$$\mathbf{H}_k^{v(i)} = \frac{\partial \mathbf{h}_k^i(\mathbf{x}_k^v, \mathbf{x}_k^i)}{\partial \mathbf{x}_k^v} = \begin{bmatrix} -\frac{\Delta x_k^i}{r_k^i} & -\frac{\Delta y_k^i}{r_k^i} & 0 \\ \frac{\Delta y_k^i}{r_k^{i^2}} & -\frac{\Delta x_k^i}{r_k^{i^2}} & -1 \end{bmatrix}. \quad (2.1.24)$$

The measurement model Jacobian of landmark i with respect to itself at timestep k is given by

$$\mathbf{H}_k^i = \frac{\partial \mathbf{h}_k^i(\mathbf{x}_k^v, \mathbf{x}_k^i)}{\partial \mathbf{x}_k^i} = \begin{bmatrix} \frac{\Delta x_k^i}{r_k^i} & \frac{\Delta y_k^i}{r_k^i} \\ -\frac{\Delta y_k^i}{r_k^{i^2}} & \frac{\Delta x_k^i}{r_k^{i^2}} \end{bmatrix}. \quad (2.1.25)$$

The final measurement Jacobian of landmark i at timestep k is given by concatenating (2.1.24) and (2.1.25) in the following manner

$$\mathbf{H}_k = \begin{bmatrix} \mathbf{H}_k^{v(1)} & \mathbf{H}_k^1 & 0 \dots & 0 \\ \mathbf{H}_k^{v(2)} & 0 & \mathbf{H}_k^2 \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{H}_k^{v(n)} & 0 & 0 \dots & \mathbf{H}_k^n \end{bmatrix}. \quad (2.1.26)$$

2.1.5 Summary of the SLAM problem

The problem can now be summarized as the task of estimating the joint states of the vehicle and of all visible landmarks (2.1.27) with respect to F_w , given the vehicle process model (2.1.28), the landmark process model (2.1.29) and the vehicle centric measurement model (2.1.30).

The joint state of the vehicle and landmarks are created by state augmentation of (2.1.1) and (2.1.13), which can be expressed as

$$\mathbf{x}_k = \begin{bmatrix} \mathbf{x}_k^v \\ \mathbf{x}_k^1 \\ \vdots \\ \mathbf{x}_k^i \end{bmatrix} = \begin{bmatrix} x_k^v & y_k^v & \theta^v & x_k^1 & y_k^1 & \dots & x_k^i & y_k^i \end{bmatrix}^T. \quad (2.1.27)$$

The non-linear vehicle process model is given by

$$\mathbf{x}_{k+1}^v = \mathbf{f}^v(\mathbf{x}_k^v, \mathbf{u}_k) + \mathbf{q}_k^v, \quad (2.1.28)$$

and the stationary landmark process model is linear of the form

$$\mathbf{x}_{k+1}^i = \mathbf{f}^l(\mathbf{x}_k^i) + \mathbf{q}_k^i. \quad (2.1.29)$$

Finally, the non-linear measurement model that obtains measurements of stationary landmark positions is given by

$$\mathbf{z}_k^i = \mathbf{h}(\mathbf{x}_k^i, \mathbf{x}_k^v) + \mathbf{w}_k^i. \quad (2.1.30)$$

The formulation of the problem is now complete for its use in the Extended Kalman Filter, which is discussed in the next section.

2.2 Extended Kalman Filter

The Kalman filter is one of the most useful estimation tools available that provides a recursive method of estimating the state of a dynamical system in the presence of noise. Since the Kalman Filter assumes that the output is a Gaussian probability density function (PDF) with a mean of $\hat{\mathbf{x}}_k$ and covariance \mathbf{P} , the estimates $\hat{\mathbf{x}}_k$ of the state vector \mathbf{x}_k and the error covariance matrix \mathbf{P} are representative of the distribution. This shows that the Kalman filter is a specific example of a more general technique known as probabilistic estimation [67]. The Kalman Filter is viewed as a predictor-corrector algorithm, that predicts based on the model of the system, and updates its prediction based on the acquired measurements.

For the current task of state estimation, the discrete time version of the Extended Kalman Filter has been used, and thus a brief introduction to the discrete time Kalman Filter is given as a predecessor to the introduction of the discrete time Extended Kalman Filter, in the following sections.

2.2.1 Introduction to the discrete time Linear Kalman Filter

Considering implementation in real time systems, the discrete time Linear Kalman Filter (LKF) is introduced, which is closely related to the continuous time Kalman Filter. The LKF is an optimal, unbiased and minimum variance observer with the

assumption that any noise is zero-mean white Gaussian and the system is of the following linear form

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{G}\mathbf{u}_k + \mathbf{q}_k \quad (2.2.1)$$

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{w}_k. \quad (2.2.2)$$

Assume that noise is zero-mean Gaussian and that the noise from the system and measurement processes are independent in the following form

$$\mathcal{E}[\mathbf{q}_k] = \mathcal{E}[\mathbf{w}_k] = \mathbf{0} \quad (2.2.3)$$

$$\mathcal{E}[\mathbf{w}_k \mathbf{q}_j^T] = \mathbf{0}, \quad (2.2.4)$$

where superscript T indicates transpose, and

$$\mathcal{E}[\mathbf{w}_k \mathbf{w}_j^T] = \begin{cases} \mathbf{R}_k, & \text{if } k = j \text{ (the measurement noise covariance)} \\ 0, & \text{otherwise.} \end{cases}$$

$$\mathcal{E}[\mathbf{q}_k \mathbf{q}_j^T] = \begin{cases} \mathbf{Q}_k, & \text{if } k = j \text{ (the process noise covariance)} \\ 0, & \text{otherwise.} \end{cases}$$

Define the state error

$$\tilde{\mathbf{x}}_k = \mathbf{x}_k - \hat{\mathbf{x}}_k \quad (2.2.5)$$

and the covariance associated with the state estimate $\hat{\mathbf{x}}_k$ as

$$\mathbf{P}_k = \mathcal{E}[\tilde{\mathbf{x}}_k \tilde{\mathbf{x}}_k^T]. \quad (2.2.6)$$

The main goal of the LKF is to simultaneously maintain an estimate $\hat{\mathbf{x}}_k$ of the state and that of the associated covariance \mathbf{P}_k . Since the filter has two phases, prediction and correction, for the state and covariance, the following notation has been adopted

- $\hat{\mathbf{x}}_k^-$ is the a priori or predicted state estimate at timestep k .
- $\hat{\mathbf{x}}_k^+$ is the a posteriori or corrected state estimate at timestep k .
- \mathbf{P}_k^- is the a priori or predicted covariance at timestep k .
- \mathbf{P}_k^+ is the a posteriori or corrected covariance at timestep k .

The filter is initialized with the following

$$\hat{\mathbf{x}}_0^+ = \mathcal{E}[\mathbf{x}_0] \quad (2.2.7)$$

$$\mathbf{P}_0^+ = \mathcal{E}[\tilde{\mathbf{x}}_0 \tilde{\mathbf{x}}_0^T]. \quad (2.2.8)$$

The LKF has been derived on the basis of Bayesian postulates, that allows for the dependency of the state (a priori) at one sample period on the state (a posteriori) at the previous sample period. This implies that the filter can run iteratively, without having to store the history of observations or states, for the functioning of the filter. The algorithm of the LKF is given as follows.

After initialization of the Kalman state and Kalman covariance as seen in equations (2.2.7) and (2.2.8), the filter performs a prediction step shown in equations

(2.2.9) and (2.2.10), which involves using the system process models to obtain an a priori estimate of the state as well as the covariance. It is assumed that the input at every timestep k , \mathbf{u}_k , the a posteriori state estimate $\hat{\mathbf{x}}_{k-1}^+$, the a posteriori covariance \mathbf{P}_{k-1}^+ are known from simulation or initialization. Process model noise and measurement noise covariance matrices, \mathbf{Q}_{k-1} and \mathbf{R}_{k-1} are user provided values, determined from the levels of confidence attributed to measurements

$$\hat{\mathbf{x}}_k^- = \mathbf{F}\hat{\mathbf{x}}_{k-1}^+ + \mathbf{G}\mathbf{u}_{k-1} \quad (2.2.9)$$

$$\mathbf{P}_k^- = \mathbf{F}\mathbf{P}_{k-1}^+\mathbf{F}^T + \mathbf{Q}_{k-1}. \quad (2.2.10)$$

Equations (2.2.9) and (2.2.10) are called *time update* equations, because they run the system ahead in time and predict the state and covariance. Following this, sensory measurements \mathbf{z}_k obtained are of the form

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{w}_k \quad (2.2.11)$$

and the predicted measurements are computed using

$$\hat{\mathbf{z}}_k = \mathbf{H}\hat{\mathbf{x}}_k. \quad (2.2.12)$$

The next step is to obtain the innovation sequence (2.2.13) and the covariance associated with it (2.2.14)

$$\mathbf{v}_k = \mathbf{z}_k - \hat{\mathbf{z}}_k \quad (2.2.13)$$

$$\mathbf{S}_k = \mathbf{H}\mathbf{P}_k^{-1}\mathbf{H}^T + \mathbf{R}_k. \quad (2.2.14)$$

The information gathered from sensory observations are used to correct the estimated state and covariance. Thus, using equation (2.2.14), the Kalman gains can be computed as follows

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}^T \mathbf{S}_k^{-1}. \quad (2.2.15)$$

The computed Kalman gains form the weights that determine how the predicted state and covariance values get updated. The final steps involving correction of the predicted state and covariance are given by

$$\hat{\mathbf{x}}_k^+ = \mathbf{K}_k \mathbf{v}_k + \hat{\mathbf{x}}_k^- \quad (2.2.16)$$

$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k \mathbf{H}_k) \mathbf{P}_k^-, \quad (2.2.17)$$

where \mathbf{I} is an $n \times n$ identity matrix. Expressions (2.2.16) and (2.2.17) are called *measurement update* equations, because they update the predicted state based on true measurements obtained using sensory observations, and provide a revised estimate of the state and the covariance, taking into account the deviation of the system's estimates from the true measurements.

2.2.2 Introduction to the discrete time Extended Kalman Filter

While the LKF works well for linear systems, it cannot accommodate non-linear systems. A modification of the LKF, called the Extended Kalman Filter(EKF) is used for non-linear systems. The EKF assumes a non-linear process and measurement

model of the following form

$$\mathbf{x}_{k+1}^v = \mathbf{f}^v(\mathbf{x}_k^v, \mathbf{u}_k) + \mathbf{q}_k^v \quad (2.2.18)$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{w}_k. \quad (2.2.19)$$

The initialization process of the EKF is the same as that of the LKF however, it differs during time update sequence. In the LKF, the time update sequence makes use of the system's models, as they are linear. In the EKF, the linearization is a forced procedure, that must take place in order to obtain a priori state and covariance estimates. Considering the problem formulation, since both the vehicle and measurement process models are non-linear, both system models will be linearized using the Jacobian

$$\mathbf{F}_k = \left. \frac{\partial \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_k)}{\partial \mathbf{x}_{k-1}} \right|_{\mathbf{x}_{k-1} = \hat{\mathbf{x}}_{k-1}^+}. \quad (2.2.20)$$

Equation (2.2.20) represents the vehicle process Jacobian, and has been computed in Section 2.1.4 and is given by the expression (2.1.23).

The measurement process Jacobian given by

$$\mathbf{H}_k = \left. \frac{\partial \mathbf{h}(\mathbf{x}_k)}{\partial \mathbf{x}_k} \right|_{\mathbf{x}_k = \hat{\mathbf{x}}_k^-}, \quad (2.2.21)$$

is computed in Section 2.1.4 and is given by the expression (2.1.26).

Correspondingly, the time update equations take a slightly different form

$$\hat{\mathbf{x}}_k^- = \mathbf{f}(\hat{\mathbf{x}}_{k-1}^+, \mathbf{u}_{k-1}) \quad (2.2.22)$$

$$\mathbf{P}_k^- = \mathbf{F}_{k-1} \mathbf{P}_{k-1}^+ \mathbf{F}_{k-1}^T + \mathbf{Q}_k. \quad (2.2.23)$$

Similarly, the computation of the innovation sequence involving $\hat{\mathbf{z}}_k$ is computed with

$$\hat{\mathbf{z}}_k = \mathbf{h}(\hat{\mathbf{x}}_{k-1}^-). \quad (2.2.24)$$

The remaining steps of the algorithm are the same as that of the LKF as seen in Section 2.2.

2.3 Observability

The SLAM problem requires the EKF to act as a state observer, and since state observers require systems to be observable [68], the SLAM problem must be formulated as observable for the EKF to provide consistent state estimates. Observability plays a major role in filtering and in reconstruction of states from inputs and outputs, so while literature on SLAM algorithms is plenty, theoretical issues relating to the fundamental properties of SLAM has seldom received as much attention, as the authors of [37] and [11] point out. There have been a few exceptions however, with the following authors focusing on observability, specifically for SLAM problems [38], [39], and [40], whose works question the validity of convergence of the filter, when the system is unobservable. Considering the importance of observability, this section focuses on defining observability, its implications for SLAM and the constraints required to keep the system observable.

2.3.1 Observability in linear and non-linear systems

A system is said to be completely observable if at a particular time instant t_0 , every element of the state, $x(t_0) = x_0$ can be reconstructed only from the available measurements/observations over a finite time interval, $t_0 \leq t \leq t_1$. A dynamic system with an unobservable state is called unobservable. The initial state x_0 can be uniquely determined from the measurements if and only if the system is observable, and this proves crucial to the understanding of its implications for reconstructing the states of the vehicle and the landmarks in SLAM.

Observability takes different forms depending on the linearity of the system. For a linear, time invariant system of the form

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{G}\mathbf{u}_k \quad (2.3.1)$$

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k, \quad (2.3.2)$$

if the observability matrix \mathbf{O} given in equation (2.3.3) is full rank, then the system is observable, and unobservable if it is not full rank,

$$\mathbf{O} = \begin{bmatrix} \mathbf{H} \\ \mathbf{HF} \\ \mathbf{HF}^2 \\ \vdots \\ \mathbf{HF}^{n-1} \end{bmatrix}. \quad (2.3.3)$$

Assuming n states and m measurements, the dimension of \mathbf{O} is $nm \times n$, implying

that the system is observable if $\text{rank}(\mathbf{O}) = n$.

Consider nonlinear, discrete time systems of the following form

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) \quad (2.3.4)$$

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k). \quad (2.3.5)$$

To arrive at a similar analysis of the observability matrix for a deterministic system, consider the following definitions of functional composition

$$\mathbf{f} \circ \mathbf{f}(\mathbf{x}) = \mathbf{f}(\mathbf{f}(\mathbf{x})) \quad (2.3.6)$$

$$\mathbf{f}^j(\mathbf{x}) = \mathbf{f} \circ \mathbf{f}^{j-1}(\mathbf{x}) \quad (2.3.7)$$

$$\mathbf{f}^0(\mathbf{x}) = \mathbf{x}. \quad (2.3.8)$$

Referring to the review in [69] for any given initial state \mathbf{x} , the system produces the following sequence of outputs given by the ordered set

$$\mathcal{O}(\mathbf{x}) = \{\mathbf{h}(\mathbf{x}), \mathbf{h} \circ (\mathbf{f}(\mathbf{x})), \mathbf{h} \circ (\mathbf{f}^2(\mathbf{x})), \dots, \mathbf{h} \circ \mathbf{f}^k(\mathbf{x}), \dots\} \quad (2.3.9)$$

Points $\bar{\mathbf{x}}$ and \mathbf{x} are considered to be *indistinguishable* if $\mathcal{O}(\mathbf{x}) = \mathcal{O}(\bar{\mathbf{x}})$. Thus the set of points that are indistinguishable from $\bar{\mathbf{x}}$ can be defined as

$$\mathcal{L}(\bar{\mathbf{x}}) = \{\mathbf{x} \mid \mathcal{O}(\mathbf{x}) = \mathcal{O}(\bar{\mathbf{x}})\}. \quad (2.3.10)$$

The system is then said to be observable at $\bar{\mathbf{x}}$ if $\mathcal{L}(\bar{\mathbf{x}}) = \{\bar{\mathbf{x}}\}$, and unobservable otherwise. Local observability can be defined in the situation when \mathbf{x} is indistinguishable from a set of isolated points $\bar{\mathbf{x}}^i, \forall i \in \{1, \dots, n_{points}\}$, then the system is locally observable about \mathbf{x} . The distance between \mathbf{x} and the closest indistinguishable point provides a measure of the minimum size of the observable subspace.

2.3.2 Fisher's Information and Cramer-Rao Lower Bounds

To determine how well the EKF performs in comparison to the optimal performance possible, Cramer-Rao Lower Bounds (CRLB) and Fisher's Information are used. The formulation and computation of the Cramer Rao Lower Bounds and Fisher's Information has been adapted from [69] and [11].

Consider a nonlinear discrete time state estimation problem with a noise free process model

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k) \quad (2.3.11)$$

and a noisy measurement model

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{w}_k, \quad (2.3.12)$$

where \mathbf{w}_k is zero mean Gaussian white noise with covariance matrix \mathbf{R}_k . Assume that an unbiased initial estimate $\hat{\mathbf{x}}_0$ of the state at time step $k = 0$ is available with covariance matrix \mathbf{P}_0 . The Cramer-Rao Lower Bounds then provide an assessment of optimal accuracy in estimation of the state \mathbf{x}_k based on the observations

$$\mathbf{Z}_k = \{\hat{\mathbf{x}}_0, \mathbf{z}_0, \mathbf{z}_1, \dots, \mathbf{z}_k\}. \quad (2.3.13)$$

The Cramer-Rao results state that the covariance matrix \mathbf{C}_k for an unbiased estimator $\hat{\mathbf{x}}_k$ of the true state \mathbf{x}_k is bounded from below by the inverse of Fisher's Information Matrix \mathbf{J}_k , [11], [69]. These lower bounds are expressed as

$$\mathbf{C}_k = \mathcal{E}[(\hat{\mathbf{x}}_k - \mathbf{x}_k)(\hat{\mathbf{x}}_k - \mathbf{x}_k)^T] \geq \mathbf{J}_k^{-1}, \quad (2.3.14)$$

where Fisher's Information Matrix is defined as the conditional expectation

$$\mathbf{J}_k = \mathcal{E} \left[\left(\frac{\partial \ln p(\mathbf{Z}_k | \mathbf{x}_k)}{\partial \mathbf{x}_k} \right)^T \left(\frac{\partial \ln p(\mathbf{Z}_k | \mathbf{x}_k)}{\partial \mathbf{x}_k} \right) \middle| \mathbf{x}_k \right], \quad (2.3.15)$$

and $\mathbf{p}(\mathbf{Z}_k | \mathbf{x}_k)$ is the joint conditional probability density function of \mathbf{Z}_k given \mathbf{x}_k . The matrix inequality $\mathbf{C}_k \geq \mathbf{J}_k^{-1}$ is equivalent to stating that $(\mathbf{C}_k - \mathbf{J}_k^{-1})$ is positive semi-definite. Since the diagonal elements of a positive semi-definite matrix are non-negative, the diagonal elements of \mathbf{J}_k^{-1} provide the estimate error variance lower bounds for the corresponding elements of $\hat{\mathbf{x}}_k$.

Due to the Gaussian nature of the measurement noise \mathbf{w}_k assumed, the calculation of the Fisher's Information Matrix \mathbf{J}_k is reduced to an iterative process with initialization

$$\mathbf{J}_0 = \mathbf{P}_0^{-1} + \mathbf{H}_0^T \mathbf{R}_0^{-1} \mathbf{H}_0, \quad (2.3.16)$$

and time updates

$$\mathbf{J}_{k+1} = \mathbf{F}_k^{-1T} \mathbf{J}_k \mathbf{F}_k^{-1} + \mathbf{H}_k^T \mathbf{R}_k^{-1} \mathbf{H}_k, \quad (2.3.17)$$

where

$$\mathbf{F}_k = \left. \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_k}, \quad (2.3.18)$$

and

$$\mathbf{H}_k = \left. \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\mathbf{x}_k}. \quad (2.3.19)$$

Note that CRLB are primarily used to address the question, "If the system follows a known trajectory through the state space and produces the defined noisy measurements, what is the lower bound on the variance for the estimation error for each element of the state vector over this trajectory?" This is based on the assumption that the trajectory is known resulting from the noise free process model, which is why the Jacobians in (2.3.18) and (2.3.19) are computed using the true state \mathbf{x}_k , unlike the EKF which uses $\hat{\mathbf{x}}_k$. An important property of Fisher's Information is that in the absence of prior information or when $\mathbf{J}_0 = 0$, if \mathbf{J}_k remains singular, then the system has a unobservable subspace along this trajectory. This implies that system carries almost no information about the parameter that \mathbf{x}_k is dependent on, which is \mathbf{z}_k , and the lower bound for the covariance is extremely high, indicating high uncertainty. The reader is directed to Appendix C of [69] for a detailed understanding of the Cramer-Rao Covariance Bounds.

2.3.3 Requirements for observable 2D SLAM

The autonomous, world centric SLAM problem as formulated and summarized in Section 2.1.5, can now be examined for observability. The authors of [39] showed that the world-centric SLAM problem is unobservable; the authors of [37] analyze the issue using differential geometry, and arrive at the number of unobservable states in the nonlinear SLAM system. A mathematical explanation adopted from [70], as described and used by Khan in Section 3.1.1 of [11], proves that the number of unobservable degrees of freedom in a 2D SLAM problem is 3. Using distinguishability of states, the proof in Section 3.1.1 of [11], demonstrates that the sequence of measurements cannot uniquely distinguish between the different initial states $(\tilde{\mathbf{x}}_0^v, \tilde{\mathbf{x}}_0^i)$ and $(\bar{\mathbf{x}}_0^v, \bar{\mathbf{x}}_0^i)$, under an arbitrary rotation matrix $\mathbf{C}(\beta)$ and an arbitrary transformation vector \mathbf{T} . Thus, there are three degrees of freedom, two through \mathbf{T} and one through $\mathbf{C}(\beta)$. This establishes that the SLAM problem's unobservable subspace has a dimension of three or that it has three unobservable states.

Figure 3 (brought forward from Subsection 2.1.1) demonstrates the SLAM problem and the observability issue. Consider the vehicle pose $[x_k^v, y_k^v, \theta_k^v]^T$ and all landmark positions $[x_k^i, y_k^i]^T$, which are defined relative to the world centric reference frame F_w . All notations in bold in Figure 2 indicate that they are with respect to F_w . The second frame of reference is the vehicle centric reference frame or F_v , and all range and bearing measurements, $[r_k^i, b_k^i]^T$ are obtained with respect to F_v . The SLAM problem aims to estimate the pose of the vehicle and that of the landmarks in F_w while using sensory information obtained with respect to F_v . The measurements obtained however are not world centric, because they do not return absolute positions of the features sensed, but measurements relative to the position of the vehicle. Intuitively, this implies that there is no unique mapping from the measurements obtained to the

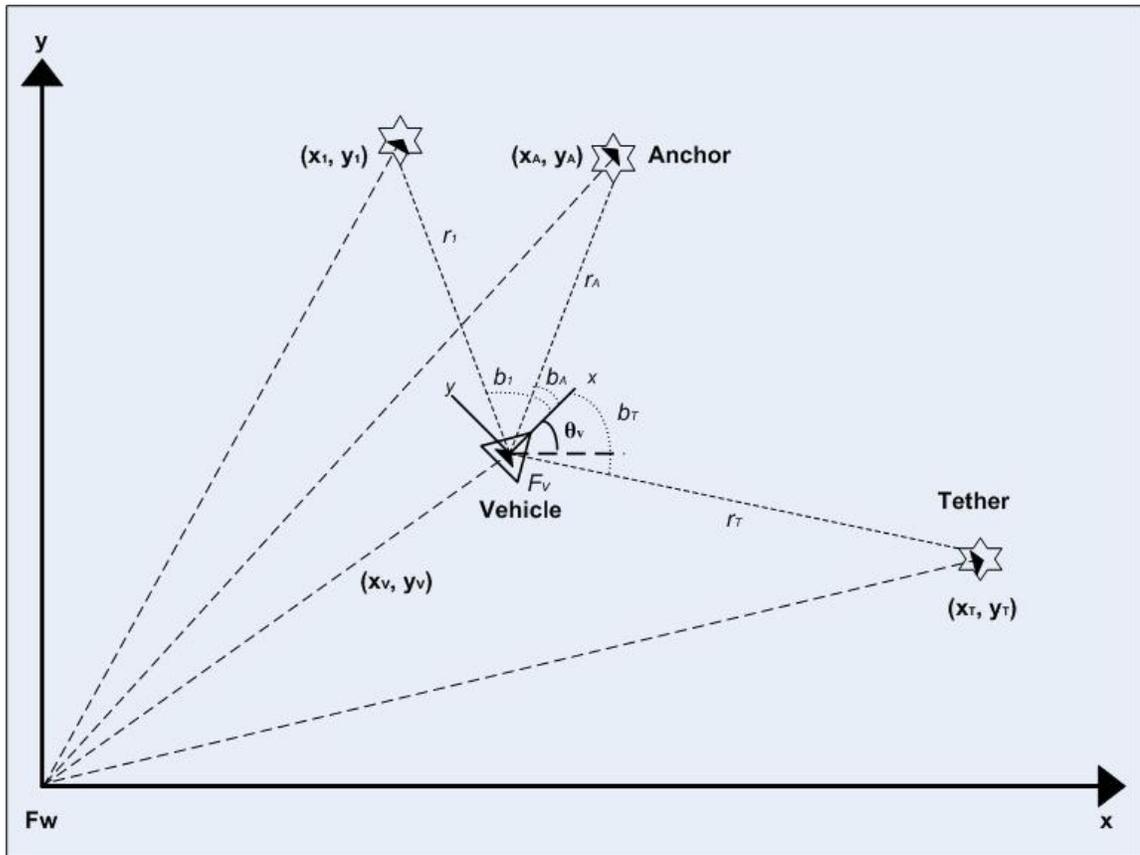


Figure 3: Relationship between World Centric and Vehicle Centric Reference Frames

absolute position of the landmarks or the vehicle in the world centric frame. The position of the vehicle expressed in F_w is the transformation vector between F_w and F_v , however, for the transformation to take effect, the origin of F_w must be defined. If the origin and orientation of F_w is not fixed, then there is no way to guarantee that measurements obtained in F_v pertain to a certain location in F_w . This is because a given measurement is expressed in F_v , and to express the measurement with respect to F_w , a transformation between the two reference frames, F_v and F_w must be defined. If no transformation is defined, then the definition of F_w is ambiguous, as there are infinite world centric reference frames to choose from. Three constraints are required to define F_w .

2.3.4 Imposing Observability

In the part of literature that did focus on observability of SLAM, there have been differences in how many unobservable states the SLAM system has, thus modifying the solutions for the problem. The authors of [39], proposed that a two dimensional SLAM problem could be made observable if the co-ordinates of one landmark in the world centric frame were known. However, as Khan of [11] and Lee et al. of [37] point out, knowledge of world centric positions of one landmark, does not render the problem observable. Lee et al. of [37] prove that knowledge of world centric positions of two landmarks makes the system observable. However, this requires prior information of the distance between two known landmarks. Khan [11], showed that the 2D SLAM problem could be made observable when three co-ordinates of two landmarks in the world centric frame are known.

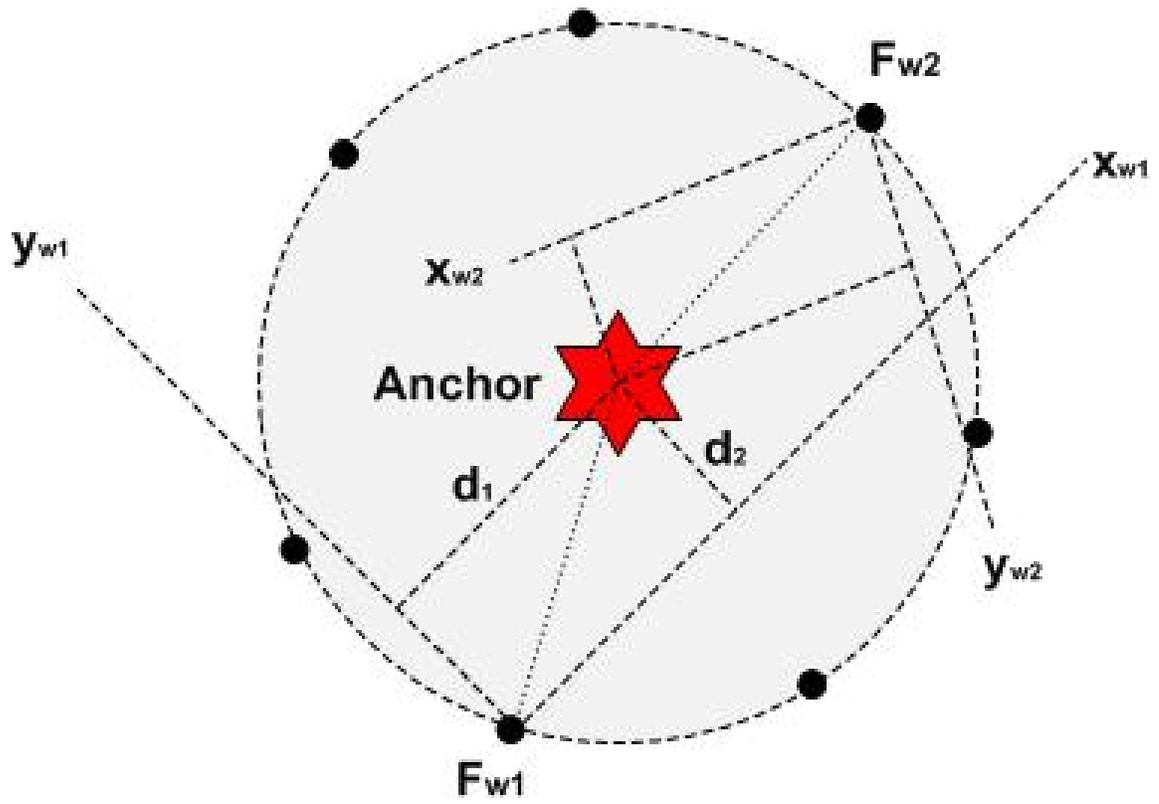


Figure 4: Impact of using only an Anchor to define F_w

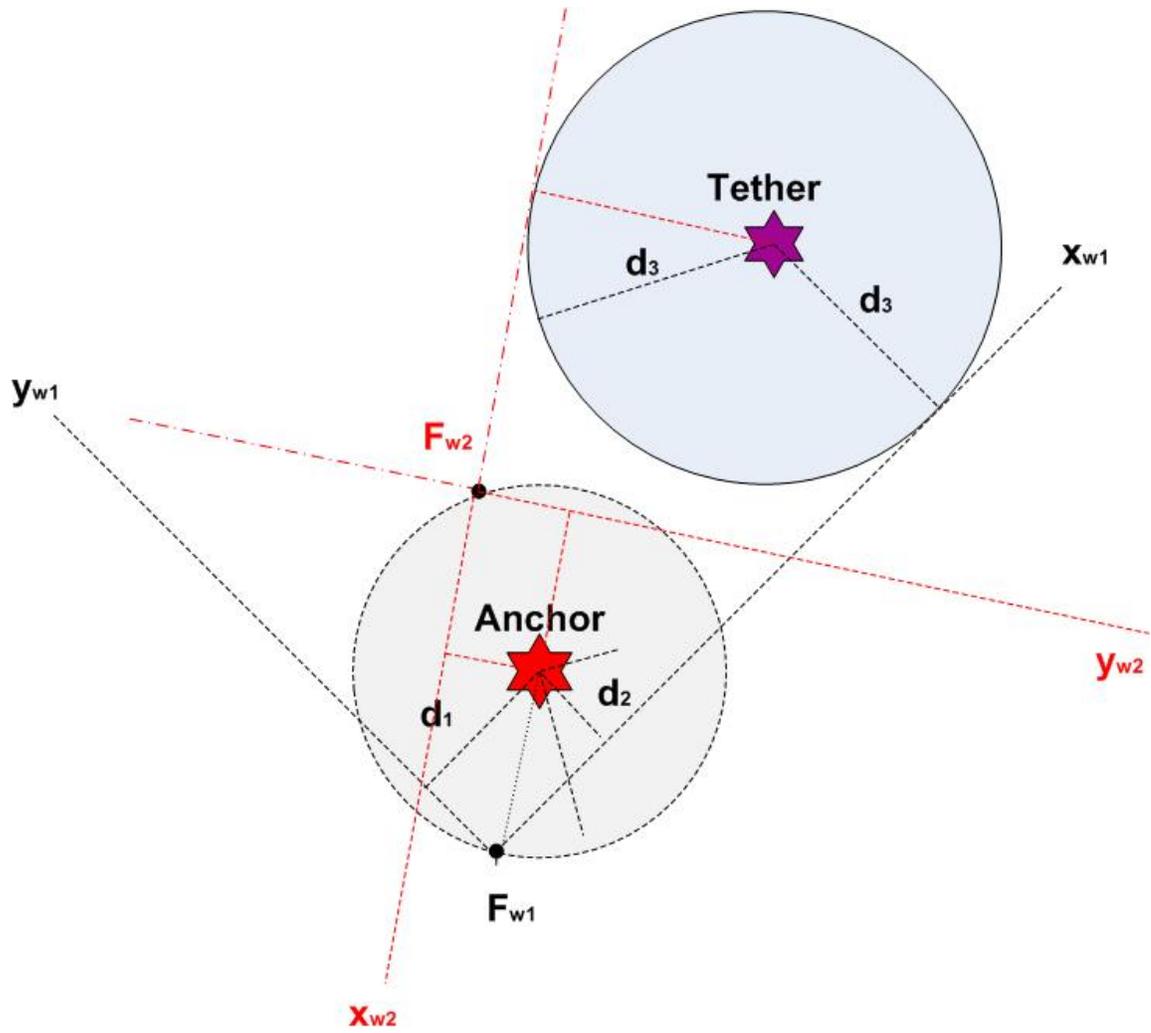


Figure 5: Impact of using an Anchor and Tether to define F_w

In the absence of knowledge of the initial vehicle position and of the world frame of reference in an autonomous SLAM problem, sensor readings from two landmarks are necessary to initialize a world frame of reference to make the system observable, and it is necessary to have at least two landmarks in sight to maintain observability. This imposes 3 constraints by holding constant 2 coordinates of one landmark and 1 coordinate of the other. The landmark whose two co-ordinates are fixed, and not estimated is referred to as an "anchor". The landmark whose one co-ordinate is fixed and not estimated, is called a "tether". The notations $\{d_1, d_2\}$ representing the position of the anchor in Figure 4 and in Figure 5, and $\{d_3\}$ representing the single coordinate of the tether that is fixed, are signed quantities.

As shown in Figure 4, fixing 2 coordinates of the anchor now restricts the origin of the newly created F_w , to lie anywhere on a circle whose radius is the distance between the origin of F_w and the anchor. Two possible F_w 's are shown and the black dots on the circle indicate that any position on the circle could possibly be the origin of F_w . Fixing a coordinate of the tether, narrows the placement of the origin from anywhere on the circle to two possible solutions, both of which are valid; both solutions are shown in Figure 5. In Figure 5, the y coordinate of the tether in the form of d_3 is fixed.

The reader is referred to Section 3.3 of [11] for a detailed description of how knowledge of three coordinates of two landmarks can be used for the autonomous initialization process that has been used in simulations. It is important to note the difference between initialization in practical applications and in simulation environments. Since simulated environments have knowledge of F_w , the true positions of the anchor and tether are used by the initialization routine. In practical applications, since there is no pre-fixed F_w , it is often preferred to assume the position of the anchor as the origin of F_w , with the tether assumed to be on either the x or y axes, and then proceeding with the initialization procedure described in Section 3.3 of [11].

2.4 Dempster-Shafer Evidential Theory and Occupancy Grids

Of all algorithms that handle mapping with known robot poses, occupancy grid maps, originally developed by the authors of [71], [72] and [73], have received popularity, because of their use of probabilistic techniques, their discretization of the environment into grid-cells and their generation of a consistent map from noisy sensor measurements [43]. The grids may be 3 dimensional, but usually tend to be 2 dimensional [60], [12]. Occupancy grid maps fundamentally use two concepts- to represent information and to effectively combine evidence from new sources to update the previously represented information in the map. The method of combining existing data with newly arrived data has been dominated by Baye's rule of combination, until an alternative method was brought to the forefront by the authors Arthur. P. Dempster of [74] and Glenn Shafer of [75], [76], called Dempster-Shafer (DS) theory of evidential reasoning. Yang of [12], [60] analyzed DS theory for occupancy grids under the assumption that the localization problem was solved. The use of Bayesian theory in grid based maps is reviewed, along with simulations in small scale, office-like environments.

The Dempster-Shafer (DS) theory, also known as the theory of belief functions, is a generalization of the Bayesian theory of subjective probability [75]. Probability's interpretation is subject to two schools of thought- one that attributes probabilities for a proposition, and another that attributes the probabilities for questions that lead to formulating, and thus interpreting the belief in the proposition. The DS theory was seen as a substantial idea because of its two fold functionality- the idea of obtaining degrees of belief for a proposition from subjective probabilities for related propositions, and Dempster's rule for combining such degrees of belief when they are based on independent items of evidence [75]. This created a powerful alternative

to the conventional Bayesian theory of combining new information to existing ones, to update the proposition's state. The remainder of this section focuses on an introduction to Dempster-Shafer theory, its comparison to, and a brief introduction to Bayesian theory.

2.4.1 Introduction to DS theory

The DS theory is modelled by considering a frame of discernment θ , consisting of all singleton propositions. In the case of grid based cells, the following singleton propositions exist

$$\theta = \{E, F\} \tag{2.4.1}$$

where E and F are the emptiness and fullness propositions respectively. The power set containing all possible combinations of elements of θ is

$$\Xi = 2^{|\theta|} = \{E, F, \theta, \phi\} \tag{2.4.2}$$

where ϕ indicates that no proposition is applicable to the cell and θ indicates that either E or F is applicable to the cell. The propositions ϕ and θ will be denoted by *neither* and *EorF* respectively. Each cell of the grid map is indexed by its row and column indices i and j , and all equations described henceforth are applicable to every cell in the grid, and thus cell indices i and j will be dropped. Every cell is assigned with a certain probability value for each proposition from the power set, called the mass or the basic probability assignment (bpa) of the cell and have to adhere to the following constraints

$$\sum_{A \in \Xi} m_A = m_E + m_F + m_{neither} + m_{EorF} = 1 \quad (2.4.3)$$

for cell $\{i, j\}$. Since neither proposition applying to the cell is improbable, $m_\phi = m_{neither} = 0$ leading to

$$\sum_{A \in \Xi} m_A = m_E + m_F + m_{EorF} = 1. \quad (2.4.4)$$

Since the cells being full or empty is determined by what is contained in the cell as well as measurements obtained from sensors, bpa's are assigned to propositions from two sources, one being the map and the other being the sensor. Both the map and the sensor contain propositions from the same frame of discernment θ and the same power set Ξ , however, they contain different information. The four bpa values attributed to every cell in the map corresponding to the four propositions of Ξ - m_E^m , m_F^m , m_{EorF}^m and $m_{neither}^m$ (superscript m denotes map), contain evidence for each proposition and is interpreted as knowledge defining the cell. When the same propositions related to the sensor - m_E^s , m_F^s , m_{EorF}^s and $m_{neither}^s$ (superscript s denotes sensor), are assigned values, they signify the degree of confidence in the sensor's ability to detect emptiness and fullness accurately.

The bpa's for initialization of all cells in the grid take the following general form

- $m_E^m = mass_E$
- $m_F^m = mass_F$
- $m_{neither}^m = 0$
- $m_{EorF}^m = 1 - mass_E - mass_F$.

This assignment suggests a certain weight associated with the trust in each proposition applicable within the cell and eventually the map, based on any previous knowledge about the map. Initialization of the grid or map usually involves assigning low bpa values to the emptiness and fullness propositions, because of lack of knowledge about the map, while correspondingly increasing the bpa value for proposition $EorF$, since a cell has to be either empty or full. It can be seen that the summation of bpa values adhere to the probability constraint in equation (2.4.3).

Assigning values to the same propositions for evidence obtained from the sensor uses the following general form

1. If the sensor detects a cell as empty, assign

- $m_E^s = \lambda_E$,
- $m_F^s = \alpha_E$,
- $m_{neither}^s = 0$,
- $m_{EorF}^s = \lambda_{EorF} = 1 - \lambda_E - \alpha_E$.

2. If the sensor detects a cell as full, assign

- $m_E^s = \alpha_F$,
- $m_F^s = \lambda_F$,
- $m_{neither}^s = 0$,
- $m_{EorF}^s = \lambda_{EorF} = 1 - \lambda_F - \alpha_F$.

The summation of bpa values adhere to the constraint in (2.4.3). The bpa values or sensor support values assigned to m_E^s and m_F^s depend on the accuracy of the sensor. In the case of an empty cell being detected, if the sensor is considered accurate, m_E^s is assigned a high value (<1), and m_F^s receives a bpa value of $\alpha_E = 0$ signifying that the

sensor is reliable and could not have inaccurately detected a full cell as being empty. The value of m_{EorF}^s then becomes $1 - \lambda_E$ and the same procedure is used to assign bpa values for sensory information detecting a full cell. Since real world sensors are seldom accurate enough to assign a high bpa value to emptiness or fullness exclusively and 0 to fullness or emptiness correspondingly, a high bpa is assigned to m_{EorF}^s and low values to both m_E^s and m_F^s .

The newly arrived sensor data can be recursively combined with previously initialized map data, by using Dempster's rule of combination given by

$$m_A^m(k+1) = m_A^s(k+1) \oplus m_A^m(k) = \frac{1}{1-K} \sum_{B \cap C = A \neq \phi} m_B^s m_C^m, \quad (2.4.5)$$

where k is the time step.

With respect to the grid, each proposition in the power set Ξ , undergoes an information update using Dempster's rule of combination as specified in (2.4.5), and the new value is assigned to the proposition with respect to the map or m_A^m . In (2.4.5), the normalizing factor $1 - K$ takes into account all evidence(masses) from two sets of sources, that favour the proposition A while attributing any contradicting evidence to the null set through K , which is given by

$$K = \sum_{B \cap C = \phi} m_B^s m_C^m. \quad (2.4.6)$$

Bpa values merely assign a proportional value of evidence that favours a proposition A while not claiming any knowledge of the subset of the proposition. The three main functions that enable DS theory to produce higher levels of interpretation are the belief, doubt and plausibility functions. The belief function expresses the level of confidence in the evidence gathered for a proposition. The belief function is related to the bpa as follows

$$Bel(A) = \sum_{B|B \subseteq A, \forall A \subseteq \theta} m_B^m. \quad (2.4.7)$$

The above relation shows that the belief function combines evidence from all sources that contribute towards the proposition being true. The plausibility function is defined as follows

$$Pl(A) = 1 - Bel(\bar{A}) = 1 - Dou(A) = \sum_{B|B \cap A \neq \phi, \forall A \subseteq \theta} m_B^m. \quad (2.4.8)$$

The plausibility function of proposition A is defined as the resulting confidence, after confidence in evidence that explicitly supports contradicting propositions is taken away from the total degree of confidence. Thus for a proposition A , the belief and the plausibility functions form lower and upper bounds on the true probability of the proposition respectively

$$Bel(A) \leq P(A) \leq Pl(A). \quad (2.4.9)$$

2.4.2 Comparison with Bayesian theory

For most grid based maps, Bayesian theory is the basis of representing information and incorporating new information into the current map. A reminder to the Bayesian framework is now provided with respect to grid based maps.

The events of concern in a grid based map are that of fullness (denoted by subscript F) and emptiness (denoted by subscript E), whose outcome will be judged by their probability assignments. Assuming that all computations take place for every cell i, j in the grid, cell indices as subscripts will be dropped. Consider

$$0 \leq P_F \leq 1, \quad (2.4.10)$$

$$0 \leq P_E \leq 1, \quad (2.4.11)$$

$$P_F + P_E = 1. \quad (2.4.12)$$

Numerical measures showing support for emptiness and fullness for each cell are represented in the form of probabilities of each event occurring. Since the numerical measures are probabilities, the values adhere to (2.4.12). For every time step $t, \forall t \in 1 : n_{time}$, as new sensor data (denoted by s) is obtained, Bayes theorem is used to incorporate it with existing knowledge

$$P(F_t|s_t) = \frac{P(s_t|F_t)P(F_t)}{P(s_t)} \quad (2.4.13)$$

$$= \frac{P(s_t|F_t)P(F_t)}{P(s_t|F_t)P(F_t) + P(s_t|E_t)P(E_t)}. \quad (2.4.14)$$

The above equation is updated recursively for each event, by setting

$$P(F_t) = P(F_{t-1}|s_{t-1}). \quad (2.4.15)$$

When a priori values are not available, during initialization of cells for instance, an unbiased value of 0.5 is assigned to the cell. This implies that the cell has a 50% chance of being occupied and 50% chance of being empty, so an unbiased equal chance of either proposition being true. Eventually, due to the Bayesian update, all visible cells should have their numerical measures for either proposition approach a value of 1, depending on whether they are empty or full.

Analyzing the initialization values, it reveals that the values may convey ambiguous interpretations; a value of 0.5 for a cell could mean either of the following - the cell has not been scanned yet, the cell's value has been updated to a degree that now has a 50% chance of being full or empty, the sensory measurements have noise that allowed for erroneous updates or that the cell is literally, half occupied and half empty. Even if it was assumed that the cell was not scanned yet, this implies that the cell is being assigned a 50% chance of either proposition, when there is no evidence suggesting either. So while the Bayesian method provides a good form of recursive updating, it does not provide a unique form of interpretation. Authors Yang et al. of [77] have provided modifications to the basic Bayesian update methods to aid in clearer interpretation, by defining a conditional probability that takes into account sensor errors, in the following form

$$p_c = P(\text{obstacle} = 1 | \text{obstacle present} = 1) \quad (2.4.16)$$

$$p_f = P(\text{obstacle} = 1 | \text{obstacle present} = 0) \quad (2.4.17)$$

Thus p_c and p_f give the correctness of obstacle detection in the cell correspondingly, and this can then be factored into the fullness and emptiness support to increase the accuracy of the numerical measures of fullness and emptiness. Additionally, the authors also included a factor that keeps track of whether each cell has been scanned, and as the number of times the cell has been scanned increases, the factor's value approaches 1.

Improvisations to the Bayesian model can yield better maps, however, the Dempster-Shafer theory of evidential reasoning has a more natural framework that attributes interpretations to numerical values. Consider the basic difference in analyzing propositions and assessing how feasible it is for all of them to occur - while the Bayesian theory assigns a numerical measure for each proposition, the DS theory

assigns a numerical measure for all questions that potentially lead to answers about each proposition. Manipulation of these answers then leads to various states of interpretation about the event, and each state considers evidence in favour of and against the event. The information that Bayesian theory considers as actual probability, DS theory breaks down into evidence, and then deduces a level of belief in that event, based on the evidence accrued. Thus, there can be no ambiguity in the interpretation of belief in different states, because all evidence has been accounted for, and there is also a measure of lower and upper bounds in the form of the belief and plausibility functions, between which the true probability of the event is bounded. Thus for data fusion in grid based maps, the DS theory has a natural advantage over Bayesian theory.

Chapter 3

Observable EKF SLAM Simulations and Results

Simulations have been conducted in two representative environments to demonstrate different aspects of the observable autonomous SLAM problem in 2 dimensions. The results are analyzed using four cases, each of which analyzes factors that cause fluctuations in the EKF's performance as an observer for the SLAM problem. A brief reminder of the cases previously considered in literature is given, followed by simulation parameters and an overview of the cases that are discussed in this section.

Authors Lee et al. of [37] and Khan of [11] have shown the autonomous SLAM problem devoid of prior landmark knowledge is unobservable with three unobservable degrees of freedom through simulations. Since it has been proved that knowledge of at least three coordinates of two landmarks is necessary for the problem to be observable [69], [11], all simulations conducted for this thesis assume that at least two landmarks are in the field of view of the vehicle at all times.

3.1 Simulation parameters

The EKF algorithm comprises of a simple range and bearing measurement model to available point landmarks. Additive Gaussian-white noise is added to both range and bearing measurements obtained to simulate real world values. Before the initialization procedure is executed for anchor and tether selection, measurements are obtained and averaged for 100 readings (the value 100 is chosen with an intent to reduce initial measurement errors). This is then followed by anchor and tether selection and initialization. All landmarks are constantly evaluated based on their distance to the vehicle (distance test) and the number of iterations for which they have been in the filter (time test). When either of the anchor-tether pair fails the distance or time tests, a re-selection is triggered, allowing for the landmark with the best success rate from the tests to become the anchor or tether. Reselecting also employs a test to determine if the landmarks are horizontal or vertical in the world frame of reference, thus allowing for switching in the estimation of the tether's coordinate.

The values used are similar to values used in SLAM literature, and is specifically adopted from [11].

The simulation uses the following parameters during initialization.

1. Kalman Covariance

The Kalman covariance is initialized to high values signifying low levels of certainty in the knowledge of the vehicle's pose

$$\mathbf{P}_v = \text{blkdiag}(5\text{m}^2, 5\text{m}^2, 5\text{rad}^2). \quad (3.1.1)$$

Similarly, a new landmark i also receives a high degree of uncertainty corresponding to its position and its covariance values are initialized to

$$\mathbf{P}_i = \text{blkdiag}(5\text{m}^2, 5\text{m}^2). \quad (3.1.2)$$

2. Vehicle Process Covariance

The covariance associated with the noise of the vehicle process model is given by

$$\mathbf{Q}_{veh} = \text{blkdiag}(0.01\text{m}^2, 0.01\text{m}^2, 0.04\text{rad}^2). \quad (3.1.3)$$

3. Landmark Process covariance

The covariance associated with the noise of the landmark process model for landmark i is given by

$$\mathbf{Q}_i = \text{blkdiag}(0.03\text{m}^2, 0.03\text{m}^2). \quad (3.1.4)$$

The covariance for all landmarks $i = 1 : n$ in the field of view are given by

$$\mathbf{Q}_{lm} = \text{blkdiag}(\mathbf{Q}_1, \mathbf{Q}_2, \dots, \mathbf{Q}_n). \quad (3.1.5)$$

The complete process model covariance is given by

$$\mathbf{Q} = \text{blkdiag}(\mathbf{Q}_{veh}, \mathbf{Q}_{lm}). \quad (3.1.6)$$

4. Measurement Model Covariance

Assuming reasonable sensor accuracy, measurement model covariance for landmark i is initialized as

$$\mathbf{R}_i = \text{blkdiag}(0.02\text{m}^2, 0.05\text{rad}^2) \quad (3.1.7)$$

and the complete measurement model covariance for all landmarks $i = 1 : n$ that are in field of view is given by

$$\mathbf{R} = \text{blkdiag}(\mathbf{R}_1, \mathbf{R}_2, \dots, \mathbf{R}_n). \quad (3.1.8)$$

5. Vehicle Linear and Angular velocities

(a) Straight Ladder Environments

$$\mathbf{u} = \begin{bmatrix} 2\text{m/s} \\ 0 \end{bmatrix} \quad (3.1.9)$$

(b) Circular Ladder Environments

$$\mathbf{u} = \begin{bmatrix} 2\text{m/s} \\ 0.05\text{rad/s} \end{bmatrix} \quad (3.1.10)$$

6. Sample period

$$T = 0.05\text{s} \quad (3.1.11)$$

7. Maximum Sensor Range

The thesis assumes that the sensor has a finite sensing range with a 360° field of view, but takes on different values of maximum sensor range for each case to

simulate the entry to and exit of landmarks from the field of view of the sensor.

(a) **Cases 1 and 2**

range = 200m bearing =360°

(b) **Case 3**

range = 80m bearing =360°

(c) **Case 4**

range = 55m bearing =360°

3.2 Overview of Simulation environments and Results

To demonstrate different concepts of the SLAM problem, an illustrative straight ladder environment as shown in Figure 6, and a circular ladder environment as shown in Figure 7 are used, where the vehicle is indicated by the black triangle. Each environment's description in terms of location of landmarks has also been provided in Table 1 and Table 2. Both environments used are adopted from [11], but are modified to include surrounding walls. The environments are reasonably large, with the straight ladder environment covering an approximate area of $200\text{m} \times 200\text{m}$ and the circular ladder covering an area of $140\text{m} \times 140\text{m}$. A brief introduction to the 4 cases that have been analyzed are given as follows, followed by a detailed analysis of each case.

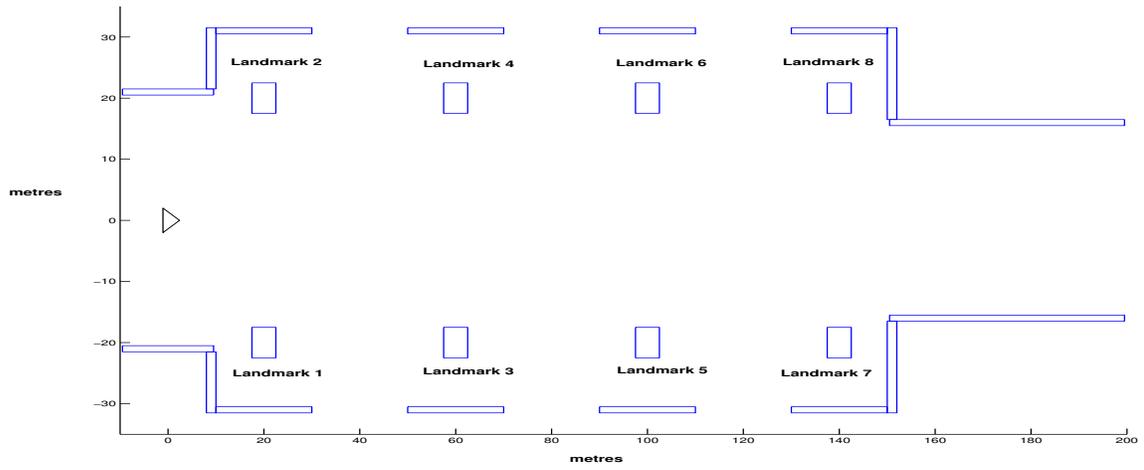


Figure 6: Straight Ladder Environment

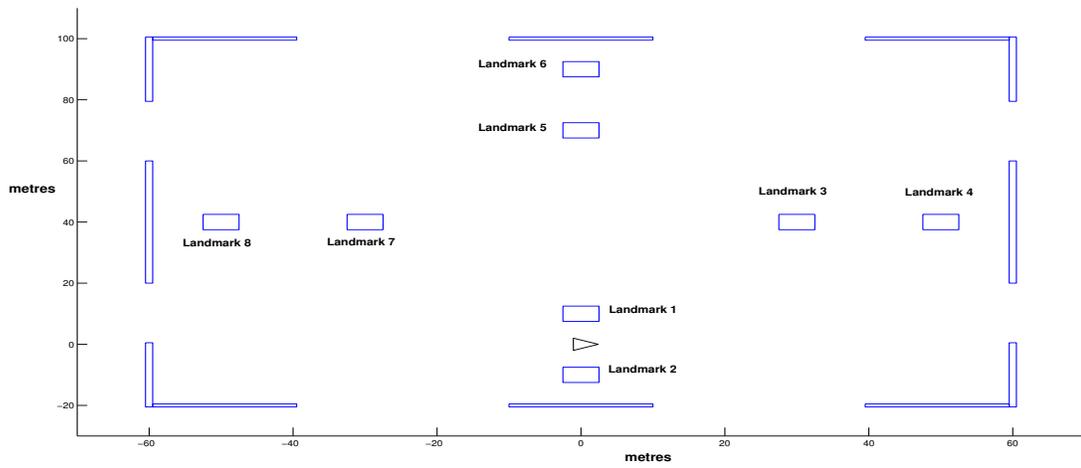


Figure 7: Circular Ladder Environment

Table 1: Straight Ladder Environment Spatial Description

Landmark Index	x coordinate in 'm'	y coordinate in 'm'
1	20	-20
2	20	20
3	60	-20
4	60	20
5	100	-20
6	100	20
7	140	-20
8	140	20

Table 2: Circular Ladder Environment Spatial Description

Landmark Index	x coordinate in 'm'	y coordinate in 'm'
1	0	10
2	0	-10
3	30	40
4	50	40
5	0	70
6	0	90
7	-30	40
8	-50	40

- **Case 1: Accuracy in Large Environments**

Using the straight ladder environment and by choosing landmarks 1 and 2 as the anchor and tether respectively, demonstrate the need for reselecting anchor and tether as the vehicle travels through a large environment.

- **Case 2: Observability Issues with Tether Estimation**

Using the straight ladder environment and by choosing landmarks 1 and 2 as the

anchor and tether respectively, demonstrate the need for switching the coordinate of estimation of the tether from x to y , and vice versa. Fisher's Information Matrix is used to analyze observability based on anchor-tether orientations in F_w .

- **Case 3: Observable Reconfigurable EKF SLAM**

Demonstrate the benefits of enabling automated re-selection of anchors and tethers, and automated switching of the estimated coordinate of the tether in the straight ladder environment.

- **Case 4: Limitations of Anchor-Tether Re-selection**

Demonstrate the benefits of enabling automated re-selection of anchors and tethers, and automated switching of the estimated coordinate of the tether in the circular ladder environment. Also examine possible drift in estimates through repeated re-selection with multiple passes through the environment.

Since both environments are large and considering a simple range and bearing sensor model, it is important to note the impact of range and bearing in F_v on the Cartesian error in F_w . While noise in the range and bearing measurements is assumed to have a constant covariance, the effective Cartesian error due to bearing uncertainty still depends on the distance between the vehicle and the landmark, as seen in Figure 8. Observing landmarks 2(a) and 2(b), it can be seen that with the same bearing value b_2 for both landmarks, the distance between the vehicle and the landmarks given by $r_{2(a)}$ and $r_{2(b)}$ have an impact on the error in Cartesian estimates. The impact is evident in the y coordinate for landmark 2(a) as it is farther away from the vehicle than landmark 2(b) is. For landmarks 1 and 3, the Cartesian error is higher in the x and y coordinates respectively. With the range values to landmarks 1 and 3- r_1 and r_3 being equal, the impact of vehicle-landmark orientation on Cartesian error is evident, as it influences the decrease in estimation accuracy in

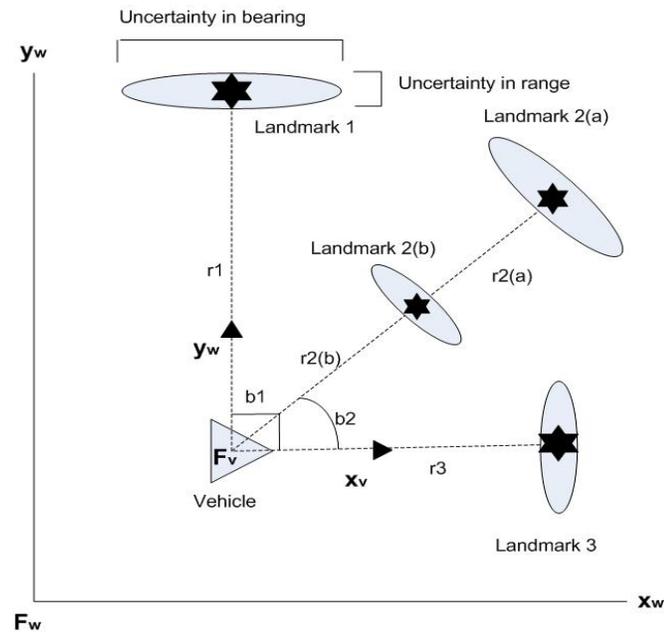


Figure 8: Effect of Range and Bearing Sensor Errors on Cartesian Error

the x and y directions respectively.

3.3 Case 1: Accuracy in Large Environments

The simulation is conducted in the straight ladder environment, with landmark 1 chosen as the anchor and landmark 2 chosen as the tether for the entire duration of the simulation, with no re-selection of the anchor and tether. With initial selection of anchor and tether, the coordinate of the tether that is estimated is y, as the anchor-tether pair is vertical in F_w . A detailed analysis of the choice of coordinate of the tether to be estimated depending on the orientation of the anchor-tether pair is given in Section 3.4. The range of the sensor on board the vehicle is assumed to have a range of 200m and a 360° field of view, thus enabling all landmarks in the environment to be in its field of view for the duration of the simulation.

Figure 9 uses rectangles to display landmarks, while the landmarks positions as used in the EKF are points located in the centre of the rectangles. Figure 9 contains

the anchor highlighted in red, the tether highlighted in magenta, the vehicle shown as a black triangle for various time steps, the true path shown in green and the estimated path shown in red. It also contains uncertainty ellipses plotted for each landmark; however, the figure represents the uncertainty ellipse plotted at the vehicle's final position. The uncertainty ellipse may look like a line due to the dimensions of the figure. For trends in the standard deviations of estimates of landmarks, the reader is directed to Figures 11 through 12.

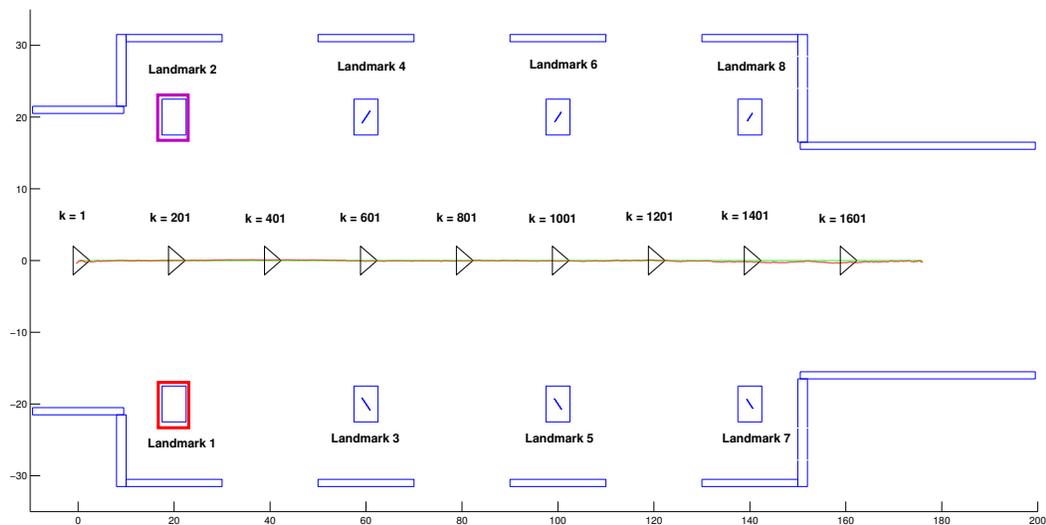


Figure 9: Case 1: Motion of Vehicle through the Straight Ladder Environment

The state error for the anchor- landmark 1, remains 0 throughout the simulation, because both its co-ordinates have been held constant without being estimated by the EKF. Since simulation environments are user defined, positions of all landmarks are known. Thus exact initialization of values has been used for the anchor's coordinates and the tether's x coordinate leading to zero initialization error. Figure 10 shows that the state error for y coordinate of the tether- landmark 2 is non zero, because it is estimated by the EKF.

Figure 10 represents the error in estimation of the vehicle pose and the standard

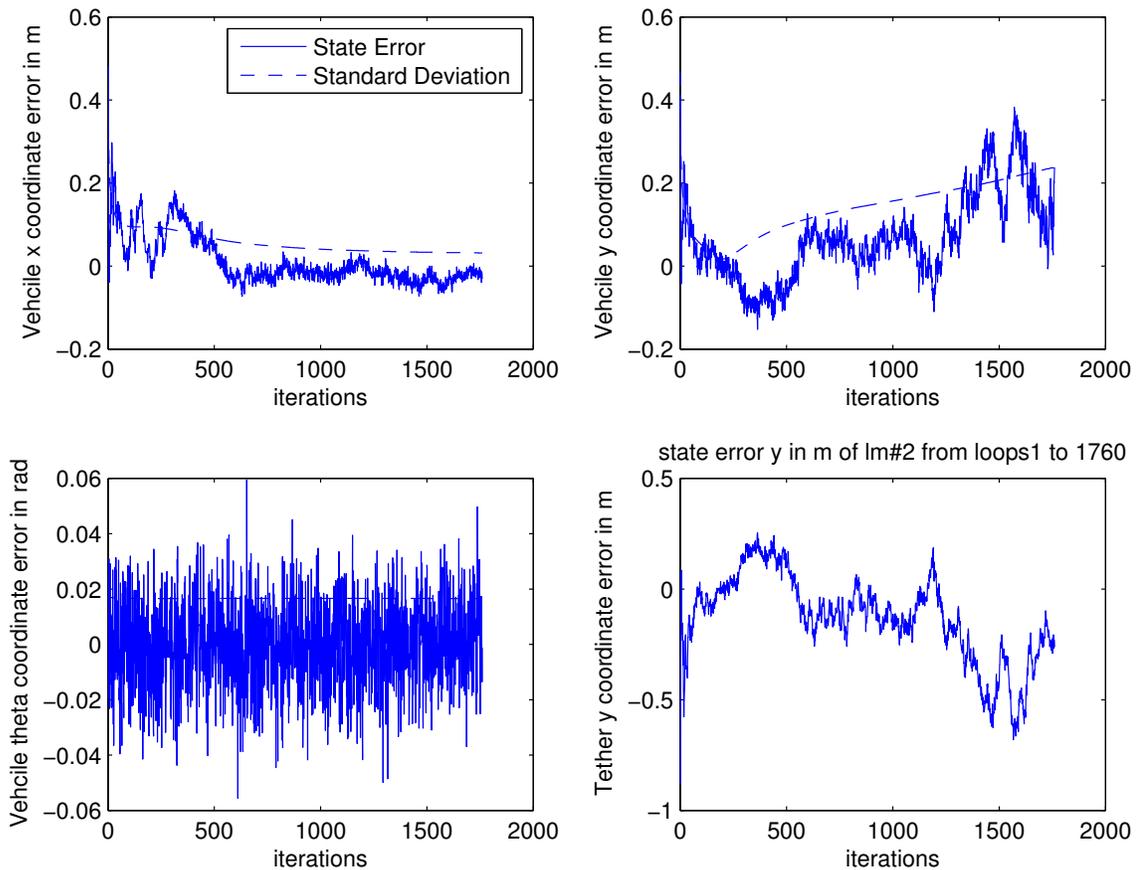


Figure 10: Case 1: State Error, Standard Deviation of Vehicle pose and Tether State Error

deviation. It can be seen that errors are at approximately at 20cm on average for the x and y coordinates, and in the order of 10^{-2} rad for θ for the entire simulation, thus indicating that the EKF is performing well without diverging. It is interesting to note the trends in the error for the x coordinate fluctuate initially before settling to an almost constant level, without seemingly including any visible bias. The error for the y coordinate however, while initially following a similar pattern in fluctuation, gradually seems to accrue a bias. This is because the vehicle moves along the x-axis of F_w and the farther away it gets from the anchor-tether pair, there is greater Cartesian

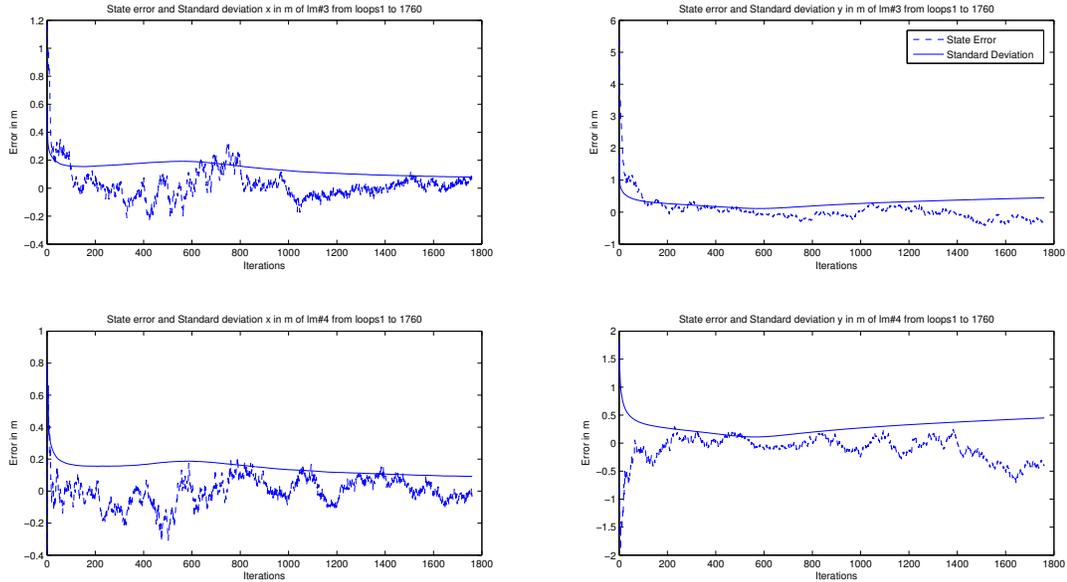


Figure 11: Case 1: State Error and Standard Deviation of Landmarks 3 and 4

error in the y coordinate.

On inspecting Figure 10, the standard deviation of the y coordinate seems to decrease from approximately iterations 0 through 200, which is when the vehicle moves closer towards the anchor and tether, and a steady ascent when the vehicle moves farther away from the anchor and tether, from approximately iterations 200 through 1760. The impact of range and bearing uncertainty on Cartesian error is evident here, as the distance between the anchor-tether pair increases, there is a clear increase in uncertainty in the y coordinate. Thus, as the distance between the anchor-tether pair and the vehicle increases, the more likely it is for the uncertainty in SLAM estimates to grow. To prevent the increase in uncertainty and bound the error estimates, a new anchor and/or tether, must be reselected at appropriate time steps, that will prevent the uncertainty from growing.

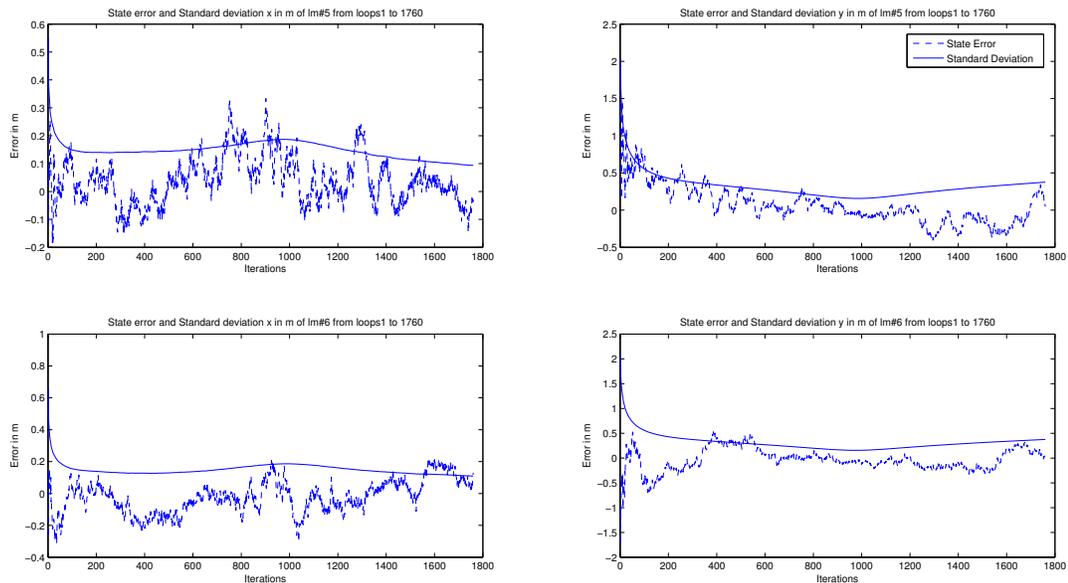


Figure 12: Case 1: State Error and Standard Deviation of Landmarks 5 and 6

State errors and standard deviations for other landmarks are given in Figures 11, 12 and 13. The errors experience a steady correction after their high initialization errors and approach errors within the cm range quickly. Since the vehicle moves along the x-axis of F_w in this case, the vehicle's position at time steps $k = 201, 601, 1001$ and 1401 in Figure 9 is approximately on the vertical line connecting the two landmarks the vehicle is currently passing. In such cases, the error in estimation of the landmarks' x coordinate is higher than the error in estimation of y coordinate. With the exception of the mentioned time steps where the error in estimating the x coordinate may be slightly higher than during the rest of the simulation, all landmarks errors show that the errors in estimating the y coordinate are greater than errors in estimating the x coordinate. As the vehicle moves along the x-axis of F_w and farther away from the tether whose y coordinate is estimated throughout the simulation, there is a constant increase in the Cartesian error in the y direction because of increase in range uncertainty or the relative distance between the vehicle

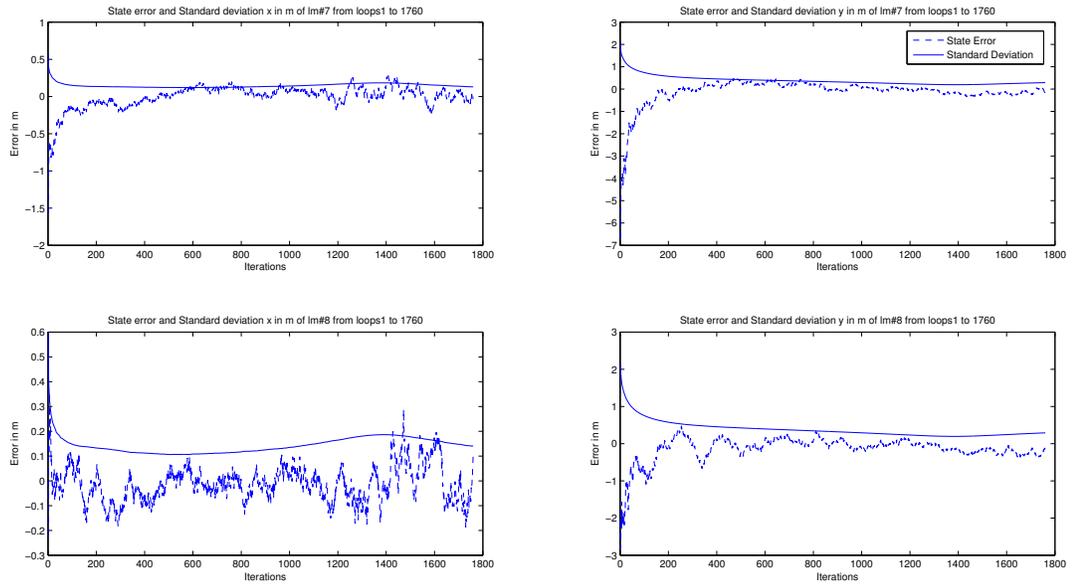


Figure 13: Case 1: State Error and Standard Deviation of Landmarks 7 and 8

and the tether. This error in turn reflects as errors in estimation for landmarks in the y direction.

3.4 Case 2: Observability Issues with Tether Estimation

Subsection 2.3.4 emphasizes the importance of choosing an anchor-tether pair to impose observability. It is however, important to realize that there are cases when imposing three constraints by holding fixed both the anchor's coordinates and either of the tether's coordinates, does not guarantee observability. Unobservability occurs when either of the following takes place.

1. The anchor-tether pair is *horizontal* in F_w and the estimated coordinate of the tether is y, or
2. The anchor-tether pair is *vertical* in F_w and the estimated coordinate of the tether is x.

The terms *horizontal* and *vertical* are defined to be parallel to x-axis and the y-axis of F_w respectively. Items 1 and 2 are analyzed using Fisher's Information Matrix. To test the hypotheses that estimating the y coordinate of the tether when the anchor-tether pair is horizontal in F_w and that estimating the x coordinate of the tether when the anchor-tether pair is vertical in F_w may cause loss of information, and thus lead to unobservability, the following simulation was conducted. Results are shown in Figure 14. The anchor is placed at the origin of F_w (the origin of the frame seen in the central plot) and the tether lies on a circle of radius 40m from the anchor. The tether's position on the circle is stepped in 1 degree increments to complete the circle. The vehicle is placed at 8 different locations, shown as V1-V8 in the central plot in Figure 14. For each of the 8 vehicle positions, the tether's position is varied, thus creating 360 different anchor-tether orientations, and Fisher's Information Matrix is computed for each orientation for 1800 iterations, for either the x or y coordinates of the tether being estimated with $\mathbf{J}_0 = 0$ or no prior information. The minimum

singular value of the Fisher's Information Matrix, which is a measure of the distance from Fisher's Information matrix to a singular matrix, is shown in 8 plots around the central plot. The positions of the surrounding subplots correspond to the 8 positions of the vehicle. For any of the 8 vehicle positions, the following observations are noticed for horizontal and vertical anchor-tether orientations respectively.

- Horizontal anchor-tether orientations corresponding to an angle (AT Angle in the figure) of 0° and 180° in F_w which is represented by the central plot's axes.
 1. The minimum singular value for estimating y (Est Ty in the figure) experiences a sudden drop to 0. This indicates that \mathbf{J} is singular and that there is loss of information from the filter, implying the presence of an unobservable subspace.
 2. The minimum singular value for estimating x (Est Tx in the figure) however, reaches a local maximum, indicating that \mathbf{J} is sufficiently far from singularity.
- Vertical anchor-tether orientations corresponding to angles of 90° and 270° in F_w which is represented by the central plot's axes.
 1. The minimum singular value for estimating x experiences a drop to 0, indicating loss of information.
 2. The minimum singular value of estimating y however, experiences peaks.

Thus, it can be seen that anchor-tether pairs that are vertical and horizontal in F_w which have the x and y coordinate of the tether estimated respectively, still does not guarantee observability, even through the three constraints required are present. The results from Fisher's Information Matrix demonstrate that the 2D SLAM problem as formulated is unobservable even with the presence of three observability constraints, when the incorrect coordinate of the tether is imposed as a constraint.

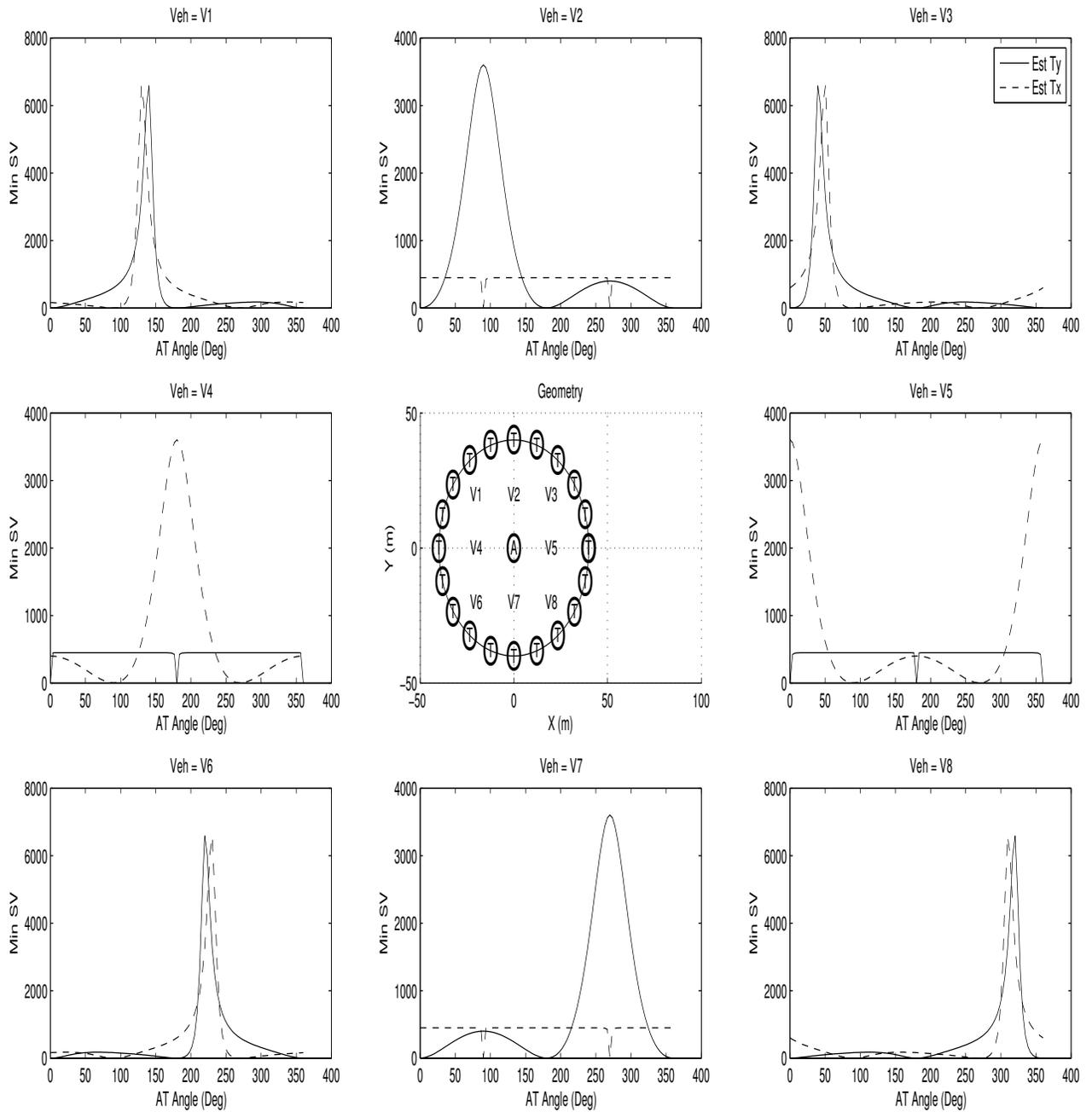


Figure 14: Fisher's Information for varying Vehicle positions and Anchor-Tether orientations

This case demonstrates the performance of the EKF with three observability constraints but with wrong coordinate of the tether being estimated. The simulation emulates a horizontal anchor-tether pair in F_w and estimates the y coordinate of the tether, which is the wrong coordinate of the tether to be estimated when the anchor-tether pair is horizontal. The simulation is conducted in the same straight ladder environment as Case 1, with the vehicle moving along the x-axis and with a maximum sensor range of 200m and a 360° field of view, enabling all landmarks to be in the vehicle's field of view. Landmark 1 still remains the anchor, however, landmark 3 is now chosen as the tether. The anchor-tether pair is now horizontal in the world frame of reference, while in Case 1, the pair was vertical. With no re-selection of the anchor-tether pair, the simulation is conducted with no re-selection.

Figure 15 contains the anchor highlighted in red, the tether highlighted in magenta, the vehicle shown as a black triangle for various time steps, the true path shown in green and the estimated path shown in red. It also contains uncertainty ellipses plotted for each landmark; however, the figure represents the uncertainty ellipse plotted at the vehicle's final position. The uncertainty ellipse may look like a line due to the dimensions of the figure. For trends in the standard deviations of estimates of landmarks, the reader is directed to Figures 17 to 18.

Figure 16, consisting of the state error for the tether- landmark 3, reveals that the initialization error is significantly high of approximately 6.7m. The magnitude of error increases throughout the simulation remaining in the order of m throughout of the simulation. The errors for other ordinary landmarks follow the same trends as well. The high magnitude of error can be attributed to two sources- initialization and holding the y coordinate constant when the anchor-tether pair is horizontal in the world frame of reference.

Referring to Figure 16, it can be seen that the errors for the estimation of the vehicle pose are extremely high in the range of m, and it is clear that the EKF is

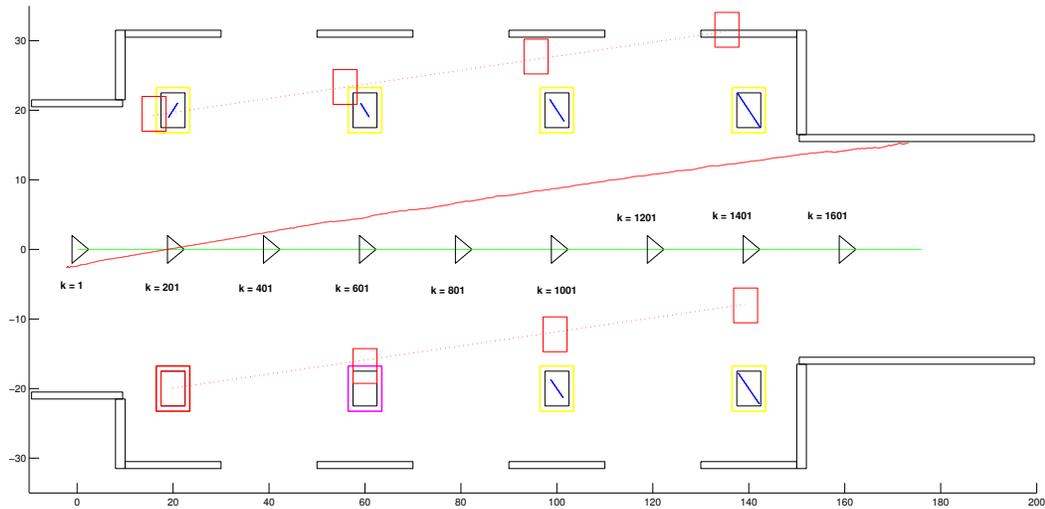


Figure 15: Case 2: Motion of Vehicle through the Straight Ladder Environment

diverging, with a steady increase in error. The standard deviation reveals the initial decrease in state error in the x and y coordinates as the vehicle moves closer towards the anchor and tether, from iterations 0 through 200, and undergoes a gradual but steady increase for coordinates x and θ , with an increase in magnitude of error in the y coordinate. This case shows that even with the knowledge of 3 coordinates of 2 landmarks, there is a possibility that the SLAM problem can become unobservable and it takes place when the wrong coordinate of the tether is estimated. Since the vehicle still moves along a straight line in the x -axis of the world centric reference frame, there is a noticeable increase in the magnitude of error in the y coordinates of all landmarks.

Thus, the results from this case support the analysis performed using Fisher's Information, and choosing the appropriate coordinate to estimate is crucial when anchors and tethers are selected. It is even more important when re-selection of anchors and tethers takes place, as that is when any ambiguity must be avoided, to maintain a consistent world frame of reference. An improvement from this situation

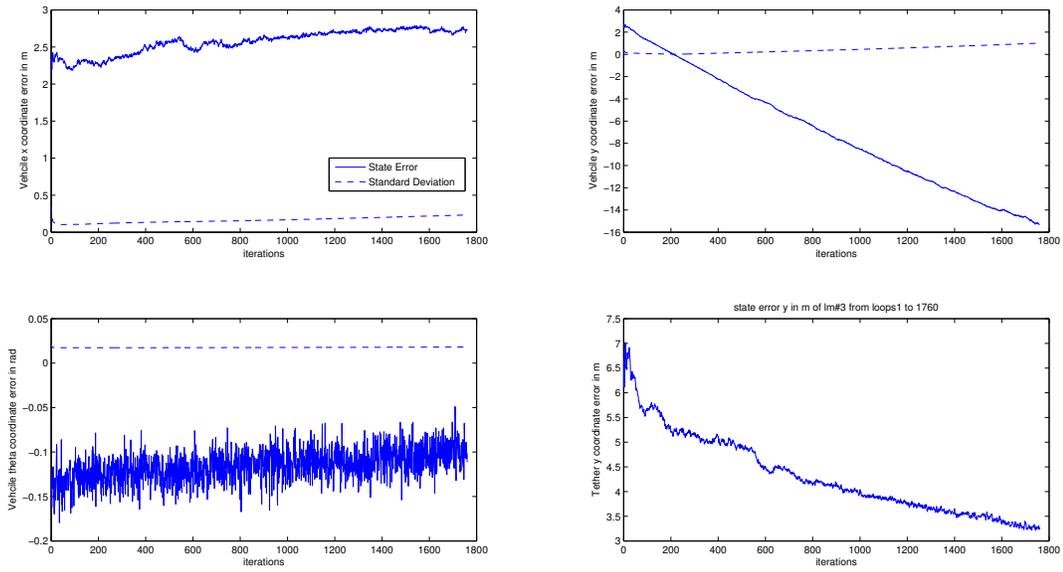


Figure 16: Case 2: State Error, Standard Deviation of Vehicle pose and Tether State Error

has been provided in the next case.

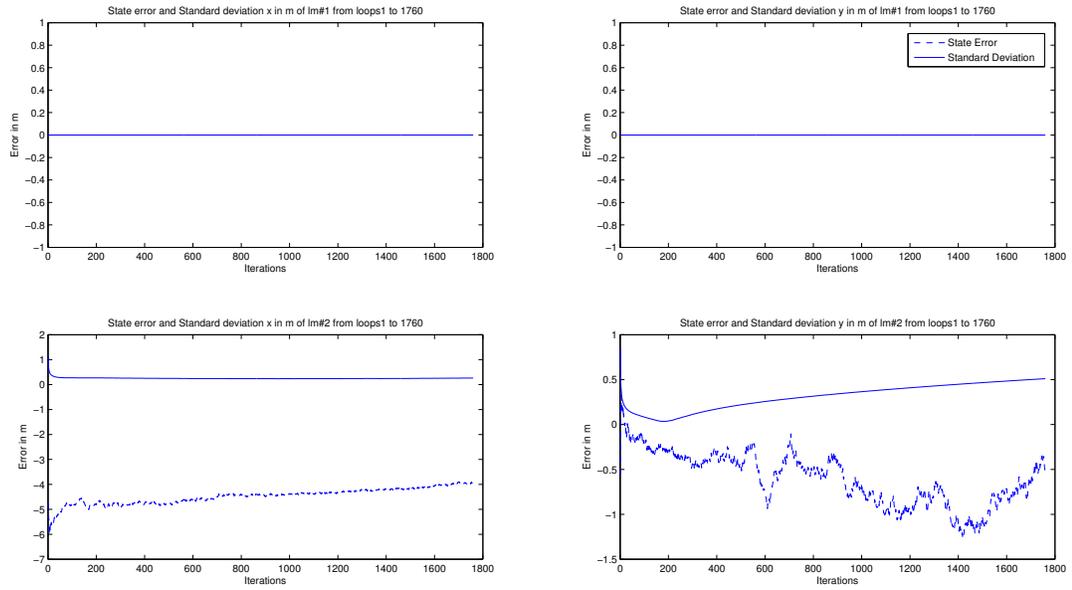


Figure 17: Case 2: State Error and Standard Deviation of Landmarks 1 and 2

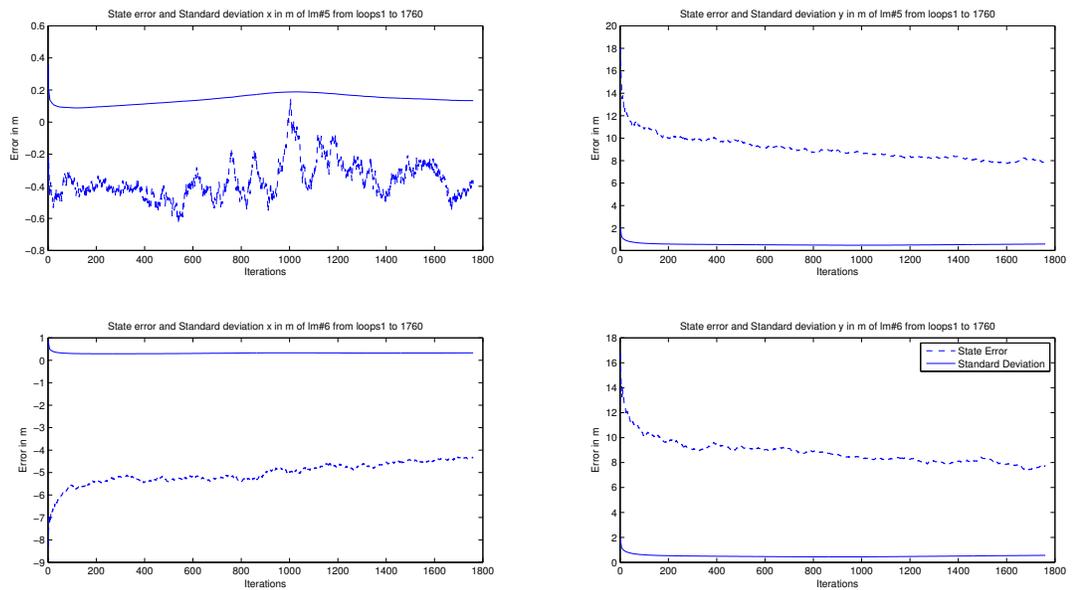


Figure 18: Case 2: State Error and Standard Deviation of Landmarks 5 and 6

3.5 Case 3: Observable Reconfigurable EKF SLAM

3.5.1 Maintaining Observability

Since at least two landmarks must be visible at all times to maintain observability, and with the possibility that the vehicle can move, it leads to the issue of the anchor and tether going out of sight. This calls for reselecting either one or two new landmarks as the anchor and tether to re-instate the world frame of reference, and to maintain observability. After initial selection of anchor and tether, any of the following could occur as landmarks enter and leave the field of view

- Reselect Anchor
- Reselect Tether
- Reselect Anchor and Tether
- Addition of New Landmarks
- Deletion of Departing Landmarks

The EKF is modified to account for all of the above mentioned possibilities, which makes the EKF continuously reconfigure its states, Jacobians and covariance, and the term *Reconfigurable EKF* is thus used.

The term "special landmark" is used as a generic term to refer to both anchor and tether for simplicity, and the term "ordinary landmarks" is used for non special landmarks. The need to reselect a special landmark can arise when either the vehicle moves increasingly farther away from the existing special landmark or when the special landmarks exit the field of view. Considering the former, tests are utilized to evaluate if existing special landmarks need to be replaced. The two tests used include

the distance test and the time test. The distance test checks the distance between the vehicle and the special landmarks and the time test checks the duration for which the landmark has been estimated in the filter, against user defined thresholds. A third test checking to see if the special landmark lies ahead of the vehicle may be used, however that assumes that the vehicle travels only in the forward direction. Similarly, ordinary landmarks are continually evaluated using the distance and time tests, and only those landmarks satisfying the criteria are possible candidates for re-selection as special landmarks. Other kinds of tests like the Mahalanobis distance test suggested and used by authors of [11] and [78], the exponential decay test evaluating the maturity of landmarks in the environment suggested by [78] and the covariance convergence test suggested by Souici et al. of [42] could also be used to evaluate which landmarks have consistently remained in view and have better estimates, thus becoming potential anchors and tethers. This thesis however, uses the distance and time tests only.

Continuous evaluation modules in the simulation algorithm enable automatic re-selection of anchors and tethers, thus always maintaining observability. Existing special landmarks in the current field of view are required to pass both tests to sustain their positions as special landmarks, failing which a new successful candidate landmark is chosen. In the absence of a successful candidate landmark, the old special landmark is retained. The flowchart in Figure 19 demonstrates a high level decision sequence of anchor-tether re-selection.

The algorithm achieves three main goals:

- It maintains observability through re-selection of special landmarks.
- It switches between estimating the x and y coordinate of the tether depending on whether the anchor-tether pair is horizontal or vertical in F_w .

- Finally, it ensures that with the loss of a special landmark, a non-special landmark is chosen only when the other special landmark fails to satisfy the tests. This avoids accumulation of error in the filter, as re-selecting an existing tether as a new anchor would decrease error in 1 dimension (since only one coordinate was estimated, the other held constant) when compared with choosing a landmark that has been estimated for period of time, thus accumulating error in both dimensions.

The flowchart in Figure 19 gives a detailed account of the situations that trigger re-selections, if and when necessary.

The simulation assumes that existing special landmarks are required to pass both the distance and time tests in order to retain their positions as special landmarks. The distance and time tests take the following form.

- If $\text{range} \geq T_r$ and $\text{iterations} \geq T_k$, then reselect special landmark, else retain special landmark.
- The values for T_r and T_k are user defined and can be changed depending on the size of the environment. The values chosen for simulation are chosen purely to demonstrate the effectiveness of the tests. The impact of different threshold values on re-selection is left for future work.

1. $T_r = 40\text{m}$

2. $T_k = 50$ iterations

As Case 2 demonstrated, it is important to evaluate the angle between the anchor-tether pair in F_w and choose the correct coordinate of the tether to estimate. The following procedure is used to determine the coordinate to be estimated.

- If the anchor's position is given by (x^A, y^A) and tether's position is given by (x^T, y^T) in F_w , compute

1. $\delta x = x^T - x^A$
 2. $\delta y = y^T - y^A$
 3. $\gamma^{AT} = \text{atan2}(\delta y, \delta x)$
- If $89^\circ \leq \gamma^{AT} \leq 91^\circ$ or $-91^\circ \leq \gamma^{AT} \leq -89^\circ$, then
 1. Anchor-tether pair is approximately vertical in F_w .
 2. Fix x^T and estimate y^T .
 - If $-1^\circ \leq \gamma^{AT} \leq 1^\circ$ or $-181^\circ \leq \gamma^{AT} \leq -179^\circ$, then
 1. Anchor-tether pair is approximately horizontal in F_w .
 2. Fix y^T and estimate x^T .

It is recommended to evaluate the orientation of the anchor-tether pair at every timestep, to take into account errors in position estimates of the anchor-tether pair. For anchor-tether pairs whose orientations do not fall into the above categories, the coordinate that was previously estimated will continue to be estimated until a switching of the coordinate is determined necessary. Since the angular thresholds defining the orientations as vertical or horizontal are strict, estimation errors of landmark positions during simulations may not be able to correctly classify the orientation of the anchor-tether pair as horizontal or vertical, while they actually are horizontal or vertical in F_w . As estimation errors decrease, orientations are determined with better accuracy enabling it to fall between the thresholds defined, and the correct coordinate of the tether to be estimated is chosen.

3.5.2 Reconfigurable EKF: Impact of Entering and Departing Landmarks on the EKF

Since the anchor and tether are used to define the world frame of reference, both coordinates of the anchor and one coordinate of the tether will remain fixed, and not be estimated. This influences entries in the state vector, the Kalman covariance and the measurement Jacobian, and the dimensions of the vehicle process Jacobian and the landmark process model noise covariance. A more detailed description follows.

The following order is used to organize the state vector, Kalman covariance and measurement Jacobian for landmarks

$$\text{Order} = \begin{bmatrix} \text{anchor} \\ \text{tether} \\ \text{old landmarks} \\ \text{old anchor} \\ \text{old tether} \\ \text{new landmarks} \end{bmatrix}. \quad (3.5.1)$$

Choosing an anchor involves fixing both co-ordinates of the landmark and the following steps are taken.

1. The two states corresponding to the anchor's position are removed from the state vector and held constant.
2. The rows and columns corresponding to the covariance associated with the state estimates of the anchor are removed from the Kalman covariance.

3. The columns corresponding to the measurement Jacobian associated with the anchor's measurements are removed, but corresponding rows are retained as the anchor is still in the field of view and measurements with respect to the anchor are received.

Choosing a tether involves fixing either the x or y coordinate of the landmark and the following steps are taken.

1. Depending on which coordinate is determined to be fixed, that state of the tether is removed from the state vector and held constant.
2. The row and column corresponding to the element (related to the coordinate that was determined to be fixed) of covariance of the tether are removed from the Kalman covariance.
3. The column corresponding to the element (related to the coordinate that was determined to be fixed) of the tether's landmark measurement Jacobian is removed.

When a landmark leaves the field of view, the same steps for choosing an anchor are followed, with the exception of the last step. While the columns corresponding to the landmark measurement Jacobian are removed, its corresponding rows are removed as well, since no measurements with respect to the landmark are received. When a new landmark enters the field of view the following steps are taken.

1. Following the order defined in (3.5.1), the states corresponding to the new landmark are augmented to the state vector.
2. In the same order, rows and columns corresponding to the covariance values associated with the landmark are inserted to the Kalman covariance.
3. The rows and columns corresponding to the landmark's measurement Jacobian are added to the measurement Jacobian.

3.5.3 Analysis of Case 3

Considering the importance of anchor and tether re-selection and that of evaluating which coordinate of the tether to estimate, this case accounts for enabling both situations. The simulation is conducted in the same straight ladder environment used in Cases 1 and 2, however with no user defined enforcement of anchor and tether. The Reconfigurable EKF equipped can now independently evaluate which landmarks are suitable to be anchor and tether based on measurements obtained and also determine which coordinate of the tether is more suitable to estimate, based on the spatial information of the anchor and tether. With iterative modifications to the information passed into the EKF through the changes mentioned in Subsection 3.5.2, the algorithm becomes a Reconfigurable EKF. The range of the sensor is reduced to 80m with a 360° field of view to ensure that there are a limited number of landmarks in the field of view. The sequence of selection of anchors and/or tethers, with their corresponding errors during re-selection is given in Table 3.

Table 3: Automated Observable SLAM in a Straight Ladder Environment

Iteration	Anchor	Tether	Tether error in 'm'	Coordinate of Tether Estimated	Anchor error x in 'm'	Anchor error y in 'm'	Landmarks in Field of View
1	1	2	-3.66E-02	y	0	0	1,2,3,4
226	'	'	-NA-	y	-NA-	-NA-	1,2,3,4,5,6
626	'	'	-NA-	y	-NA-	-NA-	1,2,3,4,5,6,7,8
975	3	5	4.60E-03	x	-6.26E-03	-8.49E-02	3,4,5,6,7,8
1375	5	7	1.40E-01	x	4.64E-02	1.29E-01	5,6,7,8

Entries in Table 3 can be observed in Figure 20. Consider iterations 1 through 226, from Table 3, where the vehicle initializes the anchor-tether pair and moves towards the anchor and tether, the state error in Figure 20 fluctuates initially but settles down as iteration 226 approaches. Row 2 in Table 3 indicates the entry of landmarks 5 and 6 into the sensor's field of view, and that can be noticed in Figure 20 with a slight decrease in standard deviation in the x coordinate, an insignificant change in y and a small decrease in θ 's standard deviation. The drop in uncertainty can be justified by

the entry of landmarks 5 and 6 at this iteration, where more measurements can be used to correct the Kalman estimates of the vehicle pose. A similar observation can be made at iteration 626, when the uncertainty in x and θ undergo yet another slight drop when landmarks 7 and 8 enter the sensor's field of view providing additional measurements to correct the Kalman estimates.

An important observation to make is that of the standard deviation of the y coordinate, which does not seem to be as influenced by the entry of landmarks into the field of view, unlike those of x and θ . It is also interesting to note that the standard deviation of y experiences a gradual and steady increase from approximately iterations 226 through 975. Iteration 975 is when the old anchor-tether pair- landmarks 1 and 2, exit the the sensor's field of view and are replaced by landmarks 3 and 5 as the new anchor-tether pair, along with a switch in the coordinate of estimation of the tether from y to x , as the anchor-tether pair is horizontal. Considering that in this simulation the vehicle moves along the x axis, as it moves away from the original anchor-tether pair (1 and 2), there is an increase in contribution to the uncertainty in the y axis due to bearing error, and as landmarks 3 and 5 are chosen, the corresponding range and bearing errors drop, thus influencing the behaviour of the standard deviation as well.

Since landmark 5- the current tether at iteration 975 has been in the field of view since iteration 226, its position has been estimated by the filter till iteration 975, at which point its most recent estimate will be initialized as the value to be held constant by the filter, which is why there is a non zero initialization error of $4.6e^{-3}$ m. Similarly, landmark 3 which is chosen as the anchor has also been estimated so far, and its initialization errors are $-6.26e^{-3}$ m and $-8.49e^{-2}$ m. The standard deviation of θ however experiences a slight increase at iteration 975, as landmarks 1 and 2 exit the field of view, and the level of certainty is not as high as when there are two extra landmarks whose measurements can support the certainty. The last re-selection

occurs at iteration 1375, when the previous anchor- landmark 3, and landmark 4 leave the field of view, and the previous tether- landmark 5 is adopted as the new anchor. It is worth noting that the new anchor-tether pair happen to be horizontal in the world frame of reference, and thus the x coordinate of the tether remains being estimated. The departing landmarks have an impact on the standard deviation of θ by increasing the uncertainty slightly, while decreasing the uncertainty in the x and y coordinates.

An interesting improvement from Cases 1 and 2 is that any initial error is corrected almost immediately due to re-selection of anchors and/or tethers, and the switching in estimation of the tether's coordinate. An observation of the state error of the vehicle pose in Figure 20, reveals that all three coordinates show traces of initialization errors which are gradually corrected in the x and θ coordinates, but retains some drift in the y coordinate. The landmark state errors along with their standard deviations provided in Figures 21 and 22 show trends similar to that of Cases 1 and 2, in that the y coordinate has a higher degree of initialization error. However, the difference between the previous cases and Case 3 is that the errors tend to approach steady state within cm range. The sequence of events that landmark 5 undergoes has been highlighted in Figure 23 and its corresponding events are given in Table 4.

Table 4: Case 3: Landmark 5 Event Chart

Iteration	Event	Notation in Figure	Colour Code
1	Not in Field of View	Not Visible	
226	Enters Field of View	Landmark	
975	Chosen as Tether; estimate x	T est x	
1375	Chosen as Anchor	Anchor	

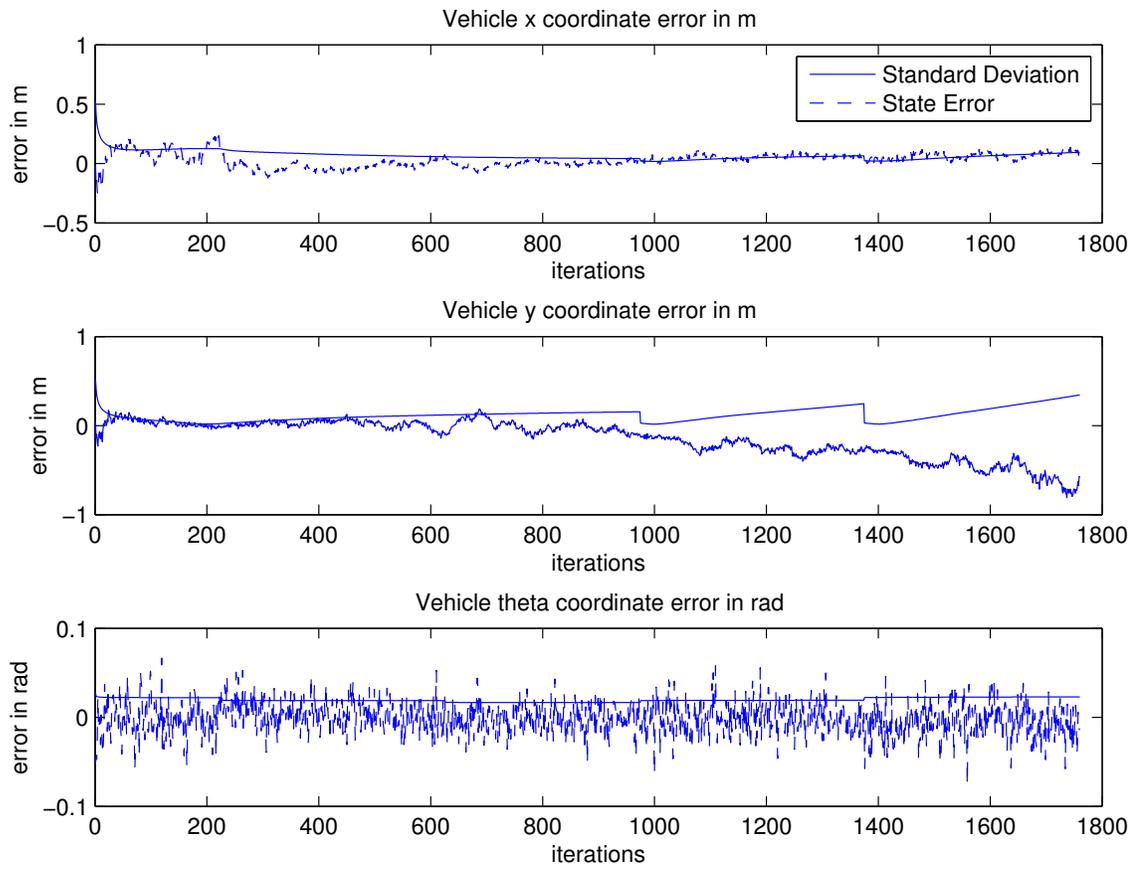


Figure 20: Case 3: State Error , Standard Deviation of Vehicle pose and Tether State Error

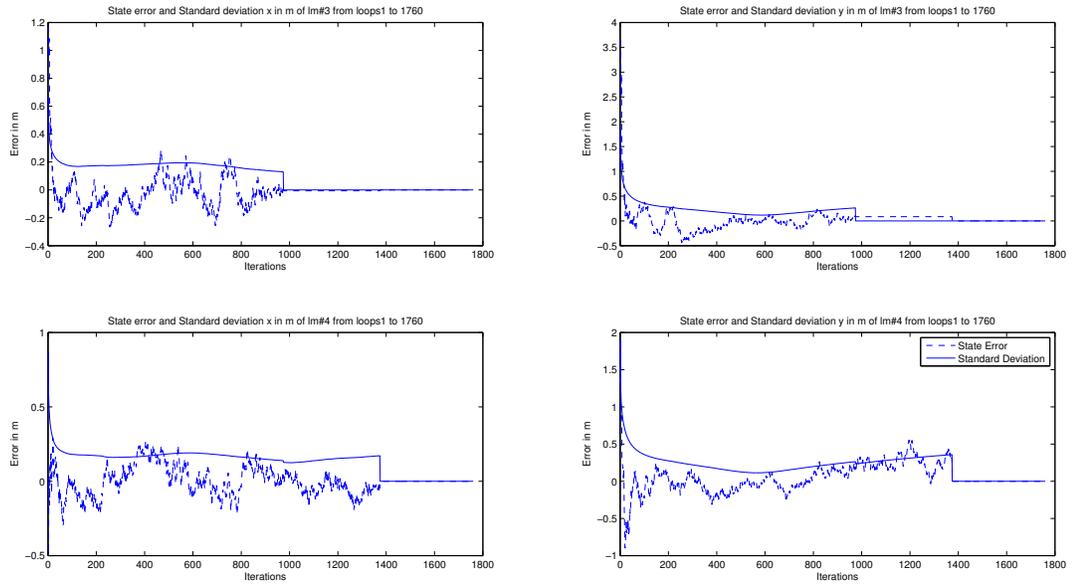


Figure 21: Case 3: State Error and Standard Deviation of Landmarks 3 and 4

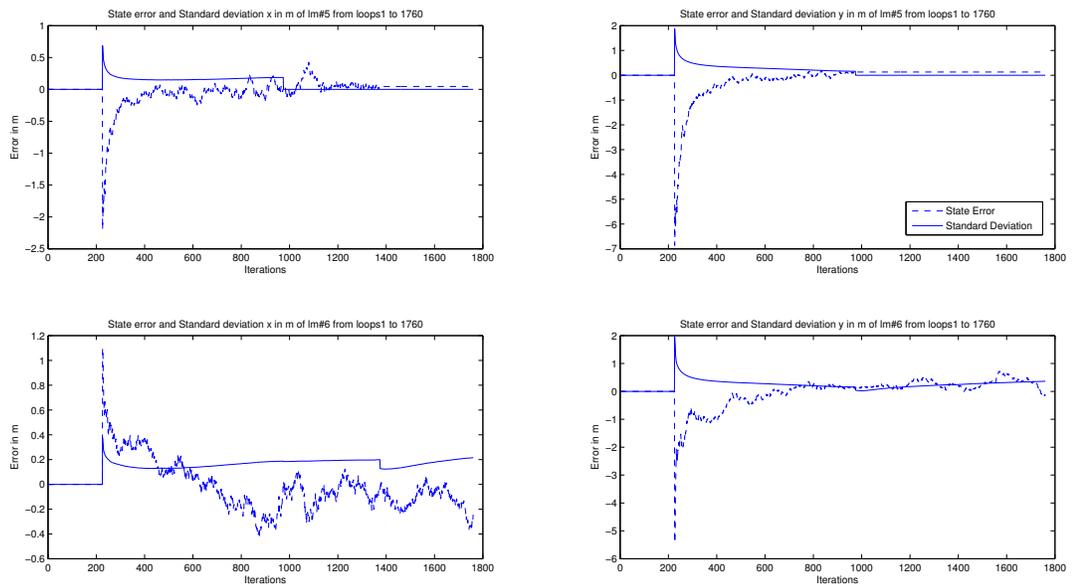


Figure 22: Case 3: State Error and Standard Deviation of Landmarks 5 and 6

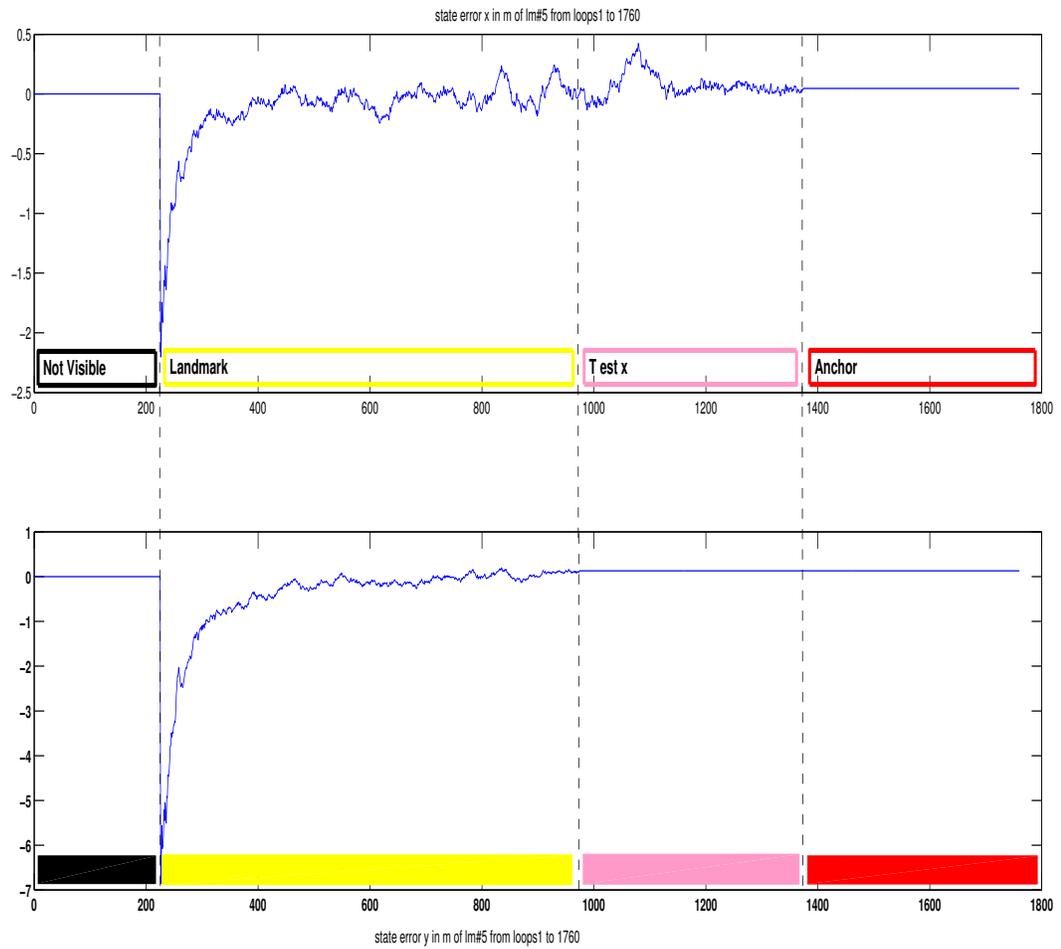


Figure 23: Case 3: Lifecycle of Landmark 5

3.6 Case 4: Limitations of Anchor-Tether Re-selection

Having tested the algorithm in a straight ladder environment, this case investigates the trends in re-selection of anchors and tethers as the vehicle repeatedly passes through the same environment. This simulation is conducted in the circular ladder environment, whose spatial descriptions are given in Table 2 and is represented in Figure 7. Automatic re-selection of anchors and tethers, along with switching between estimating the x and the y coordinate, are enabled. The range of the sensor is 55m with a 360° field of view, allowing a few landmarks at a time to be in the field of view, enabling the need to choose new anchors and tethers.

This simulation allows the vehicle to travel along its circular path about 2 and quarter times or for 6000 iterations, allowing for sufficient patterns in re-selection of anchors and tethers to be recognized. The vehicle's linear and angular velocities are initialized with values specified in Section 3.1 corresponding to the circular ladder environment. Table 5 depicts the sequence of anchor and tether selections, and the exit and entry of landmarks into the field of view. Figure 24 contains the vehicle shown as a black triangle for various time steps, the true path shown in green and the estimated path shown in red. It also contains uncertainty ellipses plotted for each landmark; however, the figure represents the uncertainty ellipse plotted at the vehicle's final position. The uncertainty ellipse may look like a line due to the dimensions of the figure.

Two cases of repeated anchor-tether pair will be considered which are the anchor-tether pair of landmarks 1 and 4 highlighted in yellow in Table 5, and that of landmarks 3 and 4, highlighted in orange. The initial selection of anchor at iteration 1 owes its zero initialization error to exact initialization as discussed in Cases 1, 2 and 3, and the tether's y coordinate is initialized with an error of $2.13e^{-2}$ m. Focusing on

Table 5: Automated Observable SLAM in a Circular Ladder Environment

Iteration	Anchor	Tether	Tether error in 'm'	Coordinate of Tether Estimated	Anchor error x in 'm'	Anchor error y in 'm'	Landmarks in Field of View
1	2	1	2.13E-02	y	0.00E+00	0.00E+00	1,2,3,7
89	"	"	-NA-	y	-NA-	-NA-	1,2,3
110	"	"	-NA-	y	-NA-	-NA-	1,2,3,4
520	1	4	9.07E-02	y	0.00E+00	2.10E-02	1,3,4
542	"	"	-NA-	y	-NA-	-NA-	1,3,4,5
718	3	4	-1.14E-01	x	-9.84E-02	-6.14E-02	3,4,5
739	"	"	-NA-	x	-NA-	-NA-	3,4,5,6
1149	3	5	-4.34E-01	x	-9.84E-02	6.14E-02	3,5,6
1170	"	"	-NA-	x	-NA-	-NA-	3,5,6,7
1346	7	5	-3.20E-01	x	-7.33E-02	-3.88E-01	5,6,7
1367	"	"	-NA-	x	-NA-	-NA-	5,6,7,8
1777	"	"	-NA-	x	-NA-	-NA-	5,7,8
1797	"	"	-NA-	x	-NA-	-NA-	1,5,7,8
1975	7	1	3.96E-01	x	-7.33E-02	-3.88E-01	1,7,8
1994	"	"	-NA-	x	-NA-	-NA-	1,2,7,8
2406	"	"	-NA-	x	-NA-	-NA-	1,2,7
2425	"	"	-NA-	x	-NA-	-NA-	1,2,3,7
2603	2	1	3.09E-01	x	7.06E-01	1.21E-01	1,2,3
2623	"	"	-NA-	x	-NA-	-NA-	1,2,3,4
2663	2	1	4.40E-02	y	7.06E-01	1.21E-01	1,2,3,4
3033	1	4	9.06E-01	y	3.63E-01	1.29E-01	1,3,4
3055	"	"	-NA-	y	-NA-	-NA-	1,3,4,5
3230	3	4	-9.63E-02	x	-1.14E-01	5.81E-01	3,4,5
3252	"	"	-NA-	x	-NA-	-NA-	3,4,5,6
3662	3	5	-1.95E-01	x	-1.14E-01	5.81E-01	3,5,6
3684	"	"	-NA-	x	-NA-	-NA-	3,5,6,7
3860	7	5	-1.59E-01	x	-3.22E-02	4.55E-01	5,6,7
3880	"	"	-NA-	x	-NA-	-NA-	5,6,7,8
4291	"	"	-NA-	x	-NA-	-NA-	5,7,8
4310	"	"	-NA-	x	-NA-	-NA-	1,5,7,8
4488	7	1	1.41E-01	x	-3.22E-02	4.55E-01	1,7,8
4508	"	"	-NA-	x	-NA-	-NA-	1,2,7,8
4919	"	"	-NA-	x	-NA-	-NA-	1,2,7
4939	"	"	-NA-	x	-NA-	-NA-	1,2,3,7
5115	2	1	7.29E-01	y	4.67E-01	6.04E-01	1,2,3
5136	"	"	-NA-	y	-NA-	-NA-	1,2,3,4
5546	1	4	1.18E+00	y	2.58E-01	6.79E-01	1,3,4
5569	"	"	-NA-	y	-NA-	-NA-	1,3,4,5
5743	3	4	-1.51E-01	x	-5.47E-02	1.01E+00	3,4,5
5765	"	"	-NA-	x	-NA-	-NA-	3,4,5,6

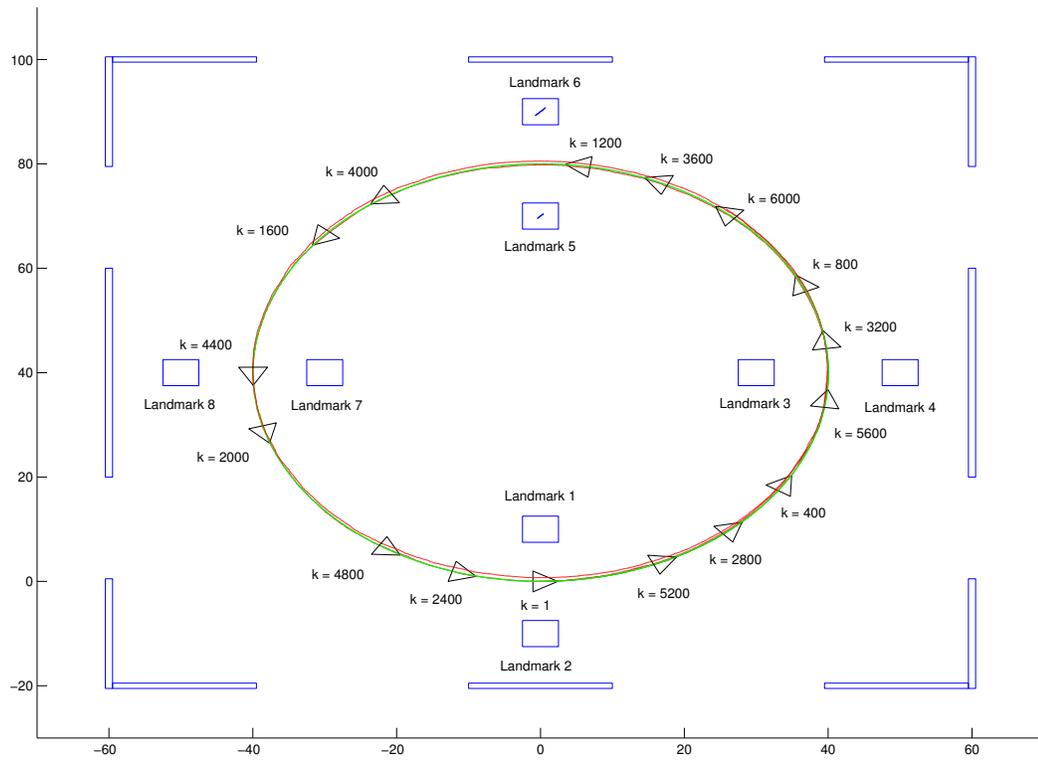


Figure 24: Case 4: Motion of Vehicle through the Circular Ladder Environment

iteration 520, when landmarks 1 and 4 are chosen as anchor and tether; landmark 1 was previously the tether and thus had its y coordinate estimated, while landmark 4 had both coordinates estimated in the filter. As a result, the anchor and tether initialization errors are non zero (excepting the anchor's x coordinate which was held constant when landmark 1 was the tether) with the tether's error of 0.0907m and the anchor's errors of 0m and 0.021m. As the vehicle makes one complete pass and reaches the same location again at iteration 3033, landmarks 1 and 4 are re-chosen as the anchor-tether pair, with initialization errors for the tether and the anchor increasing to about 0.906m and, 0.363m and 0.129m respectively.

However, there is no indication of a sudden increase in the state error of the vehicle

pose in either of the coordinates, as seen in Figure 25. Considering the third time when the vehicle arrives at the same point at iteration 5546, the initialization error of the tether increases to 1.18m, which is still comparatively small considering the size of the environment that spans an area of $140\text{m} \times 140\text{m}$. The anchor's errors are in the range of $2.58e^{-1}\text{m}$ and $6.79e^{-1}\text{m}$, similar to the last time the anchor-tether pair was chosen.

Observing the second anchor-tether pair of landmarks 3 and 4 (highlighted in orange) starting at iteration 718, the initialization errors and re-selection processes are similar to that of anchor-tether pair 1 and 4. However the main difference occurs the second time the vehicle reaches the same point at iteration 3230, and landmarks 3 and 4 are chosen as the anchor-tether pair. Landmark 4's y coordinate has been estimated as the tether's coordinate till iteration 3055 after which it's x coordinate is estimated. There is thus, a decrease in the tether's initialization error in the x coordinate, since it was held constant from its previous position as tether. This implies that the y coordinate that was previously estimated is now held constant, and the last estimated y value of landmark 4 is $-9.63e^{-2}\text{m}$. The anchor's initialization errors are in approximately $-1.14e^{-1}\text{m}$ and $5.81e^{-1}\text{m}$. While its effect can be seen in the y coordinate of the state error of the vehicle, it still has no significant impact on the errors of x and θ . At iteration 5743, when the anchor-tether pair is chosen for the third time, there is a significant increase in the anchor's initialization error in the y coordinate to 1.01m, i.e. an increase from the range of cm to m from the last time it was chosen, similar to the increase in error of landmark 4 in the anchor-tether pair of landmarks 1 and 4. Figure 25 reflects all the events that take place by revealing correlating changes in the standard deviation that form a steady pattern throughout the simulation, and can be seen at points where the same anchor-tether pairs are chosen repeatedly.

Referring to Table 5, the regions highlighted in green demonstrate the need for

switching the coordinate of the tether to be estimated, even as the anchor-tether pair remains the same. From iterations 2603 to 2663, the vehicle approaches its initial starting point at the origin of the world centric reference frame, and a switch in the coordinate of the tether to be estimated is performed. While the anchor-tether pair of landmarks 1 and 2 will always remain vertical in F_w , noisy measurements may hamper accurate determination of the orientation of the anchor-tether pair. Considering the strict thresholds used to define horizontal and vertical orientations in Subsection 3.5.1, it can be seen that noisy measurements coupled with error in estimates can lead to a delayed switching in the coordinate of the tether to be estimated. Table 6 contains the sequence of events that landmark 1 goes through as the vehicle traverses the environment, and can be seen by observing the changes in the state errors of landmark 1 in Figure 26. Landmark errors and standard deviations are provided in Figures 27 to 30.

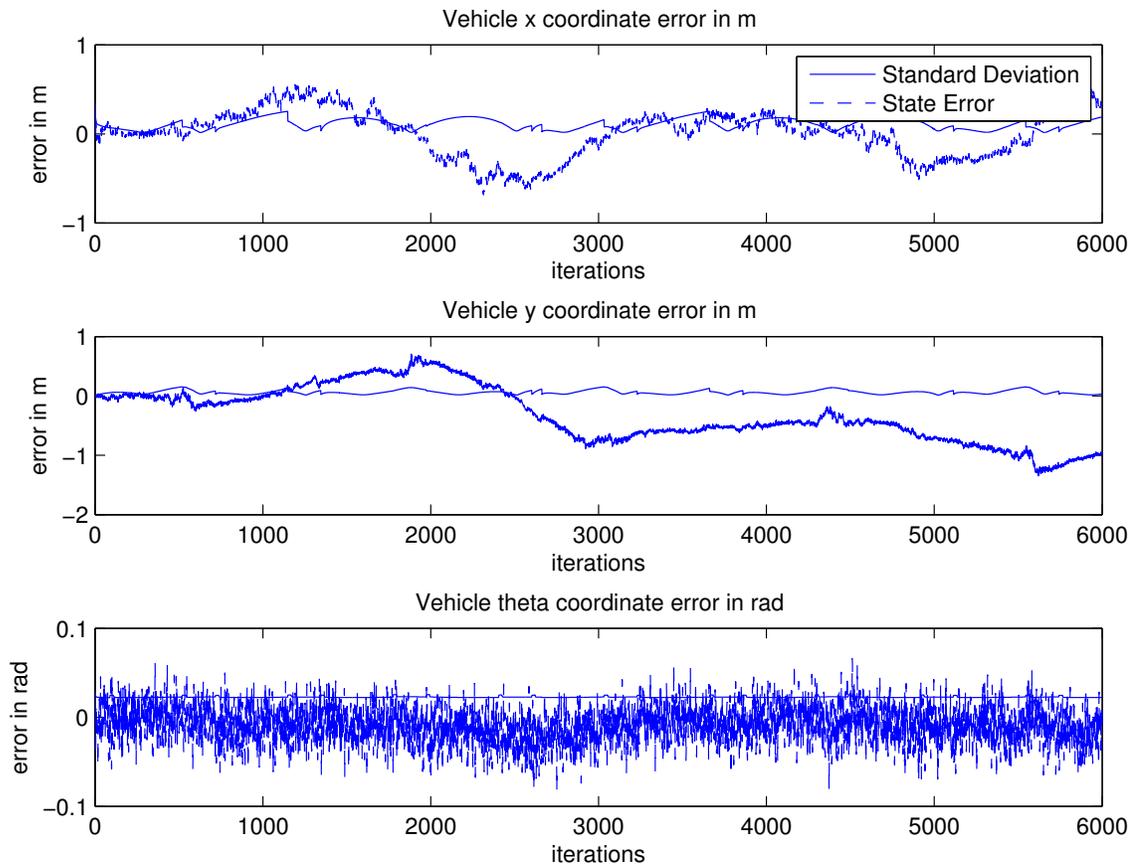


Figure 25: Case 4: State Error , Standard Deviation of Vehicle pose and Tether State Error

Table 6: Case 4: Landmark 1 Event Chart

Iteration	Event	Notation in Figure	Colour code
1	Chosen as Tether; estimate y	T est y	
520	Chosen as Anchor	A	
718	Exits FOV	Not Vis or NV	
1797	Enters FOV as ordinary landmark	L	
1975	Chosen as Tether; estimate x	T est x	
2663	Remains as Tether but estimate y	T est y	
3033	Chosen as Anchor	A	
3230	Exits FOV	Not Vis or NV	
4310	Enters FOV as ordinary landmark	L	
4488	Chosen as Tether; estimate x	T est x	
5115	Remains as Tether; estimate y	T est y	
5546	Chosen as Anchor	A	
5743	Exits FOV	Not Vis or NV	

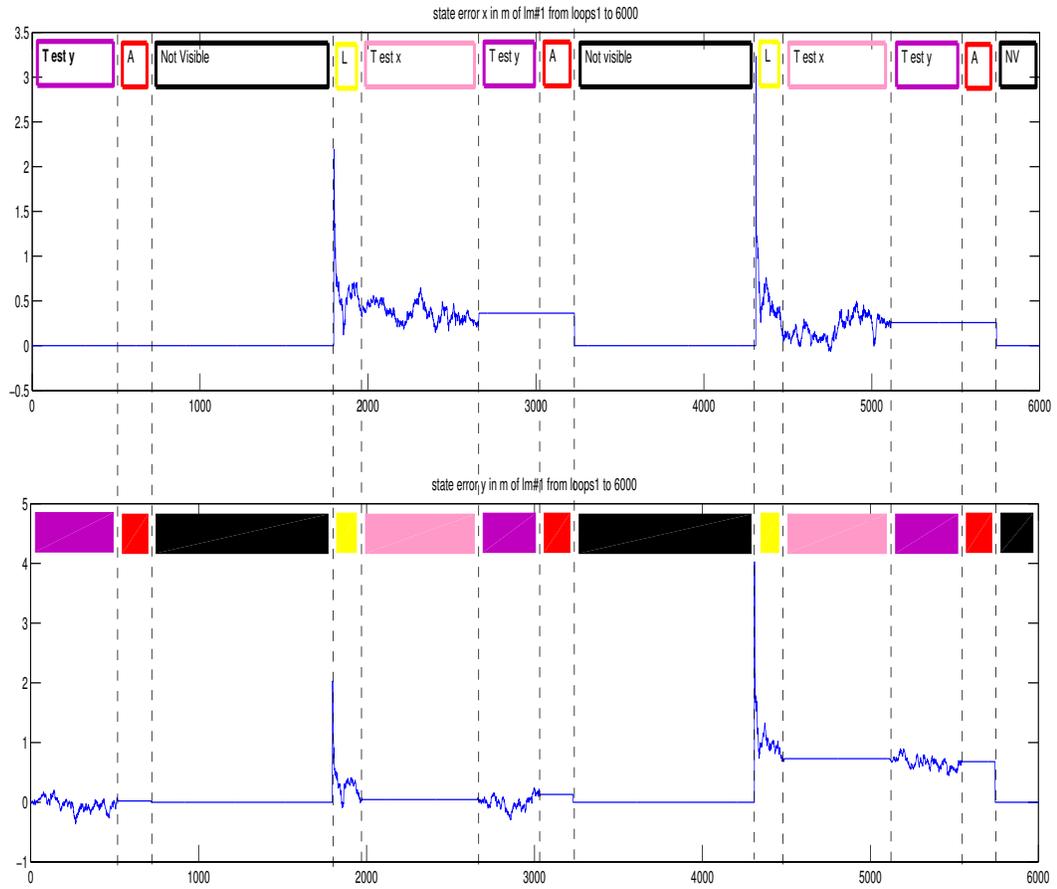


Figure 26: Case 4: Lifecycle of Landmark 1

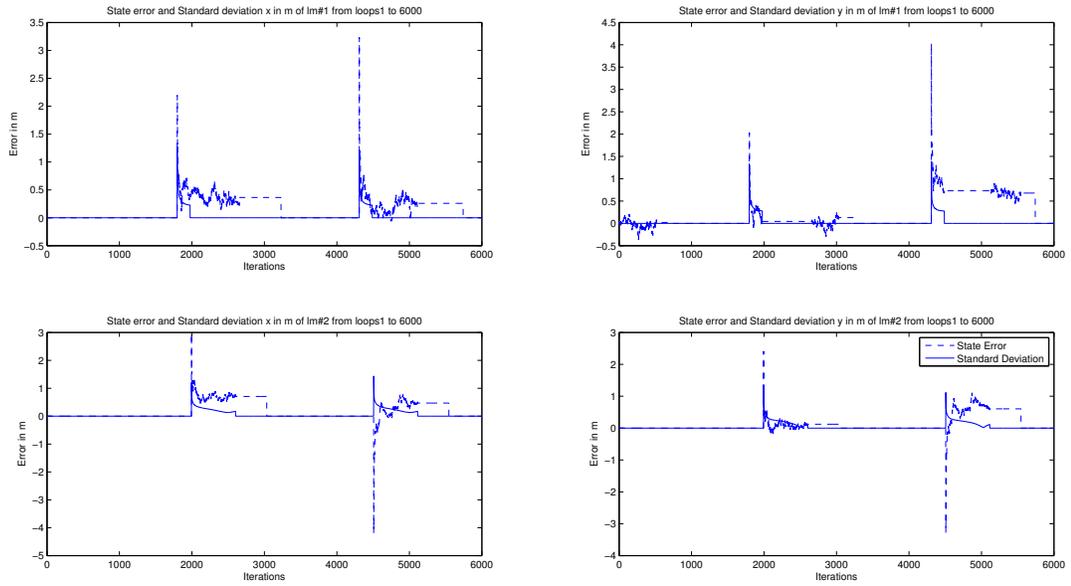


Figure 27: Case 4: State Error and Standard Deviation of Landmarks 1 and 2

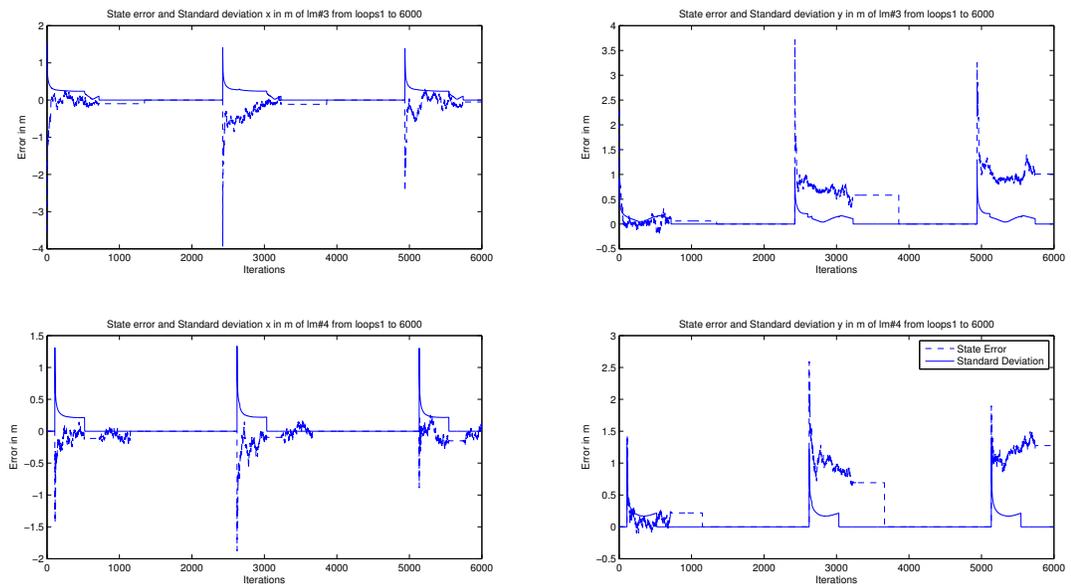


Figure 28: Case 4: State Error and Standard Deviation of Landmarks 3 and 4

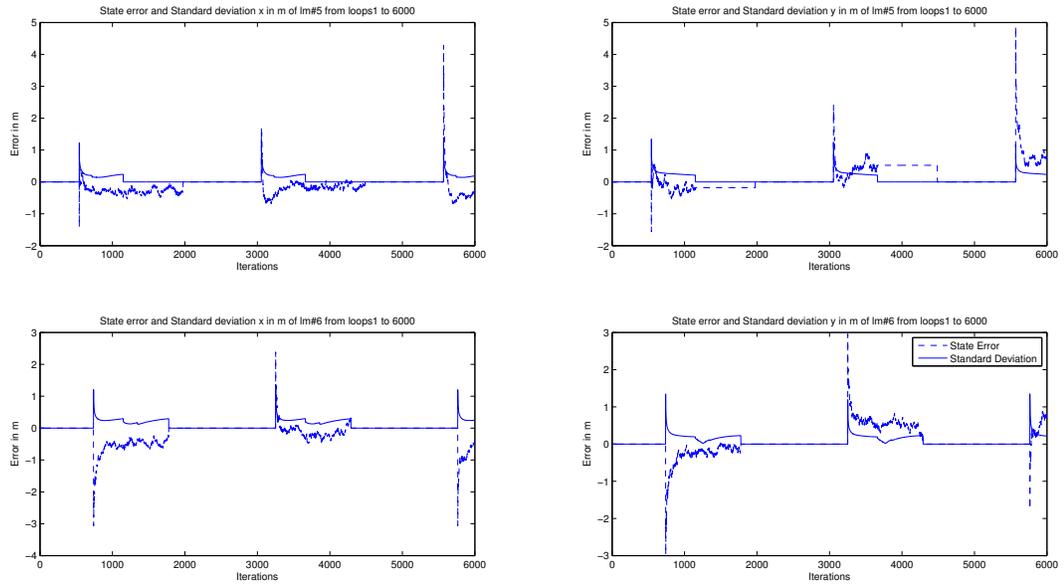


Figure 29: Case 4: State Error and Standard Deviation of Landmarks 5 and 6

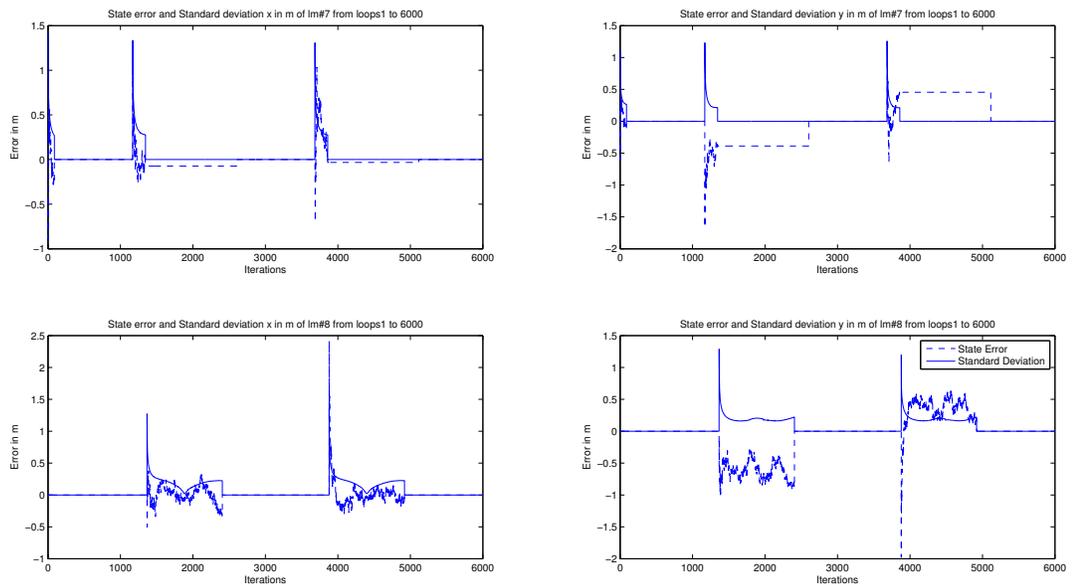


Figure 30: Case 4: State Error and Standard Deviation of Landmarks 7 and 8

Chapter 4

Dempster-Shafer Occupancy Grid Map Simulations and Results

The grid based maps are produced for both the straight ladder environment with an approximate area of $200\text{m} \times 200\text{m}$ and the circular ladder environment with an area of $140\text{m} \times 140\text{m}$. The 4 kinds of maps analyzed for each environment are the emptiness belief map, the emptiness plausibility map, fullness belief map and fullness plausibility map. All maps use grids with a cell size $1\text{m} \times 1\text{m}$. In all emptiness maps, white regions represent emptiness and black and grey regions represent non-emptiness; in all fullness maps, white regions represent fullness, while black and grey regions represent non-fullness.

The chapter provides a detailed illustration of Dempster-Shafer theory to create a grid based map using a simple environment. Occupancy grid maps of the straight ladder and circular ladder environment are provided, with a complete analysis of the emptiness and fullness, belief and plausibility maps of the circular ladder environment.

4.1 Illustration of the Dempster-Shafer theory through examples

Consider the environment in Figure 31 with all units in metres, where the stationary vehicle denoted by the black rectangle is surrounded by walls on the left, right and bottom denoted by blue rectangles. The grid(not drawn to scale) overlaid depicts the region that is initialized with DS theory bpa values, and the size of the grid is chosen to be small part of the environment for clarity in demonstrating the example. The size of each grid cell is 1×1 m. While finer resolution of grid squares creates better maps, the size chosen here is for simplicity of the example. The sensor aboard the vehicle is assumed to have a maximum range of 5m, implying that only certain regions of the grid can be sensed. The regions of the obstacles highlighted in dotted green rectangles are the parts that are detected by the sensor, beyond which the sensor provides no evidence of any proposition, because it is unable to sense beyond occupied regions.

Using Dempster-Shafer theory of evidential reasoning, the steps in developing an occupancy grid map are outlined. Initially, all the cells in the grid are initialized with the following four values

- $m_E^m = 0.1$
- $m_F^m = 0.1$
- $m_{EorF}^m = 0.8$
- $m_{neither}^m = 0$

indicating that there is no evidence to suggest whether the cells are exclusively full or empty, hence $m_E^m = 0.1$ and $m_F^m = 0.1$. Since the cells being neither empty nor full is improbable, $m_{neither}^m = 0$ and because the cells have to be either empty or full,

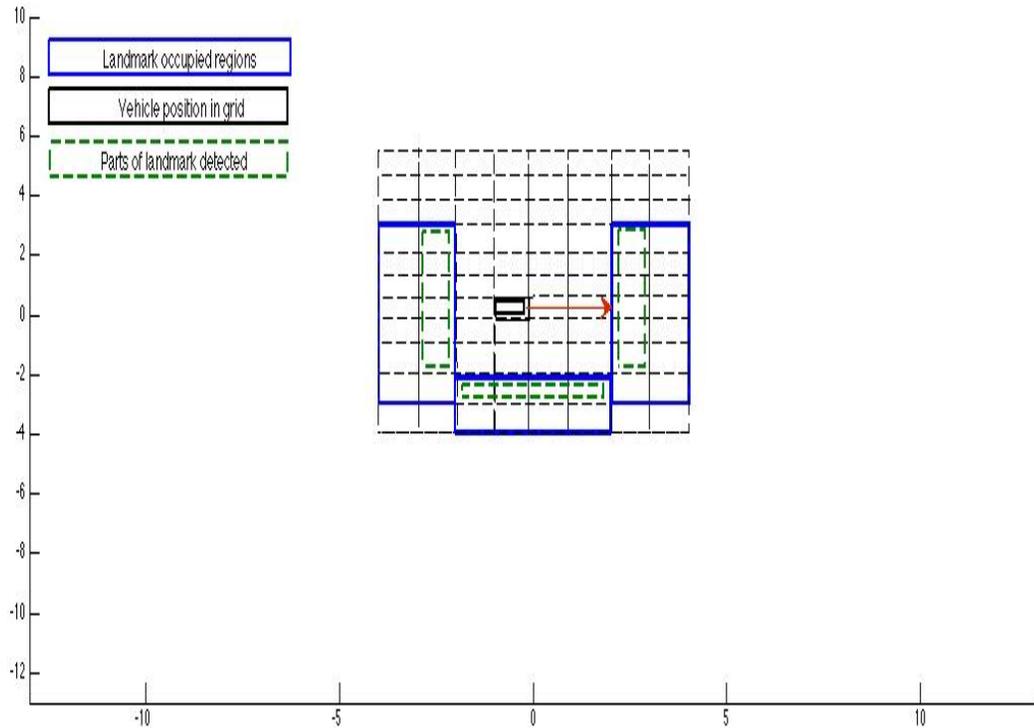


Figure 31: Example of Environment used to illustrate Dempster-Shafer theory

$m_{EorF}^m = 0.8$. After the first time the environment is sensed, every cell through which the scan line has passed will have its mass values updated, based on whether the sensor returned the cell as full or empty. The confidence in the sensor's ability to detect empty cells as empty and full cells as full are given numerical values in the following manner

- If a cell is detected as empty
 1. $\lambda_E = 0.3$
 2. $\alpha_E = 0$
 3. $\lambda_{EorF} = 0.7$

- If a cell is detected as full
 1. $\lambda_F = 0.3$
 2. $\alpha_F = 0$
 3. $\lambda_{EorF} = 0.7$.

The above assignment of values indicates a low level of confidence in the sensor's ability to detect fullness and emptiness accurately, but low confidence values are used to demonstrate that Dempster-Shafer theory is still efficient in situations where sensors may not be entirely reliable. An example of the update procedure for one event of detecting a full grid element is given, followed by a detailed interpretation of the emptiness mass values and the fullness mass values in Tables 7 and 8 respectively.

In Figure 31, consider the vehicle position outlined as a black rectangle, and assume the scan line indicated by the red arrow is the first time the environment is sensed. The empty cells through which the scan line passes receive emptiness support, and part of the obstacle that has been sensed by the scan line receives fullness support.

A cell detected as empty by the sensor and initially having parameters m_E^m, m_F^m, m_{EorF}^m and $m_{neither}^m$ are updated as follows:

1. Since it is empty, assign sensor support for emptiness.

- $m_E^s = \lambda_E = 0.3$
- $m_F^s = 0$
- $m_{EorF}^s = 1 - \lambda_E = 0.7$
- $m_{neither}^s = 0$

2. Compute normalization factor K .

- $K = m_E^m \cdot m_F^s + m_F^m \cdot m_E^s + m_{neither}^m \cdot m_{neither}^s = 0.03$

3. Using Dempster's rule of combination, combine sensory bpa values with existing map bpa values.

- $m_E^m = \frac{1}{1-K} (m_E^m \cdot m_{EorF}^s + m_{EorF}^m \cdot m_E^s + m_E^m \cdot m_E^s) = 0.3298$
- $m_F^m = \frac{1}{1-K} (m_F^m \cdot m_{EorF}^s + m_{EorF}^m \cdot m_F^s + m_F^m \cdot m_F^s) = 0.0679$
- $m_{EorF}^m = \frac{1}{1-K} (m_{EorF}^m \cdot m_{EorF}^s) = 0.5432$
- $m_{neither}^m = 0$

4. Compute belief and plausibility function values for all propositions.

- $\text{Bel}(F) = m_F^m + m_{neither}^m = 0.06794$
- $\text{Bel}(E) = m_E^m + m_{neither}^m = 0.3298$
- $\text{Bel}(E \text{ or } F) = m_{EorF}^m + m_E^m + m_F^m + m_{neither}^m = 0.9409$
- $\text{Bel}(\text{neither}) = m_{neither}^m = 0$
- $\text{Pl}(F) = m_F^m + m_{EorF}^m = 0.6111$
- $\text{Pl}(E) = m_E^m + m_{EorF}^m = 0.8730$
- $\text{Pl}(EF) = m_{EorF}^m + m_E^m + m_F^m = 0.9409$
- $\text{Pl}(\text{neither}) = m_{neither}^m = 0$

The same steps are taken when a full cell is encountered, but sensor support is now assigned to fullness and correspondingly, other values are affected.

1. Since this cell is full, assign sensor support for fullness.

- $m_E^s = 0$
- $m_F^s = \lambda_F = 0.3$
- $m_{EorF}^s = 1 - \lambda_F = 0.7$
- $m_{neither}^s = 0$

2. Compute normalization factor K .

- $K = m_E^m \cdot m_F^s + m_F^m \cdot m_E^s + m_{neither}^m \cdot m_{neither}^s = 0.03$

3. Using Dempster's rule of combination, combine sensory bpa values with existing map bpa values.

- $m_E^m = \frac{1}{1 - K} (m_E^m \cdot m_{EorF}^s + m_{EorF}^m \cdot m_E^s + m_E^m \cdot m_E^s) = 0.0679$

- $m_F^m = \frac{1}{1 - K} (m_F^m \cdot m_{EorF}^s + m_{EorF}^m \cdot m_F^s + m_F^m \cdot m_F^s) = 0.3298$

- $m_{EorF}^m = \frac{1}{1 - K} (m_{EorF}^m \cdot m_{EorF}^s) = 0.5432$

- $m_{neither}^m = 0$

4. Compute belief and plausibility function values for all propositions.

- $\text{Bel}(F) = m_F^m + m_{neither}^m = 0.3298$

- $\text{Bel}(E) = m_E^m + m_{neither}^m = 0.0679$

- $\text{Bel}(E \text{ or } F) = m_{EorF}^m + m_E^m + m_F^m + m_{neither}^m = 0.9409$

- $\text{Bel}(\text{neither}) = m_{neither}^m = 0$

- $\text{Pl}(F) = m_F^m + m_{EorF}^m = 0.8730$

- $\text{Pl}(E) = m_E^m + m_{EorF}^m = 0.6111$

- $\text{Pl}(EF) = m_{EorF}^m + m_E^m + m_F^m = 0.9409$

- $\text{Pl}(\text{neither}) = m_{neither}^m = 0$

The greater the number of times a cell is detected as empty or full, the more evidence there is in favour of one proposition or the other. The above described procedure is carried out 5 times for the environment given in Figure 31 and the emptiness and fullness mass (bpa), belief and plausibility values are provided in Tables 7 through 12, over loops 1, 2 and 5. The tables provide values only for the region in Figure 31

that is overlaid with a grid. Since the values are in tabular format, direct reference to the grid in Figure 31 is done using row and column indices, and not using Cartesian coordinate reference. Also, the different shades of grey used for highlighting values in the table are purely for demonstration purposes and are not directly related to colours produced in the final grid based map. It is important to note that since values used to define parameters are small, the regions highlighted in green in the table that are meant to depict an increase in values are not evident from the values in the table due to restrictions on the number of significant figures that can be displayed. The scaling of the axes in the grid based maps in Figures 32 and 33 correspond to row and column indices as shown across all tables.

Table 7 shows the values calculated for m_E^m using Dempster's rule of combination in (2.4.5). Loops 1, 2 and 5 in Tables 7 and 8 contain regions that have a value of 0.1, implying that the cells' bpa values have not changed after initialization. This could be because of two factors- either the region is beyond the sensor's maximum range or the region is occluded from sensor view by an obstacle. At loop 1, regions that were previously initialized to a neutral value of 0.1, have now been updated to reflect increase in evidence for emptiness. Regions highlighted in green show an increase in the mass for emptiness based on Dempster's rule of combining previous map data and newly arrived sensor support. Regions in white containing obstacles, do not receive support for emptiness, and their corresponding strength in evidence decreases and becomes lower than surrounding values of 0.1 that indicate uncertainty. Comparing the regions in dotted green rectangles in Figure 31 with values in Tables 7 through 12, it can be seen that only the inner most walls of all three obstacles are sensed, and thus receive support for fullness, in green. The outer walls remain occluded from view and thus receive no support or update.

Table 8 shows values computed for m_F^m using (2.4.5). Similar to the analysis of the emptiness mass values, at loop 1, the regions highlighted in green in Table 8 show an increase in evidence supporting fullness and uncertainty in regions that are grey. After updating the emptiness and fullness mass values, they are used to compute the belief in emptiness and fullness using (2.4.7). Table 9 shows the belief in emptiness at loop 1, and regions highlighted in green show a clear increase in value displaying strong belief for emptiness. Regions highlighted in light grey have received extremely low belief, however, not as low as 0 which implies that those regions have not been sensed at all. Similarly, Table 10 shows the numerical belief for fullness computed using (2.4.7), and the same analogy is applied to explain regions in green receiving a high degree of belief to depict occupied regions, while those in grey reflect uncertainty.

Tables 11 and 12 show emptiness and fullness plausibility values respectively. Computed using (2.4.8), loop 1 in Table 11 shows regions highlighted in green with a high plausibility for emptiness, meaning that contradicting evidence is low and regions in grey show lesser plausibility. A similar analogy can be used to describe regions highlighted in green in Table 12, that show high plausibility for fullness implying that in those regions, contradicting evidence is not high. As the number of times the region is sensed increases, the values of mass, belief and plausibility change to reflect the support given by new sensory evidence.

Loop 2 in Tables 7 and 8 show regions highlighted in green for an increase in emptiness and fullness masses accordingly, whose values are computed using Dempster's rule of combination in (2.4.5). The increase in values in the green regions reflect that more supporting evidence is present in those regions. This increase in evidence for fullness and emptiness at loop 2 in Tables 7 and 8 respectively, directly influence the increase in belief for emptiness and fullness in Tables 9 and 10, for loop 2 through (2.4.7). Regions that were previously green, now show an increase in values showing exclusive belief for emptiness and fullness respectively, while those regions that were grey in loop 1, have decreased in value in loop 2, showing an increase in certainty favouring the opposing proposition. Similarly the emptiness and fullness masses at loop 2 in Tables 7 and 8 influence the values in the green and grey regions of the emptiness and fullness plausibility at loop 2 in Tables 11 and 12 through (2.4.8).

After a few iterations, it can be seen that the values in the emptiness and fullness mass, belief and plausibility maps settle down to consistent values that describe the environment accurately as shown in the grid maps in Figures 32 and 33. For regions containing values greater than 0 but lesser than 10^{-3} do not reflect as grey regions although they do mean uncertainty in some situations, because the values are extremely small to be recognized as grey when producing maps. The grey regions are better noticed with a bigger environment and increased iterations as shown in Cases

5 and 6.

As mentioned earlier, a drastic increase in values across tables is not noticeable due to small initialization values, and thus values across tables seem unchanged. However, it is important to consider the possibility in cases when the belief and plausibility values are equal and referring back to equation (2.4.9), where belief forms the lower bound and plausibility forms the upper bound of the true probability of a proposition. In the case when belief and plausibility are equal, it implies that there is no room for ambiguity, and thus the true probability is the belief or plausibility value.

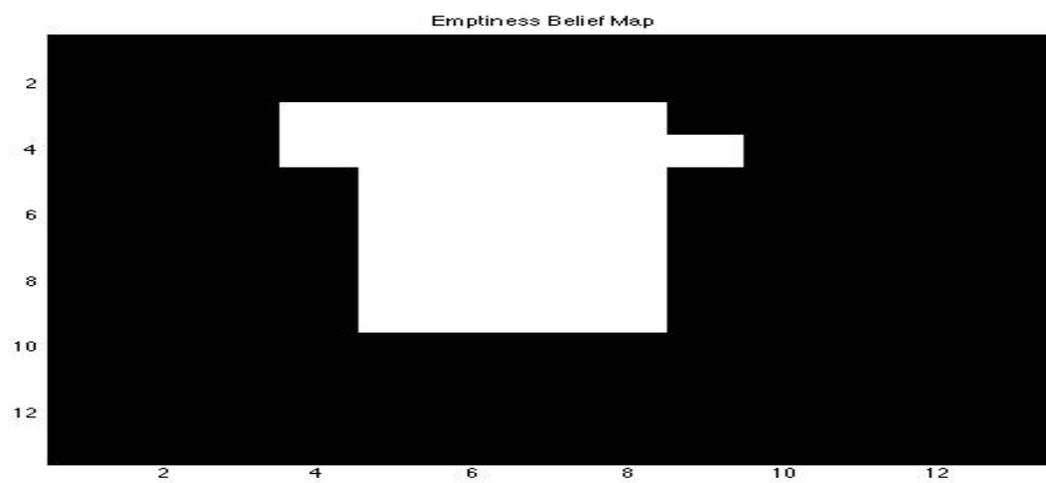


Figure 32: Emptiness belief map

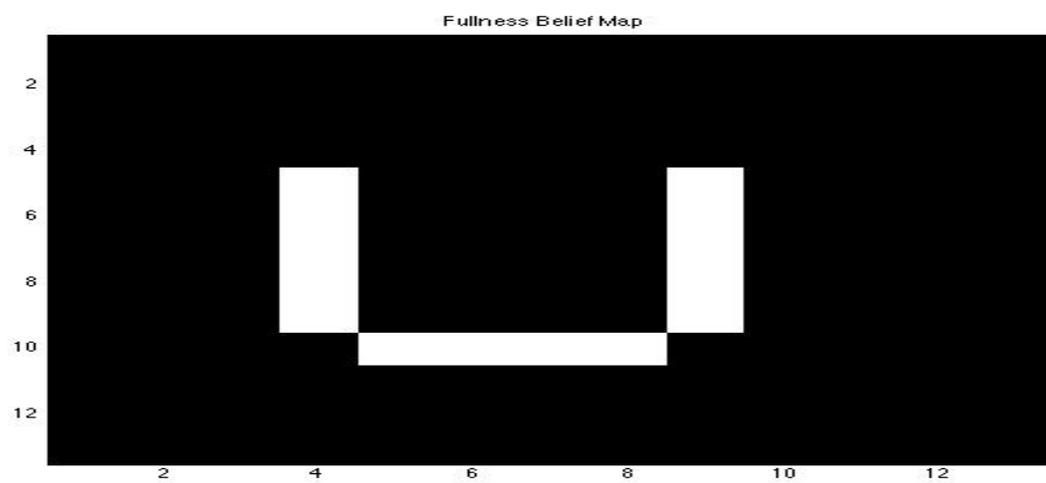


Figure 33: Fullness belief map

4.2 Case 5: Evidential Mapping of the Straight Ladder Environment

Figures 35 through 38 correspond to the straight ladder environment shown in Figure 34, where the vehicle is identified by the black triangle. Initialization values and analysis of the emptiness and fullness mass, belief and plausibility maps are analyzed in detail in Section 4.3.

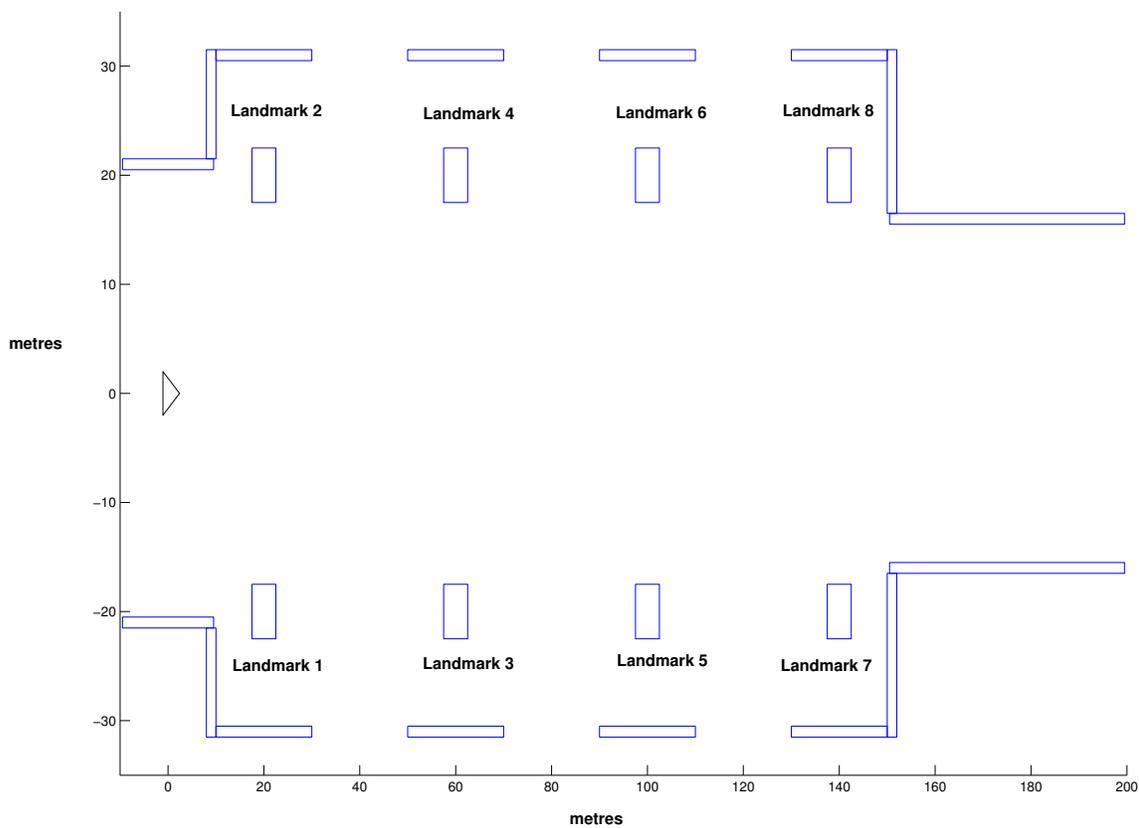


Figure 34: Straight Ladder Environment

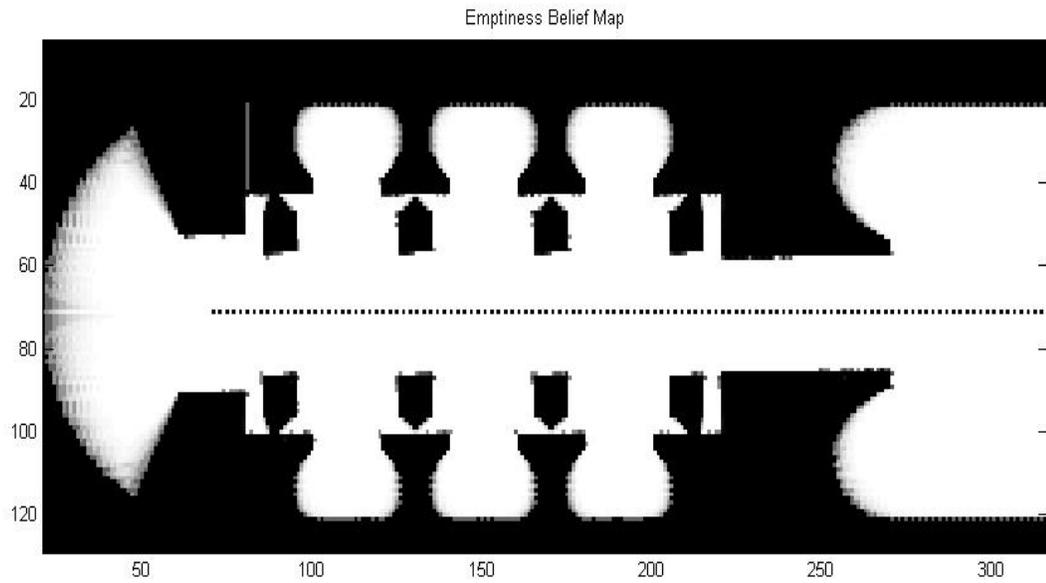


Figure 35: Case 5: Emptiness Belief Map Straight Ladder Environment

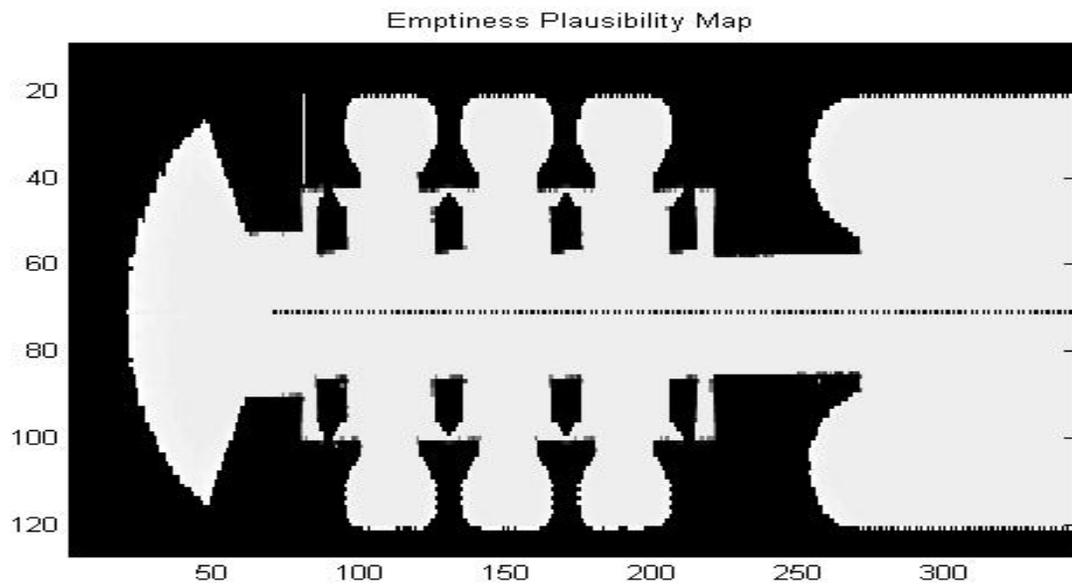


Figure 36: Case 5: Emptiness Plausibility Map Straight Ladder Environment

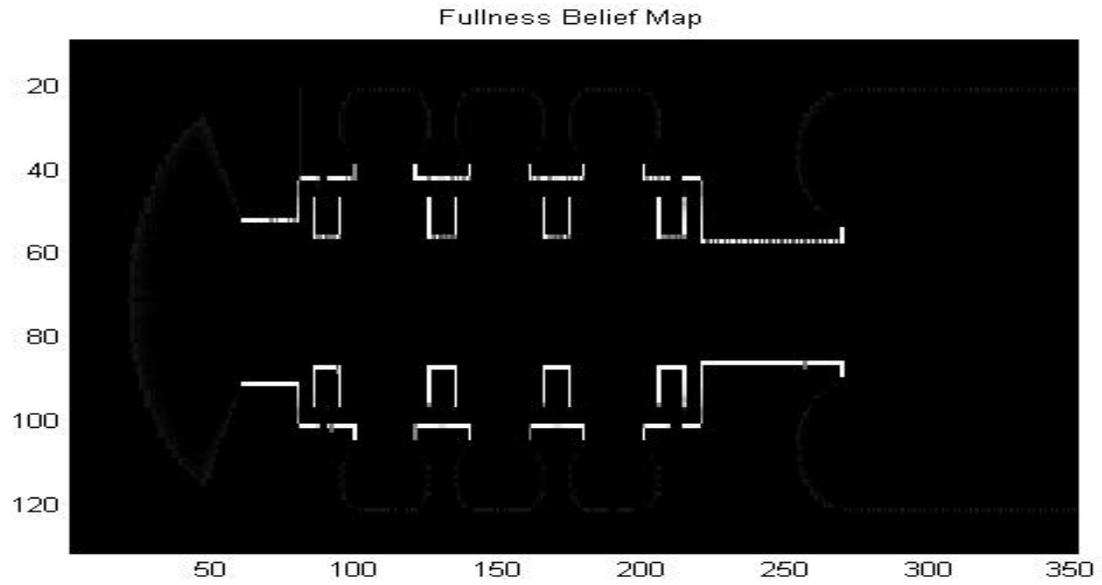


Figure 37: Case 5: Fullness Belief Map Straight Ladder Environment

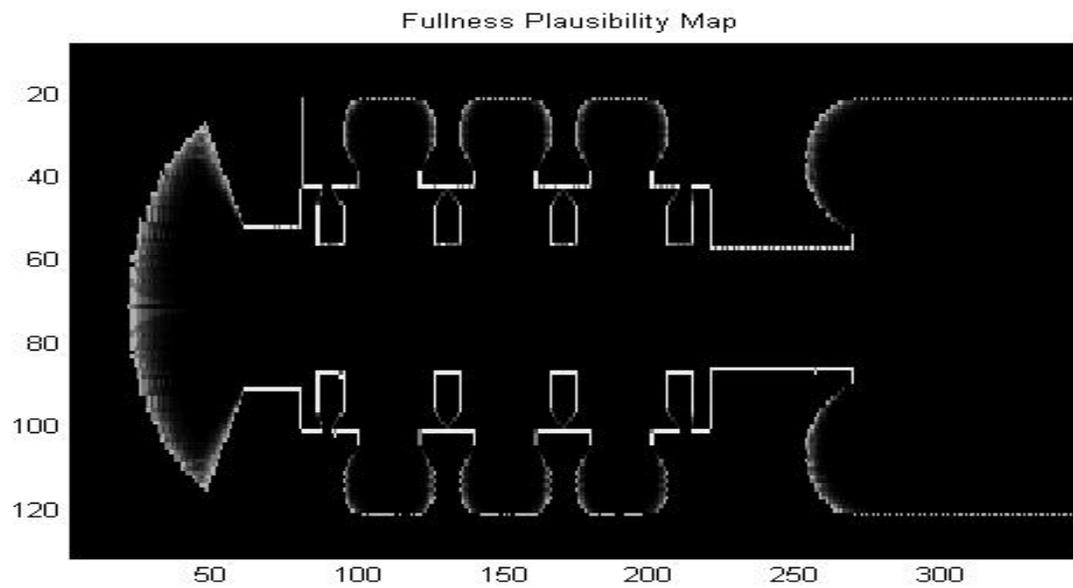


Figure 38: Case 5: Fullness Plausibility Map Straight Ladder Environment

4.3 Case 6: Evidential Mapping of the Circular Ladder Environment

Figures 40 through 43 correspond to the circular ladder environment as shown in Figure 39, where the vehicle is identified by the black triangle.

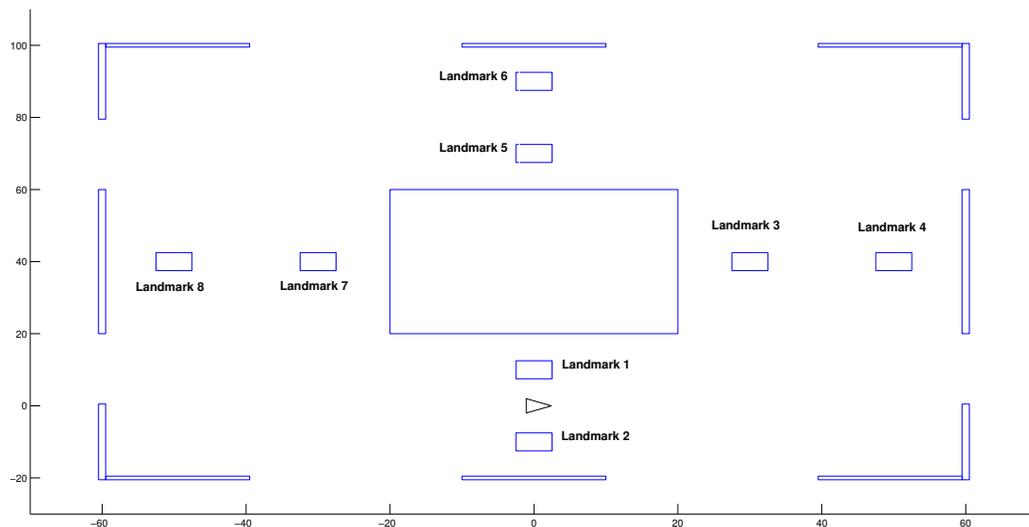


Figure 39: Circular Ladder Environment

Results of the occupancy grid are dependent on the initial values of confidence or belief probability assignment (bpa) assigned to the propositions of emptiness, fullness, either and neither. The maps use grid cells of the size $1\text{m} \times 1\text{m}$. The simulations use the following bpa values:

- $m_E^m = 0.1$
- $m_F^m = 0.1$
- $m_{EorF}^m = 0.8$
- $m_{neither}^m = 0$

When the grid is initialized at the start of simulations, each cell contains all 4 bpa values. The high bpa value for m_{EorF} signifies that all cells in the grid have to either be occupied or empty, but since there is no knowledge about how likely the cells are empty or full, the bpa values assigned to emptiness and fullness are low. Since the cells cannot be neither empty nor full, $m_{neither}$ is 0. As cells get detected the bpa values are updated accordingly to be used as evidence in belief and plausibility functions.

Since sensors are used to detect obstacles, confidence in the sensor's capability to detect correctly must be assigned as well. The bpa's assigned to the sensor's accuracy in detecting full or empty cells are given as follows:

- If cell is detected as empty

1. $\lambda_E = 0.3$

2. $\alpha_E = 0$

3. $\lambda_{EF} = 0.7$

- If cell is detected as full

1. $\lambda_F = 0.3$

2. $\alpha_F = 0$

3. $\lambda_{EF} = 0.7$

The above values signify the ability of the sensor to detect false positives. The values for λ_E and λ_F are given low values to suggest that the sensor is only 30% likely to detect empty cells as empty, and full cells as full. The results obtained thus take into account a sensor that is not highly accurate.

Referring to Figure 39 and comparing with Figure 40 demonstrates the effectiveness of the emptiness belief map. It represents the belief for the emptiness of the cell

derived from a combination of evidence that supports emptiness, which are λ_E , λ_{EF} , m_E and m_{EorF} . Similarly, the fullness belief map shown in Figure 42 represents the belief for fullness derived from a combination of supporting evidence by λ_F , λ_{EF} , m_F and m_{EorF} . Because belief maps take only supporting evidence into account, they form the lower bound in the support for the proposition. In spite of low bpa values used for trust in sensor accuracy, both maps present an almost accurate representation of the true environment. The circular path in the emptiness maps represents the path taken by the vehicle, and those cells are black because of the fullness support given to that cell, when the vehicle is in that cell.

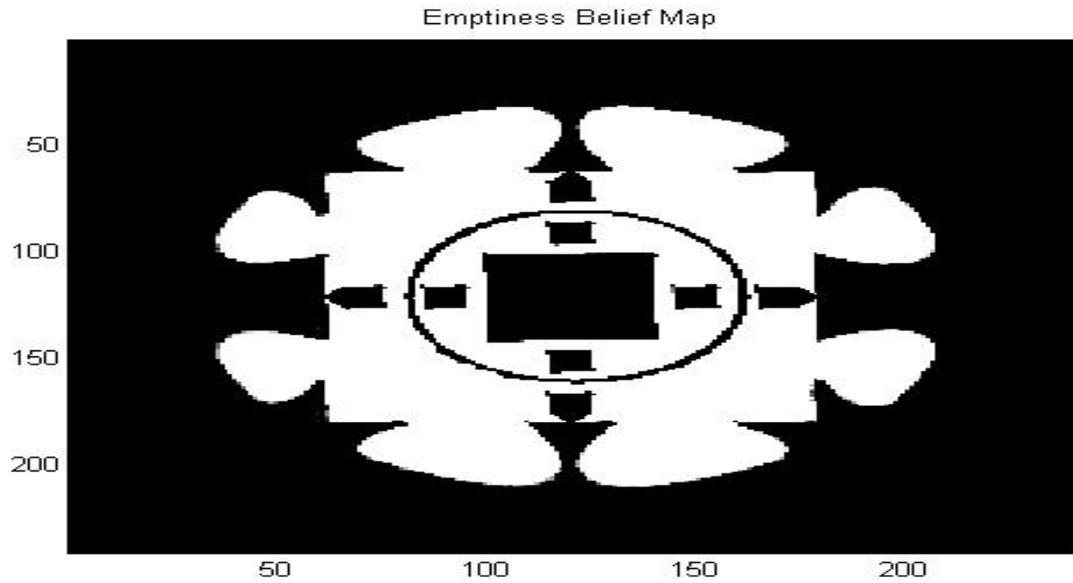


Figure 40: Case 6: Emptiness Belief Map Circular Ladder Environment

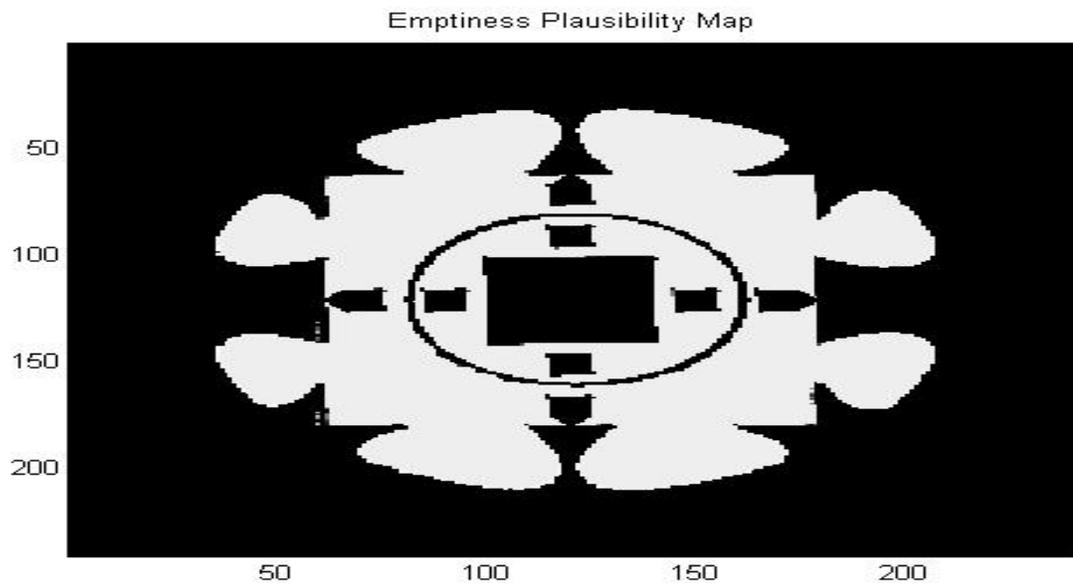


Figure 41: Case 6: Emptiness Plausibility Map Circular Ladder Environment

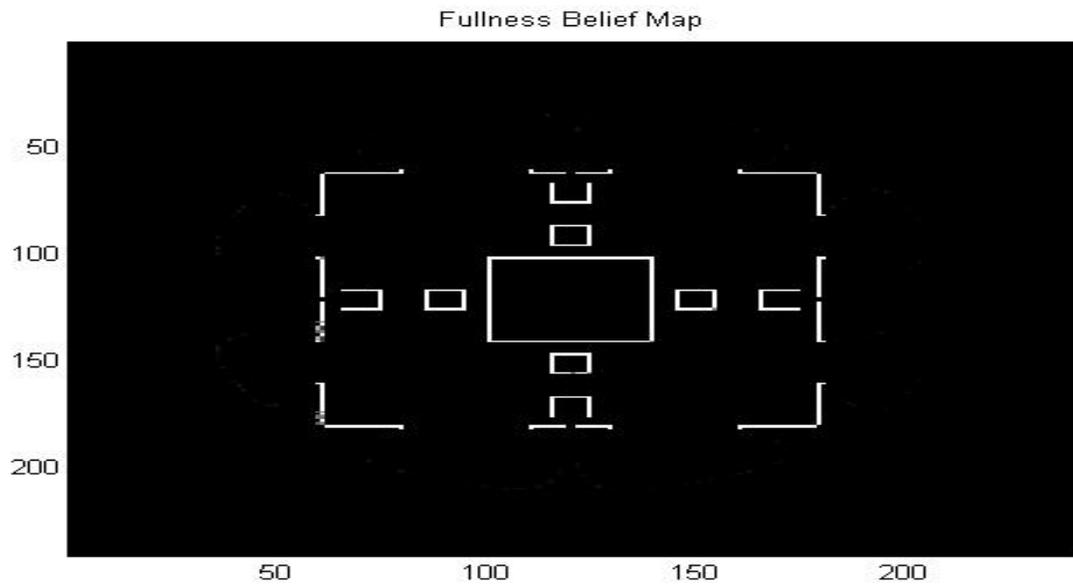


Figure 42: Case 6: Fullness Belief Map Circular Ladder Environment

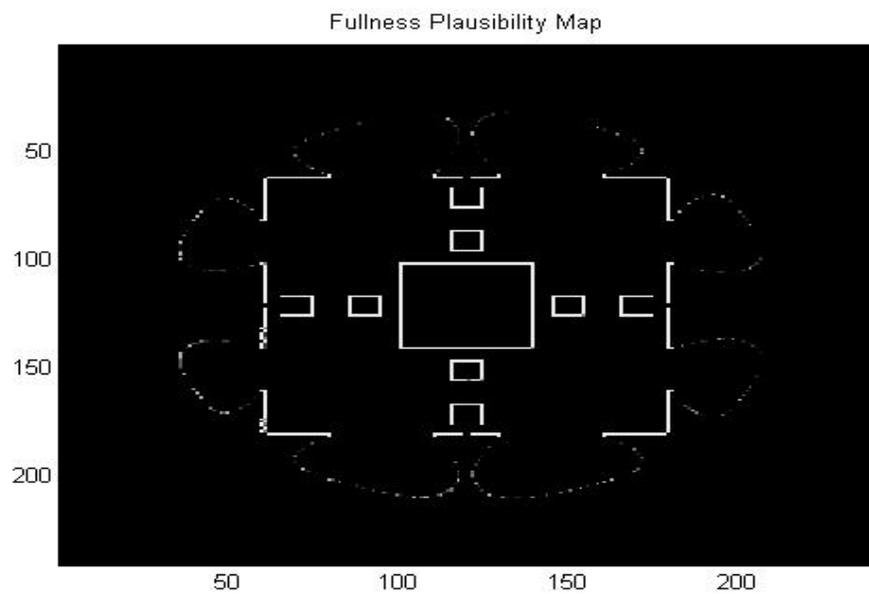


Figure 43: Case 6: Fullness Plausibility Map Circular Ladder Environment

Referring to the emptiness plausibility map in Figure 41, it represents the level of support that remains for emptiness after taking away the level of support provided by evidence that suggests the contrary, which are m_F and $m_{neither}$. Similarly, the fullness plausibility map in Figure 43 represents the level of support that remains after taking away support from evidence suggesting otherwise, which are m_E and $m_{neither}$. Because plausibility maps take into account the evidence supporting the contrary, they form the upper bound on the support for the proposition.

Comparing the emptiness belief and plausibility maps in Figure 44, it can be seen that the belief map has regions that are grey (highlighted in red), however the same regions are white in the plausibility map. This shows that the belief map is more restrictive in assigning confidence in emptiness to regions that do not possibly have as much supporting evidence. However in the plausibility map, those regions are white, indicating that plausibility shows more regions are empty than there truly are.

Similarly, comparing the fullness belief and plausibility maps in Figure 45, regions highlighted in red in the fullness plausibility map, shows more regions in grey than does the belief map. This shows that the plausibility map assumes more regions being full than there truly are. The belief map represents regions for only which it has undisputed evidence for fullness. An interesting correlation is noticeable between the highlighted regions in the emptiness belief map and the fullness plausibility map, which is representative of the definition of fullness plausibility where regions that do not have exclusive emptiness belief will be represented. The grey regions in the emptiness belief map are grey because they did not have exclusive emptiness support, and are thus picked up as regions that may be full in the fullness plausibility map.

For applications involving path planning and obstacle avoidance, the fullness plausibility map and the emptiness belief map should be used over the fullness belief and emptiness plausibility maps, as it is safer to assume more regions being occupied than there truly may be.

Figure 44: Comparison between Emptiness Belief and Plausibility Maps

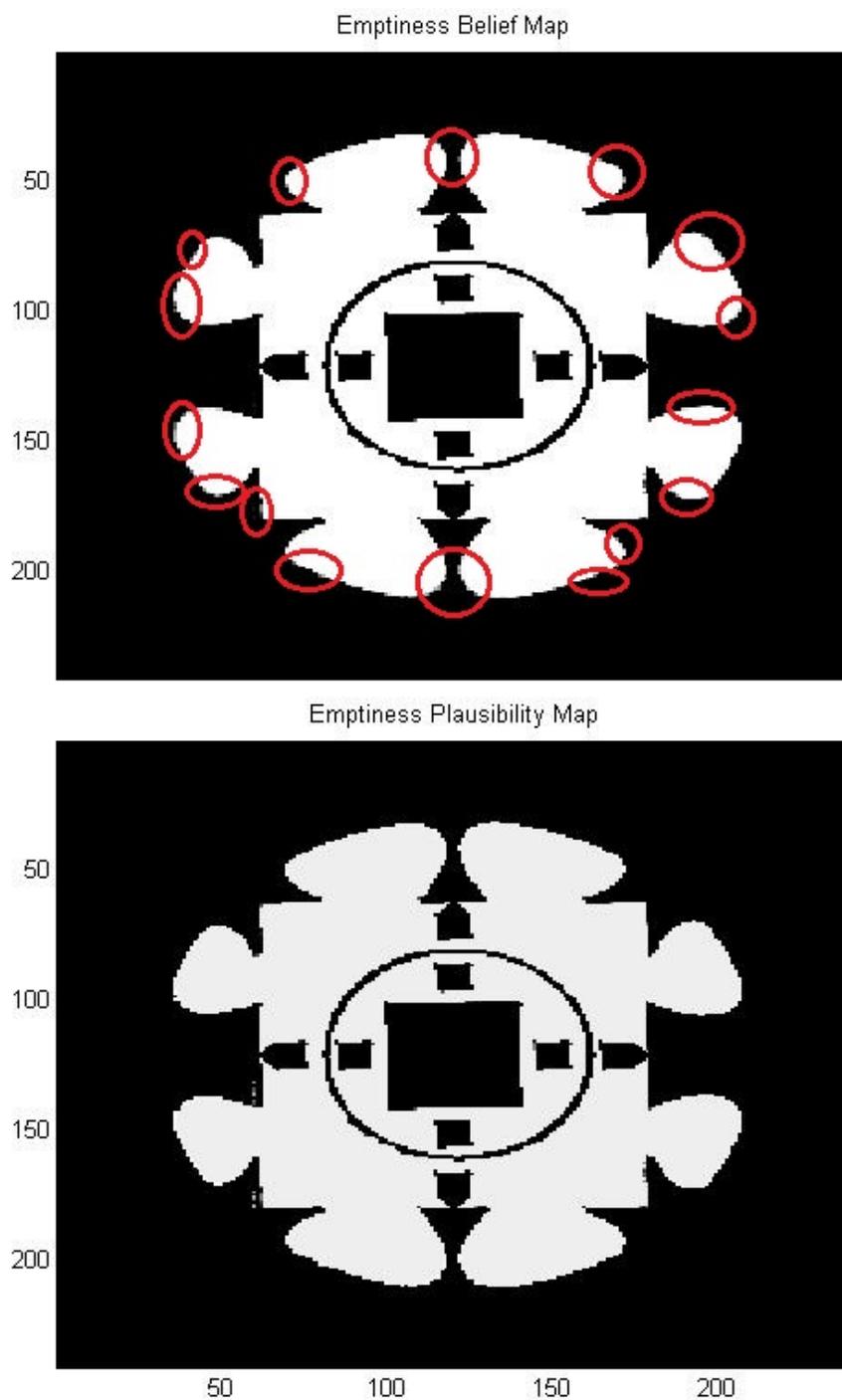
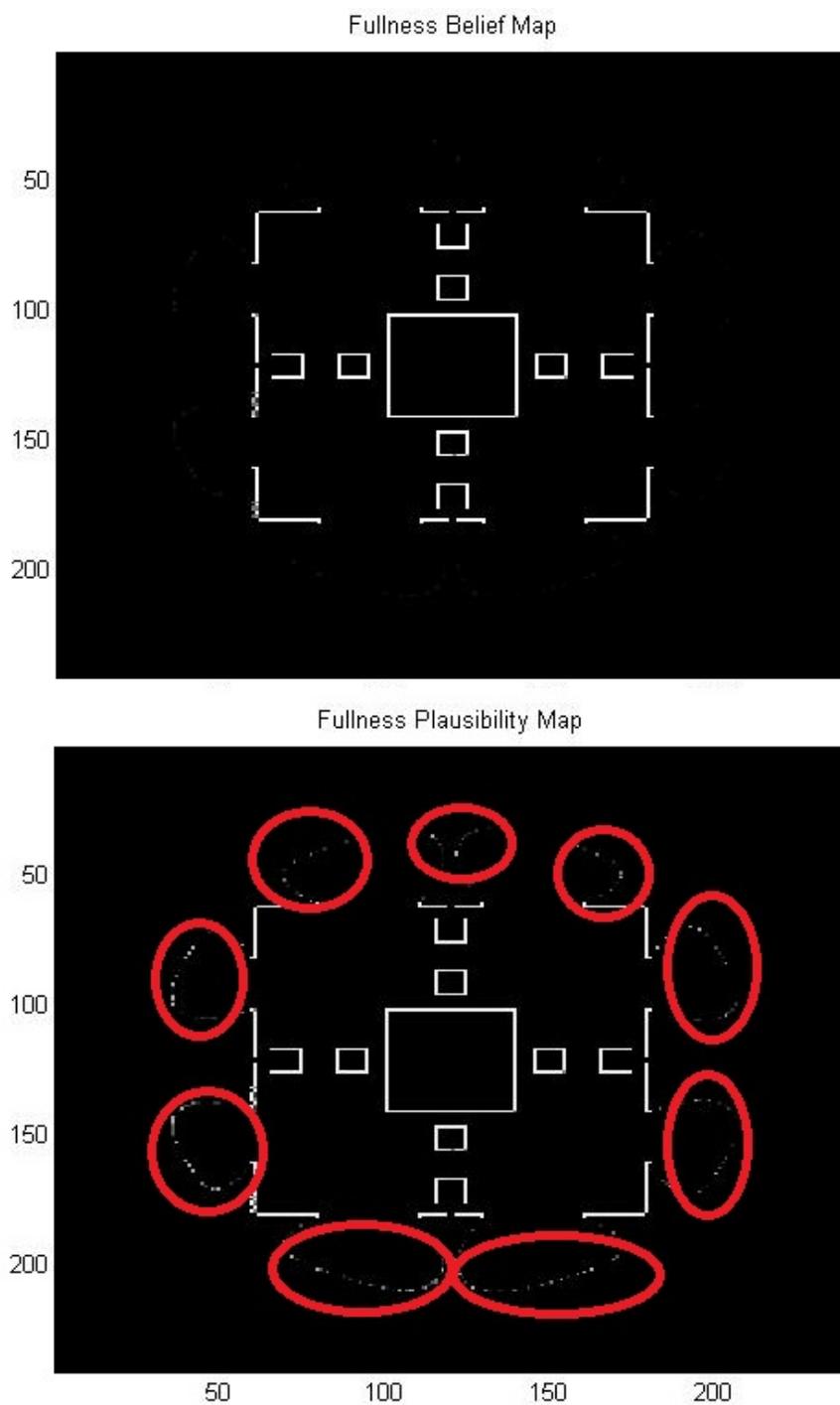


Figure 45: Comparison between Fullness Belief and Plausibility Maps

Chapter 5

Conclusions and Future work

A robot is considered completely autonomous if it achieves the following goals.

1. Estimate its position and the positions of landmarks in the environment with respect to the world centric reference frame.
2. Create a map that can be used for path planning of the robot.
3. Navigate through the environment using the map and appropriate obstacle avoidance techniques.
4. Use the map to plan exploration of unexplored territories of the environment.

This thesis focused on points 1 and 2, by using the Reconfigurable Extended Kalman Filter for state estimation and Demspster-Shafer theory based occupancy grid maps. Estimating the pose of the vehicle and the positions of landmarks in the world centric reference frame, based purely on measurements obtained in terms of range and bearing in the vehicle centric reference frame summarizes the SLAM problem. The challenges offered by the SLAM problem were that of imposing observability, maintaining observability and minimizing estimation error while imposing conditions to maintain observability. The state estimation using the EKF estimates the states of the landmarks and creates a map in terms of point landmarks, however, knowledge of

landmarks in terms of coordinates of a point is not sufficient to be used for obstacle avoidance and path planning by the robot. Producing an informative map that showed support for occupancy in regions occupied by obstacles and emptiness for those regions that were free terrain, was the next important challenge.

The thesis has contributed to addressing the above mentioned challenges by

1. Simulating a vehicle in motion with a finite range sensor, thus mimicking a practical SLAM situation where landmarks can constantly enter and exit the sensor's field of view. This setting necessitates re-selection of anchors and tethers.
2. Imposing observability by selecting an anchor and tether to define the origin and orientation of the world centric reference frame or F_w , by holding constant two coordinates of the anchor and one coordinate of the tether. This algorithm was initially used by Khan in [11] and based on initial work by Aitken in [69].
3. Maintaining observability by enabling re-selection of anchors and tethers, and thus redefining F_w periodically, as the vehicle traverses the environment. Results corresponding to Case 1 in Chapter 3, demonstrate that while observability is imposed using an anchor-tether pair, as the vehicle moves farther away, the filter is slightly resistant to bias corrections. Cases 3 and 4 show the positive impact of enabling re-selection of anchors and tethers on bias corrections, in the straight and circular ladder environments. This contribution is an implementation of the future work suggested by Khan of [11].
4. Enabling switching of estimation of the x or y coordinate of the tether, depending on whether the anchor-tether pair is horizontal or vertical in F_w , failing which the formulated SLAM problem becomes unobservable even with the presence of an anchor and tether. Results of Case 2 in Chapter 3 show the effect of

estimating the wrong coordinate of the tether, when the anchor-tether pair is horizontal in F_w ; initialization errors are extremely high and the filter does not approach steady state. Cases 3 and 4 prove that enabling automatic re-selection of anchors and tethers, aided with estimating the appropriate coordinate of the tether maintains observability as well as accuracy, in the straight and circular ladder environments. Constantly evaluating the anchor, tether and other landmarks to enable appropriate re-selection and evaluating the angle between the anchor-tether pair in F_w to determine the coordinate of the tether to be estimated, involves configuring the EKF at every time instant, thus making it a Reconfigurable EKF.

5. Creating two degrees of confidence in emptiness and fullness, by developing belief and plausibility grid based maps based on Dempster-Shafer theory of evidential reasoning, thus describing the environment effectively for obstacle avoidance. Results of Cases 5 and 6 demonstrate the accuracy and reliability of using DS theory for mapping purposes.

Through the contributions, the thesis has addressed the main issues that make the formulated SLAM problem unobservable, and has proved that the solutions for the issues discussed in Chapter 2, are indeed successful through simulations in Chapter 3, thus making the problem observable. The implementation of DS-theory based occupancy grid maps has established that grid based based maps are a better form of representing the environment rather the point landmarks and are reasonably accurate even with unreliable sensors. The main aim of testing a grid based map approach was to allow for future work to combine both the state estimation and the mapping algorithms. Successful analysis and testing of grid based maps allow for the following possible extensions to the the current work.

1. The occupancy grid based maps can be utilized for obstacle avoidance and path

planning of the autonomous vehicle.

2. Concurrently developing the grid based map with the state estimation process as the vehicle travels in the environment could be used for exploration. This is especially applicable to maps created with DS-theory rather than with those created using Bayesian theory, as unexplored or occluded regions have no support for emptiness or fullness, indicating that those regions are yet to be explored.
3. The formulated SLAM problem is currently 2 dimensional, and thus the Reconfigurable EKF as well as the occupancy grid maps are 2 dimensional. Considering the possible applications of the occupancy grid maps for obstacle avoidance, extensions to the work could involve in formulating the problem in 3 dimensions, and using the Reconfigurable EKF for a 3D environment and producing 3D occupancy grid maps.
4. Finally information from the grid based map could be used to detect and choose special landmarks for use in the state estimation algorithm. However, this approach either assumes perfect data association or will have to solve the data association problem first.

Simulations conducted and results obtained are aimed to provide an insight into the main challenge that the SLAM problem faces- observability. There are however various aspects of SLAM like front-end sensing, processing of sensor data, formulation of the mechanics of the autonomous vehicle, data association and loop closure, that continue to be researched on. This thesis aims to be the foundation for future work involving implementation and testing of information obtained from occupancy grid maps to allow for state estimation using the Reconfigurable EKF and loop closure in real environments.

References

- [1] I. Jebari, S. Bazeille, E. Battesti, H. Tekaya, M. Klein, A. Tapus, D. Filliat, C. Meyer, S.-H. Ieng, R. Benosman, E. Cizeron, J.-C. Mamanna, and B. Pothier. “Multi-sensor semantic mapping and exploration of indoor environments.” In “Technologies for Practical Robot Applications (TePRA), 2011 IEEE Conference on,” pages 151–156 (2011).
- [2] M. Sakuta, S. Takanashi, and T. Kubota. “An image based path planning scheme for exploration rover.” In “Robotics and Biomimetics (ROBIO), 2011 IEEE International Conference on,” pages 150–155 (2011).
- [3] D. Meger, M. Muja, S. Helmer, A. Gupta, C. Gamroth, T. Hoffman, M. Baumann, T. Southey, P. Fazli, W. Wohlkinger, P. Viswanathan, J. Little, D. Lowe, and J. Orwell. “Curious george: An integrated visual search platform.” In “Computer and Robot Vision (CRV), 2010 Canadian Conference on,” pages 107–114 (2010).
- [4] D. Rawlinson and R. Jarvis. “Ways to tell robots where to go - directing autonomous robots using topological instructions.” *Robotics Automation Magazine, IEEE* **15**(2), 27–36. ISSN 1070-9932 (2008).
- [5] H.-H. Ehrlicke, G. Daiber, and W. Strasser. “The vision camera: An interactive tool for volume data exploration and navigation.” In “Visualization, 1993. Visualization '93, Proceedings., IEEE Conference on,” pages 25–30 (1993).
- [6] M. Suppa, S. Kielhofer, J. Langwald, F. Hacker, K. Strobl, and G. Hirzinger. “The 3d-modeller: A multi-purpose vision platform.” In “Robotics and Automation, 2007 IEEE International Conference on,” pages 781–787. ISSN 1050-4729 (2007).
- [7] J. De Geeter, H. Van Brussel, J. De Schutter, and M. Decreton. “Recognising and locating objects with local sensors.” In “Robotics and Automation, 1996.

- Proceedings., 1996 IEEE International Conference on,” volume 4, pages 3478–3483 vol.4. ISSN 1050-4729 (1996).
- [8] Y. Arayici and A. Hamilton. “Modeling 3d scanned data to visualize the built environment.” In “Information Visualisation, 2005. Proceedings. Ninth International Conference on,” pages 509–514. ISSN 1550-6037 (2005).
- [9] A. Elfes. “Using occupancy grids for mobile robot perception and navigation.” *Computer* **22**(6), 46–57. ISSN 0018-9162 (1989).
- [10] B. Yamauchi. “A frontier-based approach for autonomous exploration.” In “Computational Intelligence in Robotics and Automation, 1997. CIRA’97., Proceedings., 1997 IEEE International Symposium on,” pages 146–151 (1997).
- [11] D. Khan. *Observable Autonomous SLAM in Two-Dimensional Dynamic Environments*. Master’s thesis, Carleton University (2011).
- [12] T. Y. Yang. *Mapping and Localization for Mobile Robots*. Master’s thesis, Carleton University (2004).
- [13] H. Durrant-Whyte and T. Bailey. “Simultaneous localization and mapping: part i.” *Robotics Automation Magazine, IEEE* **13**(2), 99–110. ISSN 1070-9932 (2006).
- [14] H. Durrant-Whyte. “Uncertain geometry in robotics.” In “Robotics and Automation. Proceedings. 1987 IEEE International Conference on,” volume 4, pages 851–856 (1987).
- [15] R. Smith, M. Self, and P. Cheeseman. “Estimating uncertain spatial relationships in robotics.” In “Robotics and Automation. Proceedings. 1987 IEEE International Conference on,” volume 4, pages 850–850 (1987).
- [16] D. Silver and A. Stentz. “Monte carlo localization and registration to prior data for outdoor navigation.” In “Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on,” pages 510–517. ISSN 2153-0858 (2011).
- [17] D. Fox, W. Burgard, F. Dellaert, and S. Thrun. “Monte carlo localization: Efficient position estimation for mobile robots.” In “Proceedings of the Sixteenth National Conference on Artificial Intelligence and the Eleventh Innovative Applications of Artificial Intelligence Conference Innovative Applications of Artificial Intelligence,” AAAI ’99/IAAI ’99, pages 343–349. American Association for Artificial Intelligence, Menlo Park, CA, USA. ISBN 0-262-51106-1 (1999).

- [18] D. Asmar, J. Zelek, and S. Abdallah. “Smartslam: localization and mapping across multi-environments.” In “Systems, Man and Cybernetics, 2004 IEEE International Conference on,” volume 6, pages 5240–5245 vol.6. ISSN 1062-922X (2004).
- [19] A. G. Technical Staff, The Analytic Sciences Corporation. *Applied Optimal Estimation*. THE M.I.T PRESS, eleventh edition (1989).
- [20] S. Chen. “Kalman filter for robot vision: A survey.” *Industrial Electronics, IEEE Transactions on* **59**(11), 4409–4420. ISSN 0278-0046 (2012).
- [21] J.-G. Kang, W.-S. Choi, S.-Y. An, and S. young Oh. “Augmented ekf based slam method for improving the accuracy of the feature map.” In “Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on,” pages 3725–3731. ISSN 2153-0858 (2010).
- [22] S. Panzieri, F. Pascucci, and R. Setola. “Interlaced extended kalman filter for real time navigation.” In “Intelligent Robots and Systems, 2005. (IROS 2005). 2005 IEEE/RSJ International Conference on,” pages 2780–2785 (2005).
- [23] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. “Fastslam: A factored solution to the simultaneous localization and mapping problem.” In “Eighteenth National Conference on Artificial Intelligence,” pages 593–598. American Association for Artificial Intelligence, Menlo Park, CA, USA. ISBN 0-262-51129-0 (2002).
- [24] C. Weyers and G. Peterson. “Improving occupancy grid fastslam by integrating navigation sensors.” In “Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on,” pages 859–864. ISSN 2153-0858 (2011).
- [25] M. Montemerlo. *FastSLAM: A Factored Solution to the Simultaneous Localization and Mapping Problem With Unknown Data Association*. Ph.D. thesis, The Robotics Institute Carnegie Mellon University (2003).
- [26] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. “FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges.” In “Eighteenth International Joint Conference on Artificial Intelligence (IJCAI),” Acapulco, Mexico (2003).
- [27] Z. Kurt-Yavuz and S. Yavuz. “A comparison of ekf, ukf, fastslam2.0, and ukf-based fastslam algorithms.” In “Intelligent Engineering Systems (INES), 2012 IEEE 16th International Conference on,” pages 37–43 (2012).

- [28] M. Montemerlo and S. Thrun. “Simultaneous localization and mapping with unknown data association using fastslam.” In “Robotics and Automation, 2003. Proceedings. ICRA '03. IEEE International Conference on,” volume 2, pages 1985–1991 vol.2. ISSN 1050-4729 (2003).
- [29] A. Monjazez, J. Sasiadek, and D. Neculescu. “Autonomous navigation among large number of nearby landmarks using fastslam and ekf-slam - a comparative study.” In “Methods and Models in Automation and Robotics (MMAR), 2011 16th International Conference on,” pages 369–374 (2011).
- [30] C. Stachniss, D. Hahnel, and W. Burgard. “Exploration with active loop-closing for fastslam.” In “Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings. 2004 IEEE/RSJ International Conference on,” volume 2, pages 1505–1510 vol.2 (2004).
- [31] M. Cummins and P. M. Newman. “Appearance-only slam at large scale with fab-map 2.0.” *I. J. Robotic Res.* **30**(9), 1100–1123 (2011).
- [32] M. C. Newman. “Fab-map: Probabilistic localization and mapping in the space of appearance.” **SECS 175**. (2008).
- [33] R. Hermann and A. J. Krener. “Nonlinear controllability and observability.” *Automatic Control, IEEE Transactions on* **22**(5), 728–740. ISSN 0018-9286 (1977).
- [34] Z. Chen. “Local observability and its application to multiple measurement estimation.” *Industrial Electronics, IEEE Transactions on* **38**(6), 491–496. ISSN 0278-0046 (1991).
- [35] H. NIJMEIJER. “Observability of autonomous discrete time non-linear systems: a geometric approach.” *International Journal of Control* **36**(5), 867–874 (1982).
- [36] Y. Song and J. Grizzle. “The extended kalman filter as a local asymptotic observer for nonlinear discrete-time systems.” In “American Control Conference, 1992,” pages 3365–3369 (1992).
- [37] K. W. Lee, W. S. Wijesoma, and J. I. Guzman. “On the observability and observability analysis of slam.” In “IROS,” pages 3569–3574. IEEE (2006).
- [38] G. Huang, A. Mourikis, and S. Roumeliotis. “Observability-based rules for designing consistent ekf slam estimators.” *The International Journal of Robotics* (2009).

- [39] J. Andrade-Cetto and A. Sanfeliu. “The effects of partial observability in slam.” In “Robotics and Automation, 2004. Proceedings. ICRA '04. 2004 IEEE International Conference on,” volume 1, pages 397–402 Vol.1. ISSN 1050-4729 (2004).
- [40] T. Vidal-Calleja, M. Bryson, S. Sukkarieh, A. Sanfeliu, and J. Andrade-Cetto. “On the observability of bearing-only slam.” In “Robotics and Automation, 2007 IEEE International Conference on,” pages 4114–4119. ISSN 1050-4729 (2007).
- [41] L. D. L. Perera, A. Melkumyan, and E. Nettleton. “On the linear and nonlinear observability analysis of the slam problem.” In “Mechatronics, 2009. ICM 2009. IEEE International Conference on,” pages 1–6 (2009).
- [42] A. Souici, M. Courdesses, A. Ouldali, and R. Chatila. “Full-observability analysis and implementation of the general slam model.” *International Journal of Systems Science* **44**(3), 568–581 (2013; 2011).
- [43] S. Thrun. “Robotic mapping: A survey.” In G. Lakemeyer and B. Nebel, editors, “Exploring Artificial Intelligence in the New Millenium,” Morgan Kaufmann. To appear (2002).
- [44] M. E. Yeap, Wai K; Jefferies. *Robotics and cognitive approaches to spatial mapping*, volume 38 of *Springer tracts in advanced robotics*. Springer (2008).
- [45] *Mobile robot localization and map building: a multisensor fusion approach*. Kluwer Academic Publishers (1999).
- [46] M. W. M. G. Dissanayake, P. Newman, S. Clark, H. Durrant-Whyte, and M. Csorba. “A solution to the simultaneous localization and map building (slam) problem.” *Robotics and Automation, IEEE Transactions on* **17**(3), 229–241. ISSN 1042-296X (2001).
- [47] J. Guivant and E. Nebot. “Optimization of the simultaneous localization and map-building algorithm for real-time implementation.” *Robotics and Automation, IEEE Transactions on* **17**(3), 242–257. ISSN 1042-296X (2001).
- [48] J. J. Leonard, H. F. Durrant-Whyte, and I. J. Cox. “Dynamic map building for an autonomous mobile robot.” *The International Journal of Robotics Research* **11**(4), 286–298 (1992).
- [49] H. Shatkay. “Learning models for robot navigation.” (1998).

- [50] H. Shatkay and L. P. Kaelbling. “Learning topological maps with weak local odometric information.” In “IN PROCEEDINGS OF IJCAI-97. IJCAI, INC,” pages 920–929 (1997).
- [51] S. Thrun. “A probabilistic online mapping algorithm for teams of mobile robots.” *International Journal of Robotics Research* **20**, 2001 (2001).
- [52] S. Thrun, W. Burgard, D. Fox, H. Hexmoor, and M. Mataric. “A probabilistic approach to concurrent mapping and localization for mobile robots.” In “Machine Learning,” pages 29–53 (1998).
- [53] I. Stamos and P. Allen. “Integration of range and image sensing for photo-realistic 3d modeling.” In “Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on,” volume 2, pages 1435–1440 vol.2. ISSN 1050-4729 (2000).
- [54] R. Bajcsy, R. Enciso, G. Kamberova, L. Nocera, and R. Sara. “3d reconstruction of environments for virtual collaboration.” In “Applications of Computer Vision, 1998. WACV '98. Proceedings., Fourth IEEE Workshop on,” pages 160–167 (1998).
- [55] R. E. Kalman. “A new approach to linear filtering and prediction problems.” (1960).
- [56] S. J. Russell and P. Norvig. *Artificial intelligence: a modern approach*. Prentice Hall. ISBN 9780132071482; 9780136042594; 0132071487; 0136042597 (2010).
- [57] L. Rabiner. “A tutorial on hidden markov models and selected applications in speech recognition.” *Proceedings of the IEEE* **77**(2), 257–286. ISSN 0018-9219 (1989).
- [58] G. E. Monahan. “A survey of partially observable markov decision processes: Theory, models, and algorithms.” *Management Science* **28**(1), pp. 1–16. ISSN 00251909 (1982).
- [59] W. Lovejoy. “A survey of algorithmic methods for partially observed markov decision processes.” *Annals of Operations Research* **28**(1), 47–65. ISSN 0254-5330 (1991).
- [60] T. Yang and V. Aitken. “Evidential mapping for mobile robots with range sensors.” *Instrumentation and Measurement, IEEE Transactions on* **55**(4), 1422–1429. ISSN 0018-9456 (2006).

- [61] F. Lu and E. Milios. “Globally consistent range scan alignment for environment mapping.” *AUTONOMOUS ROBOTS* 4, 333–349 (1997).
- [62] J.-S. Gutmann and B. Nebel. “Navigation mobiler roboter mit laserscans.” In “Autonome Mobile Systeme 1997,” Informatik aktuell, pages 36–47. Springer Berlin Heidelberg. ISBN 978-3-540-63513-0 (1997).
- [63] I. M. Rekleitis. “A particle filter tutorial for mobile robot localization.” Technical Report TR-CIM-04-02, Centre for Intelligent Machines, McGill University, 3480 University St., Montreal, Québec, CANADA H3A 2A7 (2004).
- [64] M. Cummins and P. Newman. “Appearance-only slam at large scale with fab-map 2.0.” *The International Journal of Robotics Research* (2010).
- [65] B. Yamauchi. “A frontier-based approach for autonomous exploration.” In “Computational Intelligence in Robotics and Automation, 1997. CIRA’97., Proceedings., 1997 IEEE International Symposium on,” pages 146–151 (1997).
- [66] C. Stachniss and W. Burgard. “Mapping and exploration with mobile robots using coverage maps.” In “Intelligent Robots and Systems, 2003. (IROS 2003). Proceedings. 2003 IEEE/RSJ International Conference on,” volume 1, pages 467–472 vol.1 (2003).
- [67] H. C. et al. *Principles of Robot Motion: Theory, Algorithms, and Implementation*. The MIT Press (2005).
- [68] K. Ogata. *Modern Control Engineering*. Pearson Prentice Hall, 5th edition (2010).
- [69] V. C. Aitken. *Sliding mode state estimation for nonlinear discrete-time systems: Applications in image sequence analysis*. Ph.D. thesis, Carleton University (1995).
- [70] V. Aitken. “On the 2d discrete-time slam observability problem, unpublished research notes.” (2010).
- [71] A. Elfes. “Using occupancy grids for mobile robot perception and navigation.” *Computer* 22(6), 46–57. ISSN 0018-9162 (1989).
- [72] H. Moravec. “Sensor fusion in certainty grids for mobile robots.” In A. Casals, editor, “Sensor Devices and Systems for Robotics,” volume 52 of *NATO ASI Series*, pages 253–276. Springer Berlin Heidelberg. ISBN 978-3-642-74569-0 (1989).

- [73] H. Moravec. “Certainty grids for mobile robots.” In “NASA/JPL Space Tele-robotics Workshop,” volume 1, pages 307–312 (1987).
- [74] A. Dempster. “A generalization of bayesian inference.” In R. Yager and L. Liu, editors, “Classic Works of the Dempster-Shafer Theory of Belief Functions,” volume 219 of *Studies in Fuzziness and Soft Computing*, pages 73–104. Springer Berlin Heidelberg. ISBN 978-3-540-25381-5 (2008).
- [75] G. Shafer. *Encyclopedia of Artificial Intelligence*. Wiley, second edition (1992).
- [76] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press (1976).
- [77] Y. Yang, A. Minai, and M. Polycarpou. “Evidential map-building approaches for multi-uav cooperative search.” In “American Control Conference, 2005. Proceedings of the 2005,” pages 116–121. ISSN 0743-1619 (2005).
- [78] A. S. Juan Andrade-Cetto. *Environment Learning for Indoor Mobile Robots*, volume 23 of *Springer Tracts in Advanced Robotics*. Springer Berlin Heidelberg (2006).