Bearing-Only Localization in Uncertain Environments

Master's Thesis

submitted in partial fulfillment of the requirements
for the degree of

Bachelor & Master of Technology

by

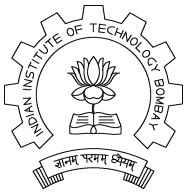
Pranav Nalin Thakkar
150010009

under the guidance of

Prof. Leena Vachhani

and

Prof. Hemendra Arya



Department of Aerospace Engineering
Indian Institute of Technology, Bombay
Mumbai - 400076.

2020

Dedicated to
my mother
Mrs. Nita Nalin Thakkar

and
my father
Mr. Nalin Harish Thakkar

Dissertation Approval Sheet

This is to certify that the dissertation titled

Bearing-Only Localization in Uncertain Environments

By

Pranav Nalin Thakkar
(150010009)

is approved for the degree of **Bachelor & Master of Technology**.

Prof. Leena Vachhani (Guide)

Prof. Hemendra Arya (Co-Guide)

Internal Examiner

External Examiner

Chairperson

Date : _____

Abstract

Since the introduction of the camera as a sensor, bearing-only navigation has become a well-researched area in robotics. One of the long-standing problems in literature has been the effect of 'uncertain' features in the environment. These features possess uncertainty by virtue of periodic motion, occasional shifts or a lack of accurate or updated knowledge about the surroundings. Several map-building strategies including EKF-SLAM have been proposed to account for such uncertainties in static environment features, and account for semi-static or dynamic ones. This work demonstrates the efficacy of an augmented state formulation paired with modified Extended Kalman Filters, towards computationally faster pose estimation in the presence of static environments where the landmark positions are not exactly known. Such a combination does not require any map building or disturbance estimation for accurate localization, as it is effective in environments with large number of features whose known positions are inaccurate. The disturbance rejection properties of these filtering techniques are also demonstrated in a multi-rate setting, for a robot model and environment in Gazebo. This thesis also discusses the observability properties of the bearing-only localization problem formulated in polar coordinates. It considers variants of the joint state and parameter estimation problem, discusses the unobservable spaces for each variant and expresses them as loci in 2-D spaces that explain the degeneracy. These results are further verified by transformations that decompose the system into observable and unobservable parts.

Keywords: robust estimation, bearing-only localization, observability analysis of robot localization

Acknowledgments

I express my sincere gratitude towards my guides **Prof. Leena Vachhani** and **Prof. Hemendra Arya** for their constant help, encouragement and inspiration throughout the project work. Without their invaluable guidance, this work would never have been a successful one. I would also like to thank my mentor, Prashant Vivek Patil, for always being a pillar of support. I am extremely grateful to members of the ARMS Lab at the Systems & Controls Engineering Department, namely Vaibhav Kadam, Kaustubh Sadekar, Dimple Bhuta, Adwaith Vijaykumar, Meghdeep Jana and Unnati Ashar, for their valuable suggestions and helpful discussions. Last, but not the least, I would like to thank the faculty and staff at the Department of Aerospace Engineering for their help during my journey here at IIT Bombay.

Pranav Nalin Thakkar
IIT Bombay
July 12, 2020

Contents

List of Figures

Chapter 1

Introduction

Localization refers to the determination of a vehicle's position and orientation at a given time instant, using sensory data and a map of the environment. It is an essential step that precedes motion planning in two-dimensional or three-dimensional space, especially when the control that is being implemented depends on the state of the robot. For mobile robots, the concept of localization is crucial when the system is modelled to possess uncertainties in various forms, such as in the sensor, in the environment or in the way the robot itself moves. These uncertainties are incorporated by using probabilistic functions for each part of the system that is non-deterministic in nature. In such situations, localization refers to the bounding of the robot pose[1] to an interval, with a certain degree of confidence.

For real-world mobile robotic systems such as unicycle-like ground robots or multicopters, localization is possible using two types of sensors:

- **Proprioceptive Sensors:** These keep a track of the internal states of a robot, such as angular velocity of motor, battery voltage, robot acceleration, etc. Sensors such as odometry sensors and inertial measurement units (IMUs) classify as proprioceptive sensors.
- **Exteroceptive Sensors:** These extract data from the environment, such as light intensity, distance from various objects, etc. Sensors such as laser-based scanners, ultrasonic transmitters & receiver modules and cameras classify as exteroceptive sensors.

[1] combined position and orientation

1.1 Motivation

The introduction of visual servoing in robotics by Espiau, Chaumette & Rives [1] led to a shift towards cameras as the primary sensor for extracting information from the environment. The compactness, accuracy and low-cost nature of vision-based sensors and the sheer volume of data made available by them paved the way for cameras to replace laser-based sensing in low-cost applications. Since cameras inherently lack range-based information, various set-ups such as stereo-vision and moving camera setups were initially required to obtain depth information from the environment. However, bearing-only system formulations were eventually shown to be tractable for several tasks since its inception. This section explains the context that this thesis is placed in and discusses the stimulus for this research.

1.1.1 An Overview of Bearing-Only Localization

The bearing-only localization problem has been recognized as a non-trivial problem in literature since its introduction by Stansfield [2] in 1947. Bearing-only target tracking using Kalman filtering has been extremely popular in aerospace, defence and marine applications, some early notable works belonging to Murphy [3] and Aidala [4]. From an image-based visual servoing perspective, bearing-only methods have been incorporated into navigation tasks such as homing. Works such as Loizou & Kumar [5] and Liu, Pradalier, Chen & Siegwart [6] make use of real-time camera feed to generate closed-loop control commands based on *a-priori* information about the environment. However, in such methods, measurement errors are not accounted for. On the other hand, robot localization techniques that utilize recursive estimation accommodate the probabilistic nature of sensor data and the velocity model. Cameras are sometimes used in tandem with other sensors due to their lack of direct depth information and, until the recent advances in techniques, the inability to extract long-term features to map. Chenavier & Crowley [7] fused bearing measurements with odometry readings, while Zingaretti & Frontoni [8] combined measurements from vision-based sensors and range-based scanners. However in contemporary literature bearing-only methods in navigation such as Krajník et al. [9] are quite popular as well.

The major leap in vision-based localization has been the establishment of correlation between the robot pose and the map features by Smith, Self & Cheeseman. [10]. The

'single state approach' towards simultaneous localization and mapping took time to gain popularity but eventually the Extended Kalman Filter was adopted as the main estimation technique in SLAM implementations. Several successful implementations such as Gutmann & Konolige [11] made use of laser range-finder sensors. Early vision-based SLAM systems attempted to tackle the scale invariance problem of cameras using different approaches. Jung & Lacroix [12] used stereo-vision and image pairs to localize and estimate map features, while Kim & Sukkarieh [13] used an IMU with the camera to perform SLAM for an unmanned aerial vehicle. Eventually, systems consisting of solely a monocular camera could perform SLAM while being able to perform map management, correct at loop closures and maintain a set of high quality features. The paper by Davison, Reid, Molton & Stasse [14] has been the most prominent attempt in this regard. Although the field has moved on to keyframe-based methods in SLAM [15–17], EKF-SLAM is still widely used today. The discussion of background of SLAM is a pertinent one since this work deals with probabilistic map features as well.

1.1.2 Relevance of Nonlinear Observability

An important aspect of localization, especially in the context of recursive estimation, is the notion of nonlinear observability. Stochastic observers such as the Extended Kalman Filter guarantee exponential convergence of the state estimate if the system is observable in nature, as demonstrated by Reif, Gunther, Yaz & Unbehauen [18]. Ham & Brown [19] demonstrate the link between system observability and the eigenvalues of the covariance matrix. For bearing-only formulations, observability of the system depends on the number of landmarks that the system measures from, usually requiring a lower bound on the number to reach a maximal dimension of the observable subspace corresponding to the system. Failure to obtain measurements from that many landmarks leads to growth in covariance values of the filter. The estimation error cannot be guaranteed to converge in that scenario.

The observability of various localization problems has been analysed in multiple contexts. Martinelli, Pont & Siegwart [20] discussed the observability of a multi-robot localization problem with access to bearing-only measurements. Martinelli, Tomatis & Siegwart [21] discussed observability properties of a simultaneous localization and calibration problem. However, significant interest has been shown in the nonlinear observability properties of SLAM systems. Huang, Mourikis & Roumeliotis [22] explained the

inconsistency in the nonlinear observability of the SLAM problem and the observability properties of the linearized EKF-SLAM. Perera, Melkumyan & Nettleton [23] discussed the nonlinear approach to observability of the SLAM problem. The unobservable space for the SLAM problem is well-documented.

There exists observability analysis in literature specifically for bearing-only systems as well. Vidal-Calleja et.al [24] discuss the observability of the bearing-only SLAM problem. However, the most comprehensive observability analysis for vision-based systems has been provided by Belo, Salaris, Fontanelli & Bicchi [25].

1.1.3 Robust State Estimation Techniques

A significant portion of this thesis deals with robust localization in presence of parameter inaccuracies. Several novel formulations exist in state estimation literature tackling this problem. Bavdekar, Gopaluni & Shah [26] discuss and compare various adaptive filters that operate in presence of parameter inaccuracies, while Varshney, Bhushan & Patwardhan [27] presents an extended version of the Kitanidis Kalman Filter, originally proposed by Kitanidis [28] as a disturbance rejector. In robotic applications, works such as Roumeliotis, Sukhatme & Bekey [29] circumvent the problem of complex dynamic modelling by using an error state EKF.

This work specifically tackles localization in environments where the landmark positions may not be accurately known. Such a situation would arise in a environment with semi-static[2] landmarks, or for a static environment with outdated knowledge of its surroundings. Standard techniques to tackle any such set of features include EKF-SLAM, however the size of the state vector becomes prohibitively large for a dense set of features. Rosen, Mason & Leonard [30] and Meyer-Delius, Hess, Grisetti & Burgard [31] both propose solutions to map building specifically when semi-static landmarks are present in the environment. However this work emphasises on robot localization and proposes a robust method to ensure better observer performance without creating a new map.

[2]landmarks whose location changes intermittently

1.2 System Modelling

This section introduces the bearing-only system and the localization problem associated with it. The chapter also discusses the augmented state space formulation for the same problem, introduced by Gupta, Arunkumar & Vachhani [32], and the associated state transition and measurement maps. The key differences between the two formulations are noted.

1.2.1 Traditional Bearing-Only State Space Model

Consider a non-holonomic unicycle robot moving on a 2-D plane as shown in the figure 1.1. The system is formulated using polar coordinates. R and θ specify the polar coordinates of the robot in 2-D space, while α denotes the orientation. These three coordinates are sufficient to describe the pose of a robot for two-dimensional motion. The unicycle is equipped with monocular vision that provides it with bearing measurements from position C. These are denoted by β_i, $i \in \{1, 2, \ldots, p\}$. Hence, the state of the robot is characterized as:

$$\mathbf{x} = \begin{bmatrix} R & \theta & \alpha \end{bmatrix}^{\mathsf{T}} \tag{1.1}$$

The system is formulated with knowledge of the environment, which exists in the form of a panoramic image captured from a reference position. This is akin to the concept of the 'home' position that exists for homing applications [33, 34]. Here, O denotes the reference position, which, without loss of generality, is considered as the origin of the X-Y reference frame. The landmarks detected in the panoramic image are indexed as L_i, $i \in \{1, 2, \ldots, p\}$ and based on these images, the relative bearing of all these landmarks w.r.t the positive X-axis from O, are calculated. These are stored *a-priori* to the simulation and are denoted by β_1^*, β_2^*, $\ldots \beta_p^*$ respectively.

The distance of the i^{th} landmark from the target position D_i^* cannot be retrieved solely from the panoramic images from the home position, since the images lack information about the scale of the landmarks that are visible. There are two ways to retrieve D_i^* for each landmark:

1. Further information regarding scale of the environment prior to performing the task can be used to find the scale of the environment. This can be used to compute the distance of each landmark from the home position. Depending on the information, just one or all D_i^* can be obtained.

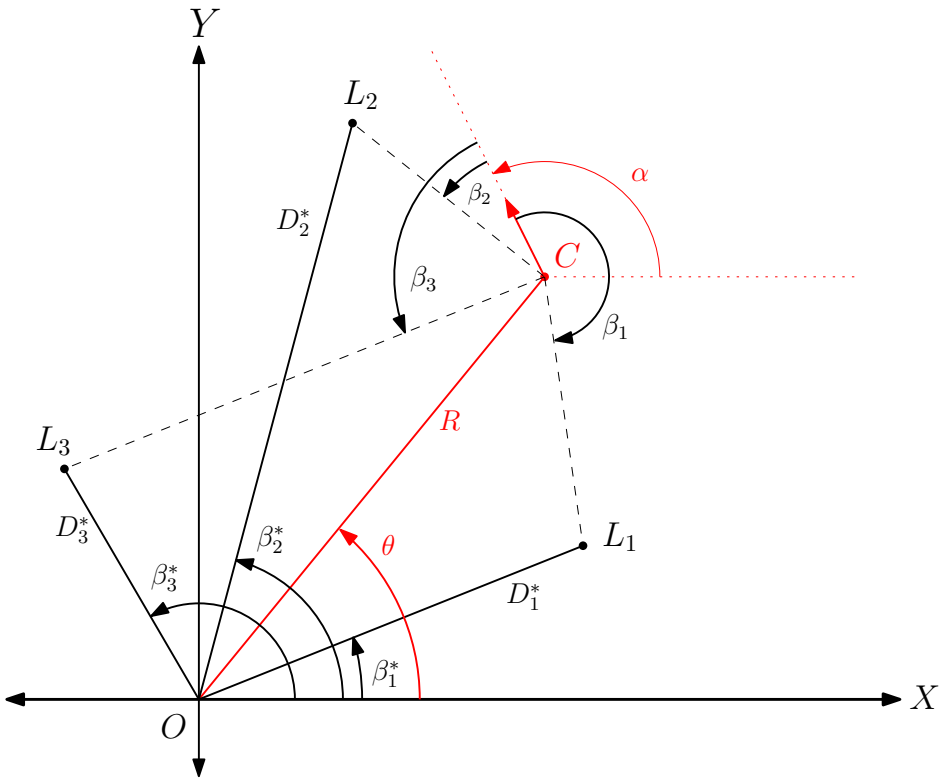


Figure 1.1: Representation of the 2-D Bearing-Only Localization Problem

2. For a robot possessing measurements from a particular landmark while it is located at any point apart from the reference position, a relation can be obtained between the obtained measurements, the robot and landmark pose. Consider $\triangle OL_1C$, $\triangle OL_2C$ and $\triangle OL_3C$ in Figure 1.1. Applying the sine law to each of them, we obtain the following relation [32]:

$$\begin{aligned} \frac{R}{\sin(\beta_i^* - (\beta_i + \alpha))} &= \frac{D_i^*}{\sin(\pi - (\theta - (\beta_i + \alpha)))} \\ D_i^* &= R\frac{\sin(\theta - (\beta_i + \alpha))}{\sin(\beta_i^* - (\beta_i + \alpha))} \end{aligned} \tag{1.2}$$

Hence, if a particular landmark from the panoramic image is found in the live feed from the monocular camera, the distance of that landmark from O can be obtained. In a real-world setting, since only estimates of the robot pose are available and the measurements can also be assumed to be noisy in nature, the D_i^* that is computed as a result has a certain probability distribution associated with it.

The general nonlinear continuous time state space model is represented as:

$$\begin{aligned} \dot{\mathbf{x}} &= f(\mathbf{x}, u) && \mathbf{x} \in M \subset \mathbb{R}^n, u \in \mathbb{R}^m \\ \mathbf{y} &= h(\mathbf{x}) && \mathbf{y} \in \mathbb{R}^p \end{aligned} \tag{1.3}$$

For the given system, $g(\mathbf{x}, u)$ and $h(\mathbf{x})$ assume the following structure:

$$f(\mathbf{x}, u) = \begin{pmatrix} \cos(\alpha - \theta) & 0 \\ \dfrac{\sin(\alpha - \theta)}{R} & 0 \\ 0 & 1 \end{pmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix} \tag{1.4}$$

$$h(\mathbf{x}) = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix} = \begin{pmatrix} \text{atan2}(y_1 - y_R, x_1 - x_R) - \alpha \\ \text{atan2}(y_2 - y_R, x_2 - x_R) - \alpha \\ \vdots \\ \text{atan2}(y_p - y_R, x_p - x_R) - \alpha \end{pmatrix} \tag{1.5}$$

where, for $i \in \{1, 2, \ldots, p\}$, the above x_R and y_R terms represent the robot position in Cartesian coordinates, while x_i and y_i represent the Cartesian coordinates of the i^{th} landmark. These can be related to the previously introduced variables in the following manner:

$$\begin{aligned} y_i &= D_i^* \sin \beta_i^* \\ x_i &= D_i^* \cos \beta_i^* \\ y_R &= R \sin \theta \\ x_R &= R \cos \theta \end{aligned} \tag{1.6}$$

Modification for Multi-rate Measurements

The measurement model given in Equation 1.5 associates a measurement with each landmark in the system output y. However, a measurement cannot be guaranteed for each landmark in the environment, for two main reasons:

1. Most RGB cameras have a limited field of view, hence only a limited set of features are available to the camera for measurement, depending on the orientation of the camera at a particular instant. We assume that the robot in consideration has the camera frame fixed to always face in the direction of forward motion.

2. For the set of features visible to the camera, not all may be correctly identified as the corresponding landmark that is detected in the panoramic image. Though various data association techniques exist that match these features with those extracted from the live feed image, none of them guarantee complete accuracy. Hence even the few measurements obtained at a given time instant have to be cross-verified using one of various techniques that exist in literature, due to the possibility of incorrect matching.

Hence, we also formulate a multi-rate version of the traditional bearing-only system. Let us assume that out of a total of p landmarks, only q landmarks are correctly identified at a particular instant. A hypothesis $\{k_1, \dots k_q\}$ is created based on the association, which links the q measurements to the p landmarks.

Hence, the measurement model for these visible landmarks, at any given time instant, can be written as follows:

$$h(\mathbf{x}) = \begin{bmatrix} \beta_{k_1} \\ \beta_{k_2} \\ \vdots \\ \beta_{k_q} \end{bmatrix} = \begin{pmatrix} \text{atan2}(y_{k_1} - y_R, x_{k_1} - x_R) - \alpha \\ \text{atan2}(y_{k_2} - y_R, x_{k_2} - x_R) - \alpha \\ \vdots \\ \text{atan2}(y_{k_q} - y_R, x_{k_q} - x_R) - \alpha \end{pmatrix} \tag{1.7}$$

where the Cartesian coordinates can be substituted to yield a function only consisting of the previously introduced variables, by using Equations 1.6.

1.2.2 Augmented State Space Model

We also discuss the augmented state space formulation presented by Misha et al. [32] that describes and solves the bearing-only localization problem using a different state space model.

Consider the same problem described by Figure 1.1. The augmented state space formulation appends the measurements obtained by every landmark to the state vector. This is done by incorporating the dynamics of the measurements into the state transition map. The resultant state vector is as follows:

$$\mathbf{x} = \begin{bmatrix} R & \theta & \phi & \beta_1 & \dots & \beta_p \end{bmatrix}^\mathsf{T} \tag{1.8}$$

where $\beta_i, i \in \{1, 2, \dots p\}$ are all the measured bearings as shown in Figure 1.1. Thus,

based on the general nonlinear state space model given in Equation 1.3, the functions $f(\mathbf{x}, u)$ and $h(\mathbf{x})$ take the following structure:

$$f(\mathbf{x}, u) = \begin{pmatrix} \cos(\alpha - \theta) & 0 \\ \frac{\sin(\alpha - \theta)}{R} & 0 \\ 0 & 1 \\ \frac{-\sin(\beta_1)}{RC_1 - D_1^* C_1^*} & -1 \\ \frac{-\sin(\beta_2)}{RC_2 - D_2^* C_2^*} & -1 \\ \vdots & \vdots \\ \frac{-\sin(\beta_p)}{RC_p - D_p^* C_p^*} & -1 \end{pmatrix} \begin{bmatrix} V \\ \omega \end{bmatrix} \tag{1.9}$$

$$= \begin{bmatrix} g_1 & g_2 \end{bmatrix} \; u \tag{1.10}$$

where, for brevity,

$$\begin{aligned} C_i &:= \cos(\theta - (\beta_i + \alpha)) \\ C_i^* &:= \cos(\theta - (\beta_i + \alpha)) \\ S_i &:= \sin(\beta_i^* - (\beta_i + \alpha)) \\ S_i^* &:= \sin(\beta_i^* - (\beta_i + \alpha)) \end{aligned} \tag{1.11}$$

The measurement equation, to measure all the landmarks is given by:

$$\mathbf{y} = h(\mathbf{x}) = \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix} = C\mathbf{x} \tag{1.12}$$

$$C = [\overline{0}_{p\times 3} \;\; I_p] \tag{1.13}$$

Modification for Multi-rate Measurements

For the multi-rate scenario, note that the state vector does not change in its form. All the landmarks registered from the panoramic image are retained as states even if they are not measured for one or many time instants. We do not consider dynamic map management

or the addition or deletion of any features as part of this formulation. The measurement model can be represented as:

$$h(\mathbf{x}) = \begin{bmatrix} \beta_{k_1} \\ \beta_{k_2} \\ \vdots \\ \beta_{k_q} \end{bmatrix} = C\mathbf{x} \tag{1.14}$$

$$C = [\bar{0}_{q\times 3} \quad K_{q\times p}] \tag{1.15}$$

The elements of K consist of 1's corresponding to the measurement associated with the matched landmark, while the rest of the elements are 0's. Hence,

$$K_{ij} = \delta_{k_i j} \tag{1.16}$$

where $\delta_{k_i j}$ represents the Kronecker delta for the tuple of indexes.

$$\delta_{k_i j} = \begin{cases} 1, & \text{if } k_i = j \\ 0, & \text{otherwise} \end{cases} \tag{1.17}$$

1.2.3 Summary

The augmented state space model differs from the traditional model solely in the way that it is formulated in, since both the models can be used to describe the same localization problem. The difference lies in where the landmark coordinates appear. For the traditional state space model, the terms D_i^* and β_i^* appear solely in the measurement map, hence the state evolution over time does not depend on these coordinates.

However, for the augmented state space, D_i^* and β_i^* are incorporated in the evolution of the measurements, since the measurements β_i's are included in the state vector. Hence the landmark coordinates are solely included in the state transition map. This has significant advantages as is discussed later in the thesis. Also, as a consequence, the measurement model is now linear in nature for each landmark that is being observed at a given time instant.

1.3 Contribution

This thesis addresses two areas in bearing-only localization: the problem of observability of joint state and parameter estimation problems for various parameter choices, and the problem of robot pose localization in presence of landmark position inaccuracies.

With regards to the first area:

- This work determines lower bounds for the number of landmarks, for a bearing-only system formulation to achieve maximal observability rank, for various choices of landmark coordinates to be estimated as parameter. It also interprets the kernel basis vectors computed for each formulation as loci of initial states in 2-dimensional space, thus demonstrating the origin of the degeneracy in the system. The analysis is in polar coordinates, which is more appropriate for expressing both robot and landmark positions, for this work. It differs from Belo et al. [25] as it specifically computes the maximal observability rank and kernel basis vectors for each choice of coordinate of the landmark position, thus providing a more intuitive understanding of the system geometry for all types of formulations.

- For the observability results obtained, the thesis presents transformations that decompose the system into observable and unobservable spaces. This enables estimator convergence for maximum number of states for that localization problem. The nature of the transformation also validates the understanding of the unobservable spaces. These transformations are ubiquitous to choice of parameter.

With regards to the second area:

- The merits of a previously-introduced augmented state space formulation over the traditional observer for bearing-only localization are demonstrated, specifically for environments where landmark positions are uncertain. It is shown that better estimates of robot pose are attainable using modified EKFs, without having to continually estimate the landmark positions. The Q-EKF, paired with the augmented state space formulation, performs better than the EKF-SLAM for low values of initial estimate error and is computationaly less consuming due to a smaller state vector than the EKF-SLAM formulation. It is shown to be effective in simulation with a robot model with a vision pipeline, in a Gazebo environment, when it is paired with an outlier rejector.

- The PI-Q-EKF is proposed to nullify errors in initial state estimate when paired with the augmented system. Its superiority in performance to the EKF-SLAM and Q-EKF is demonstrated in simulation. A search space for the proportional and integral gains is specified for this system, within which tuning of parameters would yield suitable gain matrices.

1.4 Thesis Overview

The contents of this thesis are distributed among the succeeding chapters as follows:

- Chapter 2 introduces the concept of nonlinear observability in a differential geometric context and construct algorithms to compute the observability codistribution of nonlinear system, for piecewise constant inputs and for analytic inputs. The unobservable space for each state and parameter estimation problem is computed using the symbolic computation software Mathematica. The basis vectors of this space are interpreted in 2-D space. The validity of these results are checked for potential anomalies.

- Chapter 3 validates the results obtained in the previous chapter by introducing coordinate transformations based on the intuition gathered from the observability results. These transformations decompose the system into observable and unobservable spaces. Simulation results for the transformed systems are provided.

- Chapter 4 discusses the problem of robot pose estimation in presence of inaccurate knowledge of landmark positions. Two modifications to the EKF, the Q-EKF and the PI-Q-EKF, are proposed to mitigate estimator divergence due to the inaccuracy and to eliminate any initial estimation error simultaneously. The efficacy of these modified filters paired with the augmented state space formulation is compared with EKF-SLAM for static landmarks.

- Chapter 5 presents a robot model and a planar environment in Gazebo. The camera model and computer vision algorithm is discussed and simulations are carried out for accurate as well as inaccurate knowledge of the landmarks. Comparisons are drawn between the two state space formulations and the appropriate filters used for each one.

- Chapter 6 provides concluding remarks for this thesis and discusses the possible future of this work.

Chapter 2

Nonlinear Observability Analysis

This chapter presents the theory of nonlinear observability in a differential geometric context and discusses various results presented on the basis of this theory. The contents of the chapter are mainly derived from the works of Hermann and Krener [35], Isidori [36], Sontag & Wang [37], Sussmann [38] and Anguelova [39].

2.1 Introduction to Nonlinear Observability

Consider a driftless nonlinear system affine in the control inputs, henceforth referred to as Σ. Such a system takes the following form:

$$\begin{aligned} \dot{x} &= g(x)u \qquad & x \in M \subseteq \mathbb{R}^n, u \in \mathbb{R}^m \\ y &= h(x) \qquad & y \in \mathbb{R}^p \end{aligned} \tag{2.1}$$

The control-affine part $g(x)u$ can be expressed as the summation of its individual column vectors weighted by each control input:

$$g(x)u = \sum_{i=1}^{m} g_i(x)u_i \qquad i \in \{1, 2, \ldots m\} \tag{2.2}$$

The measurement model is also expressed as the composition of various scalar-valued

functions $h_j : \mathbb{R}^n \to \mathbb{R}$ as components of the vector-valued function:

$$h(x) = \begin{bmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_p(x) \end{bmatrix} \tag{2.3}$$

We discuss the concept of distinguishability introduced by Hermann & Krener [35]. The idea arises from the need to tell apart two points in the state space, x_0 and x_1, based between their input-output realizations for a bounded control input $u(t)$. Restricting the concept to trajectories that lie completely within a subset $U \subset M$, the formal definition of distinguishability is given as follows:

Definition 2.1.1. *Let the flow of the system Σ for a known bounded input $u(t)$ be defined as $\Phi(x,t)$. A pair of points x_0 and x_1 in M are called **U-indistinguishable** if for every bounded input defined on the interval $[0,T]$ for which $\Phi(x_0,t) \in U$ and $\Phi(x_1,t) \in U \ \ \forall\, t \in [0,T]$, the condition $h_i(\Phi(x_0,t)) \neq h_i(\Phi(x_1,t)$ is satisfied $\ \forall\, t \in [0,T], \quad i \in \{1,2,\dots p\}$.*

The notation $I(x_0,U)$ is introduced to denote the set of points that are not *U-indistinguishable* from x_0. These set of points cannot be differentiated from one another based on the system output observed during the trajectory, for the same control input. We link the observability of the system to the indistinguishability property in the following manner:

Definition 2.1.2. *The system Σ is said to be **observable** at $x_0 \in M$ if $I(x_0,U) = \{x_0\}$.*

We define the system Σ to be observable on M if it is observable $\forall\ x_0 \in M$. Two important variations to this definition are introduced:

- We want to preclude a pair of points from becoming indistinguishable after an arbitrary amount of time. Hence, we restrict U to every open neighbourhood of x_0 and call the system Σ as *locally observable* if $I(x_0,U) = \{x_0\}$ for trajectories restricted within every open neighbourhood of x_0.

- In practice, we find it sufficient to distinguish x_0 from solely its neighbours. Thus, we define the system Σ to be *weakly observable* if there exists an open neighbourhood of x_0, V, $\ I(x_0) \cap V = \{x_0\}$. This is a weaker definition of observability as a system observable at x_0 over an open subset is weakly observable at x_0 as well.

Hence, we formalize the definition of local weak observability below.

Definition 2.1.3. *The system Σ can be considered* ***locally weakly observable*** *at $x_0 \in M$ if there exists an open neighbourhood V such that for every open neighborhood U that is contained in V, $I(x_0, U) = x_0$.*

The significance of local weak observability over the other definitions is the ability to evaluate it using an algebraic test. The test is described in depth in the next section.

2.2 Algebraic Test for Local Weak Observability

This section discusses the algebraic tools and the algorithm used to determine if a system Σ is locally weakly observable or not. This algebraic analysis differs slightly based on the type of control input provided to the system. However, we begin by defining the Lie Derivative of a real-valued function $\lambda(x)$, while x flows along the vector field $v(x)$:

$$\begin{aligned} L_v\lambda(x) &= \lim_{x\to 0} \frac{\lambda(\Phi_v(x,t)) - \lambda(\Phi_v(x,0))}{t} \\ &= \frac{d}{dt}\bigg|_{t=0} \lambda(\Phi_v(x,t)) \end{aligned} \tag{2.4}$$

where $\Phi_v(x,t)$ provides the flow of the vector field v at a time instant t, for a starting point x.

The local weak observability tests are described below for the following categories of control inputs:

2.2.1 Piecewise Constant Inputs

Most autonomous systems that operate in state feedback control in real time receive piecewise constant control inputs, due to the zero order hold nature of discrete-time controllers. The input remains constant for the duration of the sampling time of the controller. Hence, this analysis practically extends to all systems operating in real time.

The observable codistribution of a system is defined as follows:

Definition 2.2.1. *A nonsingular codistribution Ω is termed as the* ***observable codistribution*** *of Σ at $x_0 \in M$ if it is the smallest codistribution that spans the covector field*

spanned by $\{dh_1, dh_2, \ldots dh_p\}$ *and is invariant (to Lie differentiation) under the vector fields* $\{g_1, g_2 \ldots g_m\}$.

Isidori et al. [36] shows that the existence of an involutive distribution of dimension d that is orthogonal to the codistribution defined in Definition 2.2.1, implies the existence of a coordinate transformation for the system that allows the transformed region to be partitioned into 'slices' of dimension d such that all points on the same slice produce the same output.

The space $\mathcal{G}$ at x_0 is defined as the space containing $\{h_1, \ldots h_p\}$ and closed under Lie Differentiation by $\{g_1, \ldots g_m\}$. The elements of $\mathcal{G}$ are finite linear combinations of the form:

$$L_{g_{j_1}}(\ldots(L_{g_{j_k}}(h_i))) \tag{2.5}$$

where $g_{j_1}, \ldots g_{j_k}$ belong to span$\{g_1, \ldots g_m\}$. The gradient operator d commutes with Lie differentiation operator L_f, enabling the calculation of $d\mathcal{G}$ at x_0 by taking the gradient of each element of $\mathcal{G}$. The elements of $d\mathcal{G}$ are finite linear combinations of the form:

$$d(L_{g_{j_1}}(\ldots(L_{g_{j_k}}(h_i)))) = L_{g_{j_1}}(\ldots(L_{g_{j_k}}(dh_i))) \tag{2.6}$$

These elements from Equation 2.6 span $d\mathcal{G}$. $d\mathcal{G}$ matches the definition of the space described by Definition 2.2.1 and hence we term $d\mathcal{G}$ as the observable codistribution[1] of the system Σ.

Since the system Σ is autonomous and has piecewise constant control inputs, the Lie Derivative calculation reduces to a dot product:

$$L_{g_i} h_j = dh_j(x) \cdot g_i \tag{2.7}$$

To compute the observable codistribution at a point x_0, [36] proposed a recursive algorithm that constructs the codistribution. The following set of equations are used iteratively:

$$\Omega_0 = \text{span}\{d_x h_1, \ldots, d_x h_p\} \tag{2.8}$$

$$\Omega_k = \Omega_{k-1} + \sum_{i=1}^{m} L_{g_i} \Omega_{k-1} \tag{2.9}$$

The terminal condition specified by the algorithm is if there exists a k such that

[1] used interchangeably with the term 'observable subspace'

$\Omega_k = \Omega_{k+1}$, for which $k^* := k$. In that case, the algorithm is said to have arrived upon $d\mathcal{G}$. The system is said to be locally weakly observable if $dim(d\mathcal{G}) = p$.

This work sticks to Lie derivative calculation of exact differentials instead of working with covector fields as the above algorithm does. We find the space $\mathcal{G}$ by using iterative Lie differentiation of $\{h_1, \dots h_p\}$, which are scalar-valued functions, to arrive to a form such as (2.6). The next step is the calculation of the gradient of each element. Together, these gradients span $d\mathcal{G}(x)$ which is the cotangent space at x. These gradients, also referred to as covectors, make up the rows of the ***observability matrix***. The ***observability rank condition*** alludes to the rank of this observability matrix being equal to p for the system to be locally weakly observable at x.

Since the algebraic computation is done on Mathematica, we provide the pseudo-code:

Algorithm 1: Observability Rank Computation for Piecewise Constant Inputs

currentLieDerivatives $\leftarrow \{\}$, prevLieDerivatives $\leftarrow \{h_1, \dots h_p\}$;
currentObsRank, prevObsRank $\leftarrow 0$;
obsMatrix $\leftarrow$ **Jacobian**$(h(x), x)$;
obsRankCondn $\leftarrow 0$;
while currentObsRank $< p$ **do**
    currentLieDerivatives $\leftarrow \{\}$;
    foreach $\lambda \in$ prevLieDerivatives **do**
        foreach $i \in \{1, \dots m\}$ **do**
            curentLieDerivatives.append$(L_{g_i}\lambda)$;
            obsMatrix $\leftarrow$ **vertCat**(obsMatrix, **Jacobian**$(L_{g_i}\lambda, x)$);
        end
    end
    prevLieDerivatives $\leftarrow$ currentLieDerivatives;
    prevObsRank $\leftarrow$ currentObsRank;
    currentObsRank $\leftarrow$ **Rank**(obsMatrix);
    if prevObsRank == currentObsRank **then return**;
end
obsRankCondn $\leftarrow 1$;
return;

It is essential to note that both methods (the algorithm used above and the one

that uses Equations 2.8) essentially compute the dimension of the same space, hence we use the same terminal condition used for Equation 2.8 by manually checking the dimension of $d\mathcal{G}(x)$ after each step of Lie differentiation. The algorithm is terminated if the observability matrix rank remains the same for consecutive steps, since this is equivalent to adding covectors that already belong to the codistribution.

2.2.2 Analytic Inputs

Here we discuss the generic nonlinear system (of the form of Equation 1.3) with control inputs that are continuous functions of time, hence reverting to that notation. In fact, for this section, we assume each input to belong to C^∞.

The expression of the Lie derivative for a system with analytic input slightly differs from the one in Equation 2.7. Let the initial values of u and its derivatives be $u^{(r)}(0) = U(r)$ for $r \geq 0$ with $U(r) \in \mathbb{R}$. We start from 2.4 just as before, however it reduces to the following expression [39]:

$$L_f\, h_j = dh_j(x) \cdot f + \sum_l U^{(l+1)} \frac{\partial h_j}{\partial u} \qquad \text{for some } l \geq 0 \tag{2.10}$$

Since the inputs are continuous functions of time, this expression simplifies to:

$$L_f\, h_j = dh_j(x) \cdot f + \left.\frac{\partial h}{\partial t}\right|_{t=0} \tag{2.11}$$

Using this definition, the set of Lie derivatives obtained can be expressed as a polynomial in $U^0, U^1, \ldots$ whose coefficients are functions of x. Since the $U(r)$'s are free to vary over $\mathbb{R}$, their coefficients (which are covectors) can be interpreted as spanning a space. Sontag & Wang [37] uses this to present a different definition of the observation space, which is then proven equivalent to the cotangent space $d\mathcal{G}$ in the previous section. However, this work sticks to the previous method of computing the codistribution recursively.

Wang and Sontag [40] show the equality of the observation space for systems with piecewise constant control with those equipped with analytic control inputs. Hence we use Definition 2.2.1 for this type of system and control input as well. By iterated Lie differentation to obtain elements of the form given in (2.6), we obtain the observable codistribution $d\mathcal{G}$ at x. The only difference is the presence of the terms $U^0, U^1, \ldots$

etc in the vectors spanning $d\mathcal{G}$. A pseudo-code similar to Algorithm 1 to calculate the observability rank is given below:

Algorithm 2: Observability Rank Computation for Analytic Inputs

currentLieDerivatives $\leftarrow \{\}$, prevLieDerivatives $\leftarrow \{h_1, \dots h_p\}$;
currentObsRank, prevObsRank $\leftarrow 0$;
obsMatrix $\leftarrow$ **Jacobian**$(h(x), x)$;
obsRankCondn $\leftarrow 0$;
while currentObsRank $< p$ **do**
    currentLieDerivatives $\leftarrow \{\}$;
    foreach $\lambda \in$ prevLieDerivatives **do**
        curentLieDerivatives.append$(L_f\lambda)$;
        obsMatrix $\leftarrow$ **vertCat**(obsMatrix, **Jacobian**$(L_f\lambda, x)$);
    end
    prevLieDerivatives $\leftarrow$ currentLieDerivatives;
    prevObsRank $\leftarrow$ currentObsRank;
    currentObsRank $\leftarrow$ **Rank**(obsMatrix);
    if prevObsRank == currentObsRank **then return**;
end
obsRankCondn $\leftarrow 1$;
return;

2.3 Observability Properties for Piecewise Constant Inputs

Sufficient literature has covered the observability properties for bearing-only systems, for varying number of landmarks. It has been established by Belo et al. [41] and by Martinelli [42] that for known landmark positions, 2 landmarks are sufficient for a system to be locally weakly observable. However, most of the above work has been done for the localization problem formulated in the Cartesian coordinates. The problem presented in this thesis presupposes knowledge of the environment in the form of a panoramic image from the reference position. Hence, for any localization problem, the system has *a-priori* knowledge of β_i^*'s as shown in Figure 1.1. For such applications, the localization and subsequent closed-loop control strategy to return to the reference position can be better represented in polar coordinates. Hence polar coordinates are the choice of coordinate system for this work.

This section also introduces observability analysis for joint state and parameter formulations. From a landmark position estimation perspective, for this system, polar coordinates are the more logical option to describe the position of landmarks as well. In fact, some results that are described later in this section are obtained faster from the symbolic computation software than if Cartesian coordinates are used to describe the location of the landmarks. However, we do not claim that the results obtained for polar coordinates are different for the same system formulated in Cartesian coordinates; in fact, the results are expected to be identical irrespective of the choice of coordinate system used for the robot.

Observability properties for joint state and parameter formulations have been analyzed by Belo et al. [25], however no such analysis in literature extends to polar formulations. This work also provides geometric interpretations to the basis vectors that span the space orthogonal to the observable subspace. The emphasis remains on finding the lower bound on the number of landmarks required for the system to reach the minimum possible dimension of the aforementioned space, which is also referred to as the *unobservable space*.

All the results in this section are for piecewise constant inputs. The modifications made to the traditional bearing-only system formulation, to accommodate parameter estimation, can be grouped in the following categories:

2.3.1 Robot Pose Localization

The state vector is identical to that in Equation 1.1. It consists of the two-dimensional robot pose, with polar coordinates being used to describe the robot position. Hence it takes the following form:

$$\mathbf{x} = \begin{bmatrix} R & \theta & \alpha \end{bmatrix}^{\mathsf{T}} \tag{2.12}$$

The system has the same state transition map and the measurement map given in Equations 1.4 and 1.5 respectively.

1 landmark

The system is not observable for $p = 1$. The rank of the observability matrix is computed to be 2. Hence the unobservable space is spanned by only one vector:

$$\begin{aligned} d\mathcal{G}^{\perp} &= \operatorname*{span}_{\text{col.}} \begin{bmatrix} \sin(\beta_1^* - \theta) \\ 1 - \dfrac{D_1^* \cos(\beta_1^* - \theta)}{R} \\ 1 \end{bmatrix} \\ &\triangleq \operatorname{span} \begin{bmatrix} \mathbf{l}_1 \end{bmatrix} \end{aligned} \tag{2.13}$$

For given state $\mathbf{x_0}$, these vectors span the indistinguishable subspace around it, i.e. the set of points $\mathbf{x_0} + \epsilon_1 \mathbf{l_1}$ realize the same input-output map, for a sufficiently small value of ϵ_1. Hence, any linear combination of this basis vector can be expressed as a shift in the initial state $\mathbf{x_0}$.

$$\begin{bmatrix} \delta R \\ \delta\theta \\ \delta\alpha \end{bmatrix} = \epsilon_1 \mathbf{l_1} \tag{2.14}$$

To understand this space qualitatively, we use the following relations converting solely the components corresponding to shift in robot position, from polar to cartesian:

$$\begin{aligned} \delta R &= \delta x_R \cos\theta + \delta y_R \sin\theta \\ R\delta\theta &= -\delta x_R \sin\theta + \delta y_R \cos\theta \end{aligned} \tag{2.15}$$

The difference terms δx and δy are condensed after substituting for $\mathbf{l_1}$ from Equation 2.13, into 2.15. Expressing $\mathbf{l_1}$ in robot transformed coordinates, using Equation 1.6 yields

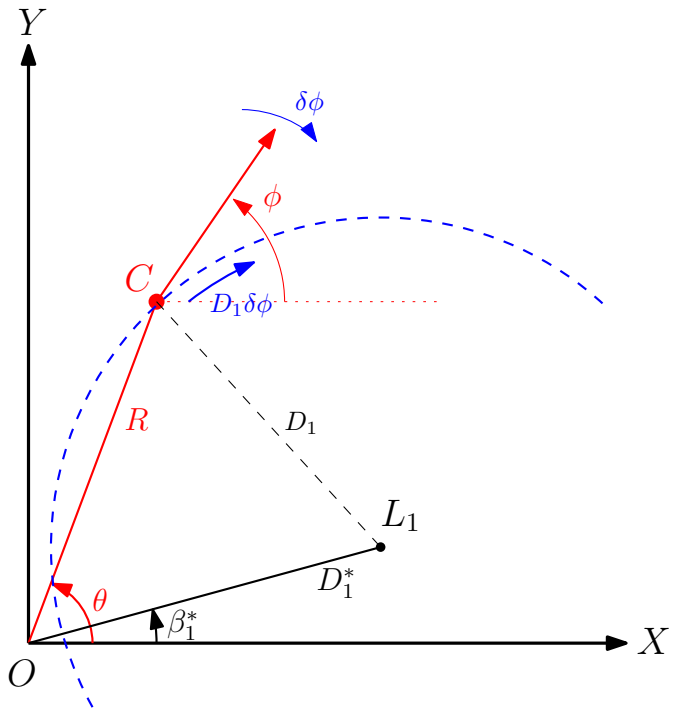


Figure 2.1: Nullspace for Robot Pose Localization ($p = 1$)

$\mathbf{l_{1_c}}$:

$$\mathbf{l_{1_c}} = [y_1 - y_R \quad -x_1 + x_R \quad 1]^\mathsf{T} \tag{2.16}$$

Now, consider the case of the robot undergoing a rotation around the landmark by a small angle α_s. Using the small angle approximation, the transformation can be represented by the following equation:

$$\begin{bmatrix} x_R^a - x_1 \\ y_R^a - y_1 \end{bmatrix} = \begin{bmatrix} 1 & -\alpha_s \\ \alpha_s & 1 \end{bmatrix} \begin{bmatrix} x_R - x_1 \\ y_R - y_1 \end{bmatrix} \tag{2.17}$$

which simplifies to:

$$\begin{bmatrix} x_R^a \\ y_R^a \end{bmatrix} = \begin{bmatrix} x_R \\ y_R \end{bmatrix} + \alpha_s \begin{bmatrix} -y_R + y_1 \\ x_R - x_1 \end{bmatrix} \tag{2.18}$$

Notice that $\mathbf{l_{1_c}}$ takes the same form as the vector in Equation 2.18, while additionally imposing the condition that the orientation α should rotate by the same angle. Hence, in the original state space, $\mathbf{x_0} + \epsilon_1 \mathbf{l_1}$ corresponds to the states achieved by slightly rotating

robot C around L_1, as demonstrated in Figure 2.1.

2.3.2 Joint Robot Pose and Landmark Distance Estimation

Given only one panoramic image from the home position (which is usually the available information for homing applications), the system does not possess landmark distances. Hence, the state vector can be appended with D_i^*'s, as shown in Figure 1.1 to have the following:

$$\mathbf{x} = \begin{bmatrix} R & \theta & \alpha & D_1^* & \dots & D_p^* \end{bmatrix}^\mathsf{T} \tag{2.19}$$

The system observability for varying number of landmarks is as follows:

1 Landmark

For $p = 1$, the system is not locally weakly observable. The observable codistribution has a rank two less than the state vector size. The nullspace has the following basis vector:

$$\begin{aligned} d\mathcal{G}^\perp &= \operatorname*{span}_{\text{col.}} \begin{bmatrix} \sin(\beta_1^* - \theta) & \cos(\beta_1^* - \theta) \\ 1 - \dfrac{D_1^* \cos(\beta_1^* - \theta)}{R} & \dfrac{\sin(\beta_1^* - \theta)}{R} \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \\ &\triangleq \operatorname{span} \begin{bmatrix} \mathbf{m}_1 & \mathbf{m}_2 \end{bmatrix} \end{aligned} \tag{2.20}$$

It is clear that the first three components of $\mathbf{m_1}$ is identical to $\mathbf{l_1}$ and does not point to a shift in D_1^*. Hence the geometrical interpretation should be the same.

The geometric interpretation for the origin of the second basis vector, $\mathbf{m_2}$, is analysed in the same manner, using Equations 2.15, 2.20 and 2.23. The transformation to $\mathbf{m_{2_c}}$ yields the following:

$$\mathbf{m_{2_c}} = \begin{bmatrix} \cos\beta_1^* & \sin\beta_1^* & 0 & 1 \end{bmatrix}^\mathsf{T} \tag{2.21}$$

It is evident from the structure of $\mathbf{m_{2_c}}$ that if the robot and landmark are shifted in parallel by equal distance, while keeping β_1^* constant, the new state is degenerate in terms of input-output behaviour. A pictorial representation of the same is provided in Figure 2.2

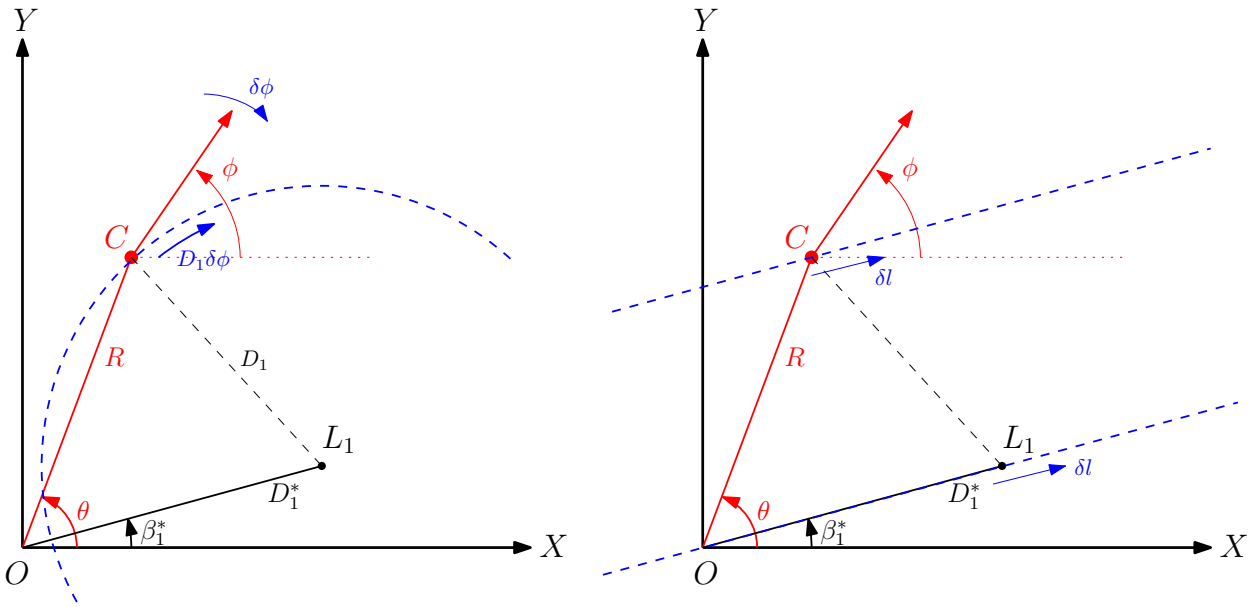


Figure 2.2: Nullspace for Joint Robot Pose and Landmark Distance Estimation ($p = 1$)

2 landmarks

For $p = 2$, the system is not locally weakly observable either. The observable codistribution has a rank one less than the state vector size. The nullspace has the following basis vector:

$$d\mathcal{G}^{\perp} = \underset{\text{col.}}{\text{span}} \begin{bmatrix} D_1^* \cos(\beta_2^* - \theta) - D_2^* \cos(\beta_1^* - \theta) \\ \sin(\beta_1^* - \beta_2^*) - \dfrac{D_1^* \sin(\beta_1^* - \theta) - D_2^* \sin(\beta_2^* - \theta)}{R} \\ \sin(\beta_1^* - \beta_2^*) \\ D_1^* \cos(\beta_1^* - \beta_2^*) - D_2^* \\ D_1^* - D_2^* \cos(\beta_1^* - \beta_2^*) \end{bmatrix} \tag{2.22}$$

$$\triangleq \text{span} \begin{bmatrix} \mathbf{n}_1 \end{bmatrix}$$

Hence, any linear combination of the basis vectors can be expressed as a shift in the states.

$$\begin{bmatrix} \delta R \\ \delta\theta \\ \delta\alpha \\ \delta D_1^* \\ \delta D_2^* \end{bmatrix} = \epsilon_3 \mathbf{n_1} \tag{2.23}$$

To understand what the space spanned by $\mathbf{n_1}$ implies, Equation 2.15 is used in the same manner to yield:

$$\begin{aligned} \delta x_R &= \epsilon_3[D_1^* \cos\beta_2^* - D_2^* \cos\beta_1^* - R\sin\theta\sin(\beta_1^* - \beta_2^*)] \\ \delta y_R &= \epsilon_3[D_1^* \sin\beta_2^* - D_2^* \sin\beta_1^* + R\cos\theta\sin(\beta_1^* - \beta_2^*)] \end{aligned} \tag{2.24}$$

for a small quantity ϵ_3. Since β_1^* and β_2^* are constants, we only discuss the radial shifts in landmarks L_1 and L_2.

Let us discuss the relative shift between the bot and a landmark in a particular direction. The relative shift in X direction is:

$$\begin{aligned} \delta x_R - \delta D_1^* \cos\beta_1^* &= \epsilon_3[D_1^* \cos\beta_2^* - D_2^* \cos\beta_1^* \\ -R\sin\theta\sin(\beta_1^* - \beta_2^*)] &- \epsilon_3[D_1^* \cos(\beta_1^* - \beta_2^*) - D_2^*]\cos\beta_1^* \\ &= \epsilon_3 \sin(\beta_1^* - \beta_2^*)(-y_R + y_1) \\ &= \epsilon_3^{'}(-y_R + y_1) \end{aligned} \tag{2.25}$$

Similarly, relative motion in Y direction:

$$\begin{aligned} \delta y_R - \delta D_1^* \sin\beta_1^* &= \epsilon_3[D_1^* \sin\beta_2^* - D_2^* \sin\beta_1^* \\ +R\cos\theta\sin(\beta_1^* - \beta_2^*)] &- \epsilon_3[D_1^* \cos(\beta_1^* - \beta_2^*) - D_2^*]\sin\beta_1^* \\ &= \epsilon_3 \sin(\beta_1^* - \beta_2^*)(x_R - x_1) \\ &= \epsilon_3^{'}(x_R - y_1) \end{aligned} \tag{2.26}$$

The same can be demonstrated for shifts relative to L_2 as well. Thus, a shift along the unobservable subspace translates to the robot and the landmarks shifting such that with respect to each landmark, the robot is rotated by a small angle around it.

The change in orientation has the same magnitude of rotation $\epsilon_3^{'}$, hence relative to both landmarks, the robot revolves around them simultaneously.

The pictorial representation of this is provided in Figure 2.3. Notice that the shift in

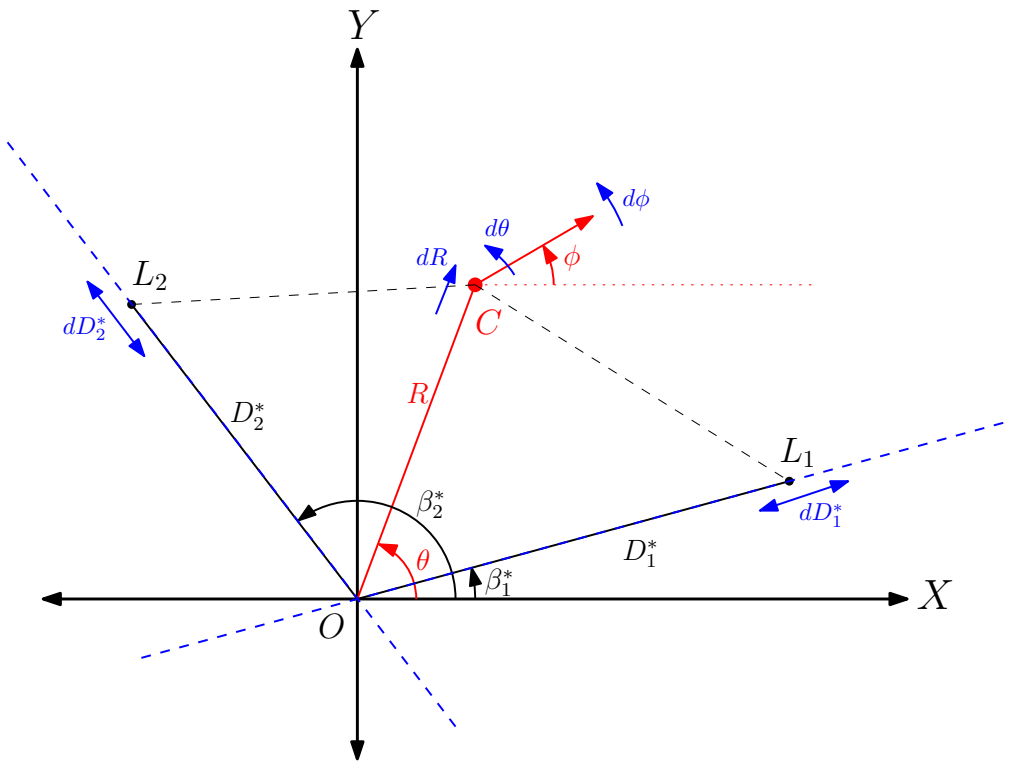


Figure 2.3: Nullspace for Joint Robot Pose and Landmark Distance Estimation ($p = 2$)

landmark distances may be in the same or opposite directions, depending on the values of β_1^* and β_2^*.

3 or more landmarks

The observable subspace has the rank equal to state vector size. Hence, for $p \geq 3$ the system is locally weakly observable, imparting the property of distinguishability of $\mathbf{x_0}$ from its neighbours. The nullspace of the subspace spans $\{\Phi\}$ as a result.

2.3.3 Joint Robot Pose and Landmark Bearing Estimation

For situations in which landmark distances can be assumed as known but the bearings may be unknown, β_i^*'s, $i \in \{1, 2, \ldots p\}$ as shown in Figure 1.1 can be appended to the state vector, resulting in the following:

$$\mathbf{x} = \begin{bmatrix} R & \theta & \alpha & \beta_1^* & \ldots & \beta_p^* \end{bmatrix}^\mathsf{T} \tag{2.27}$$

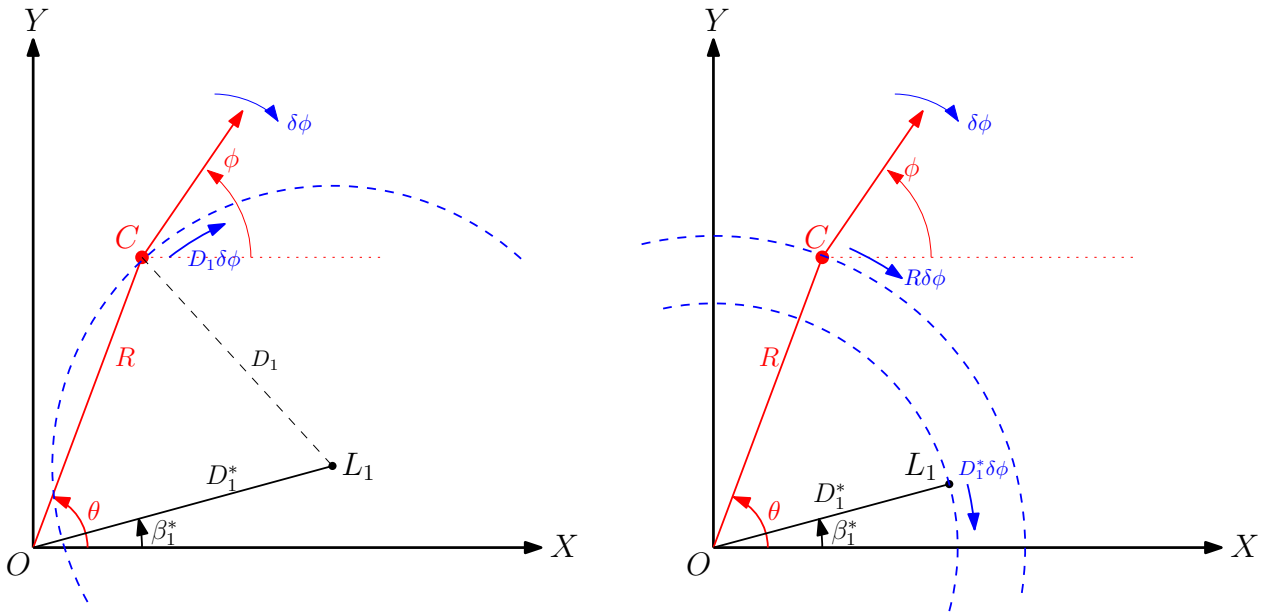


Figure 2.4: Nullspace for Joint Robot Pose and Landmark Bearing Estimation ($p = 1$)

The system observability for varying number of landmarks is as follows:

1 Landmark

For $p = 1$, the observable codistribution has a rank two less than the state vector size. Hence the system is not locally weakly observable in this case. The nullspace has the following basis vector:

$$d\mathcal{G}^{\perp} = \underset{\text{col.}}{\text{span}} \begin{bmatrix} \sin(\beta_1^* - \theta) & 0 \\ 1 - \dfrac{D_1^* \cos(\beta_1^* - \theta)}{R} & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \tag{2.28}$$
$$\triangleq \text{span} \begin{bmatrix} \mathbf{p_1} & \mathbf{p_2} \end{bmatrix}$$

The first vector $\mathbf{p_1}$ is identical to $\mathbf{m_1}$ and hence spans the same slice of the neighbourhood of $\mathbf{x_0}$. The second vector $\mathbf{p_2}$ indicates a degeneracy of all initial states corresponding to equi-angular rotation of the robot pose and landmark, around the origin. The geometric implications of the nullspace basis is depicted in Figure 2.4.

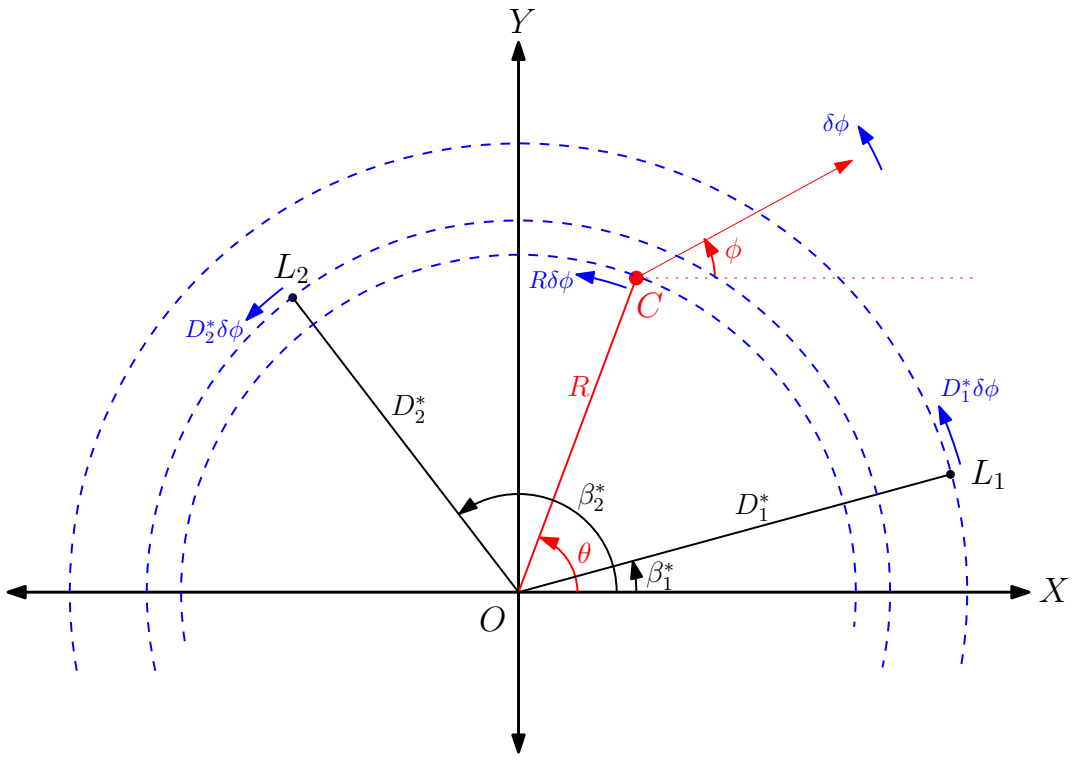


Figure 2.5: Nullspace for Joint Robot Pose and Landmark Bearing Estimation ($p = 2$)

2 or more landmarks

For $p \geq 2$, the observable codistribution has a rank one less than the state vector size. Hence the system is not locally weakly observable. The nullspace is spanned by the following basis:

$$d\mathcal{G}^{\perp} = \underset{\text{col.}}{\text{span}} \begin{bmatrix} 0 \\ 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} \triangleq \text{span} \begin{bmatrix} \mathbf{q_1} \end{bmatrix} \tag{2.29}$$

The basis vector $\mathbf{q_1}$ suggests that all initial states for which the robot pose and both landmarks are rotated by the same angle around the origin, demonstrate the same input-output behaviour. This is illustrated in Figure 2.5.

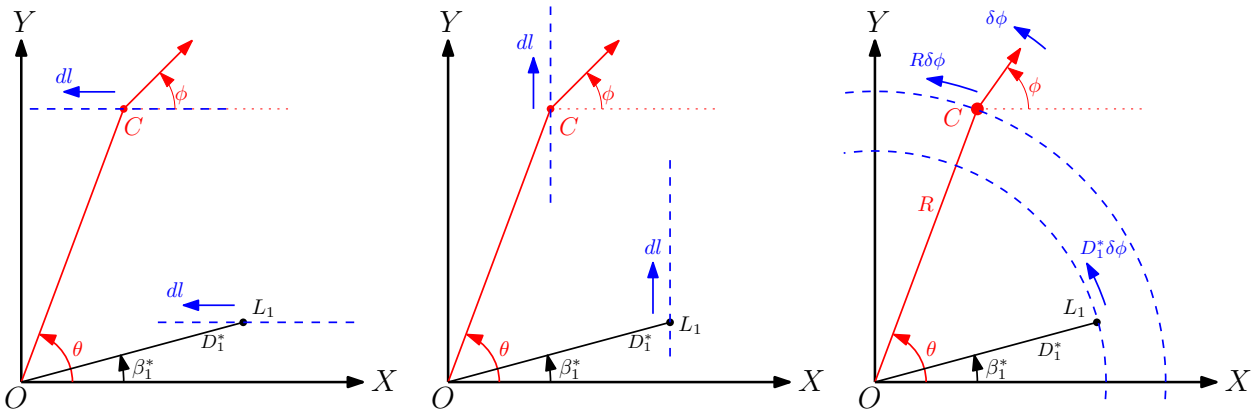


Figure 2.6: Nullspace for SLAM ($p = 1$)

2.3.4 Simultaneous Localization and Mapping

The most frequent scenario when landmark positions are probabilistic in nature is when both coordinates require estimation. This is referred to as the simultaneous localization and mapping (SLAM) problem. We present the results of the nonlinear observability analysis of the SLAM problem expressed in polar coordinates, which is represented by the following state vector:

$$\mathbf{x} = \begin{bmatrix} R & \theta & \alpha & D_1^* & \beta_1^* & \dots & D_p^* & \beta_p^* \end{bmatrix}^\mathsf{T} \tag{2.30}$$

The nonlinear observability of SLAM has been well documented in [22, 24]. However, for this particular formation, the symbolic toolbox performance does not demonstrate consistency between its rank calculation and its search for the basis spanning the nullspace. The responsibility of vetting the results from the software lies on the user, hence results from the aforementioned papers is used to manually check the nullspace of the observability matrix. Upon manual verification, we present the vectors spanning the

unobservable space:

$$
\begin{aligned}
d\mathcal{G}^{\perp} &= \underset{\text{col.}}{\text{span}} \begin{bmatrix}
\cos(\alpha-\theta) & -\sin(\alpha-\theta) & 0 \\
\dfrac{\sin(\alpha-\theta)}{R} & \dfrac{\cos(\alpha-\theta)}{R} & 1 \\
0 & 0 & 1 \\
\cos(\alpha-\beta_1^*) & -\sin(\alpha-\beta_1^*) & 0 \\
\dfrac{\sin(\alpha-\beta_1^*)}{D_1^*} & \dfrac{\cos(\alpha-\beta_1^*)}{D_1^*} & 1 \\
\vdots & \vdots & \vdots \\
\cos(\alpha-\beta_p^*) & -\sin(\alpha-\beta_p^*) & 0 \\
\dfrac{\sin(\alpha-\beta_p^*)}{D_p^*} & \dfrac{\cos(\alpha-\beta_p^*)}{D_p^*} & 1
\end{bmatrix} \\
&\triangleq \text{span}\begin{bmatrix} \mathbf{r}_1 & \mathbf{r}_2 & \mathbf{r}_3 \end{bmatrix}
\end{aligned}
\tag{2.31}
$$

The kernel structure shows degeneracy in the output map for the initial states slightly shifted in the radial or tangential direction, as long as the landmark is shifted in the same manner such that the pose relative to the landmark remains the same. Hence, for unit distance shift for the robot and all landmarks, they can be shifted in any direction in 2D space. The degeneracy extends to rotation of robot pose and landmark around the origin, by the same angle.

This result applies for all $p \geq 1$.

2.3.5 Checking for Anomalies

The use of computational tools to obtain the above results implies that even though the results are applicable for a generic point in the state space, they might not be applicable for every point in the state space. Besançon & Bornard [43] remark that the observability matrix rank computed using symbolic computation software such as Mathematica is valid for *almost all* values of x. In this section, we attempt to manually search for any deviation from the reported results. We look at two possible issues that may arise for certain points in the state space:

1. A particular element in the observability matrix becoming undefined
2. Reduction in dimension of $d\mathcal{G}$

We note that the state equations 1.4 and 1.5 do not have a denominator that would cause the former issue to occur in any of the iterated Lie derivative terms. Hence, for this system we only analyse the latter issue.

To find regions in the state space where the dimension of $d\mathcal{G}$ may be lower than usual, a list of configurations are made for which such a situation may arise. These configurations are listed below:

- For one landmark:
  1. When the robot is collinear with the origin and the landmark (i.e., $\theta = \beta_1^*$)
  2. When robot is moving towards/away from the landmark (i.e. $\alpha = \text{atan2}(y_1 - y_R, x_1 - x_R + 0/\pi)$
  3. When 1 and 2 are encountered simultaneously
- For two landmarks:
  1. When the robot is collinear with the origin and one landmark (i.e., $\theta = \beta_1^*$)
  2. When the robot is collinear with both the landmarks but not the origin (i.e., $R = \dfrac{D_1^* D_2^* \sin(\beta_1^* - \beta_2^*)}{D_2^* \sin(\theta - \beta_2^*) - D_1^* \sin(\theta - \beta_1^*)}$)
  3. When the robot is collinear with the origin and both landmark (i.e., $\theta = \beta_1^* = \beta_2^*$)
  4. When robot is moving towards/away from one landmark (i.e. $\alpha = \text{atan2}(y_1 - y_R, x_1 - x_R + 0/\pi)$
  5. When 2 and 4 are encountered simultaneously

The observability matrix is arrived at, for these configurations, by substituting the equivalent mathematical conditions in the generic matrix itself. The rank remains constant for all of these conditions, irrespective of the type of formulation. These results deem the same analysis for higher number of landmarks as redundant.

2.4 Observability Analysis for Analytic Inputs

In this section, the observability properties of the traditional state space formulation of the bearing-only localization problem are studied in the presence of analytic inputs. It can be empirically inferred that the computational power required to find the observability rank increases as the number of symbols in the formulation increases. Hence this analysis is restricted to robot pose localization problems only.

Analytic inputs are not widely used for robotic applications since most micro-controllers issuing control commands are digital in nature, and hence have an associated sampling time. However, analytic inputs are widely used in missile applications where analog circuits are used to deliver control signals.

This section considers three types of inputs provided to the system and uses Algorithm 2 to evaluate the dimension of $d\mathcal{G}$.

2.4.1 Parabolic Control Inputs

The choice of control input here is parabolic with time. The forward velocity V and the steering angular velocity ω take the following form:

$$\begin{aligned} V &= a_0 t^2 + a_1 t + a_2 \\ \omega &= b_0 t^2 + b_1 t + b_2 \end{aligned} \tag{2.32}$$

The above generic coefficients appear in the observability matrix by repeated use of Equation 2.11. However, the observability properties of this system are the same as those observed for piecewise constant inputs (in Subsection 2.3.1).

2.4.2 Sinusoidal Control Inputs

The choice of control input here is sinusoidal with time. The forward velocity V and the steering angular velocity ω take the following form:

$$\begin{aligned} V &= a_0 \sin(a_1 t + a_2) + a_3 \\ \omega &= b_0 \sin(b_1 t + b_2) + b_3 \end{aligned} \tag{2.33}$$

The above generic coefficients, now 8 in number, appear in the observability matrix by repeated use of Equation 2.11. However, the observability properties of this system are the same as those observed for piecewise constant inputs (in Subsection 2.3.1).

2.4.3 Closed-Loop Control Inputs: Homing Strategy

The choice of control input here is closed-loop, with the control strategy being that of homing. The forward velocity V is assumed to be constant, however the steering angular

velocity ω take the following form:

$$\omega = a_0 \left[\pi - (\alpha - \theta)\right] \tag{2.34}$$

The observability properties of this system are surprising, as this system is shown to be observable with just **1 landmark** to take bearing measurements from. This result is noted, however the implicit assumption made by substituting for ω in Equation 1.4 is that it is now a part of the state transition map and hence not an exogenous input anymore, which is fallacious especially when there is an observer involved and the estimation and control problems are coupled. In that case, the control input then becomes $\omega = a_0 \left[\pi - (\hat{\alpha} - \hat{\theta})\right]$, which is a different structure for which this result does not necessarily apply. Hence the problem has to be described in an appropriate manner for the result to be interpreted correctly.

2.5 Summary

This chapter introduced the concept of nonlinear observability and the relevance of local weak observability for nonlinear systems. The observable codistribution for the system is defined and algorithms are constructed to compute the observable subspace for systems, for piecewise constant inputs as well as analytic inputs. We proceed to compute the rank as well as kernel basis vectors of this observable subspace of four joint state and parameter formulations, for various choice of landmark position coordinates to be estimated. We observe that the kernel basis that spans the unobservable subspace for robot pose localization, is present in the kernel of all other formulations where a parameter is also being estimated. Another key observation is that the root of all these degeneracies is solely the relative motion of the robot pose on a circle around a landmark. These inferences shall be important in the next chapter, where an attempt is made to validate the results obtained in this chapter.

Chapter 3

Validation of Observability Analysis

In this chapter, coordinate transformations are proposed to segregate joint state and parameter formulations into observable and unobservable parts. This serves to validate the results obtained in the previous chapter and further establish the degeneracy for each system through the transformations.

3.1 Coordinate Transformation for Decomposition

The dimension of observable subspace has been computed for several variations of the state and parameter estimation problem for the non-holonomic vehicle in the previous chapter. The kernel basis is also computed and rationalized for each case. However for an individual state x_i to be observable, the unit vector corresponding to it in the state space, $e_i := [0 \dots 1_i \dots 0]^\mathsf{T}$ must be orthogonal to $d\mathcal{G}^\perp$. The estimation error corresponding to x_i converges to 0 for an observer if the above condition is met.

In this chapter we seek to find the observable states and separate them from the unobservable ones. Isidori et al. [36] prove that for a nonlinear affine system Σ for which $d\mathcal{G}$ possesses a dimension $n - d$, for every point $\mathbf{x}$ there exists a local coordinate transformation $\mathbf{z} = \Gamma(\mathbf{x})$ defined in a neighbourhood of $\mathbf{x}$ such that the transformed system can be represented in the form:

$$\begin{aligned}\dot{\xi}_1 &= \sum_{i=1}^{m} g_{1_i}(\xi_1, \xi_2)\, u_i \\ \dot{\xi}_2 &= \sum_{i=1}^{m} g_{2_i}(\xi_2)\, u_i \\ y_i &= h_i(\xi_2)\end{aligned} \tag{3.1}$$

where $\xi_1 = [z_1 \dots z_d]^\mathsf{T}$ and $\xi_2 = [z_{d+1} \dots z_n]^\mathsf{T}$

Here ξ_1 consists of all the unobservable states, whereas ξ_2 consists of all of the observable ones. We can verify that all components of ξ_2 are orthogonal to the unobservable space. This clear demarcation means that in an observer-based simulation, the estimation error corresponding to exactly $n - d$ states will converge to zero. This work attempts to find such a coordinate transformation that manages to express each joint state and parameter formulation in such a manner to verify the results obtained in the previous section. This is equivalent to the Kalman decomposition for linear systems to find its observable and unobservable parts.

The transformations discussed further in this section do not match the structure of the equations in 3.1, however the transformed space contains exactly d unobservable states. Thus observer convergence for the maximum number of x_i's is guaranteed.

To arrive at a transformation, the physical meaning of a degeneracy common to all formulations in the previous section, is discussed. Except for $\mathbf{m_2}$ in Equation 2.20, where there is no relative motion between R and L_1, all kernel basis vectors entail that relative to the landmark, the initial robot pose must shift on a circle for the initial state vector to remain indistinguishable, for any joint state and parameter estimation formulation. Hence, the coordinate transformation is chosen as the shifting of the origin to one landmark, without loss of generality. A notable characteristic of the transformations are that they are ubiquitous to choice of parameter, D_i^* or β_i^*.

3.2 Simulation Results

This work shows convergence of estimate error for the observable components, under input and measurement noise. Hence an Extended Kalman Filter is used as the observer. The initial estimate is deviated from the system state at the first time instant. For the components decoupled from the kernel, the estimation error converges to zero until it

reaches an error bound significantly smaller than the initial estimate error. The state estimation problem is also sensitive to landmark position, making it crucial to have a good choice of landmark position. The parameters for the simulations carried out further in this section are provided in Table 3.1.

Simulation Parameters	Value
Constant Velocity (V)	12 cm/s
Steering Angular Velocity (ω)	0.2 rad/s
Wheel Velocity Noise	$\mathcal{N}(0, 0.001)$
Measurement Noise	$\mathcal{N}(0, 0.0001)$
Sampling Time	0.025 s

Table 3.1: Parameters for EKF-based Simulation

However, the inital covariance matrix is the most crucial tuning paramter for this simulation. The initial covariance matrix $P_{0|0}$ is always a diagonal matrix, with the diagonal values provieded in Table 3.2.

Diagonal Element	Value
σ_R^2	1
σ_θ^2	0.1
σ_α^2	0.1
$\sigma_{D_1^*}^2$	1
$\sigma_{\beta_1^*}^2$	0.1

Table 3.2: Diagonal Elements of $P_{0|0}$ for EKF-based Simulation

3.2.1 1 landmark

The transformation for one landmark is a straightforward shift of origin to L_1, akin to that provided in [25]. The mapping is given by $\mathbf{z} = \Gamma(\mathbf{x}) = [\rho \;\; \beta \;\; \phi \;\; P_1]$ with the following relations:

$$\begin{bmatrix} \rho \\ \gamma \\ \phi \\ P_1 \end{bmatrix} = \begin{bmatrix} \sqrt{(x_r - x_1)^2 + (y_r - y_1)^2} \\ \text{atan2}(y_r - y_1, x_r - x_1) - \alpha \\ \text{atan2}(y_r - y_1, x_r - x_1) \\ D_1^* \text{ or } \beta_1^* \end{bmatrix} \tag{3.2}$$

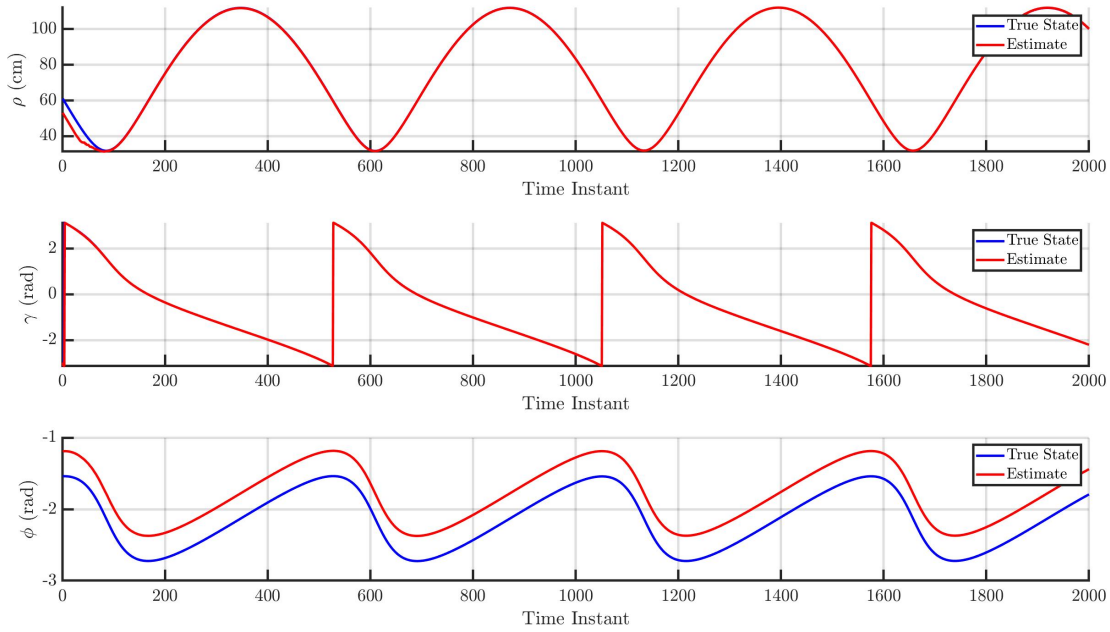


Figure 3.1: Pose Estimation for Transformed System (1 landmark)

where the Cartesian coordinates can be obtained from 1.6, and P_1 is the polar parameter to be . The dynamics of this system are as follows:

$$\dot{\mathbf{z}} = \begin{bmatrix} \cos(\gamma) \\ -\dfrac{\sin(\gamma)}{\rho} \\ -\dfrac{\sin(\gamma)}{\rho} \\ 0 \end{bmatrix} V + \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \end{bmatrix} \omega \tag{3.3}$$

$$\mathbf{y_z} = \gamma \tag{3.4}$$

The observable subspace of this system has a kernel of dimension 2, spanned the vectors $[\,0\,0\,1\,0\,]^\mathsf{T}$ and $[\,0\,0\,0\,1\,]^\mathsf{T}$. The coupling of this kernel basis with only ϕ and P_1 allows the rest of the components to be observable and hence ensure guaranteed convergence of the estimates corresponding to those components.

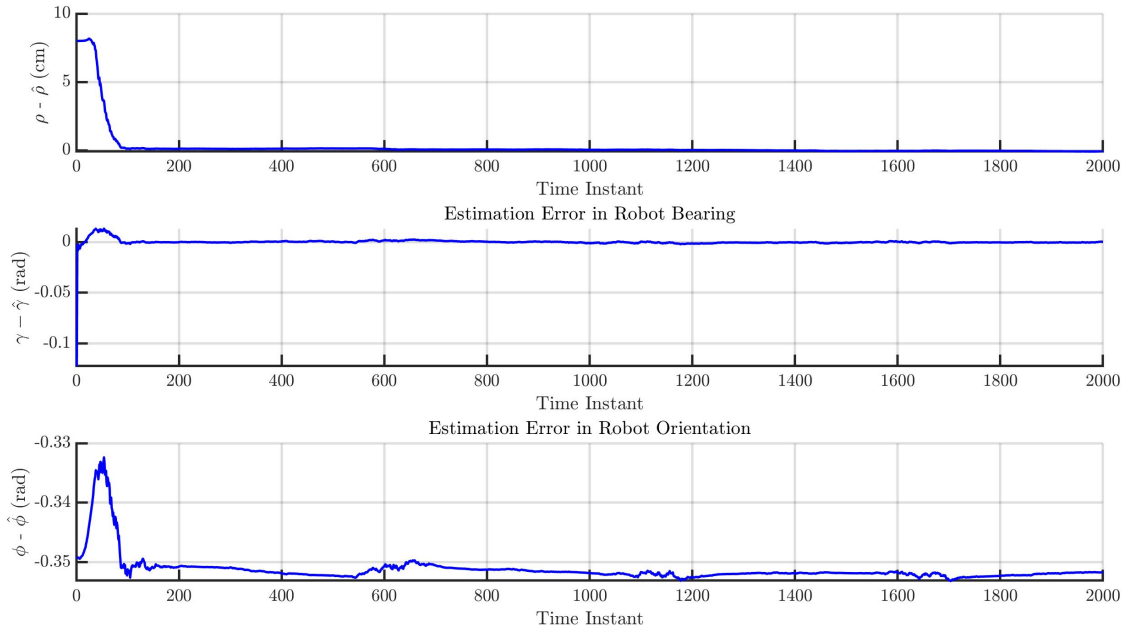


Figure 3.2: Pose Estimation Error for Transformed System (1 landmark)

We simulate this system for the choice of P_1 as β_1^*. For a robot initially located at $(22, 39)$ in the world-centric Cartesian coordinate system, we position L_1 at $(20, 100)$. For this setup, the coordinate transformation (3.2) is applied and the EKF is applied. The results are shown in Figures 3.1 and 3.2.

The estimation error for states ρ and γ converge to 0 rapidly. It is important to note that since the system is non-deterministic in nature, the estimate obtained from the EKF is also a random variable. Hence it cannot drive the estimation error completely to zero, however it reaches an error bound which is much smaller than the initial error. Notice that such a phenomenon does not occur for the unobservable state ϕ, for which estimation error remains as large as the initial value and does not go to zero. The same happens for the parameter, since not only is it unobservable but it does not appear in the state or parameter equations. This is visible in Figure 3.3

The transformation and results are identical for choice of P_1 as D_1^* as the parameter does not appear in the state transition map or the output map.

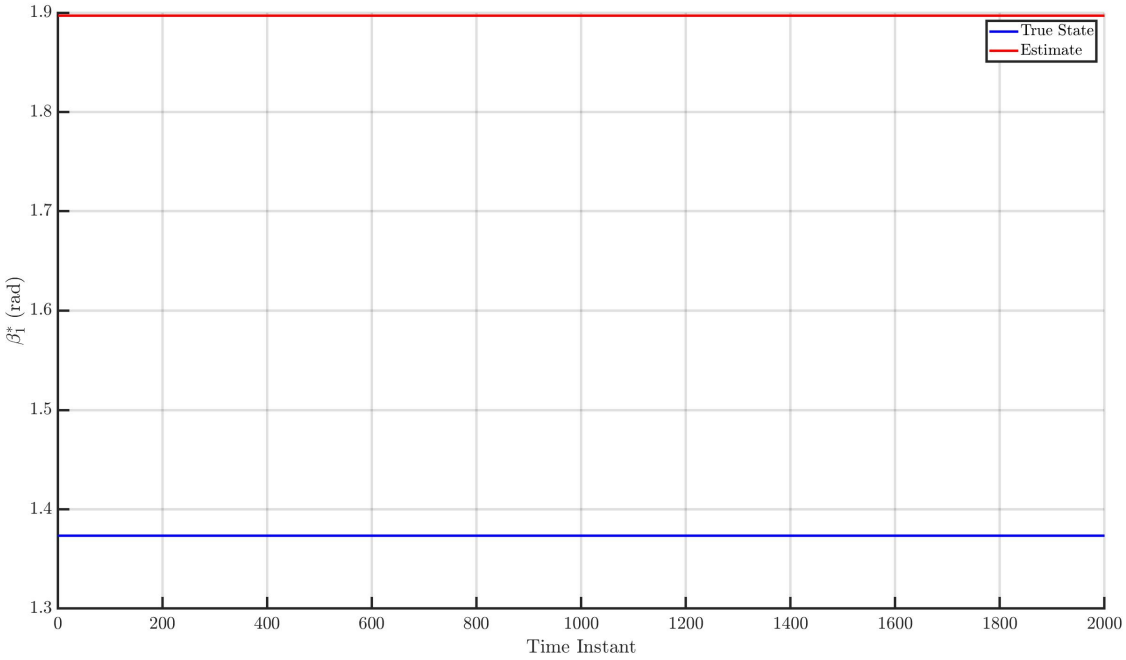


Figure 3.3: Parameter Estimation for Transformed System (1 landmark)

3.2.2 2 landmarks

For 2 landmarks, the transformation resembles the shift of origin to one of the landmarks, without loss of generality. This transformation can be used while estimating D_i^*'s or β_i^*'s alike. The mapping is given by $\mathbf{z} = \Gamma(\mathbf{x}) = [\rho\ \gamma\ \phi\ D_{21}\ \theta_{21r}]$ with the following relations:

$$\begin{bmatrix} \rho \\ \gamma \\ \phi \\ D_{21} \\ \theta_{21r} \end{bmatrix} = \begin{bmatrix} \sqrt{(x_r - x_1)^2 + (y_r - y_1)^2} \\ \text{atan2}(y_r - y_1, x_r - x_1) - \alpha \\ \text{atan2}(y_r - y_1, x_r - x_1) \\ \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \\ \text{atan2}(y_2 - y_1, x_2 - x_1) - \text{atan2}(y_r - y_1, x_r - x_1) \end{bmatrix} \tag{3.5}$$

where the Cartesian coordinates can be obtained from (1.6). The equations governing the state evolution and output are:

$$\dot{\mathbf{z}} = \begin{bmatrix} \cos(\gamma) \\ -\frac{\sin(\gamma)}{\rho} \\ -\frac{\sin(\gamma)}{\rho} \\ 0 \\ 0 \end{bmatrix} V + \begin{bmatrix} 0 \\ -1 \\ 0 \\ 0 \\ 0 \end{bmatrix} \omega \tag{3.6}$$

$$\mathbf{y} = \begin{bmatrix} \gamma \\ \operatorname{atan2}\left(D_{21}\sin(\theta_{21r}+\phi) - \rho\sin(\phi), D_{21}\cos(\theta_{21r}+\phi) - \rho\cos(\phi)\right) - (\phi - \gamma) \end{bmatrix} \tag{3.7}$$

The observable subspace of this system has a kernel of dimension 1, spanned the vector $[\,0\;0\;1\;0\;0\,]^{\mathsf{T}}$. The coupling of this kernel basis vector with only one state allows the other states to be observable and hence ensure guaranteed convergence of those state estimates. Notice that the parameters appended for this transformation are the distance between the two landmarks, D_{21}, and the angle subtended by L_2 and R at L_1, θ_{21r}.

The implication of this result is that the relative distance and bearing between two landmarks can be estimated, even in the absence of any other landmarks to take bearing-only measurements from. For the simulation, the same robot initial location $(22, 39)$ in the world-centric Cartesian coordinate system, is considered. The two landmarks L_1 and L_2 are positioned at $(-40, 40)$ and $(20, 100)$ respectively. The coordinate transformation is applied and the EKF is used. The results are depicted in Figures 3.4 and 3.5.

Apart from ϕ, the estimation error converges for all the coordinates of $\mathbf{z}$. Note that for the transformed system, the parameters D_{21} and θ_{21r} are observable. The parameter estimation results are demonstrated in Figures 3.6 and 3.7.

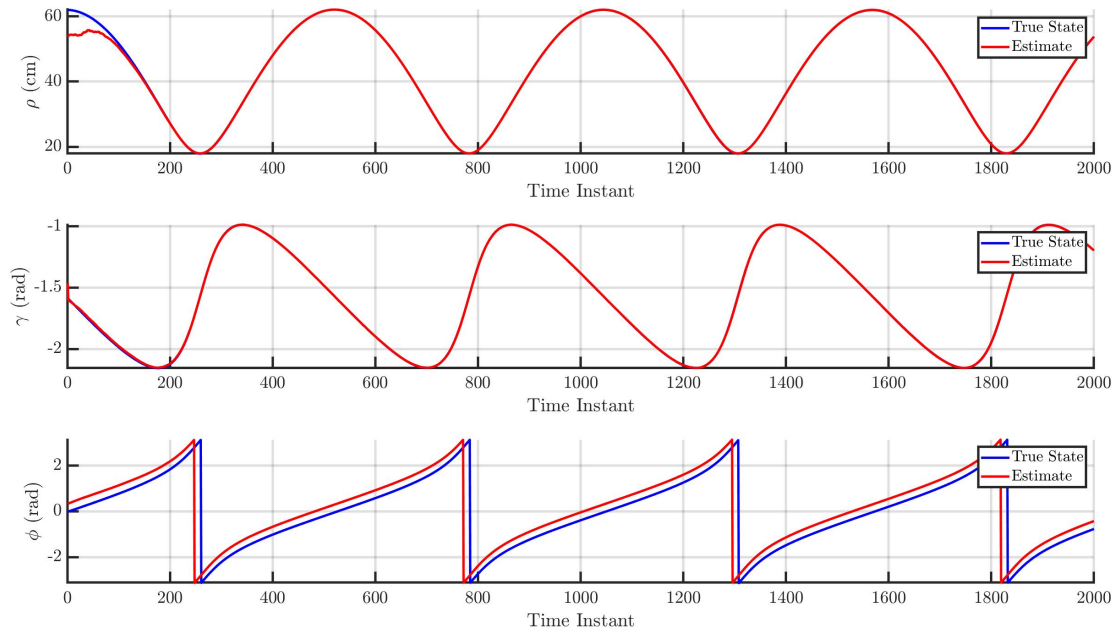


Figure 3.4: Pose Estimation for Transformed System (2 landmarks)

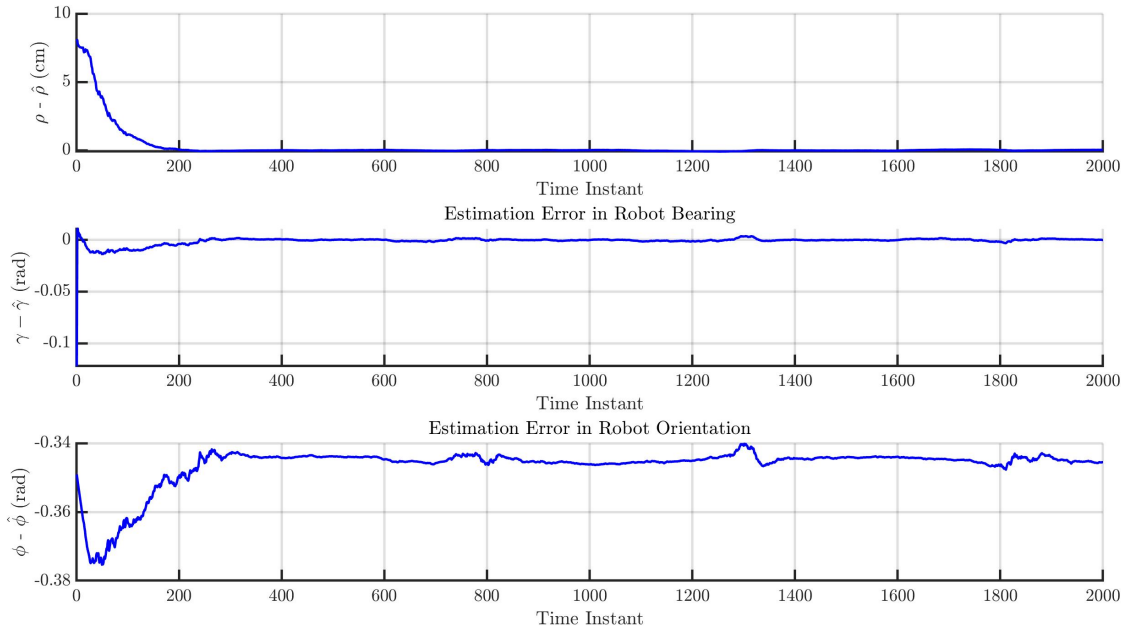


Figure 3.5: Pose Estimation Error for Transformed System (2 landmarks)

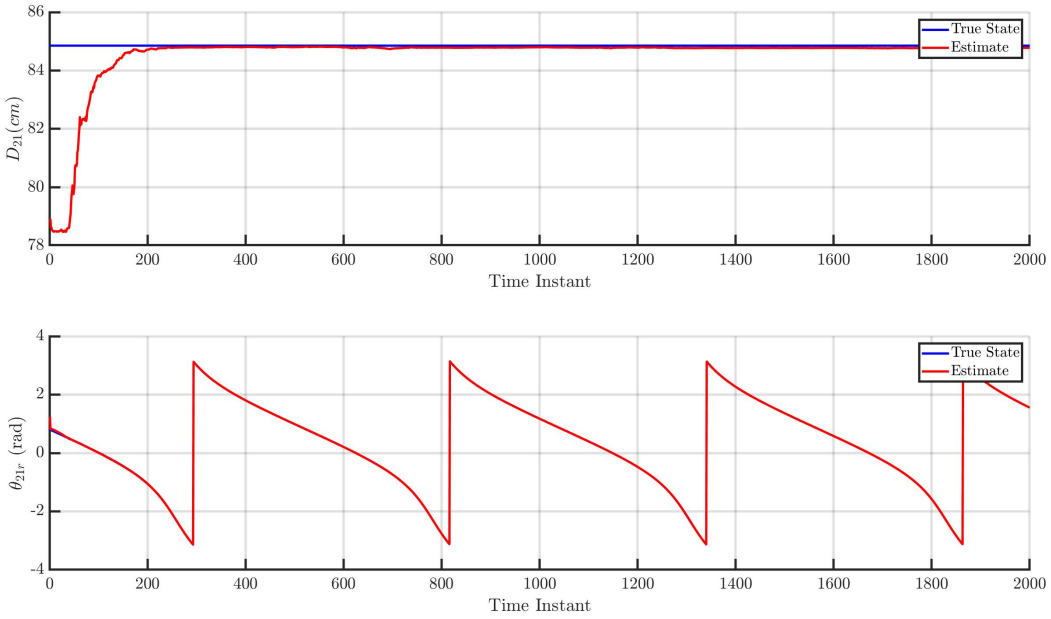


Figure 3.6: Parameter Estimation for Transformed System (2 landmarks)

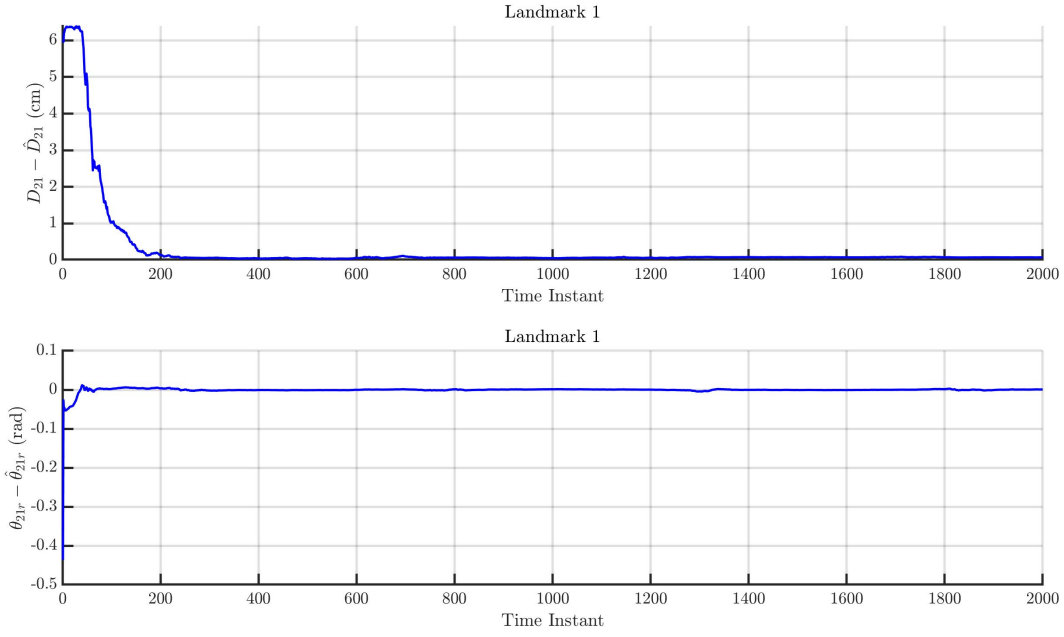


Figure 3.7: Parameter Estimation Error for Transformed System (2 landmarks)

3.3 Summary

This chapter validates the observability results obtained previously by means of system transformations that yield the observable and unobservable parts. These transformations validate our understanding of the unobservable subspace by virtue of their structure. We demonstrate the convergence of the observable state estimation errors in presence of input & measurement noise. However, estimation of landmarks can be computationally expensive and does not specifically focus on robot pose localization. In situations where the system has to be robust to uncertainties in landmark position and still has to manage to localize within tight error bounds, estimating landmark positions may not give the best results. The next chapter discusses this problem and possible solutions towards the same.

Chapter 4

Robot Localization in Uncertain Environments

In this chapter, we discuss state estimation to determine robot pose in scenarios where knowledge of the environment features is inaccurate. The emphasis is placed on robot pose localization. Two techniques are used to mitigate the resultant estimator divergence: the Q-EKF and the PI-Q-EKF. These are described and the performance is compared with EKF-SLAM, the standard solution to uncertain landmarks in literature.

4.1 Estimation in Uncertain Conditions

Consider the system described in Section 1.2.1, but with a panoramic image that contains outdated information about the features in the environment. Hence the β_1^*'s and D_i^*'s obtained from it are inaccurate. For the traditional state space formulation, these parameters are part of the measurement map. The go-to solution for this problem is usually SLAM, since the robot pose computed is correlated to the feature positions. The landmark position coordinates are appended to the state to give the state vector mentioned in Equation 2.30. The size of this state vector is $2p+3$, thus if p is large, the state vector becomes very large merely to accommodate incremental changes in the landmark positions that occurred since the panoramic image was taken.

4.1.1 Q-EKF: Characterizing Uncertainty as Process Noise

However, when the augmented state space formulation (Section 1.2.2) is used to model the same problem, the landmark coordinates shift to the state transition map. This allows any inaccuracy in the state transition map to be modelled as structured process noise. Consider the true landmark coordinates, whose values are not known, to be represented as (D_i^*, β_i^*). The landmark coordinates, made available by the panoramic image and the information that determines the environment scale, are $(\bar{D}_i^*, \bar{\beta}_i^*)$. Accounting for this bias, the true state dynamics can be expressed as:

$$\dot{\mathbf{x}} = f(\mathbf{x}, u) \approx \bar{f}(\mathbf{x}, u) + \sum_{i=1}^{p} J_{D_i^*}(D_i^* - \bar{D}_i^*) + \sum_{i=1}^{p} J_{\beta_i^*}(\beta_i^* - \bar{\beta}_i^*) \tag{4.1}$$

where

$$J_{D_i^*} = \frac{\partial f(\mathbf{x}, u)}{\partial \bar{D}_i^*},\ J_{\beta_i^*} = \frac{\partial f(\mathbf{x}, u)}{\partial \bar{\beta}_i^*} \tag{4.2}$$

$$\bar{f}(\mathbf{x}, u) = \begin{pmatrix} \cos(\alpha - \theta) & 0 \\ \frac{\sin(\alpha - \theta)}{R} & 0 \\ 0 & 1 \\ \frac{-\sin(\beta_1)}{RC_1 - \bar{D}_1^* C_1^*} & -1 \\ \frac{-\sin(\beta_2)}{RC_2 - \bar{D}_2^* C_2^*} & -1 \\ \vdots & \vdots \\ \frac{-\sin(\beta_q)}{RC_p - \bar{D}_p^* C_p^*} & -1 \end{pmatrix} u \tag{4.3}$$

$$= \begin{bmatrix} g_1 & g_2 \end{bmatrix} u \tag{4.4}$$

where $C_i = \cos(\theta - (\beta_i + \alpha))$ and $\bar{C}_i^* = \cos(\bar{\beta}_i^* - (\beta_i + \alpha))$.

The landmark coordinate errors $(D_i^* - \bar{D}_i^*)$ and $(\beta_i^* - \bar{\beta}_i^*)$ are fixed biases during a simulation if the object is not semi-static in nature. We still treat it as process noise, not using it as as corrective term to predict $\hat{\mathbf{x}}_{k+1|k}$, but instead incorporating the uncertainty as an additive term while predicting $P_{k+1|k}$. The aim is not to estimate the disturbance in the system, but merely to reject it. The addition of an extra process noise covariance

term while calculating $P_{k+1|k}$ reduces the weightage of the filter on the prediction step and gives more value to the measurements, hence mitigating estimator divergence.

Thus, after accounting for zero-mean Gaussian input noise, the prediction steps of the EKF are:

$$\hat{\mathbf{x}}_{k+1} = F(\hat{\mathbf{x}}_k, \bar{u}_k) \tag{4.5}$$

$$P_{k+1|k} = \mathbf{\Phi} P_{k|k} \mathbf{\Phi}' \ + \ \Gamma_u Q_d \Gamma'_u \ + \ Q_{D^*} + Q_{\beta^*} \tag{4.6}$$

where

$$F(\hat{\mathbf{x}}_k, \bar{u}_k) = \hat{\mathbf{x}}_k + \int_{t_k}^{t_{k+1}} f(\hat{\mathbf{x}}(\tau), \bar{u}_k) d\tau \tag{4.7}$$

$$\mathbf{\Phi_k} = \frac{\partial F(\hat{\mathbf{x}}_k, \bar{u}_k)}{\partial \hat{\mathbf{x}}} \tag{4.8}$$

$$Q_d = \begin{bmatrix} \sigma_V^2 & 0 \\ 0 & \sigma_\omega^2 \end{bmatrix} \tag{4.9}$$

$$\mathbf{J}_{D_i^*} = \frac{\partial \mathbf{F}(\mathbf{x}, u)}{\partial \bar{D}_i^*}, \ \mathbf{J}_{\beta_i^*} = \frac{\partial \mathbf{F}(\mathbf{x}, u)}{\partial \bar{\beta}_i^*} \tag{4.10}$$

$$Q_{D^*} = \sum_{i=1}^{p} \mathbf{J}_{D_i^*} (D_i^* - \bar{D}_i^*)^2 \mathbf{J}_{D_i^*} \tag{4.11}$$

$$Q_{\beta^*} = \sum_{i=1}^{p} \mathbf{J}_{\beta_i^*} (\beta_i^* - \bar{\beta}_i^*)^2 \mathbf{J}'_{\beta_i^*} \tag{4.12}$$

The matrices Q_{D^*} and Q_{β^*} do not have be calculated; rather, based on approximate knowledge of the absolute error in each of the coordinates, they can be tuned to get the desired performance. Depending on the diagonal values, these matrices increase the variance of each $(\hat{\beta}_i)_{k+1|k}$. However, since this is merely a disturbance rejection method, it works best when the difference between initial true state and and initial estimate is low. The lack of a corrective term in the prediction step means that the Q-EKF cannot filter out pre-existing estimation errors.

4.1.2 PI-EKF: Proportional Integral Extended Kalman Filter

To alleviate this problem, we present an extension of the Proportional-Integral Kalman Filter, presented by Linder & Shafai [44] and Shenoy [45], for nonlinear systems. The Proportional Integral Extended Kalman Filter (henceforth PI-EKF) introduces an artificial bias in the prediction step of the filter to mitigate errors introduced due to model-plant mismatch. The bias is made up by summing up innovations over time. These innovations are accumulated after being scaled by a 'forgetting factor', henceforth referred to as the integral gain matrix. The effect of this bias on the predicted estimate further depends on a proportional gain matrix.

The revised prediction step of the filter is:

$$\begin{aligned} \hat{\mathbf{x}}_{k+1|k} &= \mathbf{F}(\hat{\mathbf{x}}_{k|k}, \bar{u}_k) + M\mathbf{q}_k \\ \hat{\mathbf{y}}_{k+1|k} &= h(\hat{\mathbf{x}}_{k+1|k}) \end{aligned} \tag{4.13}$$

The covariance prediction step and the Kalman update step remain the same as the EKF. $\mathbf{q}_{k-1} \in \mathbb{R}^n$ represents the bias constructed by accumulated innovations. It is updated with the regular update step in the following manner:

$$\mathbf{q}_{k+1} = \mathbf{q}_k + N\left(\mathbf{y}_k - \hat{\mathbf{y}}_{k|k-1}\right) \tag{4.14}$$

Here, $M \in \mathbb{R}^{n\times n}$ and $N \in \mathbb{R}^{n\times p}$ are the proportional and integral gain matrices. The aforementioned matrices are chosen prior to the filtering and tuning of these matrices considerably affects the quality of the estimates [26].

The PI-EKF can potentially correct biases due to a model-sensor mismatch that may cause faulty measurements, however it is not equipped to do so directly. The Kalman gain, which arises from an optimization problem for minimum state variance for the linear Kalman filter, is not specifically designed to treat such biases, although some system structures may render sufficiently capable. Hence this filter cannot be used for the traditional state space model. However on using the augmented state space model along with the additive covariance matrices $Q_{D^*} + Q_{\beta^*}$, the PI-Q-EKF can be demonstrated as effective.

4.2 Simulation Results

The efficacy of using the augmented formulation in combination with the aforementioned techniques is demonstrated in simulation. The bearing-only robot is simulated to move for constant velocity inputs, with three landmarks providing bearing measurements. The initial pose of the robot is fixed at $(22, 39)$ with a yaw angle of 90. The parameters of the EKF simulation are provided in Table 4.1. The initial covariance matrix is constructed using diagonal elements from Table 4.2.

Simulation Parameters	Value
Constant Velocity (V)	10 cm/s
Steering Angular Velocity (ω)	0.5 rad/s
Wheel Velocity Noise	$\mathcal{N}(0, 0.01)$
Measurement Noise	$\mathcal{N}(0, 0.0025)$
Sampling Time	0.025 s
Simulation Length	1200 steps

Table 4.1: Parameters for EKF-based Simulation

The bearing-only localization problem is sensitive to positioning of landmarks in the environment [46]. Hence we demonstrate simulation results for two sets of landmark positions:

Diagonal Element	Value
σ_R^2	0.1
σ_θ^2	0.01
σ_α^2	0.01
$\sigma_{\beta_1}^2$	0.1
$\sigma_{D_1^*}^2$	1
$\sigma_{\beta_1^*}^2$	0.1

Table 4.2: Diagonal Elements of $P_{0|0}$ for Simulations

The landmark coordinates are given in Table 4.3, thus ensuring that the robot always remains inside the polygon made by the landmarks. For such localization problems, the estimate is generally well-behaved.

Coordinate	Value	Coordinate	Value
D_1^*	160.3122	$\bar{D}_1^*$	150.3122
β_1^*	1.5084	$\bar{\beta}_1^*$	1.7084
D_2^*	107.0047	$\bar{D}_2^*$	117.6713
β_2^*	-0.6528	$\bar{\beta}_2^*$	-0.3528
D_3^*	91.9239	$\bar{D}_3^*$	78.5905
β_1^*	-2.3562	$\bar{\beta}_1^*$	-2.5262

Table 4.3: Parameters for EKF-based Simulation

4.2.1 EKF-SLAM

We show the results of using SLAM for this setup in Figure 4.1 and 4.2

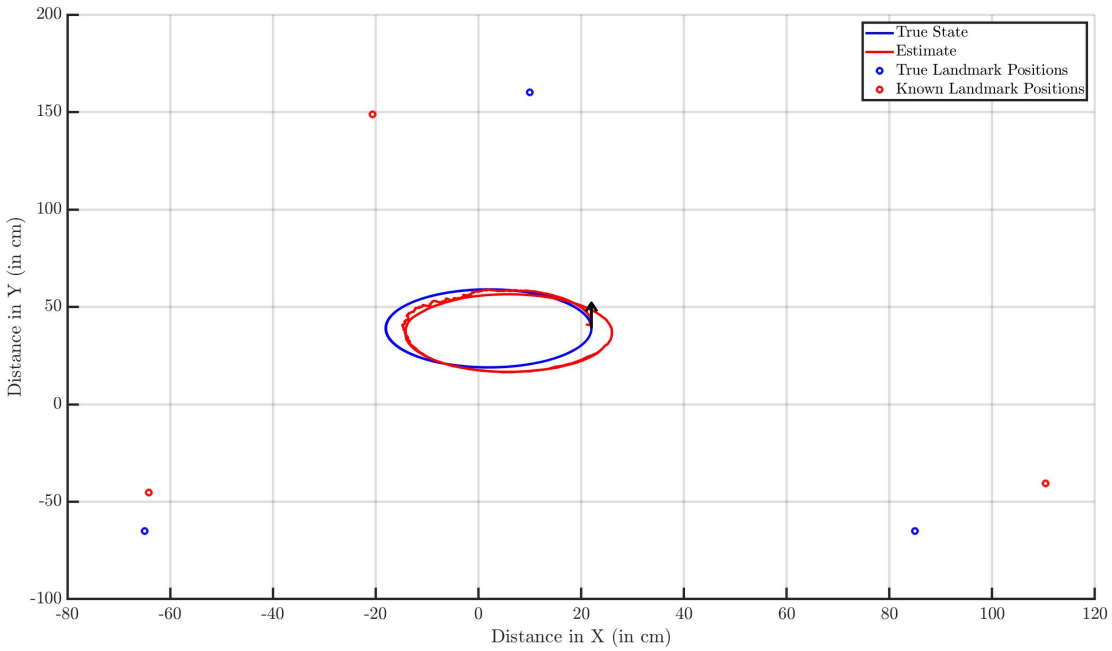


Figure 4.1: 2-D Trajectory of True State & Estimate for EKF-SLAM

Notice that the estimation error significantly increased throughout the simulation. This increase was considerably reflected in the Cartesian coordinates as well. Figures 4.3 and 4.4 show that the estimates of the landmark position coordinates did not converge within the given time frame as well. Hence, for the time duration of the simulation,

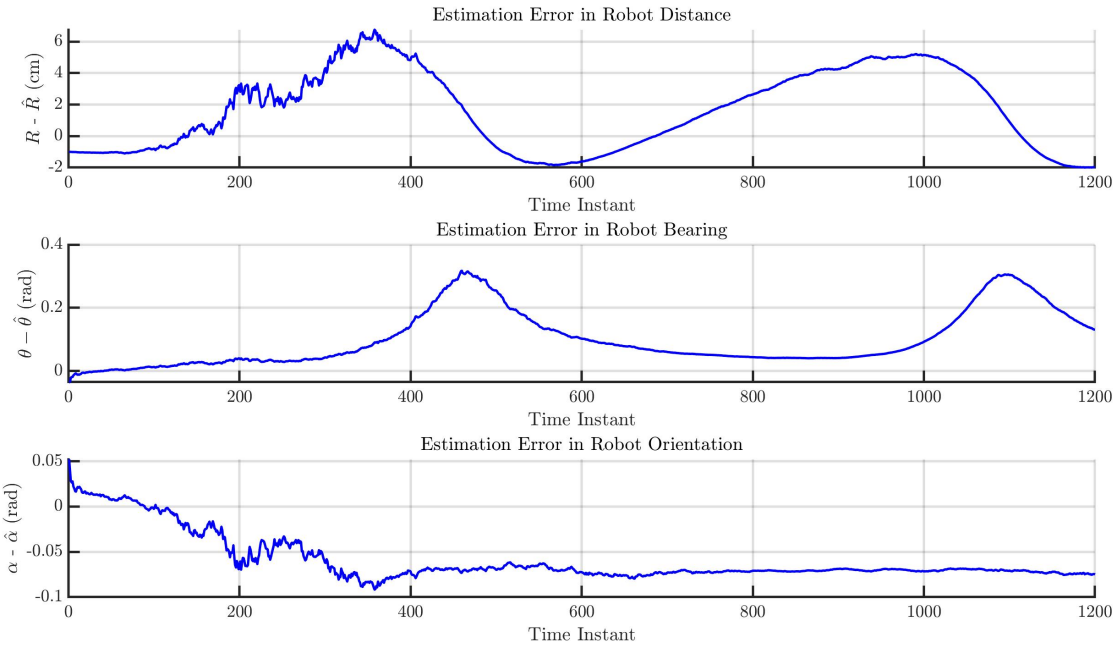


Figure 4.2: Estimation Error of Robot Pose for EKF-SLAM

EKF-SLAM was neither able to find accurate robot pose estimates nor was it able to estimate the parameters to a satisfactory degree.

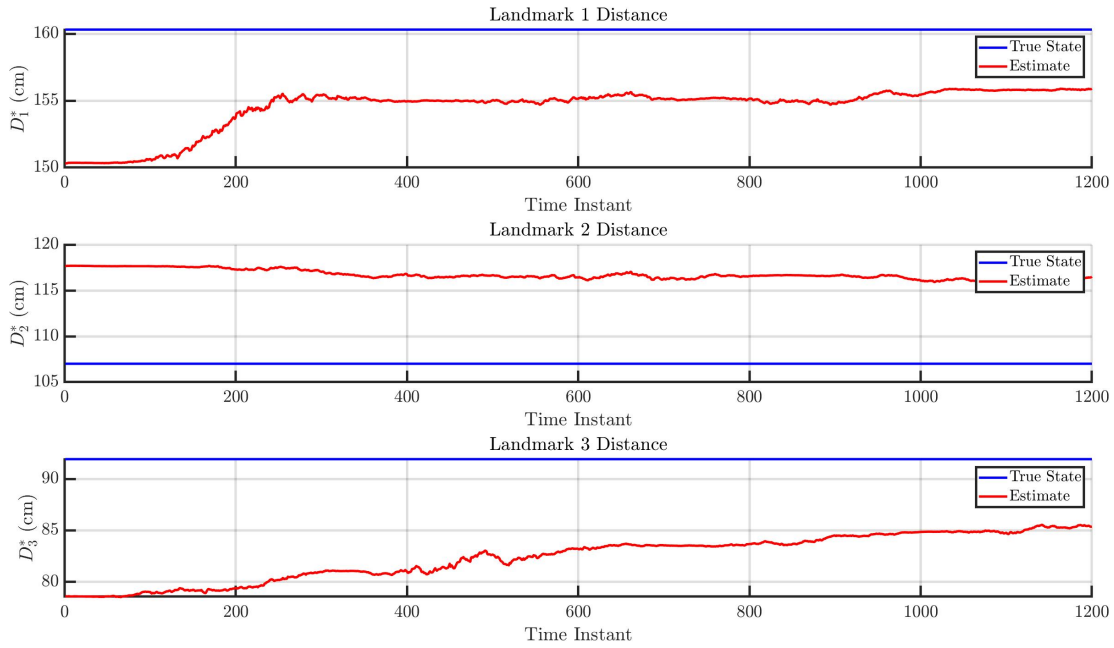


Figure 4.3: Estimation of D_i^* for EKF-SLAM

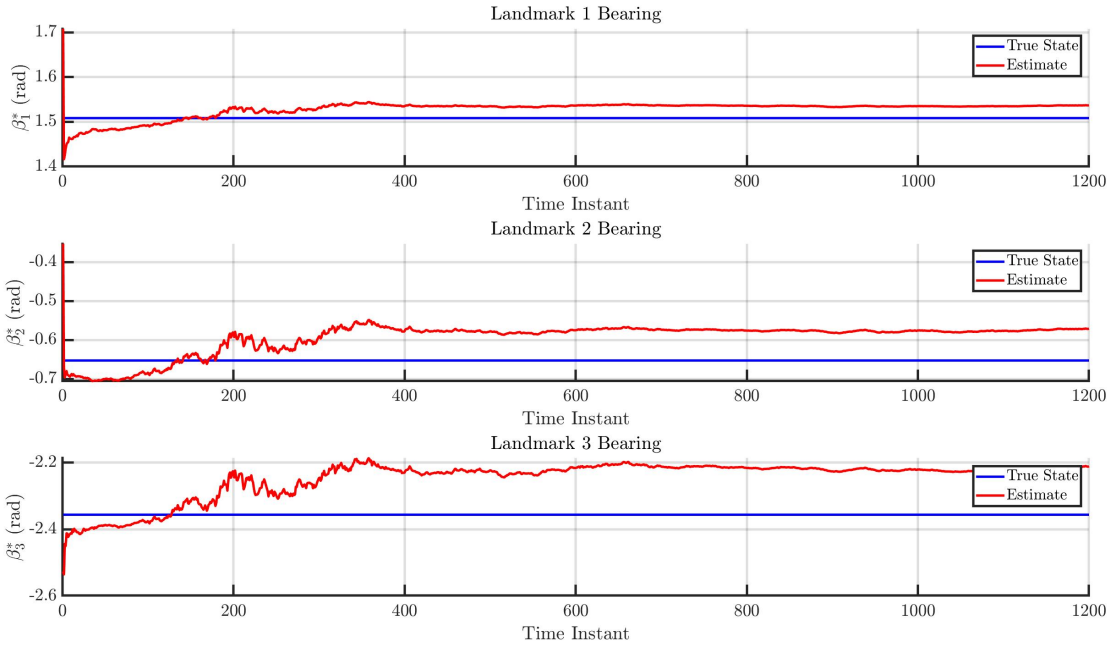


Figure 4.4: Estimation of β_i^* for EKF-SLAM

4.2.2 Augmented State + Q-EKF

We simulate the augmented state space model for the same setup. For the covariance matrices Q_{D^*} and Q_{β^*}, it is sufficient to tune the sum of the matrices. The choice of $Q_{D^*} + Q_{\beta^*}$ for the simulation is taken as $0.1I_{6\times6}$. The results are shown in Figures 4.5, 4.6 and 4.7.

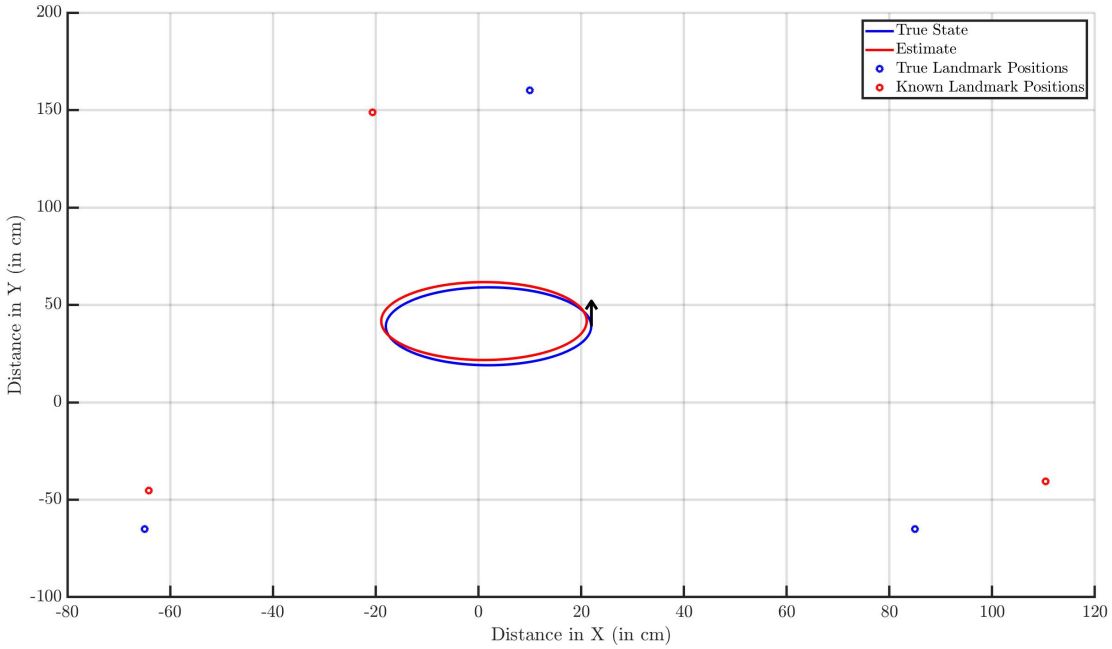


Figure 4.5: 2-D Trajectory of True State & Estimate for Augmented State + Q-EKF

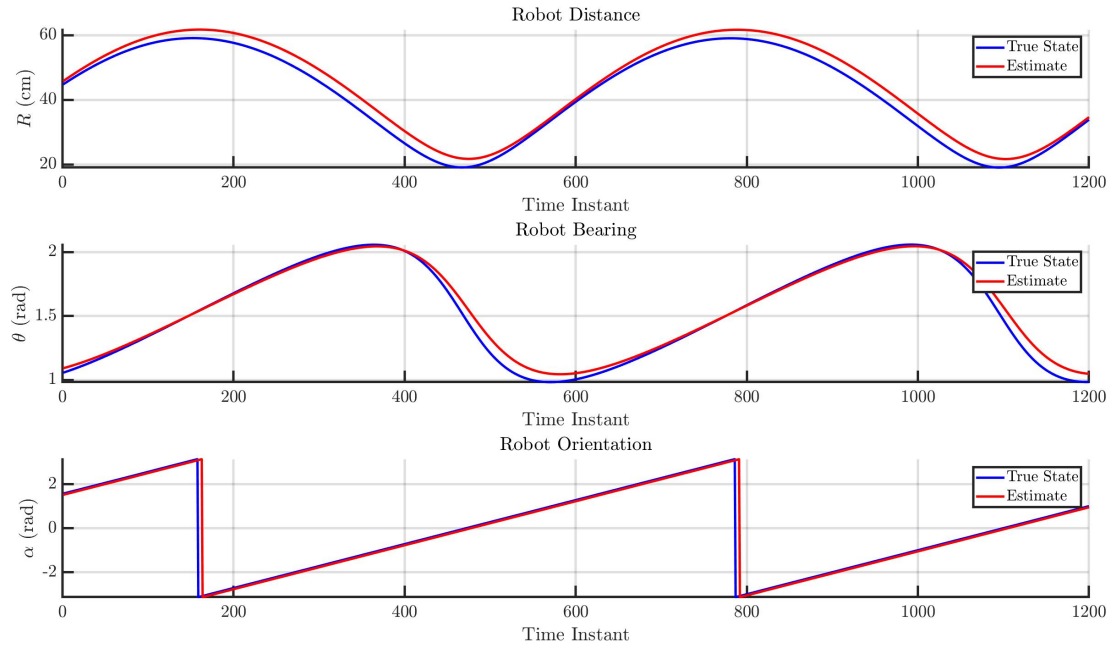


Figure 4.6: Robot Pose Estimation for Augmented System + Q-EKF

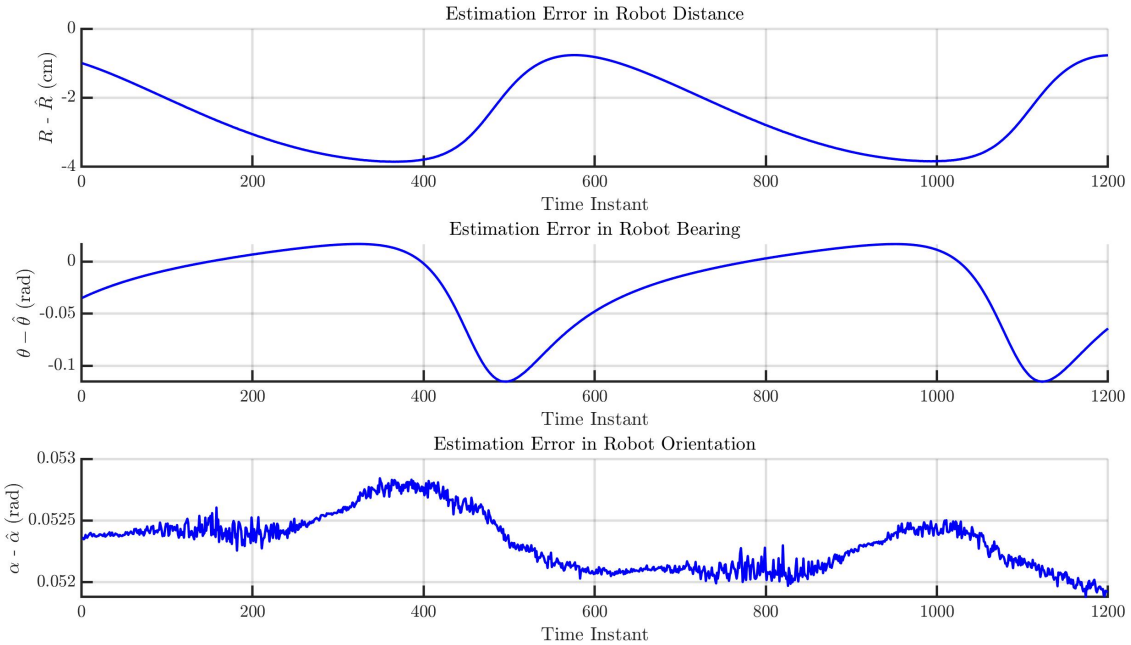


Figure 4.7: Robot Pose Estimation Error for Augmented System + Q-EKF

We observe that the expected divergence in true state and estimate is considerably mitigated by the addition of the extra covariance terms. However, the initial estimate error is not rectified and a deviant, slightly warped circular trajectory is traced by the estimate.

Another remark that is relevant here is regarding to the disturbance rejection property that this choice of system formulation and EKF observer possesses. If the initial estimate matches the true state, for the augmented model paired, the Q-EKF can keep the estimation error close enough to 0. However, on the other hand, if the initial estimation error is high, it is outperformed by EKF-SLAM. Fortunately, the PI-Q-EKF corrects that.

4.2.3 Augmented State + PI-Q-EKF

The same setup is simulated for the augmented model paired with the PI-EKF along with the added process covariance matrices, along with the extra covariance terms Q_{D^*} and Q_{β^*}. The choice of $Q_{D^*} + Q_{\beta^*}$ for the simulation is taken as $0.1I_{6\times6}$. The more important tuning parameters in this simulation are M and N. They are chosen as:

$$N = \begin{bmatrix} 2.2 & 2.2 & 2.2 \\ 0.055 & 0.055 & 0.055 \\ -0.07 & -0.07 & -0.07 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{4.15}$$

$$M = 0.4I_{6\times6} \tag{4.16}$$

For these set of matrices, along with the parameters mentioned in Tables 4.1, 4.2 and 4.3, the results are given in Figures 4.8, 4.9 and 4.10.

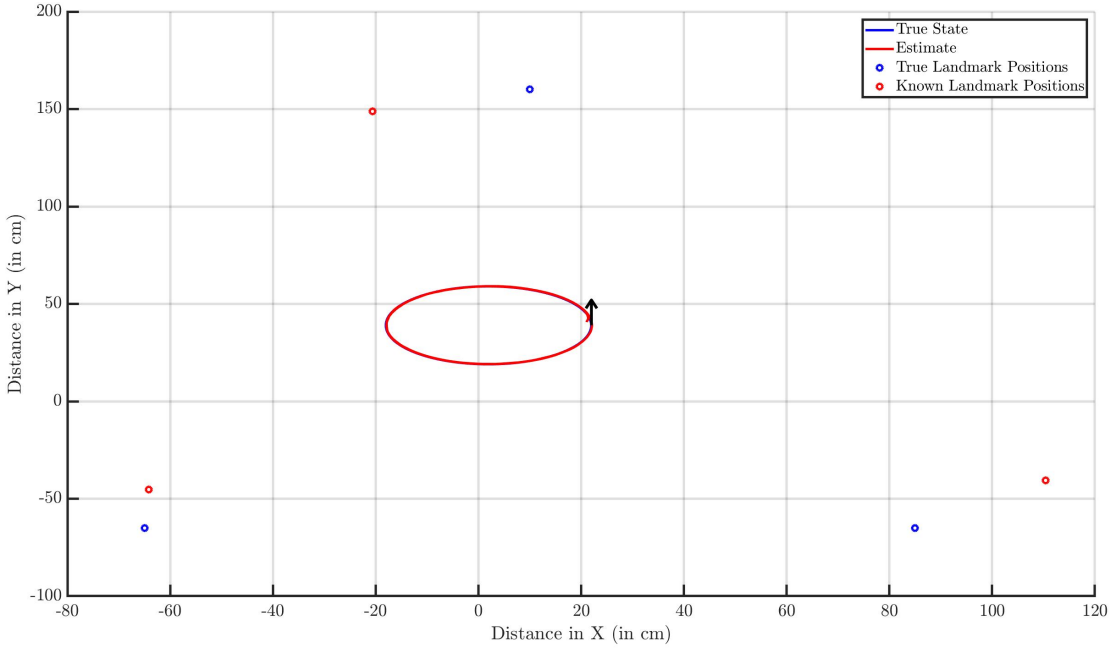


Figure 4.8: 2-D Trajectory of True State & Estimate for Augmented State + PI-Q-EKF

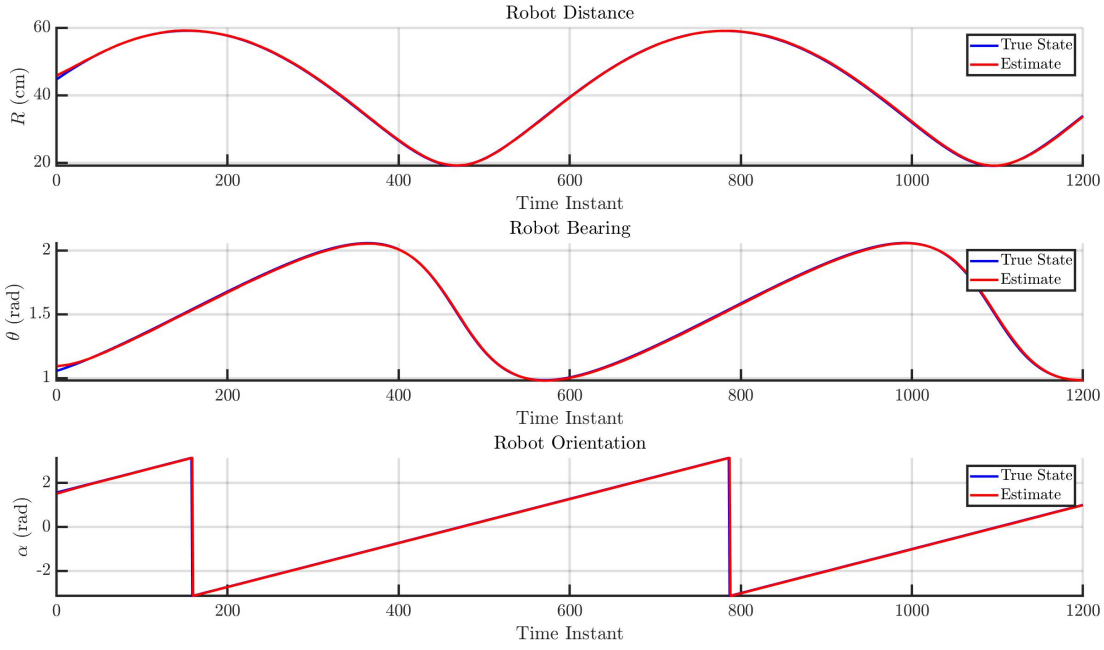


Figure 4.9: Robot Pose Estimation for Augmented System + PI-Q-EKF

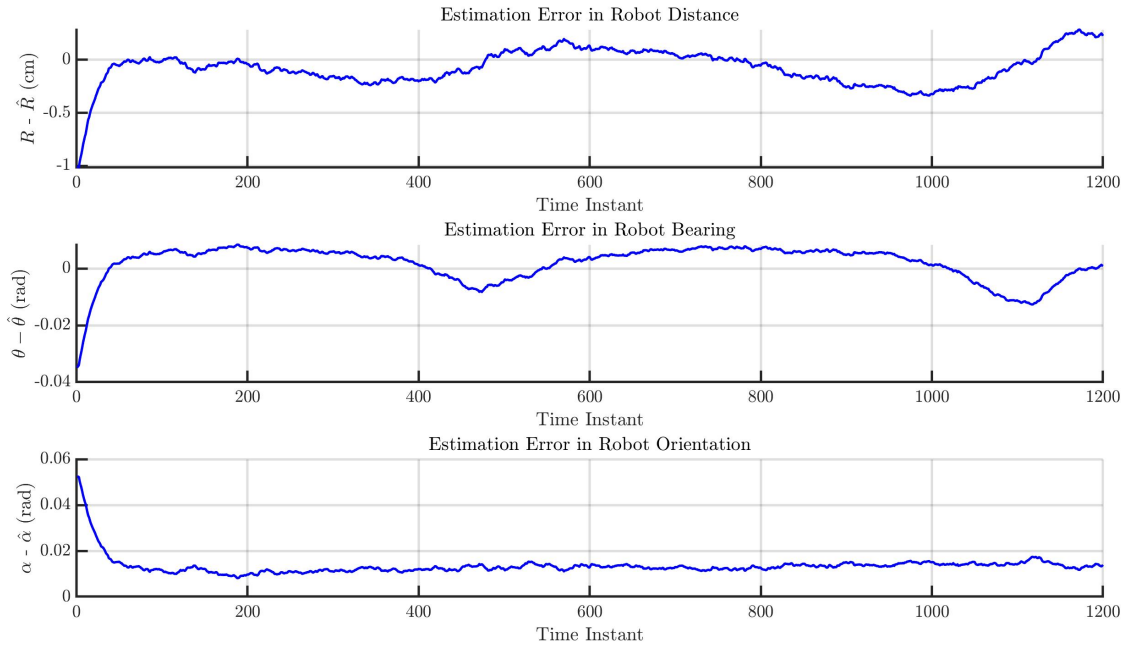


Figure 4.10: Robot Pose Estimation Error for Augmented System + PI-Q-EKF

The estimation error reduces considerably. However, this is sensitive to the choice of the elements of the integral gain matrix. Restricting the search to matrices of the form $N = n_0[\vec{n_1} \;\; \vec{n_2} \;\; \vec{n_3} \;\; n_4 I_{p\times p}]^\mathsf{T}$ where $\vec{n_i} = n_i[1 \;\; 1 \;\; 1]^\mathsf{T}, i \in \{1, 2, 3\}$ seems sufficient, however the search for $n_i, i \in \{0, 1, 2, 3, 4\}$ is the crucial step. The choice of M as $m_0 I_{n\times n}$ is also sufficient. The selection of m_0 and n_0 is primarily the search for an upper bound, above which the state estimates stop converging further and instead start oscillating around the convergent estimate value at that time instant.

4.3 Summary

This chapter proposes solutions to better and faster robot pose localization in conditions where the robot does not have exact knowledge of its surroundings. Two modifications to the EKF are presented, which when paired with the augmented state space formulation, lead to better results. The augmented state space formulation has a state vector size of $p + 3$ as compared to $2p + 3$ for EKF-SLAM. This points to a large difference in computational time for a large number of features in the environment, which is the case for a robot moving around in real-world scenarios. In the next chapter, we observe the

performance of these filters in a simulation environment that mimics the real world quite closely.

Chapter 5

Simulations in ROS-Gazebo Environment

This chapter applies the Extended Kalman Filters discussed in the previous section, to a robot model possessing vision capabilities, in a virtual world created on Gazebo. This simulation study enables performance comparisons between them for a robot and environment model that emulate real-life behaviour to a great extent. It also provides an opportunity to analyse how they behave in a multi-rate measurement scenario. However, most importantly, an implementation of these filters in conjunction with a vision pipeline helps to evaluate the joint performance of each combination.

In this work, we discuss localization scenarios for accurately as well as inaccurately known landmark positions. For accurately known landmarks, the comparison is made between an EKF implemented on traditional and augmented state space formulations. In the latter case, a comparative is drawn between the Q-EKF used with the augmented state, and EKF-SLAM used with the standard formulation. The Individual Compatibility Nearest Neighbours test is incorporated into each filter to use measurement predictions to eliminate spurious matches.

5.1 Camera Model

5.1.1 Transformation to Image Space

The raw measurement obtained from the camera is in the image space. To establish a measurement model, a coordinate transformation between the world frame and camera

frame. The robot model consists of two wheels and two casters. hence it fits the description of a non-holonomic unicycle bot. The camera is located radially outward from the robot on its cylindrical base, as shown in Figure 5.1

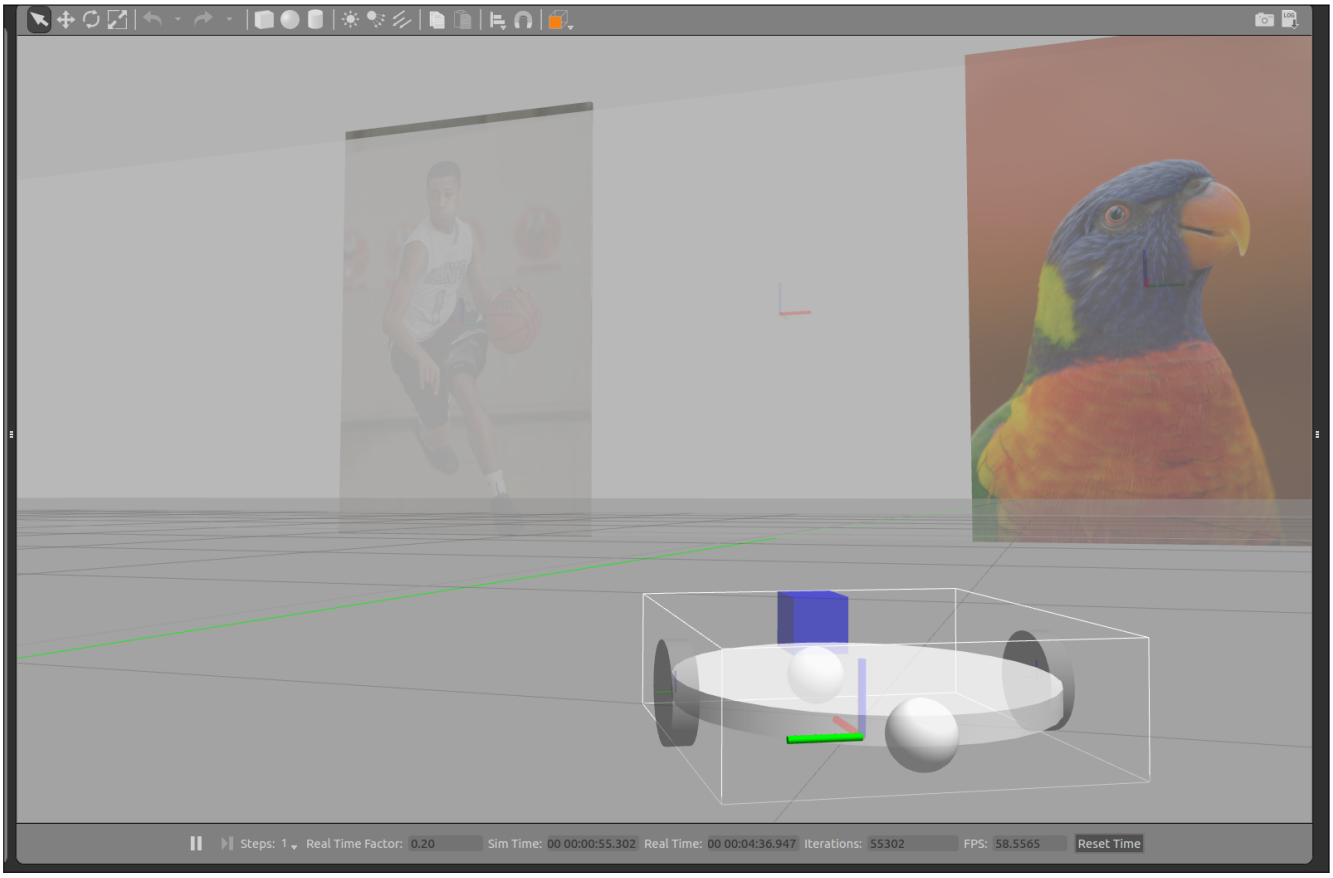


Figure 5.1: Wheeled Robot Model

The axes for the camera are chosen such that the forward direction is chosen as the z axis, while the x and y axes are along the horizontal and vertical directions in the plane perpendicular to the velocity vector, respectively.

Hence the transformation of a feature point in world-centric coordinates, to camera coordinates are:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} \sin\alpha & -\cos\alpha & 0 \\ 0 & 0 & 1 \\ \cos\alpha & \sin\alpha & 0 \end{bmatrix} \left(\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} - \begin{bmatrix} x_R + r_{cam}\cos\alpha \\ y_R + r_{cam}\cos\alpha \\ z_R + z_{cam} \end{bmatrix} \right) \tag{5.1}$$

where (x_R, y_R, z_R) are the coordinates of the robot base centroid, while $(r_{cam}\cos\alpha, r_{cam}\sin\alpha, z_{cam})$ are the coordinates of the camera relative to the base centroid.

The origin of the image frame is located at the top left corner. Hence, the second transformation to the image frame is as follows:

$$\begin{bmatrix} u' \\ v' \end{bmatrix} = \begin{bmatrix} -f_x\dfrac{x_c}{z_c} + c_x \\ -f_y\dfrac{y_c}{z_c} + c_y \end{bmatrix} \tag{5.2}$$

$$= \begin{bmatrix} -f_x\tan\beta_i + c_x \\ -f_y\tan\gamma_i + c_y \end{bmatrix} \tag{5.3}$$

where f_x, f_y are the vertical and horizontal focal lengths of the camera and c_x, c_y are the horizontal and vertical coordinates of the image centroid.

The image space has pixel values within the range $1 \leq u' \leq 2c_x - 1\,, u' \in \mathbb{N}$ and $1 \leq v \leq 2c_y - 1\,, v \in \mathbb{N}$. Hence this transformation is defined only for $|f_x\dfrac{x_c}{z_c}| < c_x - 1$, $|f_y\dfrac{y_c}{z_c}| < c_y - 1$, which is called the camera's *field of view*. A point in camera coordinates that does not lie within this field of view is not visible to the camera. The concept of multi-rate measurements arises due to a limited field of view of the camera.

5.1.2 Choice of Measurement

The raw measurements u' and v' are available to the camera for a feature as long as it is visible and matched correctly. These coordinates can be transformed to observed in-plane bearing angle β_i and elevation angle γ_i respectively. However, for this set of simulations, the conversion to the relative angles observed for a feature are used as measurements for the EKFs. We only use β_i for outlier rejection and to update the predicted estimate. However the use of β_i and γ_i from every observed feature would enable more accurate outlier rejection and arguably make the estimate bounds tighter.

5.2 Gazebo Model & Environment

Gazebo possesses the ability to create 3-dimensional dynamic environments and mimic real-world physics to a sufficiently realistic degree [47]. The choice of environment is

crucial for any vision-based task [48] since a lot of computer vision algorithms pose advantages towards achieving specific goals in specific situations. Depending on criteria such as feature density and how far the features are from the action space, considerations are made for computational cost and accuracy.

The environment chosen for this work is planar for the following reasons:

- A planar environment permits repeated feature detection for an interest point as well as accurate feature matching for a larger field of view than a non-planar environment.
- The scale of a feature cannot be determined from a single view. With one view available from the image, D_i^* can be found for a feature only when it is in the field of view and matched correctly. Real-time vision can entail incorrect data association here since lack of knowledge of the landmark position implies that outlier rejection methods used later in this work cannot be used here. However for a planar environment, *a-priori* knowledge of the scale of the environment and the features solves the problem of scale by determining one coordinate.

8 textured posters are inserted on 4 walls such that a sufficiently field of view is covered. The the square room created by the wall is 10 m in size. A 360 panoramic view of the surroundings is available to the robot in the form of concatenated images taken at the reference position with various orientations. For this world, the panoramic image is provided in Figure 5.2.

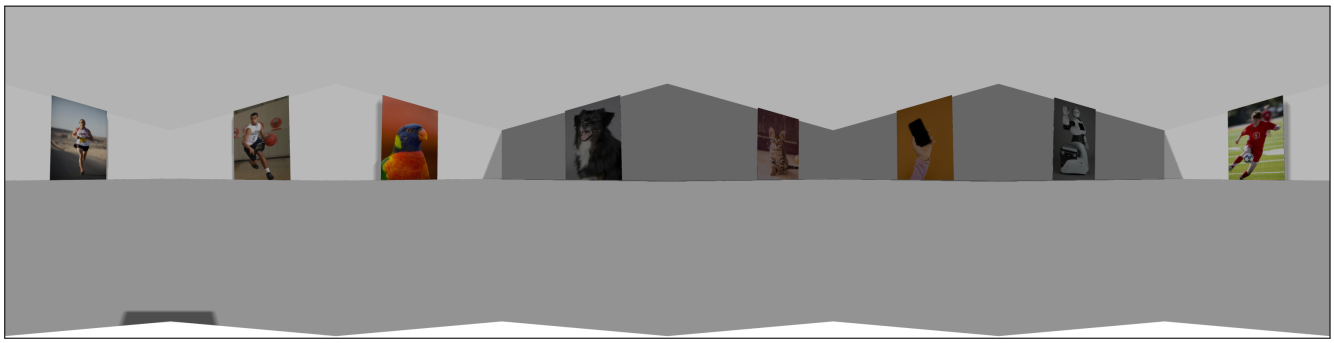


Figure 5.2: Panoramic Image available from Reference Position

Apart from the posters, the wall edges and corners are also detected as feature points, as as the visible perimeter of the robot base. The quantity and quality of the former,

in terms of repeatability and strength, depends on the type of feature detector used. However any feature belonging to the latter category is eliminated from consideration in feature matching.

5.3 Vision Pipeline

5.3.1 Feature Detection & Extraction

A simple vision pipeline is used to enable the camera to detect previously identified features. These features are initialized after being detected from the panoramic image shown in Figure 5.2. The choice of feature detector used is FAST, and the description of these features are stored in SURF. This combination requires much less computation time than that while using SIFT [49], while it detects many more interest points than ORB. However, it is less robust to projective transformations, hence when the robot is close to the wall the interest point may not be detected as a feature. The problem of detecting multiple features in adjacent locations is mitigated by enabling non-maximal suppression, which picks the 'strongest' features from 16 surrounding pixels [50]. The threshold value that the FAST algorithm takes as a parameter also allows tuning of the number of features extracted from the environment.

5.3.2 Feature Matching

The first step towards data association is not a maximum likelihood approach, as there is valuable information captured from the interest points using which can be used to obtain an initial hypothesis. Hence, a suitable matching algorithm is used to find the nearest neighbours of the live feed image feature descriptors, from the set of feature descriptors obtained from the panoramic image. The feature matcher used is the Brute Force Matcher, as it is more accurate in performance than FLANN Matching and for the number of features detected in the environment, the computation time for both matching algorithms is comparable. kNN Matching is used to get the best 2 neighbours for a query image feature, after which Lowe's ratio test [51] is used for $r = 0.65$ to choose the accurate matches.

The detection of all the scenes features for a scene is not guaranteed. This is because for the Gazebo camera model, the pixel values of each point on the live feed do not

remain constant in time, even for a static robot. As a consequence, the number of matches between the live feed and the panoramic image vary from time to time even for a still robot. However, large deviations from the norm do not occur and number of matches for various robot poses do not vary beyond a small bound.

5.3.3 Outlier Rejection

This work uses the Individual Compatibility test for nearest neighbours (ICNN) test to reject spurious matches by using the predicted measurement and the covariance of the innovations from the filter, creating a gate for measurements. Let p_v be the number of initial matches, while q denotes the number of matches accepted by the outliter rejection step. An initial hypothesis $\mathcal{H} = \{j_1, j_2, \dots j_{p_v}\}$ is generated from the feature matching algorithm, based on nearest neighbour distances of the feature descriptors for the training and query image. This hypothesis creates the measurement y_i to the landmark L_{j_i}. The likelihood of this match can be determined using the Mahalonobis distance:

$$M_{ij} \;=\; (y_i - \hat{y_{j_i}}) S^{-1} (y_i - \hat{y_{j_i}})^{\mathsf{T}} \tag{5.4}$$

where, at the k^{th} time instant, $S_k = C_k P_{k|k-1} C_k^{\mathsf{T}} + R_k$. Since the innovation $\upsilon_{ij} = (y_i - \hat{y_{j_i}})$ is χ^2 distributed, for a confidence interval α, the hypothesis H_i holds true if:

$$M_{ij} \;=\; \upsilon_{ij} S^{-1} \upsilon_{ij}^{\mathsf{T}} \;<\; \chi^2_{\alpha,d} \tag{5.5}$$

where $y_i \in \mathbb{R}$.

The match is termed as a 'correct match' henceforth in the chapter if for that particular hypothesis, Equation 5.5 holds true. Hence a 'correct match' may indeed be a false positive if a hypothesis incorrect in reality passes the aforementioned test.

Since for these set of simulations we only take one measurement from each landmark, β_i, $d = 1$. We require a 99% confidence interval for the square of the Mahalonobis distance be less than the corresponding value of the inverse of the χ^2 cumulative distribution function. If M_{ij} does not lie within these bounds, H_{ij} is deemed false and y_i is not included in the final list of measurements.

The individual compatibility test lends itself to a simple test once a hypothesis has been established. However Neira & Tardos [52] demonstrate the shortcomings of using individual compatibility for a nearest neighbours search.

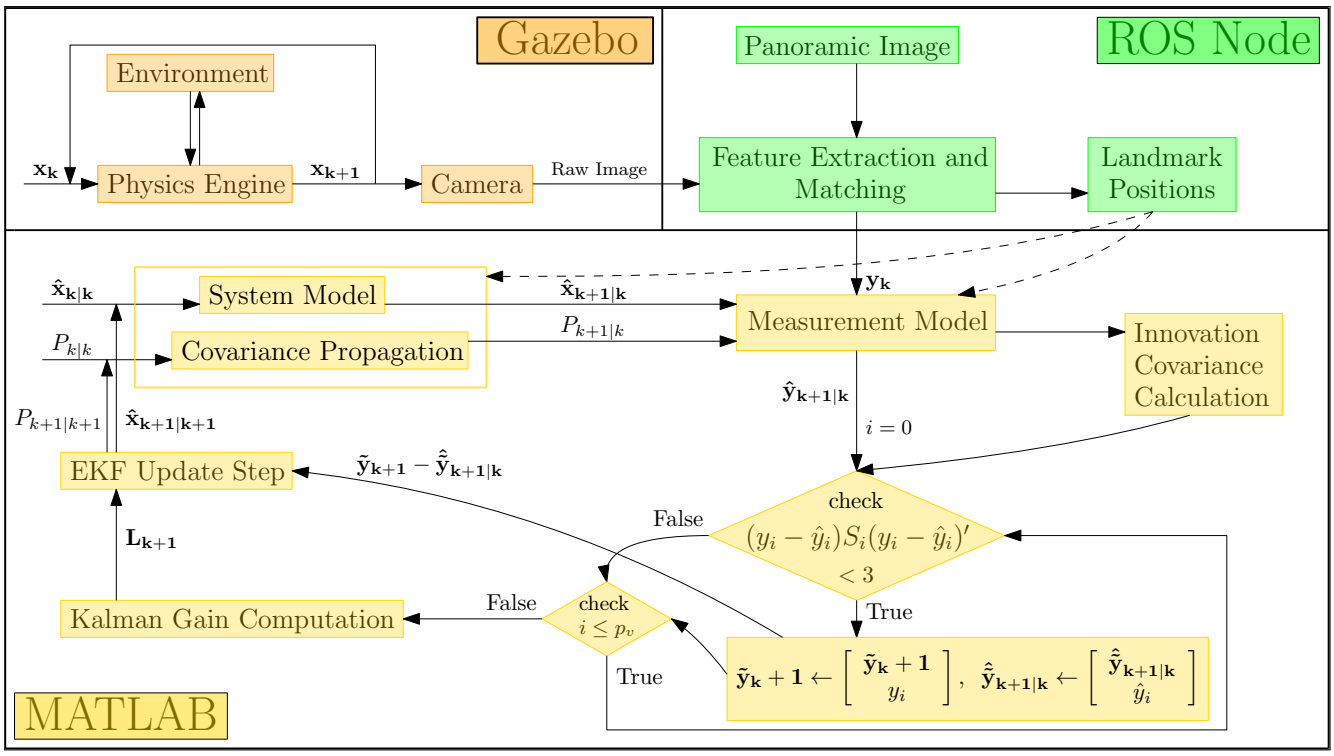


Figure 5.3: Flowchart for Gazebo-based Simulations

5.4 Simulations for Accurately Known Landmark Positions

We discuss the simulations performed when the panoramic image is able to capture landmark positions accurately. The simulations are performed with the computation distributed between a C++ ROS node and MATLAB. The flowchart for the simulation is provided in Figure 5.3.

The same environment and panoramic image is used for all the simulations in this chapter. The environment is completely static. Further details regarding the environment and tuning parameters for the simulations are provided in Table 5.1. The internal parameters of the camera are provided in Table 5.2, while Table 5.3 contains the important dimensions of the robot.

Simulation Parameters	Value
Constant Velocity (V)	0.1 m/s
Steering Angular Velocity (ω)	0.1 rad/s
Wheel Velocity Noise	$\mathcal{N}(0, 0.001)$
Measurement Noise	$\mathcal{N}(0, 0.0025)$
Total Number of Features (p)	695

Table 5.1: Parameters for Gazebo Simulation

Camera Parameters	Value
Camera field of view	$\pi/2$
Horizontal Focal Length (f_x)	399.998
Vertical Focal Length (f_y)	399.998
Horizontal Offset (c_x)	400.5
Vertical Offset (c_x)	400.5

Table 5.2: Camera Parameters

Robot Dimensions	Value
r_{cam}	0.125 m
z_R	0.4 m
z_{cam}	0.0725 m
Lateral wheel separation	0.3 m

Table 5.3: Robot dimensions

Due to the nature of the code, the real time factor of Gazebo is kept at 0.02 for this simulation. The EKF for augmented state takes significantly longer to complete one iteration than that for the traditional state, hence the sampling time is limited to 11Hz (in Gazebo time). The figure on the left is the live feed, while matches are drawn with randomly coloured horizontal lines

5.4.1 Traditional State + EKF

The results of an EKF used as an observer for the traditional bearing-only state space formulation can be seen in Figures 5.4, 5.5, 5.6 and 5.7:

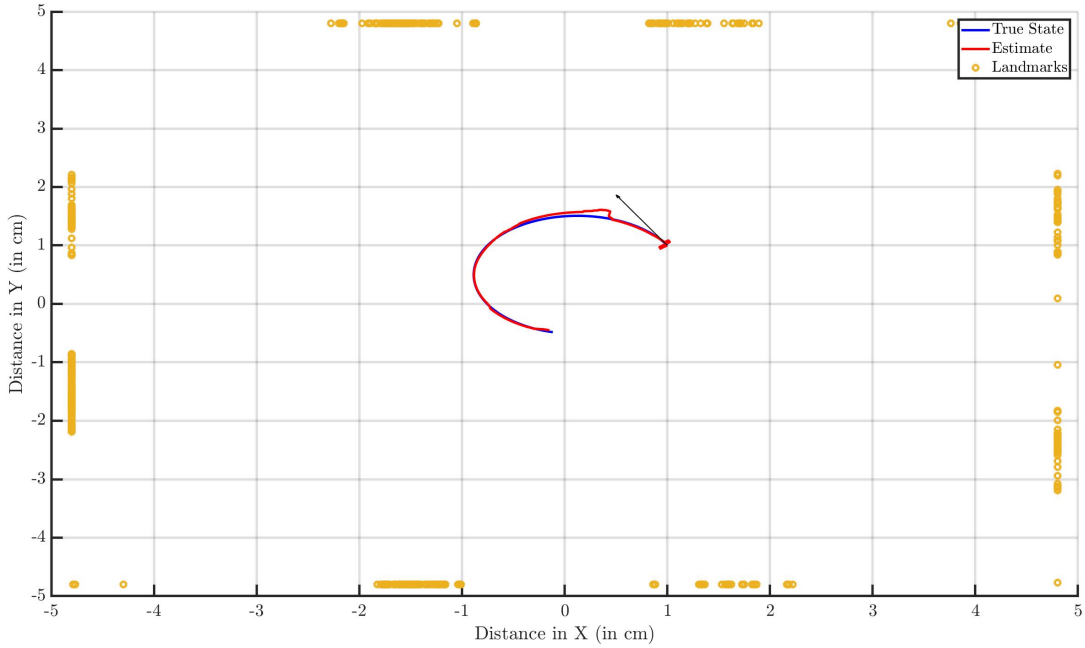


Figure 5.4: 2-D Trajectory of True State & Estimate for Traditional State + EKF

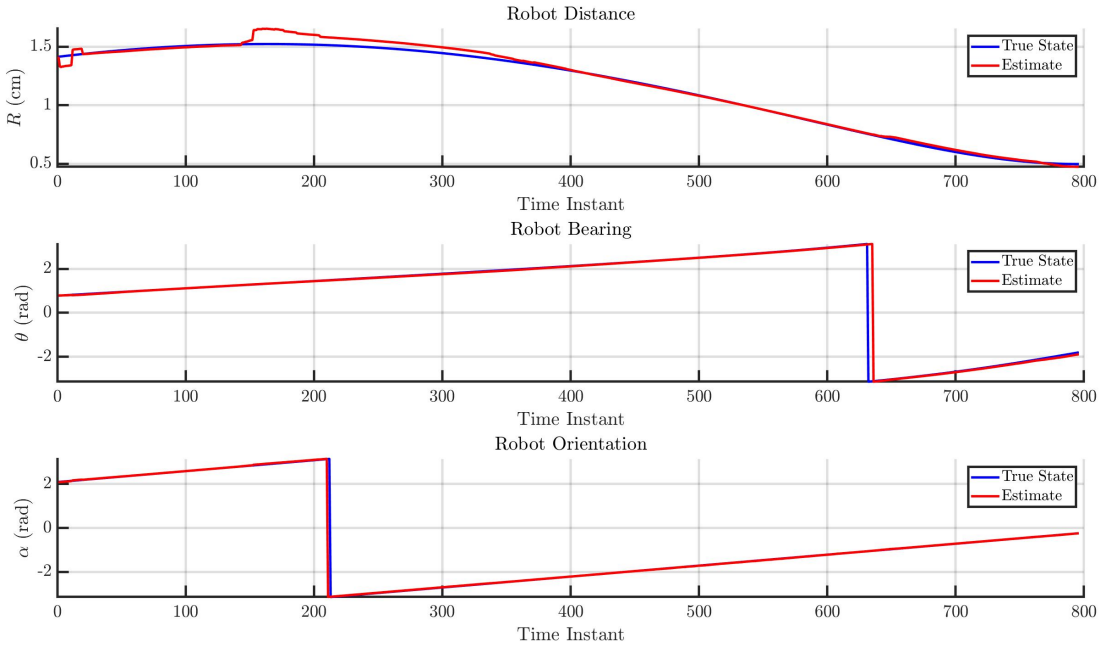


Figure 5.5: Pose Estimation for Traditional State + EKF

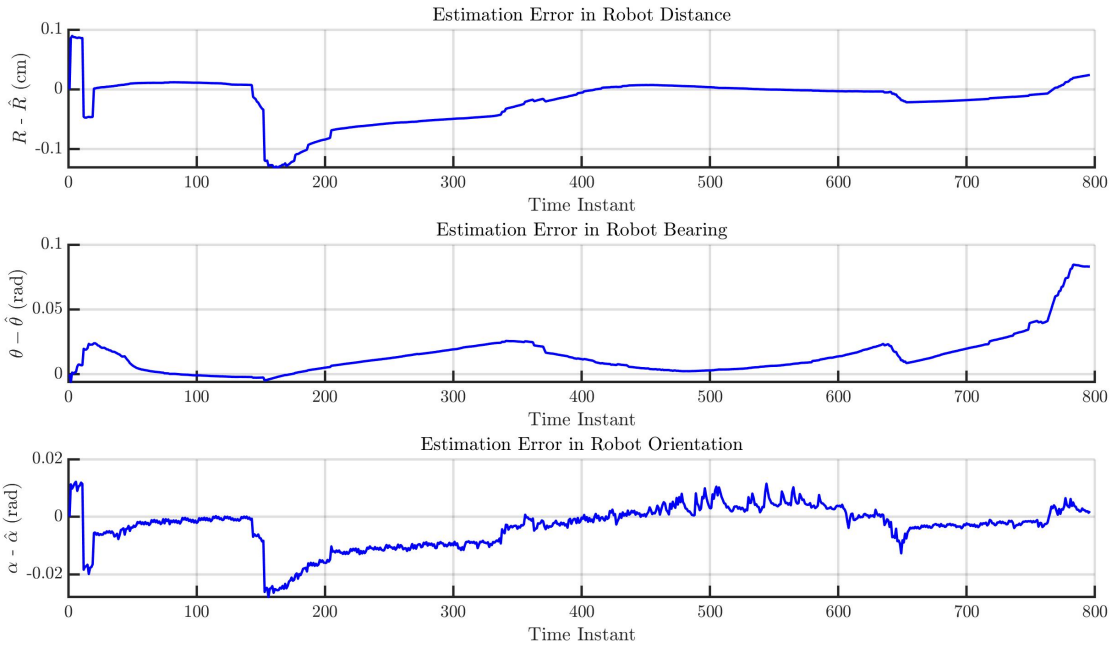


Figure 5.6: Pose Estimation Error for Traditional State + EKF

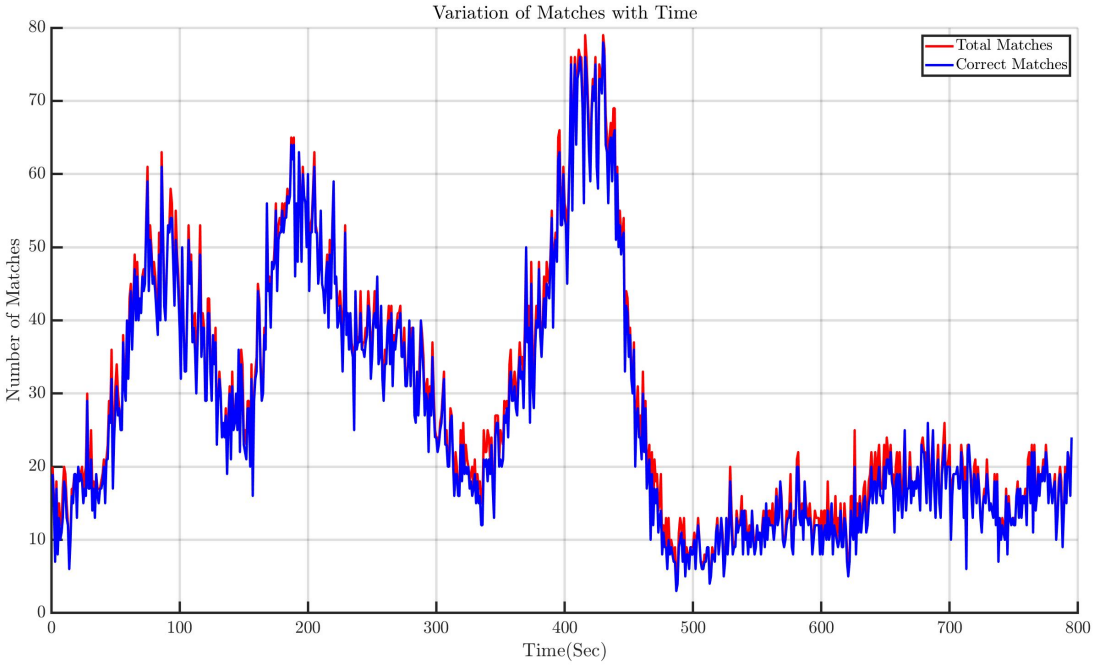


Figure 5.7: Total & Correctly Matched Features for Traditional State + EKF

Figure 5.8 shows the location of the landmarks as known to the system. Figure 5.9 shows the matched features at the first time instant, with the live feed on the left and the panoramic image on the right. Since for all the simulations in this section the pose at $t = 0$ is the same, these features are available to the robot in all the simulations and the number of matches roughly remain the same.

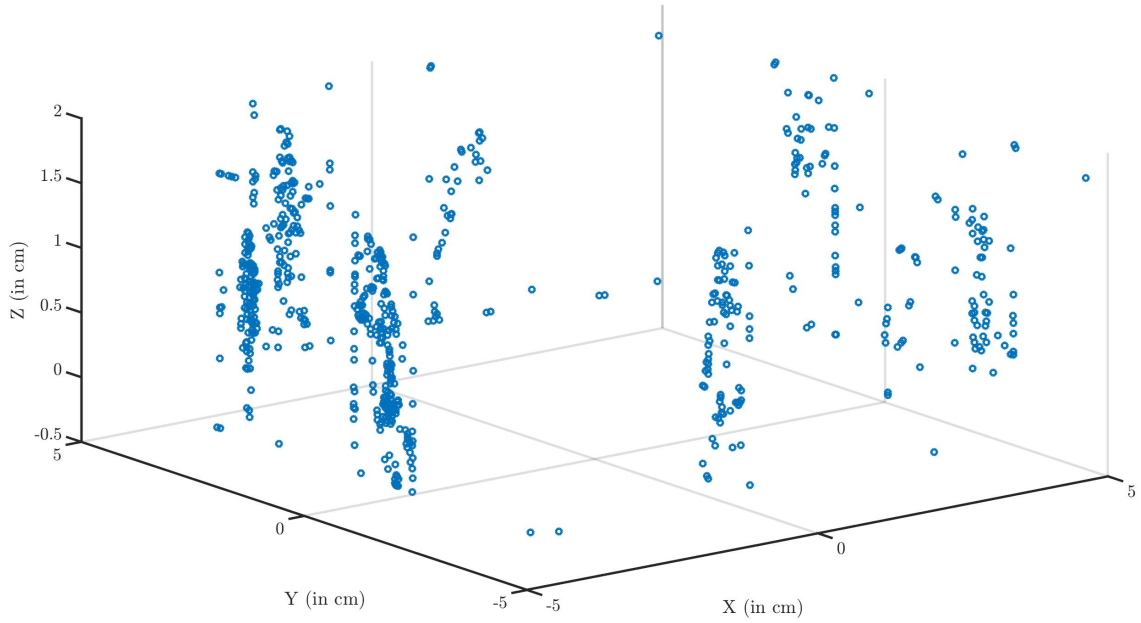


Figure 5.8: 3-D Location of Landmarks as known to System

Figure 5.9: Matched Features at Initial Time Instant

The ratio of correct matches to total matches is very high. This implies that most of the correct feature matches are passing the outlier rejection test. The estimates are not divergent in nature, however jumps are observed at certain time instants in the simulation.

5.4.2 Augmented State + EKF

The results of an EKF simulated for the augmented state space system can be seen in Figures 5.10, 5.11, 5.12 and 5.13:

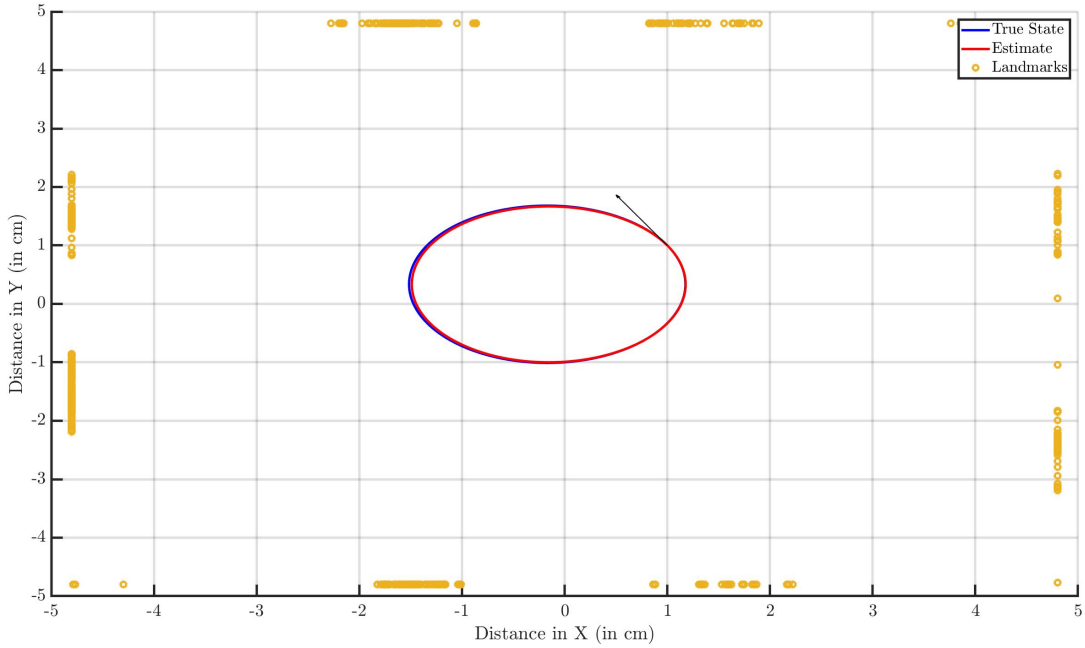


Figure 5.10: 2-D Trajectory of True State & Estimate for Traditional State + EKF

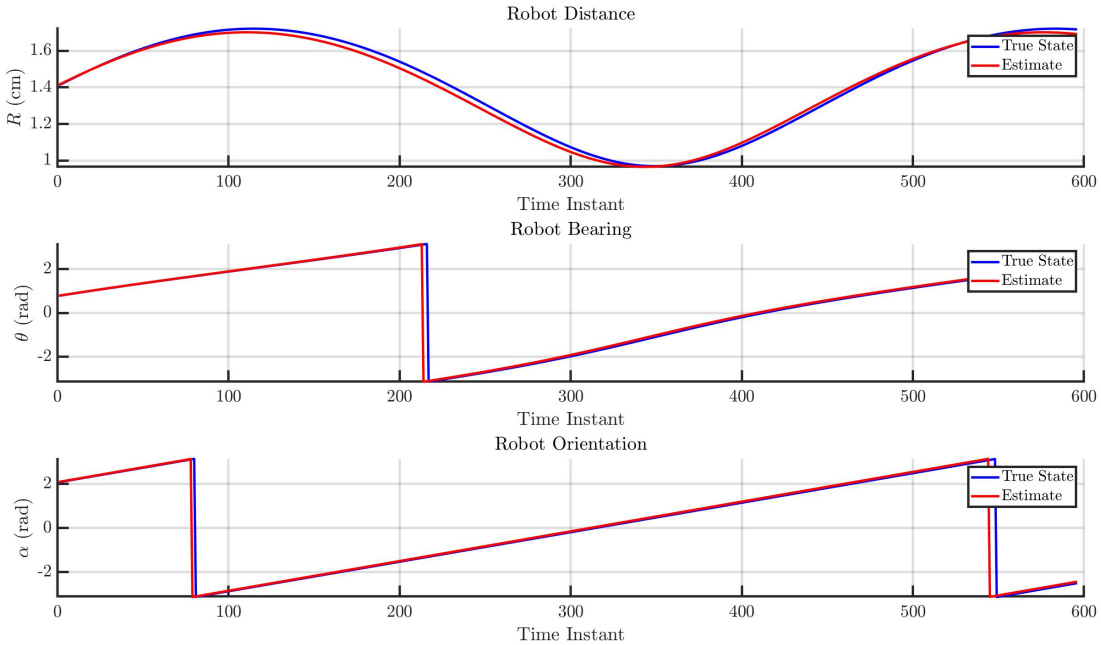


Figure 5.11: Pose Estimation for Traditional State + EKF

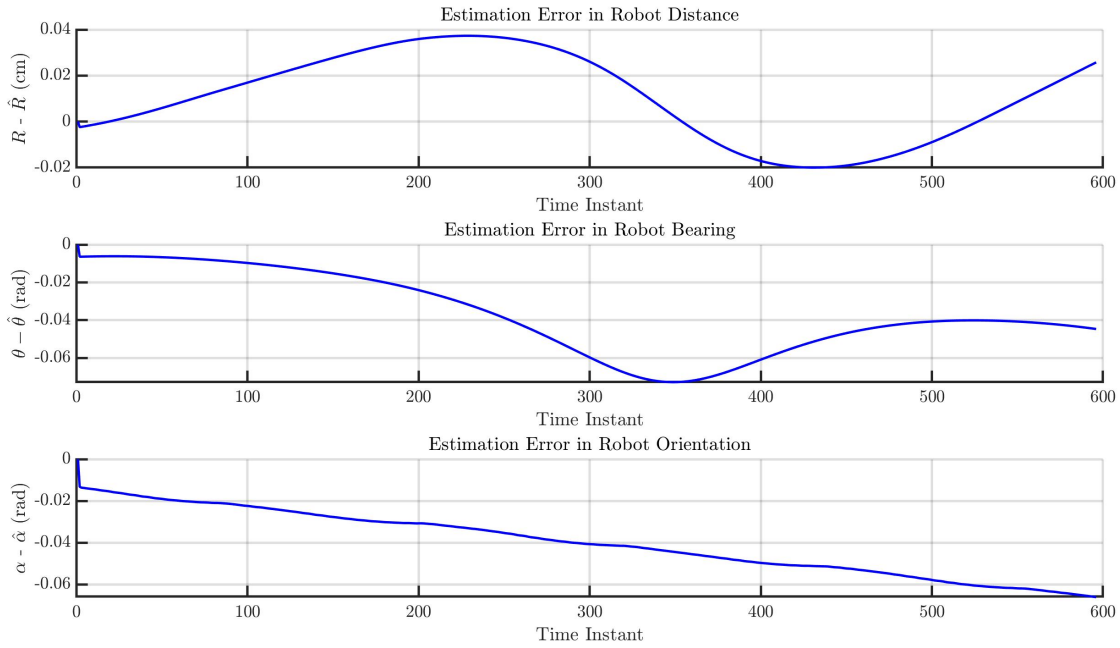


Figure 5.12: Pose Estimation Error for Traditional State + EKF

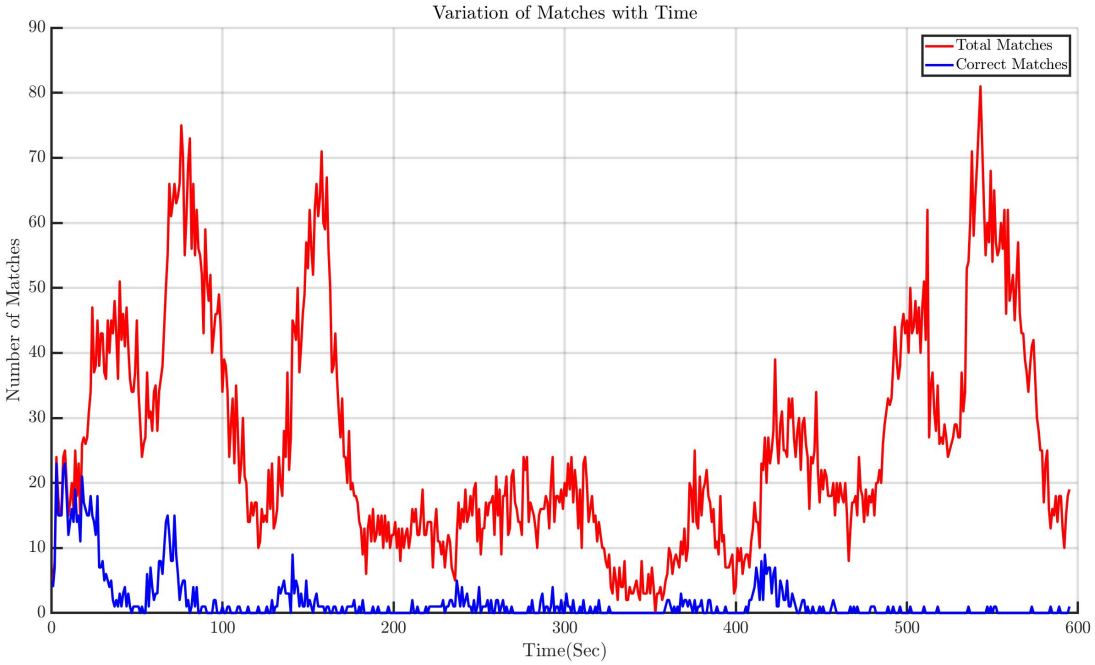


Figure 5.13: Total & Correctly Matched Features for Traditional State + EKF

As discussed before, the pose estimation error remains low when the augmented state space formulation is used and the error in in the initial state is low. Hence, for this simulation, the EKF used with the augmented state outperforms the EKF used with the traditional state. However, the filter rejects measurements after some time into the simulation. This is because of the low variance values of β_i in the augmented state covariance matrix, which increases the Mahalonobis distance of the innovations. As the trajectory taken by the robot is the same, a lot of correctly matched features are falsely rejected by the filter after some time into the simulation.

5.5 Simulations for Inaccurately Known Landmark Positions

This section discusses simulations where the system does not possess the correct landmark positions. A comparative study is conducted between the EKF-SLAM technique and the augmented state used with the Q-EKF. The results are then compared below, in terms of estimation error and matching accuracy. The enormous sizes of the state vectors for both cases require the simulations to be run at 2 Hz while Gazebo runs at the same real time factor of 0.03. The knowledge of D_i^*'s and β_i^*'s are corrupted by random zero-mean uniform noise with ranges 0.8 m and 0.15 rad respectively.

The state vector sizes for EKF-SLAM and the Q-EKF paired with the augmented states are $2p + 3$ and $p + 3$ and for these set of simulations, according to Table 5.1, $p = 695$. Hence the latter runs much faster than the former, however the comparison is performed at equal sampling times.

5.5.1 EKF-SLAM

The simulation results for EKF-SLAM for the conditions given in Tables 5.1, 5.2 and 5.3 are given below in Figures 5.14, 5.15, 5.16 and 5.17.

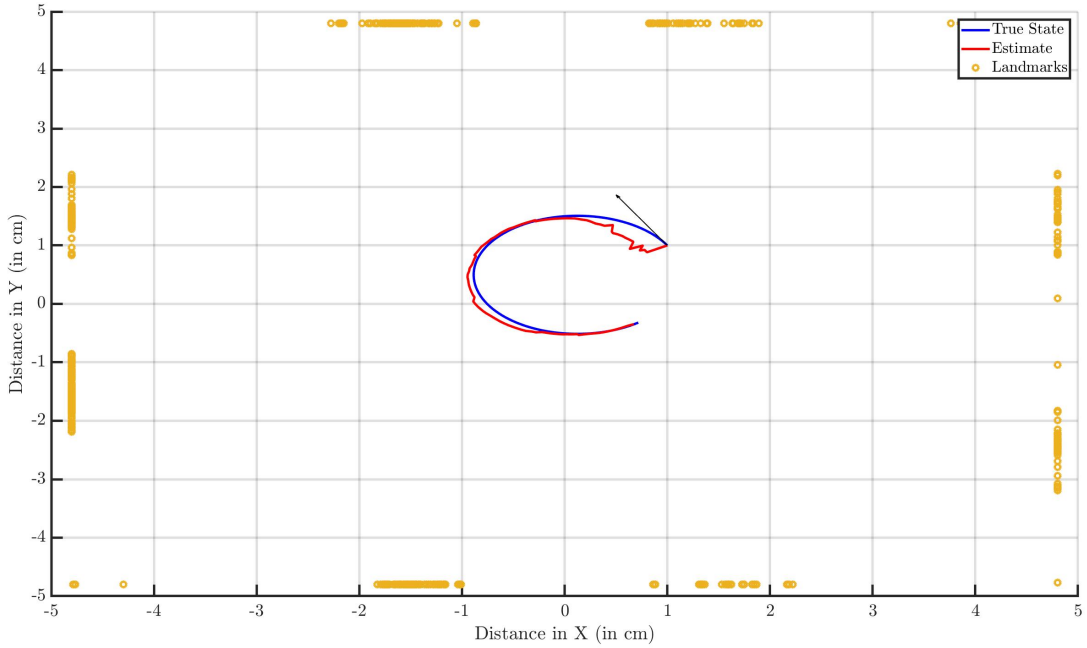


Figure 5.14: 2-D Trajectory of True State & Estimate for EKF-SLAM

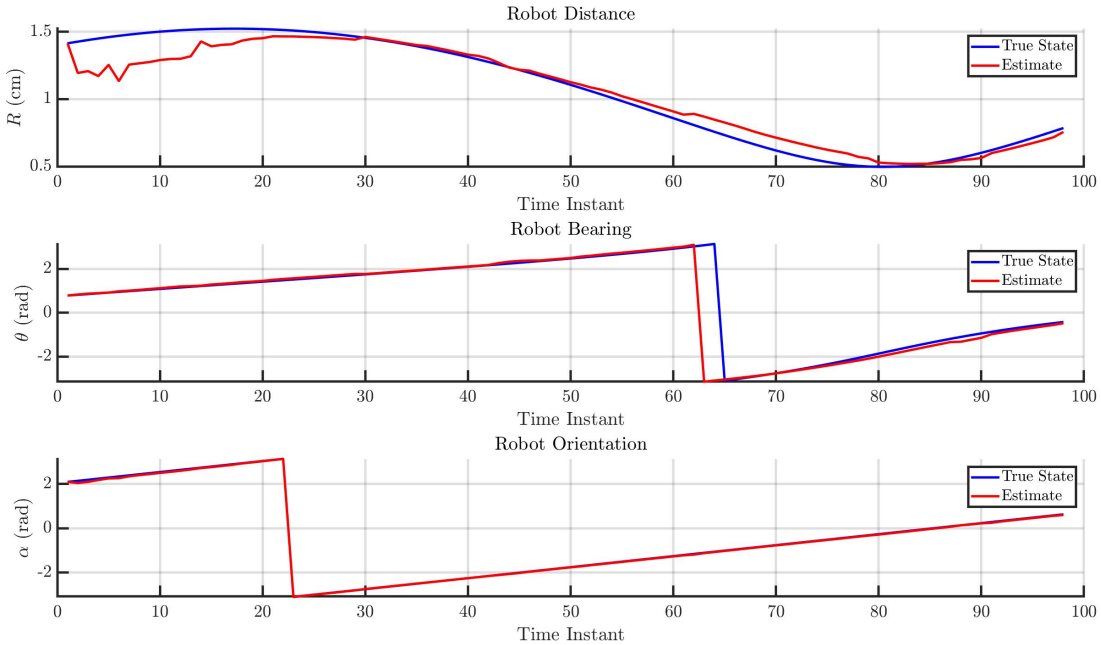


Figure 5.15: Pose Estimation for EKF-SLAM

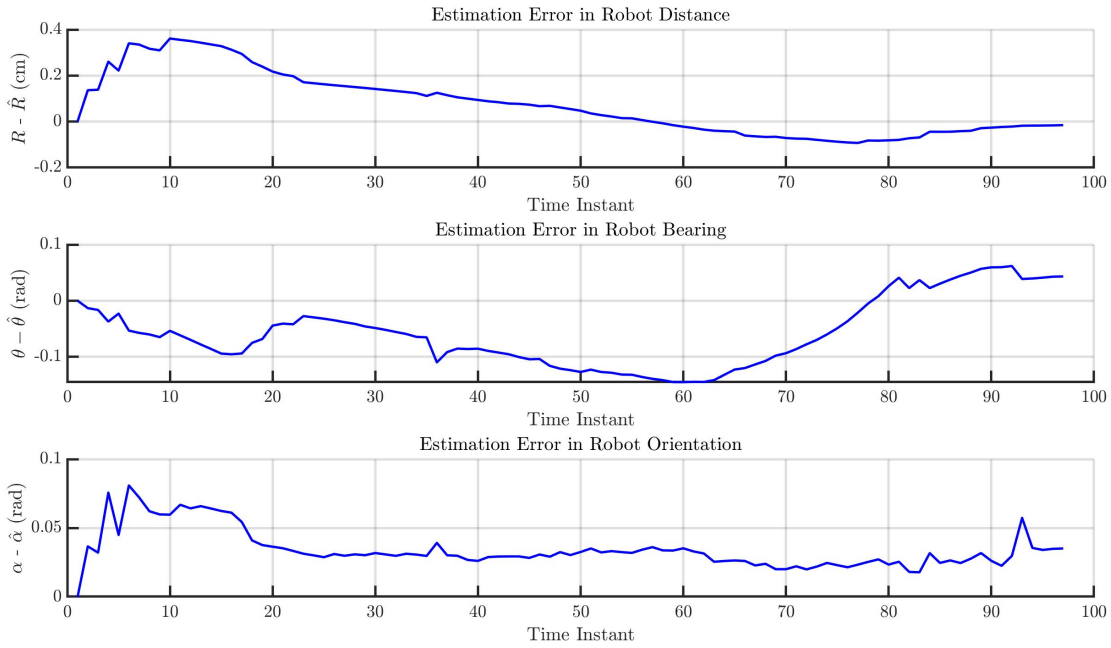


Figure 5.16: Pose Estimation Error for EKF-SLAM

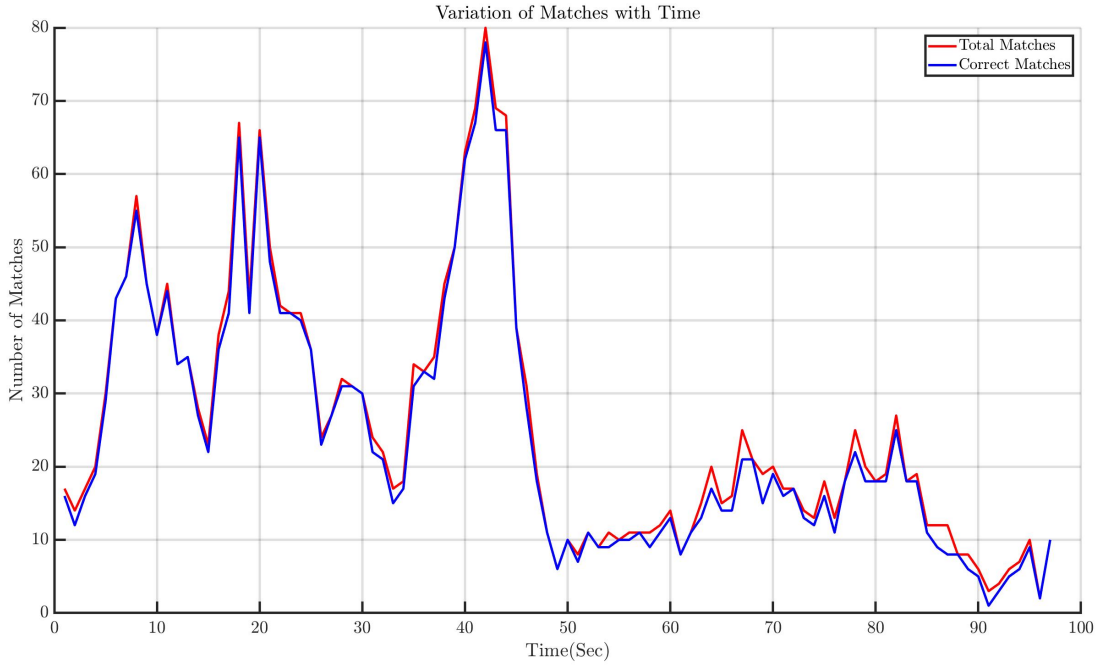


Figure 5.17: Total & Correctly Matched Features for EKF-SLAM

The pose estimates initially diverge but then remain in proximity of the true state. The feature matching has a very high ratio of correct matches to total matches as well.

5.5.2 Augmented State + Q-EKF

The augmented state is simulated with the Q-EKF as an observer. The simulations are run for two choices of $Q_{D^*} + Q_{\beta^*}$: $0.01 I_{n \times n}$ and $0.1 I_{n \times n}$. The results for the former tuning value are demonstrated in Figures 5.18, 5.19, 5.20 and 5.21

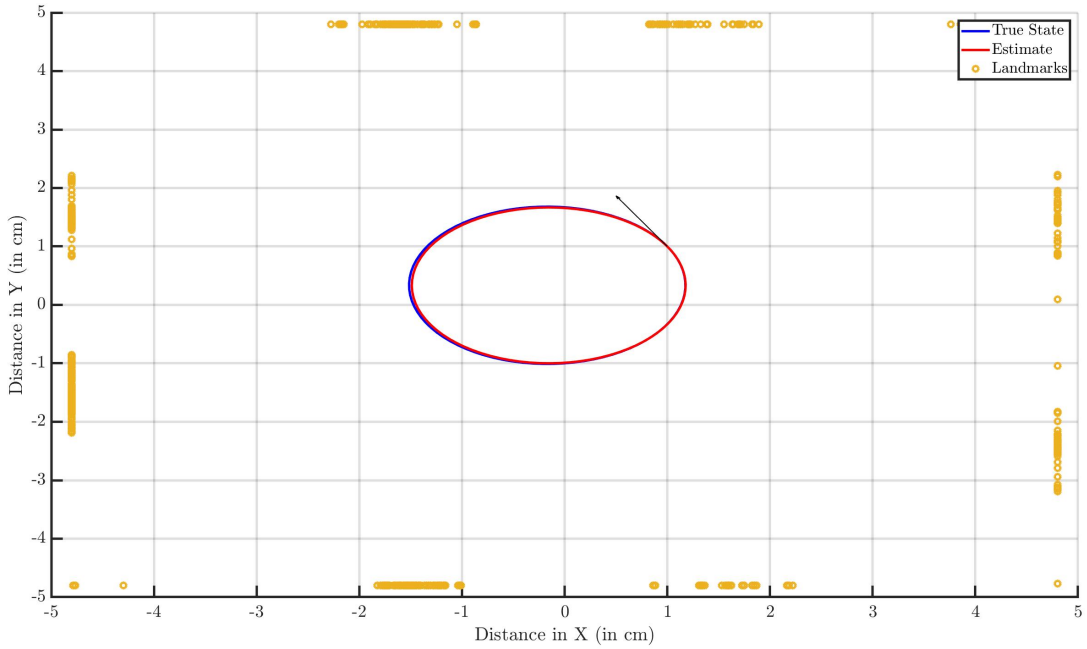


Figure 5.18: 2-D Trajectory of True State & Estimate for Augmented State + Q-EKF

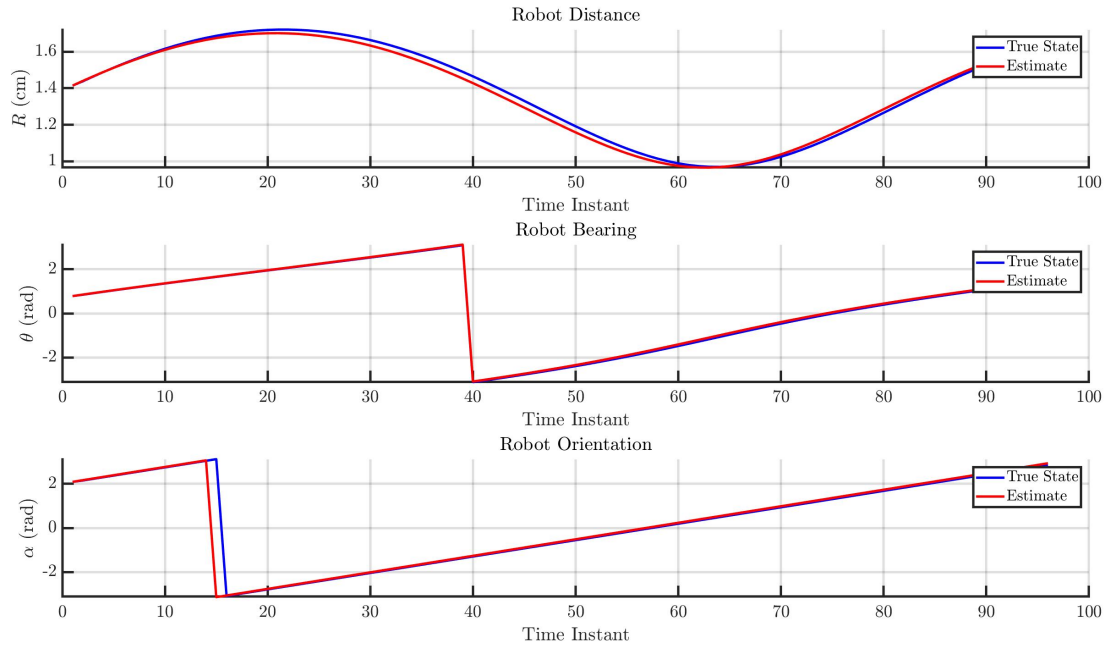


Figure 5.19: Pose Estimation for Augmented State + Q-EKF

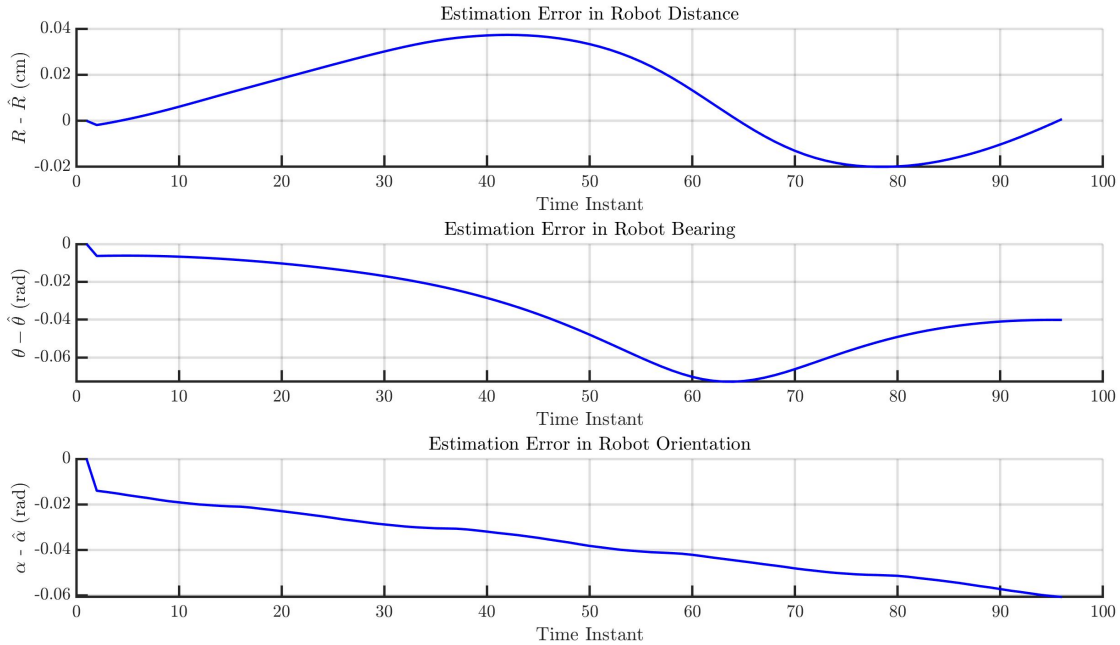


Figure 5.20: Pose Estimation Error for Augmented State + Q-EKF

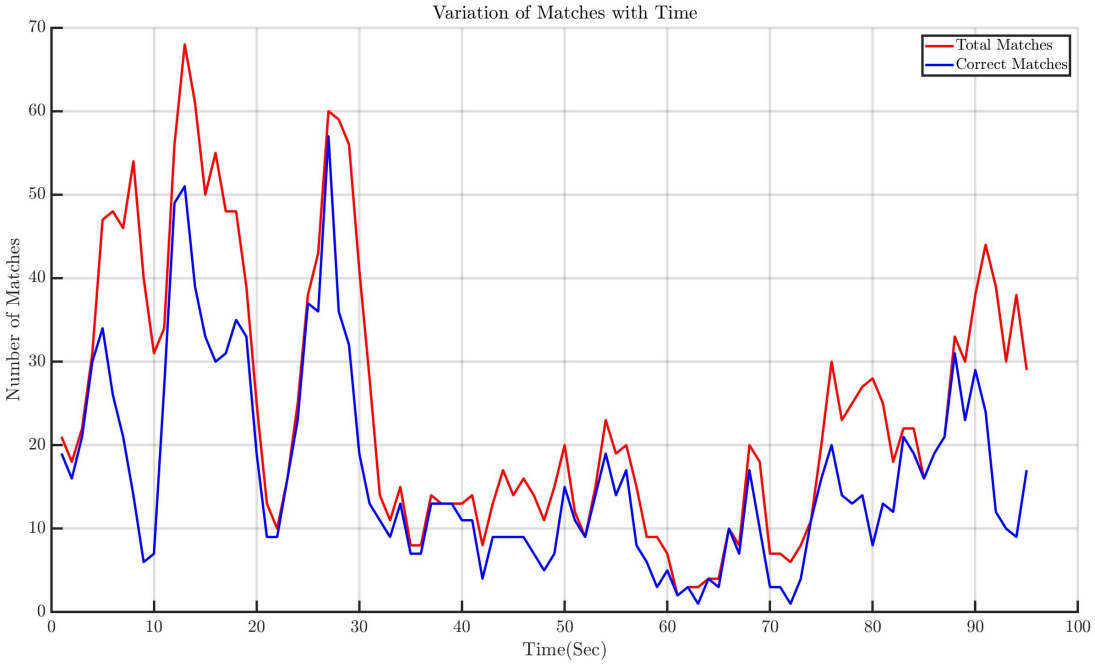


Figure 5.21: Total & Correctly Matched Features for Augmented State + Q-EKF

The Q-EKF clearly demonstrates superior performance in terms of estimation error. However, the extra benefit of the added process noise covariance matrix is that it boosts the diagonal values of β_i's. Hence the outlier rejection gate gets wider and more correct measurements are accepted than in Figure 5.13. Thus the addition of $Q_{D^*} + Q_{\beta^*}$ in the covariance propagation step is beneficial in two ways.

The performance of the augmented state + Q-EKF is compared for the two aforementioned values of the tuning parameter $Q_{D^*} + Q_{\beta^*}$. While the estimation errors are practically identical, there is a big difference between the proportion of correct matches. The identical estimation errors also imply that none of the additional correct matches obtained by increasing the covariance matrix diagonal values are in reality false correspondences.

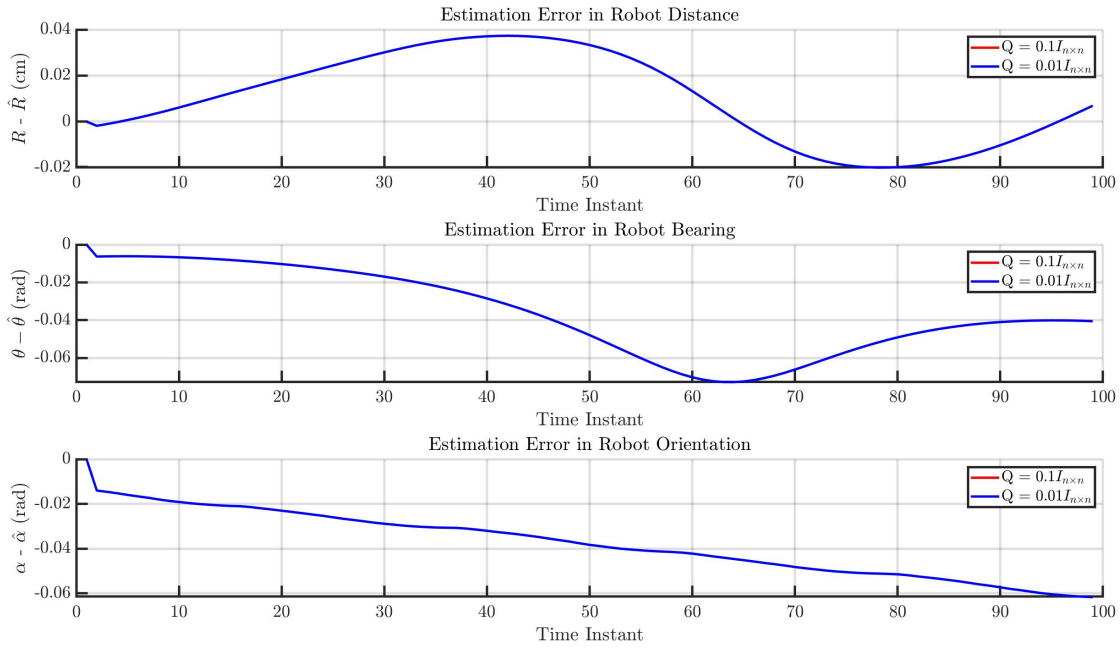


Figure 5.22: Estimation Error for different values of $Q_{D^*} + Q_{\beta^*}$

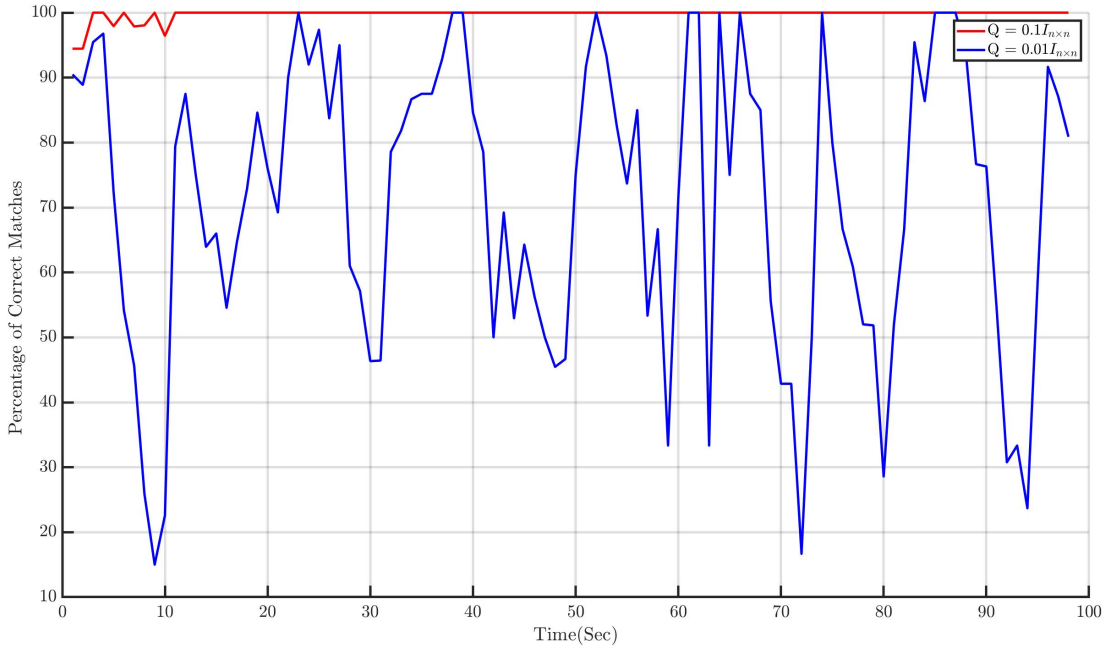


Figure 5.23: Percentage of Feature Matches for different values of $Q_{D^*} + Q_{\beta^*}$

5.6 Summary

In this chapter, the efficacy of the two state space models are compared for a robot model in a Gazebo environment. The addition of a vision pipeline and an outlier rejection step in the filter demonstrates performance in more realistic settings. For a low error in the initial state estimate, the augmented state space model with an appropriate choice of filter works much better than the traditional state space model, regardless of the accuracy of the robot's knowlegde of the landmark positions in the environment. The better choice is evident if the knowledge is not accurate, however if it is, then the traditional formulation does not require any parameter estimation. In that case, the augmented state space formulation is much more computationally expensive. Hence the trade-off lies in the quantification of the error in the information available to the robot, in terms of the position of features in the environment.

Chapter 6

Conclusion & Future Work

This chapter concludes the study on bearing-only localization in uncertain environments. The avenues for further work are discussed and the opportunities and potential pitfalls of each research direction are briefly stated.

6.1 Conclusion

This study deals with two different sub-problems of the bearing-only localization problem, in four parts. The first part establishes the concept of observability for piecewise constant as well as analytic inputs. It uses symbolic computation to calculate the kernel basis for the observable space for various joint state and parameter estimation problems. Lower bounds on the number of landmarks required for such formulations to achieve maximal observability rank, are established. The basis vectors are then interpreted to yield a physically intuitive loci of initial states that are degenerate in their input-output maps. The second part validates the observability results by presenting Kalman decomposition-like transformation for joint state and parameter formulations for $p = 1, 2$. The convergence of state estimate is demonstrated for the observable states and the transformed unobservable states reinforce the intuition derived from the results in the previous part.

The third part discusses the problem of robust localization for environments where information about landmarks is not accurate. The emphasis is placed solely on retaining accurate estimates and not on map building. Towards that end, a disturbance rejection technique is proposed, consisting of a filter with added process noise covariance (the Q-EKF) used as an observer for the augmented state space formulation. Such a technique

mitigates estimation error divergence but is incapable of correcting any initial errors. The PI-Q-EKF uses an integral gain to eliminate this bias. To ensure unbiased estimation errors, structure for the proportional and integral gains is provided with reasoning for the same. The fourth part carries these simulations onto a ROS-Gazebo environment for a mobile wheeled robot with a camera. The effects of the former two techniques working alongside a compatibility test for outlier rejection are elaborated upon.

6.2 Future Work

Several research directions from this point forward offer considerable promise. Some of them are listed below:

- **Extension to Semi-Static and Dynamic Landmarks**: This work can be extended to semi-static or dynamic landmarks by characterizing the intermittent shift or motion in position respectively and accounting for their dynamics in the system equations appropriately.

- **Generalizing Coordinate Transformation for p Landmarks**: The transformations provided in Chapter 4 are only applicable for $p = 1, 2$. A different set of transformations is required that can be generalized to an arbitrary number of landmarks, for various joint state and parameter estimation problems. Isidori [36] proves that such a transformation exists. The consequence would be further validation of observability results for $p > 2$.

- **PI-Q-EKF Gain Structure for Multi-Rate Measurements**: In Chapter 5, the gains of PI-EKF were constant throughout the simulation, as was the number of landmarks observed at each time instant. However, with intermittent observations available from each landmark in a setting such as in Chapter 6, the gains must be tuned for each trajectory to account for the variation in the number of correctly matched features throughout the simulation. However, the effect of a constant gain structure for a multi-rate structure is unknown as of now. It is also possible that constant proportional and integral gains may not yield estimator convergence. In that scenario, calculation of adaptive gains such as in [53] can be looked into as a possible solution.

- **Extending Gazebo Simulations to Non-Planar Environment**: As discussed before, the planarity combined with previously known information about environment scale helps us identify the coordinates of all the features in the environment. In a situation where prior information about surroundings is not available, such as in a non-planar environment, multiple view geometry can help us find the scale of landmarks that are being observed at a given time instant. However, more robust vision techniques such as active feature search [54] are required to ensure more suitable data association techniques for landmarks on 3-D objects. The concept of initialization, such as in contemporary SLAM, would also be crucial because the quantities used to determine scale (such as robot pose) are probabilistic nature, hence requiring a number of time instants before the scale can be bound within a tight interval.

Bibliography

[1] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Transactions on Robotics and Automation*, vol. 8, pp. 313–326, June 1992.

[2] R. G. Stansfield, "Statistical theory of d.f. fixing," *Journal of the Institution of Electrical Engineers - Part IIIA: Radiocommunication*, vol. 94, no. 15, pp. 762–770, 1947.

[3] D. J. Murphy, *Noisy bearings-only target motion analysis*. PhD thesis, Northeastern University, 1970.

[4] V. J. Aidala, "Kalman Filter Behavior in Bearings-Only Tracking Applications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-15, no. 1, pp. 29–39, 1979.

[5] S. G. Loizou and V. Kumar, "Biologically inspired bearing-only navigation and tracking," in *2007 46th IEEE Conference on Decision and Control*, pp. 1386–1391, 2007.

[6] M. Liu, C. Pradalier, Q. Chen, and R. Siegwart, "A bearing-only 2D/3D-homing method under a visual servoing framework," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 4062–4067, IEEE, 2010.

[7] F. Chenavier and J. L. Crowley, "Position estimation for a mobile robot using vision and odometry," in *Robotics and Automation, 1992. Proceedings., 1992 IEEE International Conference on*, pp. 2588–2593, IEEE, 1992.

[8] P. Zingaretti and E. Frontoni, "Vision and sonar sensor fusion for mobile robot localization in aliased environments," in *2006 2nd IEEE/ASME International Conference on Mechatronics and Embedded Systems and Applications*, pp. 1–6, 2006.

[9] T. Krajník, J. Faigl, V. Vonásek, K. Košnar, M. Kulich, and L. Přeučil, "Simple yet stable bearing-only navigation," *Journal of Field Robotics*, vol. 27, no. 5, pp. 511–533, 2010.

[10] R. Smith, M. Self, and P. Cheeseman, "A stochastic map for uncertain spatial relationships," pp. 467–474, 01 1987.

[11] J. Gutmann and K. Konolige, "Incremental mapping of large cyclic environments," in *Proceedings 1999 IEEE International Symposium on Computational Intelligence in Robotics and Automation. CIRA'99 (Cat. No.99EX375)*, pp. 318–325, 1999.

[12] Il-Kyun Jung and Lacroix, "High resolution terrain mapping using low attitude aerial stereo imagery," in *Proceedings Ninth IEEE International Conference on Computer Vision*, pp. 946–951 vol.2, 2003.

[13] Jong-Hyuk Kim and S. Sukkarieh, "Airborne simultaneous localisation and map building," in *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, vol. 1, pp. 406–411 vol.1, 2003.

[14] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-Time Single Camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, 2007.

[15] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardós, "ORB-SLAM: A Versatile and Accurate Monocular SLAM System," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.

[16] J. Engel, J. Stückler, and D. Cremers, "Large-scale direct SLAM with stereo cameras," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1935–1942, 2015.

[17] A. Rosinol, H. Rebecq, T. Horstschaefer, and D. Scaramuzza, "Ultimate SLAM? Combining Events, Images, and IMU for Robust Visual SLAM in HDR and High Speed Scenarios," *IEEE Robotics and Automation Letters*, vol. PP, pp. 1–1, 01 2018.

[18] K. Reif, S. Gunther, E. Yaz, and R. Unbehauen, "Stochastic stability of the discrete-time extended Kalman filter," *IEEE Transactions on Automatic Control*, vol. 44, no. 4, pp. 714–728, 1999.

[19] F. M. Ham and R. G. Brown, "Observability, Eigenvalues, and Kalman Filtering," *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-19, no. 2, pp. 269–273, 1983.

[20] A. Martinelli, F. Pont, and R. Siegwart, "Multi-Robot Localization Using Relative Observations," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 2797–2802, 2005.

[21] A. Martinelli, N. Tomatis, and R. Siegwart, "Simultaneous localization and odometry self calibration for mobile robot," *Auton. Robots*, vol. 22, pp. 75–85, 01 2007.

[22] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "Observability-based Rules for Designing Consistent EKF SLAM Estimators," *The International Journal of Robotics Research*, vol. 29, no. 5, pp. 502–528, 2010.

[23] L. D. L. Perera, A. Melkumyan, and E. Nettleton, "On the linear and nonlinear observability analysis of the slam problem," in *2009 IEEE International Conference on Mechatronics*, pp. 1–6, 2009.

[24] T. Vidal-Calleja, M. Bryson, S. Sukkarieh, A. Sanfeliu, and J. Andrade-Cetto, "on the observability of bearing-only slam," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*.

[25] F. A. W. Belo, P. Salaris, D. Fontanelli, and A. Bicchi, "A Complete Observability Analysis of the Planar Bearing Localization and Mapping for Visual Servoing with Known Camera Velocities," *International Journal of Advanced Robotic Systems*, vol. 10, 2013.

[26] V. A. Bavdekar, R. B. Gopaluni, and S. L. Shah, "Evaluation of Adaptive Extended Kalman Filter Algorithms for State Estimation in Presence of Model-Plant Mismatch," *IFAC Proceedings Volumes*, vol. 46, no. 32, pp. 184 – 189, 2013. 10th IFAC International Symposium on Dynamics and Control of Process Systems.

[27] D. Varshney, M. Bhushan, and S. C. Patwardhan, "State and parameter estimation using extended Kitanidis Kalman filter," *Journal of Process Control*, vol. 76, pp. 98 – 111, 2019.

[28] P. K. Kitanidis, "Unbiased minimum-variance linear state estimation," *Automatica*, vol. 23, no. 6, pp. 775 – 778, 1987.

[29] S. I. Roumeliotis, G. S. Sukhatme, and G. A. Bekey, "Circumventing dynamic modeling: evaluation of the error-state Kalman filter applied to mobile robot localization," in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, vol. 2, pp. 1656–1663 vol.2, 1999.

[30] D. M. Rosen, J. Mason, and J. J. Leonard, "Towards lifelong feature-based mapping in semi-static environments," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1063–1070, May 2016.

[31] D. Meyer-Delius, J. Hess, G. Grisetti, and W. Burgard, "Temporary maps for robust localization in semi-static environments," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 5750–5755, Oct 2010.

[32] M. Gupta, G. K. Arunkumar, and L. Vachhani, "Bearing only visual homing: Observer based approach," in *2017 25th Mediterranean Conference on Control and Automation (MED)*, pp. 358–363, July 2017.

[33] A. Argyros, K. Bekris, S. Orphanoudakis, and L. Kavraki, "Robot homing by exploiting panoramic vision," *Autonomous Robots*, vol. 19, pp. 7–25, 07 2005.

[34] K. Bekris, A. Argyros, and L. Kavraki, *Exploiting Panoramic Vision for Bearing-Only Robot Homing*, pp. 229–251. 01 2006.

[35] R. Hermann and A. Krener, "Nonlinear controllability and observability," *IEEE Transactions on Automatic Control*, vol. 22, pp. 728–740, October 1977.

[36] A. Isidori, M. Thoma, E. D. Sontag, B. W. Dickinson, A. Fettweis, J. L. Massey, and J. W. Modestino, *Nonlinear Control Systems*. Berlin, Heidelberg: Springer-Verlag, 3rd ed., 1995.

[37] E. D. Sontag and Y. Wang, "I/O equations for nonlinear systems and observation spaces," in *[1991] Proceedings of the 30th IEEE Conference on Decision and Control*, pp. 720–725 vol.1, 1991.

[38] H. Sussmann, "Single-input observability of continuous-time systems," *Theory of Computing Systems*, vol. 12, pp. 371–393, Dec. 1978.

[39] M. Anguelova, *Observability and Identifiability of Nonlinear Systems with Applications in Biology*. Doktorsavhandlingar vid Chalmers Tekniska Högskola: Chalmers Tekniska Högskola, Chalmers University of Technology, 2007.

[40] Y. Wang and E. D. Sontag, "On two definitions of observation spaces," *Systems Control Letters*, vol. 13, no. 4, pp. 279 – 289, 1989.

[41] F. Belo, P. Salaris, and A. Bicchi, "3 known landmarks are enough for solving planar bearing slam and fully reconstruct unknown inputs," pp. 2539–2545, 10 2010.

[42] A. Martinelli, "Nonlinear unknown input observability: Extension of the observability rank condition," *IEEE Transactions on Automatic Control*, vol. 64, no. 1, pp. 222–237, 2019.

[43] G. Besançon and G. Bornard, "Performances and Limitations of Symbolic Computation in Nonlinear Analysis and Control: An Example using Mathematica," *IFAC Proceedings Volumes*, vol. 28, no. 8, pp. 567 – 572, 1995. IFAC Conference on System Structure and Control 1995, Nantes, France, 5-7 July 1995.

[44] S. P. Linder and B. Shafai, "Rejecting disturbances to flexible structures using PI Kalman filters," in *Proceedings of the 1997 IEEE International Conference on Control Applications*, pp. 475–477, 1997.

[45] A. V. Shenoy, "Unconstrained nonlinear state estimation for chemical processes," 2010.

[46] K. T. Sutherland and W. B. Thompson, "Localizing in unstructured environments: dealing with the errors," *IEEE Transactions on Robotics and Automation*, vol. 10, no. 6, pp. 740–754, 1994.

[47] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, vol. 3, pp. 2149–2154 vol.3, 2004.

[48] G. N. DeSouza and A. C. Kak, "Vision for mobile robot navigation: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 2, pp. 237–267, 2002.

[49] A. Li, W. Jiang, W. Yuan, D. Dai, S. Zhang, and Z. Wei, "An Improved FAST+SURF Fast Matching Algorithm," *Procedia Computer Science*, vol. 107, pp. 306–312, 12 2017.

[50] E. Rosten, R. Porter, and T. Drummond, "Faster and better: A machine learning approach to corner detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, pp. 105–19, 01 2010.

[51] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–, 11 2004.

[52] J. Neira and J. D. Tardos, "Data association in stochastic mapping using the joint compatibility test," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 890–897, 2001.

[53] L. Bodizs, B. Srinivasan, and D. Bonvin, "On the design of integral observers for unbiased output estimation in the presence of uncertainty," *Journal of Process Control*, vol. 21, pp. 379–390, 03 2011.

[54] Davison, "Real-time simultaneous localisation and mapping with a single camera," in *Proceedings Ninth IEEE International Conference on Computer Vision*, pp. 1403–1410 vol.2, Oct 2003.