2011-11-16

Bearing-Only Cooperative-Localization and Path-Planning of Ground and Aerial Robots

Rajnikant Sharma
Brigham Young University - Provo

Bearing-only Cooperative-Localization and Path-Planning of Ground and Aerial Robots

Rajnikant Sharma

A dissertation submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

Randal W. Beard, Chair
Wynn C. Stirling
Tim W. McLain
Mark B. Colton
D. J. Lee

Department of Electrical and Computer Engineering

Brigham Young University

December 2011

ABSTRACT

Bearing-only Cooperative-Localization and Path-Planning
of Ground and Aerial Robots

Rajnikant Sharma
Department of Electrical and Computer Engineering, BYU
Doctor of Philosophy

In this dissertation, we focus on two fundamental problems related to the navigation of ground robots and small Unmanned Aerial Vehicle (UAVs): cooperative localization and path planning. The theme running through in all of the work is the use of bearing only sensors, with a focus on monocular video cameras mounted on ground robots and UAVs.

To begin with, we derive the conditions for the complete observability of the bearing-only cooperative localization problem. The key element of this analysis is the Relative Position Measurement Graph (RPMG). The nodes of an RPMG represent vehicle states and the edges represent bearing measurements between nodes. We show that graph theoretic properties like the connectivity and the existence of a path between two nodes can be used to explain the observability of the system. We obtain the maximum rank of the observability matrix without global information and derive conditions under which the maximum rank can be achieved. Furthermore, we show that for the complete observability, all of the nodes in the graph must have a path to at least two different landmarks of known location. The complete observability can also be obtained without landmarks if the RPMG is connected and at least one of the robots has a sensor which can measure its global pose, for example a GPS receiver. We validate these conditions by simulation and experimental results. The theoretical conditions to attain complete observability in a localization system is an important step towards reliable and efficient design of localization and path planning algorithms. With such conditions, a designer does not need to resort to exhaustive simulations and/or experimentation to verify whether a given selection of a control strategy, topology of the sensor network, and sensor measurements meets the observability requirements of the system. In turn, this leads to decreased requirements of time, cost, and effort for designing a localization algorithms. We use these observability conditions to develop a technique, for camera equipped UAVs, to cooperatively geo-localize a ground target in an urban terrain. We show that the bearing-only cooperative geo-localization technique overcomes the limitation of requiring a low-flying UAV to maintain line-of-sight while flying high enough to maintain GPS lock. We design a distributed path planning algorithm using receding horizon control that improves the localization accuracy of the target and of all of the UAVs while satisfying the observability conditions.

Next, we use the observability analysis to explicitly design an active local path planning algorithm for UAVs. The algorithm minimizes the uncertainties in the time-to-collision (TTC) and bearing estimates while simultaneously avoiding obstacles. Using observability analysis we show that maximizing the observability and collision avoidance are complementary tasks. We provide sufficient conditions of the environment which maximizes the chances obstacle avoidance and UAV reaching the goal. Finally, we develop a reactive path planner for UAVs using sliding mode control such that it does not require range from the obstacle, and uses bearing to obstacle to

avoid cylindrical obstacles and follow straight and curved walls. The reactive guidance strategy is fast, computationally inexpensive, and guarantees collision avoidance.

Keywords: cooperative localization, path planning, graph theory, nonlinear system theory, sliding mode control, extended kalman filter, lie derivatives, vision based estimation and control nolistoftables

ACKNOWLEDGMENTS

I would like to thank my advisor and committee chair, Prof. Randy Beard, for his guidance, funding, and enthusiasm for the research. Special thanks to Dr. Clark Taylor for inviting me to come to BYU for the PhD programme and providing the initial help for my research. Prof. Tim McLain was a great help in completing the work in this dissertation, and I thank him for being on my committee and providing me funding through TA opportunities. Also, thanks to Prof. Wynn C. Sterling, Prof. D.J. Lee, and Prof. Mark Colton for being on my committee

Thanks very much to the other graduate students of MAGICC Lab who have helped me to mature as a researcher. In particular: Travis Millet, Stephen Quebe, John MacDonald, Huili Yu, and Solomon Sun. I especially thank the department staff for making the administrative life of every ECEN student easy. Also I would like to thank Dr. P.B Sujit for the important research related discussions.

I would not be here if it wasn't for my family, and I would not be able to stay if it weren't for my beautiful wife, Suruchi. This work is dedicated to my wife and the glorious memories I have of my uncle Krishan Kumar.

TABLE OF CONTENTS

LIST OF FIGURES

NOMENCLATURE

n	Number of vehicles used in cooperative localization
l	Number of landmarks used in cooperative localization
G_n^l	Relative position measurement graph (RPMG) with n vehicle nodes and l landmark nodes
m	Number of edges in an RPMG
η_{ij}	Bearing measurement from i^{th} vehicle to j^{th} vehicle in the horizontal plane
γ_{ij}	Bearing measurement from i^{th} vehicle to j^{th} vehicle in the vertical plane
η_{ik}	Bearing measurement from i^{th} vehicle to k^{th} landmark in the horizontal plane
N_i^C	Set of communicating neighbors for $i^{th} vehicle$
N_i^M	Set of vehicles from which the i^{th} vehicle can measure bearing
$\mathscr{V}_{n,l}$	Set of nodes in the RPMG G_l^n
$\mathscr{E}_{n,l}$	Set of edges in the RPMG G_l^n
$\mathscr{V}_n$	Set of n vehicle nodes in the RPMG G_l^n
$\mathscr{V}_l$	Set of l landmark nodes in the RPMG G_l^n
X_i	State of the i^{th} vehicle
$[x_i, y_i, z_i]^\top$	Position of i^{th} vehicle in the Inertial frame
$Xl_k = [x_k, y_k]^\top$	Position of k^{th} landmark in the Inertial frame
$[\phi_i, \theta_i, \psi_i]^\top$	Attitude (roll, pitch, and heading) of the i^{th} vehicle
u_i	Control input vector for i^{th} vehicle
V_i	Forward velocity of the i^{th} vehicle
ω_i	Turn rate of the i^{th} vehicle
g	Gravitational acceleration constant
X	Combined state vector for n vehicles
Xl	Combined position vector for l landmarks
u	Combined control vector for n vehicles
R_{ij}	Distance between i^{th} and j^{th} vehicle in horizontal plane
$x_{ij} = x_i - x_j$	Difference in x direction between i^{th} and j^{th} vehicle
$y_{ij} = y_i - y_j$	Difference in y direction between i^{th} and j^{th} vehicle
$z_{ij} = z_i - z_j$	Difference in z direction between i^{th} and j^{th} vehicle
R_{ik}	Distance between i^{th} vehicle and k^{th} landmark in horizontal plane
$x_{ik} = x_i - x_k$	Difference in x direction between i^{th} vehicle and k^{th} landmark in horizontal plane
$y_{ik} = y_i - y_k$	Difference in y direction between i^{th} vehicle and k^{th} landmark in horizontal plane
$z_{ik} = z_i - z_k$	Difference in z direction between i^{th} vehicle and k^{th} landmark in horizontal plane
ρ_{ij}	Distance between i^{th} and j^{th} vehicle
ρ_{ik}	Distance between i^{th} vehicle and k^{th} landmark
O	Actual nonlinear observability matrix an RPMG

O^p	Nonlinear observability matrix of a proper RPMG
E	Elementary operation matrix
U	Row reduced matrix
f_i	Motion model of the i^{th} vehicle
h_p	Measurement model for the p^{th} edge
f	Combined motion model for n vehicles
h	Combined measurement model for m edges in the RPMG
R_{sensor}	Sensor range
Y	Information matrix
$\hat{y}$	Information vector
$\hat{x}$	Estimated state
t	Time
P	Covariance matrix
V_w	Projection of UAV's velocity on the wall
η	Bearing measurement to a obstacle measured from the UAV
ρ	Distance between UAV and a obstacle
$\hat{\eta}$	Bearing from the edge of a obstacle
$\hat{\rho}$	Distance from the edge of a obstacle
η_d	Desired bearing angle

CHAPTER 1. INTRODUCTION

In this dissertation, we focus on two fundamental problems related to the navigation of ground robots and unmanned aerial vehicles (UAVs): localization and path planning. The theme running through in all of the work is the use of bearing only sensors, with a focus on monocular video cameras mounted on ground robots and UAVs.

Localization is a process of estimating the robot's position and orientation, known together as the robot's pose, in real time using sensor measurements. Without knowledge of its pose, a robot cannot navigate and interact with its environment in a meaningful way. It should be pointed out that animals and humans themselves possess impressive navigational skills. Through millions of years of evolution, species have developed mechanisms enabling them to determine their location by processing visual information [1, 2], measurements of electric [3] and magnetic [4] fields, echolocation [5], and other sensory input. The objective of localization methods in the field of robotics is to endow robots with similar capabilities, which are a prerequisite for robots to be able to operate autonomously.

Recently, the use UAVs in several civil and military applications has increased significantly. There are many applications for smaller UAVs that require them to operate in urban terrains. To navigate in urban environments, the UAVs must have the capability to autonomously plan paths from point A to B such that they do not collide with buildings, trees or other obstacles.

The sensors employed for mobile robot navigation can be broadly divided into two categories: proprioceptive and exteroceptive sensors. The proprioceptive sensors provide measurements related to the robot's motion, such as velocity and acceleration. Examples of proprioceptive sensors are wheel encoders known as odometry sensors [6], which provide measurements of linear and rotational velocity for ground vehicles, and inertial measurement units (IMUs) [7], which measure rotational velocities and linear accelerations. On the other hand, exteroceptive sensors measure quantities of interest in the robot's surroundings, such as the range and bearing to

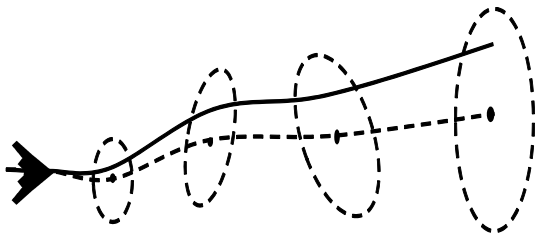


Figure 1.1: DR illustration. The solid line represents the true vehicle path, while the dashed line represents the estimated one. The ellipses represent the uncertainty in the position estimates. During DR, the uncertainty of the estimates continuously increases, asymptotically approaching infinity.

a nearby object. Examples of exteroceptive sensors typically employed in robotics applications include cameras [8], sonars [9], laser range finders [10], and GPS receivers [11]. Among the suite of possible exteroceptive sensors, a video camera is inexpensive, lightweight, and fits the physical requirements of small ground robots and UAVs. Furthermore, a camera has high information to weight ratio and can be used for several applications other than localization. A camera can directly measure bearing to objects in its image plane with very low computational work. In this dissertation, we explore how bearing-only measurements can be efficiently used for localization and path planning of ground robots and UAVs.

Due to disturbances, modeling errors, and other factors, the sensor measurements available to a robot are inevitably corrupted by noise. The presence of noise implies that the true pose, i.e., position and orientation, of a robot cannot, in general, be estimated with zero error; some uncertainty will always exist about the robot pose. Thus, the goal of a localization algorithm is to fuse the available proprioceptive and exteroceptive measurements in real time, to compute an estimate for the robot pose, as well as a measure of the uncertainty of this estimate. Depending on the sensors available onboard a robot, several localization approaches are possible [12].

1.1 Single vehicle localization

The simplest localization technique is dead reckoning (DR). The DR technique consists of propagating the pose estimates of a mobile robot by integrating the measurements of its proprioceptive sensors. However, due to the integration of the measurement noise, the pose estimates quickly drift from their true values, and eventually become unusable [6], as shown in Fig. 1.1.

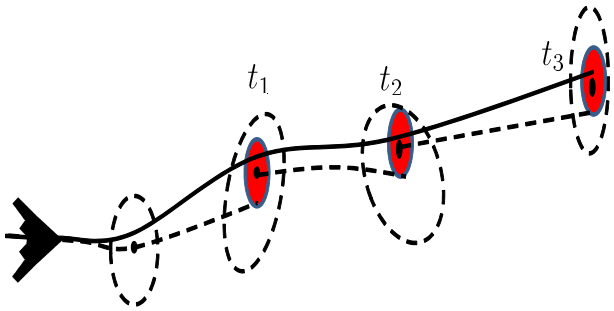


Figure 1.2: Illustration of GPS based localization. At time instants t_1, t_2, and t_3, GPS measurements are recorded, and used for updating the vehicle's pose estimate. Every time an absolute position measurement is processes, the uncertainty of the resulting estimate is reduced, as shown in shaded ellipses. If the robot periodically uses GPS measurements for position updates, drift is prevented, and at steady state the uncertainty of the pose estimate is bounded.

Therefore, for most practical purposes, additional information from exteroceptive sensors must be processed, in order to reduce or even eliminate the growth of DR errors. One of the most common exteroceptive sensors is the GPS receiver. Vehicles moving outdoors can obtain measurements of their absolute position, by measuring range to the GPS satellites [11]. The availability of the robot's position prevents drift, and maintains bounded position errors over time, e.g., see Fig. 1.2. However, in many environments of interest, GPS signals are unavailable or unreliable, e.g., indoors, underwater, on other planets, in the urban canyons. Additionally, the accuracy of pose estimates based on GPS may be insufficient for some applications, e.g., autonomous navigation in traffic. Therefore, it is important to develop methods for localization either in the absence or in the addition to GPS.

When robots operate in unknown environments, they can process their sensor measurements to detect features (also known as landmarks), which can be used to aid localization. The term feature in this context refers to a distinctive point of interest, which can be detected in the sensor data by applying an interest operator. In images, for example, features are typically selected as points where the image intensity gradient is large [13, 14], while in laser scanner data, corners and line segments are the most commonly detected features [15]. When the positions of the features in the robot's surroundings are not known in advance, they can be estimated along with the robot's trajectory (cf. Fig. 1.3). This is the task of Simultaneous Localization and Mapping (SLAM) [16, 17]. The main benefit of SLAM is that, since the point features are static, they serve

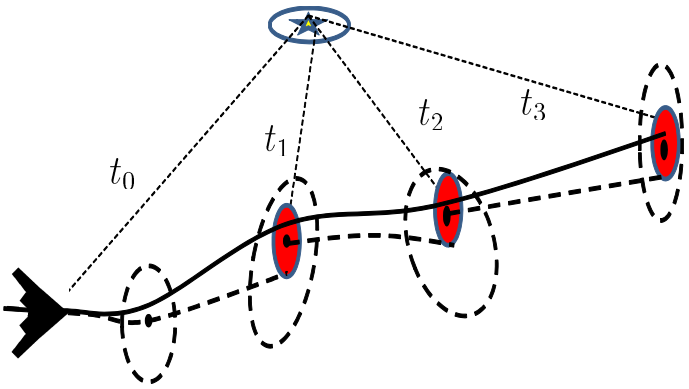


Figure 1.3: Illustration of SLAM: At time t_0, the robot observes for the first time a static feature, which is denoted by a star, and computes an estimate of its position. At later time instants, t_1, t_2, and t_3, the vehicle re-observes the same feature, and employs these measurements in order to update its pose estimates, as well as the estimate of the features position. Every re-observation results in a reduction of the uncertainty, and asymptotically, the errors of the vehicle's pose remains bounded. A similar situation arises when multiple cooperating robots localize as a group, while observing static features.

as localization beacons or landmarks. Observing features over multiple time instants, or when the robot re-visits an area, this is often termed loop closing, results in bounded estimation errors over long time periods [18]. Thus, SLAM permits accurate, long term localization in unknown environments, and is considered an enabling technology for robot autonomy [19].

1.2 Cooperative localization

It has generally been acknowledged within the autonomous robot and UAV research community that there are many advantages with multiple cooperating robots for tasks, e.g., object transportation [20], construction [21], exploration [22], surveillance [23], reconnaissance [24], search [25], and fire monitoring [26, 27]. When multiple autonomous vehicles operate in the same area, one attractive approach for aiding the localization of all robots is for them to record relative, i.e., robot-to-robot measurements, such as distance and bearing, and to combine this information with their odometric estimates as shown in Fig. 1.4. When robots record, communicate, and process such relative pose information for localization purposes, the process is termed cooperative localization (CL) [28]. CL provides the means for an implicit sensor sharing, as shown in Fig. 1.5, as localization information is dissipated over a wireless network to all the members of the

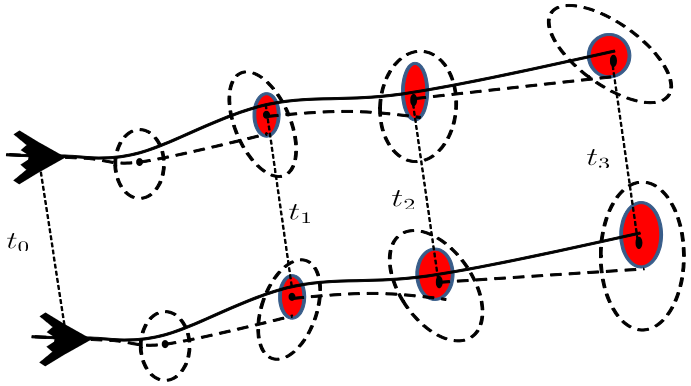


Figure 1.4: Illustration of CL. At time instants t_0, t_1, t_2, and t_3, Vehicle 2 records measurements of the relative position of Vehicle 1, which is denoted by dotted arrows. Using these measurements both robots' pose pose estimates are updated, and the resulting estimates have reduced uncertainty. However, in the absence of any absolute position information, the relative measurements cannot fully compensate for position drift. Therefore, the position uncertainty gradually increases over time, albeit at a rate lower than that of DR

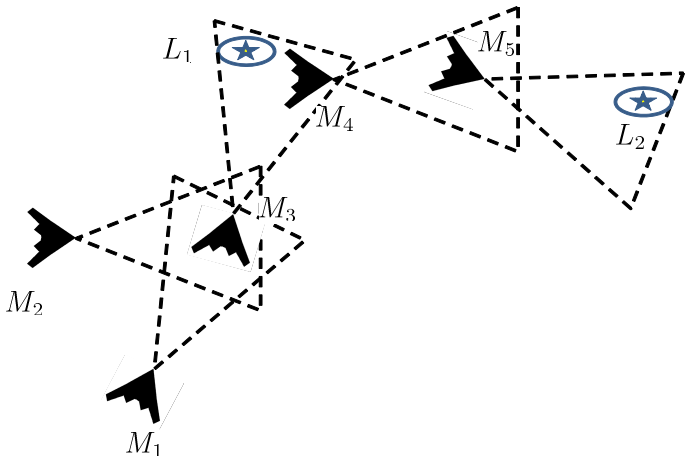


Figure 1.5: Illustration bearing-only CL. UAVs ($M_1 - M_5$) measure bearing from other UAVs and landmarks (L_1, L_2), which are in their image plane. Cooperative localization creates a joint synthetic sensor plane which is larger and flexible then the single UAV sensor plane. In CL a UAV requires two landmarks to keep its localization error bounded(Lemma 2.3.6 in Chapter 2). Although, due to limited field-of-view a single UAV, itself, cannot measure bearing from two landmarks, the joint sensor plane, however, can overcomes the limitation by keeping the sensor network connected(Theorem 2.3.2 in Chapter 2)

group. Cooperative localization has been an active area of research, e.g., [15, 29–35], because it provides several potential advantages, including increased localization accuracy, sensor coverage, robustness, efficiency, and flexibility. Recently, estimation algorithms such as the extended kalman filter (EKF) [36], minimum mean square estimator (MMSE) [30], maximum likelihood estimation (MLE) [37], particle pilter [38], and maximum a posteriori (MAP) [39], have been used to solve the cooperative localization problem. These algorithms can be used either in a centralized [32] or decentralized manner [30, 36, 39].

1.3 Path planning

The path planning and obstacle avoidance problems for UAVs have received significant attention [40–44]. The path planning problem can be grouped into global path planning and local path planning. Global path planning requires complete knowledge about the environment and a static terrain. In that setting a collision-free path from the start to the destination configuration is generated before the vehicle starts its motion [45]. The global path planning problem has been addressed by many researchers with common solutions being probability road map (PRM) [46–48] methods and rapidly-expanding random Tree (RRT) [49,50] methods. On the other hand, local path planning is executed in real-time during flight. The basic idea is to first sense the obstacles in the environment and then determine a collision-free path [40]. The local path planing algorithms are more suited for unknown and dynamic environments, which are discussed in this work.

1.4 Research overview

For the localization errors to be bounded in CL and C-SLAM, it is required that the system be observable [51], independent of the estimation technique being used. Therefore, it is necessary to conduct an observability analysis of the system. Several authors have carried out observability analysis of the cooperative localization problem [36, 52–54]. However, these efforts have focused primarily on finding observable and unobservable states of linear measurement consisting of two robots. As a result, the issue of constraints imposed by the observability of the system on the motion of robots, connectivity of the sensor network, and type and numbers of sensor have largely been left unexplored. However, the lack of theoretical conditions to attain complete observability

in a localization system is a significant impediment to reliable and efficient design of localization and path planning algorithms. Without such conditions, a designer must resort to exhaustive simulations and/or experimentation to verify whether a given selection of a control strategy, topology of the sensor network, and sensor measurements meets the observability requirements of the system. In turn, this leads to increased requirements of time, cost, and effort for designing localization algorithms.

One of the important applications of camera equipped UAVs is determining the location of a ground target when imaged from UAVs. The pixel location of the target in the image, the position and attitude of the air vehicles, the camera pose angles, and knowledge of the terrain elevation are used to geo-localize the object. Previous target localization work using a camera equipped UAV is reported in [55–58] and references therein. The existing work on vision based geo-localization successfully demonstrates the target localization concept and provides several techniques to improve the accuracy of geo-localization. However, the limitations associated with geo-localizing a target in urban environments are not addressed. All of the existing methods require the UAV's position and attitude to geo-localize a target. Furthermore, a camera is a line of sight (LOS) sensor and there may exist many occlusions in the urban environment, e.g., buildings, trees, which can lead to unreliable tracking of the target.

In environments where pop-up threats are common, a reactive planner with fixed computational cost is appropriate. Reactive obstacle avoidance methods have been developed in previous work using dynamic replanning [59], potential fields [60], simulated annealing [61], predefined maneuvers [62], and deterministic collision avoidance guidance strategy for stationery cylindrical objects [63]. The existing methods, however, require range and bearing measurements to compute the control law. Since a monocular camera is a bearing-only sensor, either the range from the object should be estimated using bearing measurement or the control law should avoid obstacles using bearing-only measurements. Also, previous methods are only designed for obstacles modeled as cylindrical objects, however, large obstacles like a wall cannot be modeled as a cylindrical obstacle.

In this work, we address the aforementioned limitations by (1) performing a rigorous nonlinear observability analysis of bearing-only cooperative localization, (2) using observability conditions of bearing-only localization to geo-localize a target in the urban terrain using a team of

UAVs, and (3) developing a path planing technique which improves the localization accuracy while satisfying the observability constraints and avoiding collision from obstacles using bearing-only measurement. We use mathematical techniques from graph theory, linear algebra, and nonlinear systems theory. The main contributions of this dissertation are as follows.

- We derive conditions for complete observability of the bearing-only cooperative localization problem. The key element of this analysis is the Relative Position Measurement Graph (RPMG). The nodes of an RPMG represent vehicle states and the edges represent bearing measurements between nodes. We show that graph theoretic properties like the connectivity and the existence of a path between two nodes can be used to explain the observability of the system. We obtain the maximum rank of the observability matrix without global information and derive conditions under which the maximum rank can be achieved. Furthermore, we show that for the complete observability, all of the nodes in the graph must have a path to at least two different landmarks of known location. The complete observability can also be obtained without landmarks if the RPMG is connected and at least one of the robot has a sensor which can measure its global pose, for example a GPS receiver. These results are reported in [64–66].

- We develop a technique, for camera equipped UAVs, to cooperatively geo-localize a ground target in an urban terrain. We use observability conditions for bearing-only localization to overcome the limitation of requiring a low-flying UAV to maintain line-of-sight while flying high enough to maintain GPS lock. We design a distributed path planning algorithm using receding horizon control that improves the localization accuracy of the target and of all of the UAVs while satisfying the observability conditions. These results are reported in [67].

- We use the observability analysis to explicitly design an active local path planning algorithm for UAVs. The algorithm minimizes the uncertainties in the time-to-collision (TTC) and bearing estimates while simultaneously avoiding obstacles. Using observability analysis we show that maximizing the observability and collision avoidance are complementary tasks. We provide sufficient conditions of the environment which maximizes the chances that obstacles are avoided and that UAV reaches the goal. These results are reported in [68, 69].

- We develop a reactive path planner for UAVs using sliding mode control such that it does not require range from the obstacle, and uses only bearing to obstacle to avoid cylindrical obstacles and follow straight and curved walls. The reactive guidance strategy is fast, computationally inexpensive, and guarantee collision avoidance. These results are reported in [70, 71].

1.5 Organization of the manuscript

Chapter 2, includes graph-based observability analysis of bearing-only cooperative localization. In Chapter 3, we present bearing-only cooperative geo-localization. In Chapter 4, we present observability based path planing for UAVs using bearing-only measurements. In Chapter 5, we present bearing-only reactive path planing technique for UAVs. Finally, Chapter 6 summarizes the key results of this work, and identifies interesting avenues for future research.

CHAPTER 2. GRAPH-BASED OBSERVABILITY ANALYSIS OF BEARING-ONLY COOPERATIVE LOCALIZATION

2.1 Introduction

In cooperative localization a group of robots exchange relative position measurements from their exteroceptive sensors (e.g., camera, laser, etc.) and their motion information (velocity and turn rate) from interoceptive sensors (e.g., IMU, encoders, etc.) to collectively estimate their states. Cooperative localization has been an active area of research (e.g., [15, 29–35]) because it provides several potential advantages, including increased localization accuracy, sensor coverage, robustness, efficiency, and flexibility.

Recently, estimation algorithms such as the extended kalman filter (EKF) [36], minimum mean square estimator (MMSE) [30], maximum likelihood estimation (MLE) [37], particle filter [38], and maximum a posteriori (MAP) [39], have been used to solve the cooperative localization problem. These algorithms can be used either in centralized [32] or decentralized manner [30, 36, 39]. For the localization errors to be bounded, it is required that the system be observable, independent of the estimation technique being used.

Several authors have carried out observability analysis of the cooperative localization problem. Initial results regarding the observability of cooperative localization were reported by Roumeliotis and Bekey [36]. They used linear observability analysis to show that the states of the robots performing cooperative localization are unobservable, but can be made observable by providing global positioning information to one of the robots. In [36] it was assumed that the absolute vehicle heading is measured directly and does not need to be estimated. Furthermore, linear approximation of a nonlinear system can provide different structural properties regarding the observability [52, 53]. Martinelli *et al*. [54] investigates the nonlinear observability of cooperative localization for two robots without heading measurements. They compared the observability properties of range and bearing measurements and showed that with either type of measurement, the

maximum rank of the observability matrix is three, i.e., not fully observable. The analysis in [54] shows that relative bearing is the best observation between the robots. The part of the system which is observable is in general larger than for the other relative observations (relative distance and relative orientation). Accordingly, [54] uses polar coordinates for the observability analysis. Although polar coordinates simplify the analysis for two robots, we use a global coordinate system because it is more appropriate for graph level ($n > 2$) observability analysis.

In this chapter, we extend the observability analysis presented in [54] from 2 to n robots, with bearing-only measurements. The extension for $n > 2$ is not obvious because of the dynamically changing set of $n(n-1)/2$ different relative bearing measurements leading to $2^{n(n-1)/2}$ possible configurations. Furthermore, since the robot states in [54] are not observable with respect to a global reference frame, and since it has been shown that two landmarks are needed for the observability of a single vehicle [72–74], in this chapter we derive the number of landmarks needed for full observability of a group of n robots performing cooperative localization. In contrast to [36], we also assume that the heading of each robot is not directly measured but must be estimated.

To represent a group of robots, we will use the Relative Position Measurement Graph (RPMG) introduced in [18]. The nodes of an RPMG represent vehicle states and the edges represent bearing measurements between nodes. We establish a relationship between the graph properties of the RPMG and the rank of the system observability matrix. We prove that for a connected RPMG, the observability matrix for a team of n robots, which has size $3n \times 3n$ will have rank $3(n-1)$. We also derive conditions under which landmarks observed by a subset of robots enable the system to become fully observable.

The chapter is organized as follows. In Section 2.2 we describe bearing-only cooperative localization and formulate the problem. In Section 2.3 we perform the nonlinear observability analysis. The simulation and experimental results are presented in Section 2.4. In Section 2.5 we give our conclusions.

2.2 Bearing-only cooperative localization

Consider n robots moving in a horizontal plane performing cooperative localization in the vicinity of l known landmarks. We can write the equations of motion for the i^{th} robot as

$$\dot{X}_i = f_i(X_i, u_i) \triangleq \begin{pmatrix} V_i \cos\psi_i \\ V_i \sin\psi_i \\ \omega_i \end{pmatrix}, \tag{2.1}$$

where $X_i = [x_i \; y_i \; \psi_i]^\top \in \mathbb{R}^3$ is the robot state, including robot location $(x_i,\ y_i)$ and robot heading ψ_i, and $u_i = [V_i, \omega_i]^\top$ is the control input vector. We assume that onboard introspective sensors (e.g., encoders) measure the linear speed V_i and angular speed ω_i of the robot. Each vehicle has an exteroceptive sensor to measure relative bearing to other vehicles and known landmarks that are in the field-of-view of the sensor. Bearing from the i^{th} robot to the j^{th} robot can be written as

$$\eta_{ij} = \tan^{-1}\left(\frac{y_j - y_i}{x_j - x_i}\right) - \psi_i, \tag{2.2}$$

and bearing from the i^{th} robot to the k^{th} landmark can be written as

$$\eta_{ik} = \tan^{-1}\left(\frac{y_k - y_i}{x_k - x_i}\right) - \psi_i, \tag{2.3}$$

where $Xl_k = [x_k \; y_k]\top$ is the position vector of the k^{th} landmark.

For cooperative localization, each robot exchanges its local sensor measurements (velocity, turn rate, and bearing to landmarks and other robots) with their neighbors. Let N_i^M be the set of neighbors for which robot i can obtain bearing measurements, and let N_i^C be the set of neighbors with which robot i can communicate. In this chapter, we assume that $N_i^M = N_i^C$ and we will therefore denote the set of neighbors as N_i. To represent the connection topology of the robots we use a relative position measurement graph (RPMG) [18] which is defined as follows.

Definition 2.2.1 *An RPMG for n robots performing cooperative localization with l different known landmarks is a directed graph* $G_n^l \triangleq \{\mathcal{V}_{n,l}, \mathcal{E}_{n,l}\}$, *where* $\mathcal{V}_{n,l} = \{\mathcal{V}_n,\ \mathcal{V}_l\}$ *is the node set which consists of vehicle node set* $\mathcal{V}_n = \{1, \cdots, n\}$ *and landmark node set* $\mathcal{V}_l = \{1, \cdots, l\}$, *and* $\mathcal{E}_{n,l}(t) \subset \{\mathcal{V}_n \times \mathcal{V}_n\} \cup \{\mathcal{V}_n \times \mathcal{V}_l\}$ *is the edge set of m bearing measurements at time t. Indexes* $i, j \in \mathcal{V}_n$

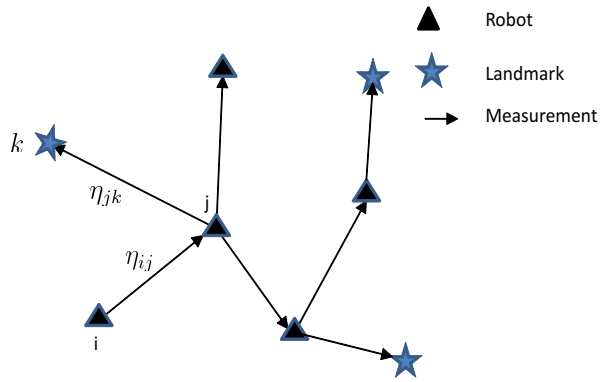


Figure 2.1: Relative position measurement graph (RPMG). The nodes ofan RPMG represent vehicle states and the edges represent bearing measurements between nodes.

represent the i^{th} and j^{th} vehicle nodes, index $k \in \mathscr{V}_l$ represents the k^{th} landmark, and index $p \in \mathscr{E}_{n,l}$ represents the p^{th} edge between two nodes. An example RPMG (G_5^3 with $m = 7$) is shown in Fig. 2.1.

Additionally, without loss of generality we assume that robots maintain a safe distance from each other and from landmarks, i.e., $R_{ij} > 0,\ \forall i, j \in \mathscr{V}_n$ and $R_{ik} > 0,\ \forall i \in \mathscr{V}_n$ and $\forall k \in \mathscr{V}_l$. Also, we assume that the robot sensors have limited sensor range R_{sensor} and limited field-of-view. Therefore, agents can only measure the bearing of those robots and landmarks that are located within the footprint of the sensor. Therefore, the graph G_n^l will likely have a time varying topology.

2.2.1 Cooperative localization implementation

The objective of the cooperative localization is to estimate the combined state $\hat{X}(k) = [\hat{X}_1(k), \cdots, \hat{X}_n(k)]^\top$. We use an extended information filter(EIF) to implement the bearing-only cooperative localization. In the information filter instead of state $\hat{X}$ and covariance $P(k)$ the information vector $\hat{y}(k)$ and information matrix $Y(k)$ is updated. The information matrix and information vector can be computed as

$$Y(k) = P(k)^{-1}, \tag{2.4}$$

$$\hat{y}(k) = Y(k)\hat{X}(k). \tag{2.5}$$

Similar to an extended kalman filter (EKF) the EIF has two steps. The first is the prediction step, which is given below.

$$Y(k+1|k) = (F(k)Y(k|k)^{-1}F(k)^{\top} + B(k)Q(k)B(k)^{\top})^{-1}, \tag{2.6}$$

$$\hat{y}(k+1|k) = Y(k+1|k)\hat{X}(k+1|k), \tag{2.7}$$

$$\hat{X}(k+1|k) = X(k|k) + T_s f(\hat{X}(k|k), u(k)), \tag{2.8}$$

where $F_k = \begin{pmatrix} F_1 & 0 & \cdots & 0 \\ 0 & F_2 & \cdots & 0 \\ \vdots & \cdots & \ddots & 0 \\ 0 & 0 & \cdots & F_n \end{pmatrix}$, $B(k) = \begin{pmatrix} B_1 \\ \vdots \\ B_n \end{pmatrix}$, and $Q(k) = \begin{pmatrix} Q_i(k) & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & Q_n(k) \end{pmatrix}$ is covariance of noise in the control input. The matrix F_i and B_i are the system jacobian with respect to state X_i and control u_i, which are given below

$$F_i = I_3 + T_s \frac{\partial f_i}{\partial X_i}|_{X_i = X_i(k)} = \begin{bmatrix} 1 & 0 & -V_i T_s \sin\psi(k) \\ 0 & 1 & V_i T_s \cos\psi(k) \\ 0 & 0 & 1 \end{bmatrix}, \tag{2.9}$$

$$B_i = T_s \frac{\partial f_i}{\partial u_i}|_{u_i = u_i(k)} \begin{bmatrix} T_s \cos\psi_k & 0 \\ T_s \sin\psi_k & 0 \\ 0 & T_s \end{bmatrix}, \tag{2.10}$$

and $Q_i(k) = \begin{pmatrix} \sigma_{v_i}^2 & 0 \\ 0 & \sigma_{\omega_i}^2 \end{pmatrix}$, where σ_{v_i} and σ_{ω_i} are the standard deviation in velocity input and turn rate input respectively.

The measurement update step is given as

$$Y(k+1|k+1) = Y(k+1|k) + \sum H_{ij}(k)^{\top} \sigma_{\eta_{ij}}^{-2} H_{ij}(k),$$
$$\hat{y}(k+1|k+1) = \hat{y}(k+1|k) + \sum H_{ij}(k)^{\top} R_{ij}^{-1} (\mu_{ij} + H_{ij}\hat{X}(k+1|k)).$$

The scalar μ_{ij} represents the innovation

$$\mu_{ij} = \eta_{ij} - h_{ij}(x(k+1|k)), \tag{2.11}$$

and $\sigma_{\eta_{ij}}$ is standard deviation of the noise in the bearing measurement. The row vector H_{ij} is the measurement jacobian

$$H_{ij}(k) = \frac{\partial h_{ij}}{\partial X}|_{X=X(k)}. \tag{2.12}$$

The EIF is dual of the EKF and the EKF is a quasi-local asymptotic observer for nonlinear systems and its convergence and boundedness are achieved when the system is fully observable [51].

2.2.2 Lie derivatives and nonlinear observability

To determine the observability of the entire system represented by the RPMG we use the nonlinear observability rank criteria developed by Hermann and Krener [75] which is summarized in the next paragraph.

Consider a system model with the following form

$$\Sigma: \begin{array}{c} \dot{X(t)} = f(X(t),u(t)) = [f_1^\top(X_1(t),u_1(t)),\cdots,f_n^\top(X_n(t),u_n(t))]^\top, \\ Y(t) = h(X(t),Xl) = [h_1^\top(X(t),Xl)\cdots h_m^\top(X(t),Xl)]^\top, \end{array} \tag{2.13}$$

where $X(t) = [X_1(t)^\top,\cdots,X_n(t)^\top]^\top \in \mathbb{R}^{3n}$ is the state of the system at time t, $Y(t) \in \mathbb{R}^m$ is the system output vector, $Xl = [Xl_1^\top,\cdots,Xl_l^\top]^\top \in \mathbb{R}^{2l}$ is the position vector of all landmarks, $Xl_k = [x_k\ y_k]^\top$ is the position vector of k^{th} landmark, $h_p : \mathbb{R}^{3n} \times \mathbb{R}^{2l} \mapsto \mathbb{R}$, is the measurement model of the p^{th} bearing measurement, $u \in \Lambda \subseteq \mathbb{R}^{2n}$ is the control input vector, $f : \mathbb{R}^{3n} \times \Lambda \mapsto \mathbb{R}^{3n}$, and $\Lambda \subseteq \mathbb{R}^{2n}$ is the set of feasible control inputs. We can write the process model in (2.13) as,

$$\dot{X} = f(X(t),u(t)) = \sum_{i=1}^{n} f_{v_i}(X(t))V_i + \sum_{i=1}^{n} f_{\omega_i}(X(t))\omega_i, \tag{2.14}$$

where

$$f_{v_i} = [0_{3(i-1)\times 1},\ \cos\psi_i(t,\ \sin\psi_i,\ 0,\ 0_{3(n-i)\times 1}]^\top, \tag{2.15}$$

$$f_{\omega_i} = [0_{3(i-1)\times 1},\ 0,\ 0,\ -1,\ 0_{3(n-i)\times 1}]^\top, \tag{2.16}$$

are vector fields on $\mathbb{R}^{3n}$. Two features of this system can be observed. First, the system is time invariant, in that there is no explicit dependence on time. Second, the system above is linear in the control. In this dissertation, we use Lie derivatives, which comes under the field of differential geometry, to answer an important behavior of the system: Is it observable?, in the sense that, for each state there exists at least one corresponding input which permits us to discriminate between this state all nearby states. We use Lie derivatives to obtain the observability matrix. A Lie derivative can be interpreted as the derivative of a scalar along integral curves of the vector field. Some basics of differential geometric methods and Lie derivatives are provided in Appendix A.

The zeroth-order Lie derivative of the p^{th} bearing measurement h_p is the function itself, i.e., $L^0 h_p = h_p$. Using the definition of Lie derivative in (A.4), the first-order Lie derivative of function h_p along with vector field f_{v_i} is given by

$$L^1_{f_{v_i}} h_p = \nabla L^0 h_p \cdot f_{v_i}, \tag{2.17}$$

where ∇ represents the gradient operator, and $\cdot$ denotes the vector inner product. Considering that $L^1_{f_{v_i}} h_p$ is a scalar function itself, the second-order Lie derivative of $L^1_{f_{v_i}} h_p$ along with vector field f_{v_i} is given by

$$L^2_{f_{v_i} f_{v_i}} h_p = \nabla L^1_{f_{v_i}} h_p \cdot f_{v_i}. \tag{2.18}$$

Higher order Lie derivatives are computed similarly. Additionally, it is possible to define mixed Lie derivatives, i.e., with respect to different functions of the process model. For example, the second-order Lie derivative of $L^1_{f_{v_i}} h_p$ along with vector field f_{v_j} is given by

$$L^2_{f_{v_i} f_{v_j}} h_p = \nabla L^1_{f_{v_i}} h_p \cdot f_{v_j}. \tag{2.19}$$

Based on the preceding expressions for the Lie derivatives the observability matrix is defined as the matrix with rows

$$O \triangleq \left\{ \nabla L^d_{f_{v_i},\cdots,f_{v_j},f_{\omega_i},\cdots,f_{\omega_j}} h_p \right\}, \tag{2.20}$$

where $i, j \in \mathscr{V}_n,\ p \in \mathscr{E}_{n,l}$, and $d \in \mathbb{N}$. The important role of this matrix in the observability analysis of a nonlinear system is demonstrated by next theorem, which is proved in [76].

Theorem 2.2.1 ([76]) *A system is locally weakly observable if its observability matrix whose rows are given in* (2.20) *has full rank, e.g., in our case rank*$(O) = 3n$.

2.3 Graph-based observability analysis

In this section, we obtain the conditions for the observability of the graph G^l_n. We derive explicit conditions that establish the rank of the observability matrix of the graph G^0_n without landmarks, and the number of landmarks needed for the full rank of the observability matrix of the graph G^l_n.

2.3.1 Rows in the observability matrix due to an edge

In a graph G^l_n there are two types of edges: an edge between two robots, and an edge between a robot and a landmark. We derive the maximum number of linearly independent rows in the observability sub-matrix of an edge and the conditions for the maximum rank of the observability sub-matrix of an edge. The linearly independent rows of the observability sub-matrix of an edge serve as building block for the observability conditions for the graph G^l_n.

Edge between two robots

First we derive the linearly independent rows in the observability matrix for an edge η_{ij} between two robots and derive the conditions under which maximum number of linearly independent rows can be obtained.

We first find the Lie derivatives of η_{ij}. We rearrange the nonlinear kinematic equations in the following convenient form for computing Lie derivatives:

$$\dot{X} = \begin{bmatrix} \dot{X}_i \\ \dot{X}_j \end{bmatrix} = f_{v_i} V_i + f_{\omega_i} \omega_i + f_{v_j} V_j + f_{\omega_j} \omega_j, \tag{2.21}$$

where $f_{v_i} = [c_{\psi_i}\ s_{\psi_i}\ 0\ 0\ 0\ 0]^\top$, $f_{\omega_i} = [0\ 0\ 1\ 0\ 0\ 0]^\top$, $f_{v_j} = [0\ 0\ 0\ c_{\psi_j}\ s_{\psi_j}\ 0]^\top$, $f_{\omega_j} = [0\ 0\ 0\ 0\ 0\ 1]^\top$, $c_{\psi_i} \triangleq \cos\psi_i$, and $s_{\psi_i} \triangleq \sin\psi_i$. We hereafter compute the necessary Lie derivatives of η_{ij} and their gradients. Since an edge consists of only one measurement, for clarity, we denote h^p by h.

Zeroth-order Lie derivative

$$L^0 h = \eta_{ij},$$

and gradient scaled by R_{ij}^2 is given by

$$\nabla L^0 h = \begin{bmatrix} -y_{ij} & x_{ij} & -R_{ij}^2 & y_{ij} & -x_{ij} & 0 \end{bmatrix},$$

where $x_{ij} \triangleq x_i - x_j$, $y_{ij} \triangleq y_i - y_j$, and $R_{ij}^2 = (x_{ij})^2 + (y_{ij})^2$.

Remark 2.3.1 *The scaling by R_{ij}^2 is an elementary row operation, therefore, it does not change the space spanned by the rows of the observability matrix. Also, it simplifies the computation of the higher order Lie derivatives.*

First-order Lie derivatives

$$\begin{aligned}
L^1_{f_{v_i}} h &= \nabla L^0 h \cdot f_{v_i} = x_{ij} s_{\psi_i} - y_{ij} c_{\psi_i}, \\
L^1_{f_{v_j}} h &= \nabla L^0 h \cdot f_{v_j} = -(x_{ij} s_{\psi_j} - y_{ij} c_{\psi_j}), \\
L^1_{f_{\omega_i}} h &= \nabla L^0 h \cdot f_{\omega_i} = -R_{ij}^2, \\
L^1_{f_{\omega_j}} h &= \nabla L^0 h \cdot f_{\omega_j} = 0,
\end{aligned}$$

with gradients given by

$$\nabla L^1_{f_{v_i}} h = \begin{bmatrix} s_{\psi_i} & -c_{\psi_i} & J_i^+ & -s_{\psi_i} & c_{\psi_i} & 0 \end{bmatrix},$$
$$\nabla L^1_{f_{v_j}} h = \begin{bmatrix} -s_{\psi_j} & c_{\psi_j} & 0 & s_{\psi_j} & -c_{\psi_j} & -J_j^+ \end{bmatrix},$$
$$\nabla L^1_{f_{\omega_i}} h = 2\begin{bmatrix} -x_{ij} & -y_{ij} & 0 & x_{ij} & y_{ij} & 0 \end{bmatrix},$$

where $J_i^+ \triangleq x_{ij}c_{\psi_i} + y_{ij}s_{\psi_i}$ and $J_j^+ \triangleq x_{ij}c_{\psi_j} + y_{ij}s_{\psi_j}$.

Second-order Lie derivatives

$$L^2_{f_{v_i}f_{v_i}} h = \nabla L^1_{f_{v_i}} h \cdot f_{v_i} = s_{\psi_i}c_{\psi_i} - s_{\psi_i}c_{\psi_i} = 0,$$
$$L^2_{f_{v_j}f_{v_j}} h = \nabla L^1_{f_{v_j}} h \cdot f_{v_j} = s_{\psi_j}c_{\psi_j} - s_{\psi_j}c_{\psi_j} = 0,$$
$$L^2_{f_{v_i}f_{v_j}} h = \nabla L^1_{f_{v_i}} h \cdot f_{v_j} = -s_{\psi_i}c_{\psi_j} + s_{\psi_j}c_{\psi_i},$$
$$L^2_{f_{v_i}f_{\omega_i}} h = \nabla L^1_{f_{v_i}} h \cdot f_{\omega_i} = J_i^+,$$
$$L^2_{f_{v_j}f_{\omega_j}} h = \nabla L^1_{f_{v_j}} h \cdot f_{\omega_j} = -J_j^+,$$
$$L^2_{f_{\omega_i}f_{v_i}} h = \nabla L^1_{f_{\omega_i}} h \cdot f_{v_i} = -2J_i^+,$$
$$L^2_{f_{\omega_i}f_{v_j}} h = \nabla L^1_{f_{\omega_i}} h \cdot f_{v_j} = 2J_j^+,$$

with gradients given by

$$\nabla L^2_{f_{v_i}f_{v_j}} h = \begin{bmatrix} 0 & 0 & -J_\psi & 0 & 0 & J_\psi \end{bmatrix},$$
$$\nabla L^2_{f_{v_i}f_{\omega_i}} h = \begin{bmatrix} c_{\psi_i} & s_{\psi_i} & J_i^- & c_{\psi_i} & s_{\psi_i} & 0 \end{bmatrix},$$
$$\nabla L^2_{f_{v_j}f_{\omega_j}} h = \begin{bmatrix} -c_{\psi_j} & -s_{\psi_j} & 0 & s_{\psi_j} & -c_{\psi_j} & -J_j^- \end{bmatrix},$$

where $J_\psi \triangleq c_{\psi_i}c_{\psi_j} + s_{\psi_i}s_{\psi_j}$, $J_i^- \triangleq y_{ij}c_{\psi_i} - x_{ij}s_{\psi_i}$, and $J_j^- \triangleq y_{ij}c_{\psi_j} - x_{ij}s_{\psi_j}$.

Remark 2.3.2 *Gradients of $L^2_{f_{\omega_i}f_{v_i}} h$ and $L^2_{f_{\omega_i}f_{v_j}} h$ are not included because they are linearly dependent on $\nabla L^2_{f_{v_i}f_{\omega_i}} h$ and $\nabla L^2_{f_{v_j}f_{\omega_j}} h$ respectively.*

Third-order Lie derivatives

$$
\begin{aligned}
L^3_{f_{v_i} f_{v_j} f_{\omega_i}} h &= \nabla L^2_{f_{v_i} f_{v_j}} h \cdot f_{\omega_i} = -(c_{\psi_i} c_{\psi_j} + s_{\psi_i} s_{\psi_j}), \\
L^3_{f_{v_i} f_{v_j} f_{\omega_j}} h &= \nabla L^2_{f_{v_i} f_{v_j}} h \cdot f_{\omega_j} = (c_{\psi_i} c_{\psi_j} + s_{\psi_i} s_{\psi_j}), \\
L^3_{f_{v_i} f_{\omega_i} f_{v_i}} h &= \nabla L^2_{f_{v_i} f_{\omega_i}} h \cdot f_{v_i} = 1, \\
L^3_{f_{v_j} f_{\omega_j} f_{v_j}} h &= \nabla L^2_{f_{v_i} f_{\omega_j}} h \cdot f_{v_j} = 1, \\
L^3_{f_{v_i} f_{\omega_i} f_{\omega_i}} h &= \nabla L^2_{f_{v_i} f_{\omega_i}} h \cdot f_{\omega_i} = -(x_{ij} s_{\psi_i} - y_{ij} c_{\psi_i}), \\
L^3_{f_{v_j} f_{\omega_j} f_{\omega_j}} h &= \nabla L^2_{f_{v_i} f_{\omega_j}} h \cdot f_{\omega_j} = x_{ij} s_{\psi_j} - y_{ij} c_{\psi_j},
\end{aligned}
$$

with gradients given by

$$
\begin{aligned}
\nabla L^3_{f_{v_i} f_{v_j} f_{\omega_i}} h &= a_1 \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix} = -\frac{a_1}{a_2} \nabla L^2_{f_{v_i} f_{v_j}} h, \\
\nabla L^3_{f_{v_i} f_{v_j} f_{\omega_j}} h &= \frac{a_1}{a_2} \nabla L^2_{f_{v_i} f_{v_j}} h, \\
\nabla L^3_{f_{v_i} f_{\omega_i} f_{\omega_i}} h &= -(x_{ij} s_{\psi_i} - y_{ij} c_{\psi_i}) = -\nabla L^1_{f_{v_i}} h, \\
\nabla L^3_{f_{v_j} f_{\omega_j} f_{\omega_j}} h &= x_{ij} s_{\psi_j} - y_{ij} c_{\psi_j} = -\nabla L^1_{f_{v_j}} h,
\end{aligned}
$$

where $a_1 \triangleq (s_{\psi_i} c_{\psi_j} - c_{\psi_i} s_{\psi_j})$, and $a_2 \triangleq c_{\psi_i} c_{\psi_j} + s_{\psi_i} s_{\psi_j}$.

Clearly, third and higher order Lie derivatives are linearly dependent on the gradients of second and lower order Lie derivatives. Therefore, with all the non-zero inputs the observability matrix of an edge between two robots can be written using gradients of Lie derivatives up to second-order as

$$
O_{ij} = \begin{bmatrix}
-y_{ij} & x_{ij} & -R^2_{ij} & y_{ij} & -x_{ij} & 0 \\
s_{\psi_i} & -c_{\psi_i} & J^+_i & -s_{\psi_i} & c_{\psi_i} & 0 \\
-s_{\psi_j} & c_{\psi_j} & 0 & s_{\psi_j} & -c_{\psi_j} & -J^+_j \\
-2x_{ij} & -2y_{ij} & 0 & 2x_{ij} & 2y_{ij} & 0 \\
0 & 0 & -J_\psi & 0 & 0 & J_\psi \\
c_{\psi_i} & s_{\psi_i} & J^-_i & -c_{\psi_i} & -s_{\psi_i} & 0 \\
-c_{\psi_j} & -s_{\psi_j} & 0 & c_{\psi_j} & s_{\psi_j} & -J^-_j
\end{bmatrix}. \tag{2.22}
$$

Our objective is to find the number of linearly independent rows in O_{ij}. Therefore, we transform O_{ij} into reduced row echelon form (RREF). RREF is the simplest possible form of a matrix, which directly provides the number of linearly independent rows in the matrix. Since RREF is the backbone of the analysis presented in this chapter we state the next lemma, which explains the properties of a RREF matrix.

Lemma 2.3.1 ([77]) *A matrix $A \in \mathbb{R}^{m\times n}$, by means of a finite sequence of elementary row operations, can be transformed to a row reduced echelon form $U \in \mathbb{R}^{m\times n}$ such that*

$$EA = U, \tag{2.23}$$

where $E \in \mathbb{R}^{m\times m}$ is the elementary operation matrix. If the rank of A is r then

1.
$$U = \begin{bmatrix} \mathbf{I}_r & B \\ \mathbf{0}_{(m-r)\times r} & \mathbf{0}_{(m-r)\times(n-r)} \end{bmatrix}, \tag{2.24}$$

   where $\mathbf{I}_r$ is the Identity matrix of size r, $\mathbf{0}$ represents matrix of zeros, and $B \in \mathbb{R}^{r\times(n-r)}$,

2. *the first r rows of matrix U are linearly independent,*

3. *the non zero rows of the matrix U spans the same row space spanned by A,*

4. *if A is an invertible matrix ($r = m = n$) then U is the Identity matrix.*

The next lemma provides conditions for the maximum rank of the observability matrix of an edge between two robots.

Lemma 2.3.2 *The rank of O_{ij} given by* (2.22) *(edge between two robots) is three if*

1. $V_i > 0$,

2. $V_j > 0$,

3. *the i^{th} robot, which is measuring the bearing, does not move along the line joining the two robots,*

4. the j^{th} robot does not move perpendicular to the line joining the two robots.

Proof: To prove the lemma, first we write J_i^- and J_j^+ as

$$J_i^- = v_1^\top v_i^{'} = y_{ij}\cos\psi_i - y_{ij}\sin\psi_i, \tag{2.25}$$

$$J_j^+ = v_1^\top v_j = x_{ij}\cos\psi_j + y_{ij}\sin\psi_j, \tag{2.26}$$

where $v_1 \triangleq [x_{ij}\ y_{ij}]^\top$ is a vector along the line between the two robots, $v_j \triangleq [\cos\psi_j\ \sin\psi_j]^\top$ is the heading vector of the j^{th} robot, and $v_i^{'} \triangleq [-\sin\psi_i\ \cos\psi_i]^\top$ is a vector perpendicular to the heading vector of the i^{th} robot.

From (2.25) and (2.26), we can verify that if the i^{th} robot, which is measuring the bearing, does not move along the line joining the two robots then $J_i^- \neq 0$, and if the j^{th} robot does not move perpendicular to the line joining the two robots then $J_j^+ \neq 0$. We then use the elementary operation matrix

$$E_{ij} = \begin{bmatrix} -\frac{c_{\psi_j}J_i^+}{J_i^-J_j^+} & -\frac{c_{\psi_j}R_{ij}^2}{J_i^-J_j^+} & -\frac{y_{ij}}{J_j^+} & 0 & 0 & 0 & 0 \\ -\frac{s_{\psi_j}J_i^+}{J_i^-J_j^+} & -\frac{s_{\psi_j}R_{ij}^2}{J_i^-J_j^+} & \frac{x_{ij}}{J_j^+} & 0 & 0 & 0 & 0 \\ -\frac{s(\psi_j-\psi_i)}{J_i^-J_j^+} & \frac{J_j^-}{J_i^-J_j^+} & \frac{1}{J_j^+} & 0 & 0 & 0 & 0 \\ -\frac{2J_i^+}{J_i^-} & -\frac{2R_{ij}^2}{J_i^-} & 0 & 1 & 0 & 0 & 0 \\ \frac{1}{2}\frac{s(2\psi_i-2\psi_j)}{J_i^-J_j^+} & \frac{c(\psi_j-\psi_i)J_j^-}{J_i^-J_j^+} & \frac{c(\psi_i-\psi_j)}{J_j^+} & 0 & 1 & 0 & 0 \\ \frac{1}{J_i^-} & \frac{J_i^+}{J_i^-} & 0 & 0 & 0 & 1 & 0 \\ -\frac{J_i^+}{J_i^-J_j^+} & -\frac{R_{ij}^2}{J_i^-J_j^+} & -\frac{J_j^-}{J_j^+} & 0 & 0 & 0 & 1 \end{bmatrix},$$

where $s(\psi_j-\psi_i) \triangleq \sin(\psi_j-\psi_i)$, $c(\psi_j-\psi_i) \triangleq \cos(\psi_j-\psi_i)$, and $s(2\psi_i-2\psi_j) \triangleq \sin(2\psi_i-2\psi_j)$. E_{ij} transforms O_{ij} as

$$E_{ij}O_{ij} = U_{ij} = \begin{bmatrix} \mathbf{I}_3 & \bar{O}_{ij} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{4\times3} \end{bmatrix}, \tag{2.27}$$

where

$$\bar{O}^{ij} \triangleq \begin{bmatrix} -1 & 0 & y_{ij} \\ 0 & -1 & -x_{ij} \\ 0 & 0 & -1 \end{bmatrix}. \tag{2.28}$$

From Lemma 2.3.1 we can say that RREF matrix U_{ij} has three linearly independent rows and these rows span the same observability space spanned by rows of O_{ij}, therefore, maximum rank of O_{ij} is three. It should be noted that the top three non-zero rows in U_{ij} corresponds to L^0h, $L^1_{f_{v_i}}h$, and $L^1_{f_{v_j}}h$, therefore, conditions of the Lemma 2.3.2 are the sufficient conditions for $rank(O_{ij}) = 3$. ■

Edge between a robot and a landmark

In this section, we derive the linearly independent rows in the observability matrix for an edge η_{ik} between a robot and a landmark and derive the conditions under which maximum number of linearly independent rows can be obtained. We rearrange the nonlinear kinematic equations in the following convenient form for computing Lie derivatives:

$$\dot{X}_i = f_{v_i} V_i + f_{\omega_i} \omega_i, \tag{2.29}$$

where $f_{v_i} = [c_{\psi_i}\ s_{\psi_i}\ 0]^\top$ and $f_{\omega_i} = [0\ 0\ 1]^\top$. We hereafter compute the necessary Lie derivatives of η_{ik} and their gradients.

Zeroth-order Lie derivative

$$L^0h = \eta_{ik},$$

and its gradient, scaled by R^2_{ik} is given by

$$\nabla L^0h = \begin{bmatrix} -y_{ik} & x_{ik} & -R^2_{ik} \end{bmatrix},$$

where $x_{ik} \triangleq x_i - x_k$, $y_{ij} \triangleq y_i - y_k$, and $R^2_{ik} = (x_{ik})^2 + (y_{ik})^2$.

First-order Lie derivatives

$$L^1_{f_{v_i}} h = x_{ik} s_{\psi_i} - y_{ik} c_{\psi_i},$$
$$L^1_{f_{\omega_i}} h = -R^2_{ik},$$

with gradient given by

$$\nabla L^1_{f_{v_i}} h = \begin{bmatrix} s_{\psi_i} & -c_{\psi_i} & x_{ik} c_{\psi_i} + y_{ik} s_{\psi_i} \end{bmatrix},$$
$$\nabla L^1_{f_{\omega_i}} h = 2 \begin{bmatrix} -x_{ik} & -y_{ik} & 0 \end{bmatrix}.$$

Second-order Lie derivatives

$$L^2_{f_{v_i} f_{v_i}} h = s_{\psi_i} c_{\psi_i} - s_{\psi_i} c_{\psi_i} = 0,$$
$$L^2_{f_{v_i} f_{\omega_i}} h = x_{ik} c_{\psi_i} + y_{ik} s_{\psi_i},$$
$$L^2_{f_{\omega_i} f_{v_i}} h = -2 L^2_{f_{v_i} f_{\omega_i}} h,$$

with gradients given by

$$\nabla L^2_{f_{v_i} f_{\omega_i}} h = \begin{bmatrix} c_{\psi_i} & s_{\psi_i} & y_{ik} c_{\psi_i} - x_{ik} s_{\psi_i} \end{bmatrix}.$$

Remark 2.3.3 *Gradient of $L^2_{f_{\omega_i} f_{v_i}} h$ is not included because it is linearly dependent on $\nabla L^2_{f_{v_i} f_{\omega_i}} h$.*

Third-order Lie derivatives

$$L^3_{f_{v_i} f_{\omega_i} f_{v_i}} h = 1,$$
$$L^3_{f_{v_i} f_{\omega_i} f_{\omega_i}} h = -(x_{ik} s_{\psi_i} - y_{ik} c_{\psi_i}) = -L^1_{f_{v_i}} h.$$

Clearly, the gradients of third and higher order Lie derivatives are linearly dependent on the rows of the observability matrix corresponding to second and lower order Lie derivatives. Therefore, we can write the rows of the observability matrix corresponding to an edge between a robot and a

landmark, using the gradients of Lie derivatives up to second order, as

$$O_{ik} = \begin{bmatrix} -y_{ik} & x_{ik} & -R_{ik}^2 \\ s_{\psi_i} & -c_{\psi_i} & J^+ \\ -2x_{ik} & -2y_{ik} & 0 \\ c_{\psi_i} & s_{\psi_i} & J^- \end{bmatrix}, \tag{2.30}$$

where $J^+ \triangleq x_{ik}c_{\psi_i} + y_{ik}s_{\psi_i}$ and $J^- \triangleq y_{ik}c_{\psi_i} - x_{ik}s_{\psi_i}$.

Lemma 2.3.3 *The rank O_{ik} given by* (2.30)*(edge between a robot and a landmark) is two if*

1. *$V_i > 0$,*
2. *the robot does not move along the line joining the robot and the landmark.*

Proof: If the robot does not move along the line joining the robot and the landmark then $J^- \neq 0$ (see proof of Lemma 2.3.2) and the elementary operation matrix

$$E_{ik} = \begin{bmatrix} \frac{-c_{\psi_i}}{J^-} & \frac{-x_{ik}}{J^-} & 0 & 0 \\ \frac{-s_{\psi_i}}{J^-} & \frac{-y_{ik}}{J^-} & 0 & 0 \\ \frac{-2J^+}{J^-} & \frac{-2R_{ik}^2}{J^-} & 1 & 0 \\ \frac{1}{J^-} & \frac{J^+}{J^-} & 0 & 1 \end{bmatrix},$$

transforms O_{ik} as

$$E_{ik}O_{ik} = U_{ik} = \begin{bmatrix} \bar{O}l_{ik} \\ \mathbf{0}_{2\times 3} \end{bmatrix}, \tag{2.31}$$

where

$$\bar{O}l_{ik} \triangleq \begin{bmatrix} 1 & 0 & y_{ik} \\ 0 & 1 & -x_{ik} \end{bmatrix}. \tag{2.32}$$

It should be noted that the top two non-zero rows in the observability matrix are linearly independent (from Lemma 2.3.1) and they correspond to L^0h and $L^1_{f_{v_i}}h$. Therefore, $V_i > 0$ and $J^- \neq 0$ are the sufficient conditions for the rank of the observability matrix being two. ■

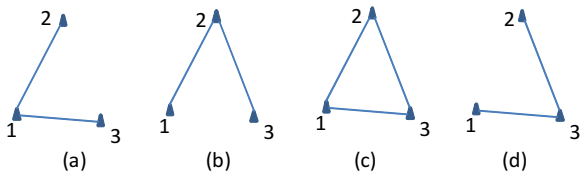


Figure 2.2: The observability conditions between these four possible configurations of a connected, 3-node RPMG are identical.

Definition 2.3.1 *An RPMG G_n^l (Definition 2.2.1) is called a proper RPMG if all of the edges between robot nodes satisfy the conditions of Lemma 2.3.2 and all the edges between robots and landmarks satisfy Lemma 2.3.3.*

In a proper RPM each edge η_{ij} between two robots contribute three linearly independent rows to the observability matrix of a proper RPMG and each edge η_{ik} between a robot and a landmark contributes two linearly independent rows to the observability matrix of a proper RPMG. Using three linearly independent rows of U_{ij} in (2.27) and two linearly independent rows of U_{ik} in (2.31), we can write the observability matrix of a proper RPMG G_n^l as

$$O^p = \left\{ \begin{array}{c} O_{ij}^p \\ O_{ik}^p \end{array} \right\}, \; i,j \in \mathscr{V}_n, \; k \in \mathscr{V}_l, \tag{2.33}$$

where $O_{ij}^p \triangleq \left[0_{3\times 3(i-1)} \; \mathbf{I}_3 \; 0_{3\times(3(j-1)-3i)} \; \bar{O}_{ij} \; 0_{3\times 3(n-j)}\right]$ and $O_{ik}^p \triangleq \left[0_{2\times 3(i-1)}) \; \bar{O}l_{ik} \; 0_{2\times 3(n-i)}\right]$.

Remark 2.3.4 *The observability matrix O^p in* (2.33) *is not the original observability matrix of the graph G_n^l. Since the rows of* (2.33) *consist of the linearly independent rows after elementary row operations, from Lemma 2.3.1 we know that the rows of the observability matrix in* (2.33) *span the same observable space spanned by the original observability matrix.*

2.3.2 Observability analysis without landmarks

In this section we derive the conditions for achieving the maximum rank of the observability matrix for the graph G_n^0 without landmarks. We first discuss the observability properties for a $3-$ node graph G_3^0.

Lemma 2.3.4 *If a three node proper RPMG G_3^0 is connected, then the rank of the observability matrix is six.*

Proof: There are four possible configurations of a connected graph G_3^0, shown as sub-figures (a) through (d) in Fig. 2.2. We can write the transformed observability matrix for these configurations using (2.33) as

$$O_a^p = \begin{bmatrix} \mathbf{I}_3 & \bar{O}_{12} & \mathbf{0} \\ \mathbf{I}_3 & \mathbf{0} & \bar{O}_{13} \end{bmatrix}, O_b^p = \begin{bmatrix} \mathbf{I}_3 & \bar{O}_{12} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_3 & \bar{O}_{23} \end{bmatrix},$$
$$O_c^p = \begin{bmatrix} \mathbf{I}_3 & \bar{O}_{12} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_3 & \bar{O}_{23} \\ \mathbf{I}_3 & \mathbf{0} & \bar{O}_{13} \end{bmatrix}, O_d^p = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} & \bar{O}_{13} \\ \mathbf{0} & \mathbf{I}_3 & \bar{O}_{23} \end{bmatrix}.$$

We perform elementary operation on O_a^p, O_b^p, O_c^p, and O_d^p by multiplying them by elementary operation matrices $E_a = \begin{bmatrix} \mathbf{0}_3 & \mathbf{I}_3 \\ \bar{O}_{12} & -\bar{O}_{12} \end{bmatrix}$, $E_b = \begin{bmatrix} \mathbf{I}_3 & -\bar{O}_{12} \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix}$, $E_c = \begin{bmatrix} \mathbf{I}_3 & -\bar{O}_{12} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ -\mathbf{I}_3 & \bar{O}_{12} & \mathbf{I}_3 \end{bmatrix}$, and $E_d = \mathbf{I}_6$ respectively to get

$$E_a O_a^p = E_b O_b^p = E_d O_d^p = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} & \bar{O}_{13} \\ \mathbf{0} & \mathbf{I}_3 & \bar{O}_{23} \end{bmatrix},$$
$$E_c O_c^p = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} & \bar{O}_{13} \\ \mathbf{0} & \mathbf{I}_3 & \bar{O}_{23} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

Therefore, Lemma 2.3.1 implies that the observability sub-matrix of all the four configurations have six linearly independent rows and that these rows span the same observable space. ■

Remark 2.3.5 *The elementary operation matrix E_d for observability matrix O_d^p in Lemma 2.3.4 is Identity because O_d^p is already in a reduced row echelon form.*

From Lemma 2.3.4 we can say that the rows of two edges for a proper RPMG with a common node are independent. The following lemma is an extension of this idea.

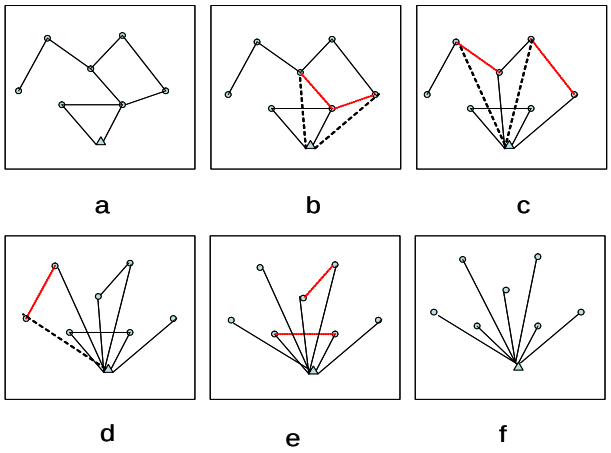


Figure 2.3: An example of converting an arbitrary connected RPMG to a 2-level tree.

Lemma 2.3.5 *If a graph G_n^0 is a proper RPMG and has the form of a 2-level tree (see figure 2.3(f)) which consists of a root node and $n-1$ leafs directly connected to the root, then the rank of the associated observability matrix is $3(n-1)$.*

Proof: Without loss of generality, assume that the root node of the 2-level tree is labelled as n. The system observability matrix will then be of the form

$$O^p_{2-level} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} & \mathbf{0} & \cdots & \bar{O}_{1n} \\ \mathbf{0} & \mathbf{I}_3 & \mathbf{0} & \cdots & \bar{O}_{2n} \\ \vdots & & \ddots & & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{I}_3 & \bar{O}_{n-1,n} \end{bmatrix}. \tag{2.34}$$

Clearly, the $rank(O^p_{2-level}) = 3(n-1)$. ■

Theorem 2.3.1 *If the graph G_n^0 is a proper connected RPMG then the rank of the associated observability matrix is $3(n-1)$.*

Proof: Using Lemma 2.3.4, any connected 3-node subgraph in the larger graph can be replaced with any other connected 3-node subgraph, without affecting the rank of the system observability matrix because their associated observability sub-matrices span the same observable sub-space. A connected graph G_n^0 can be transformed to a 2-level tree using following algorithm.

1. Choose any node and label it as the root as shown in Fig. 2.3 (a).

2. Select the nodes whose path from the root consists of two edges(three nodes including root) as shown in Fig. 2.3 (b). Each such path can be represented as a 3-node subgraph G_3^0.

3. If a valid 3-node subgraph is found, perform a subgraph replacement (see Fig. 2.3 (c)) so that nodes j and i are both a distance of one away from the root node and repeat step 2. If a valid subgraph G_3^0 is not found, continue to step 4.

4. Search for a three node subgraph G_3^0 that includes the root node and two nodes (nodes j and i) distance one away from the root node that contain an edge between these two nodes (see Fig. 2.3 (d)).

5. If a valid 3-node subgraph was found, perform a subgraph replacement that maintains the edges between the root node and nodes i and j, but removes the edge between nodes i and j (see Fig. 2.3 (e)). Repeat step 4 until G_0^n is transformed into a 2-level tree.

To show that this algorithm transforms a connected proper RPMG to a 2-level tree, first consider steps 2 and 3. Every time step 2 finds a valid subgraph, the distance of node i to the root node will be decreased from two to one. Because the graph is connected, the root node will be connected to any other node within a finite number of steps. Therefore, as steps 2 and 3 continue to execute, all nodes will be brought to a maximum of distance one away from the root node. This is similar to the graph shown in Fig. 2.3(e). Steps 4 and 5 simply remove any redundant edges. Therefore, this algorithm converts any connected graph to a 2-level tree. The algorithm is also a recursive way of performing elementary row operations on the rows of the observability matrices of sub-graphs G_3^0 to show that the observability matrix of the connected and proper RPMG G_n^0 is equivalent to the observability matrix of a 2-level tree. Furthermore, we can say that the basis of the observable space of a connected proper RPMG G_n^0 is the rows of $O_{2-level}^p$ and from Lemma 2.3.5, the rank of the observability matrix is $3(n-1)$. ■

2.3.3 Observability analysis with known landmarks

In this subsection, we assume that landmarks of known location are observed by robots within the network, providing information about the global coordinate system. We derive condi-

tions for complete observability of the graph G_n^l. First we derive the conditions for the observability of a single robot.

Lemma 2.3.6 *The rank of the observability matrix of a proper RPMG G_1^l (one robot and l landmarks) is three if there are at least two landmarks ($l \geq 2$) and the robot and two landmarks are not on the same line (i.e., $\eta_{i1} \neq \eta_{i2}$).*

Proof: Consider a proper RPMG G_1^2 with one vehicle and two landmarks such that $\eta_{i1} \neq \eta_{i2}$. Using (2.33) the observability matrix of graph G_1^2 can be written as, $O_{i12}^p = \begin{bmatrix} \bar{O}l_{i1} \\ \bar{O}l_{i2} \end{bmatrix}$. To find the number of linearly independent rows we perform elementary row operations on O by multiplying by $E_{i12} = \begin{bmatrix} \frac{y_{i2}}{y_{12}} & 0 & -\frac{y_{i1}}{y_{12}} & 0 \\ -\frac{x_{i1}}{y_{12}} & 1 & \frac{x_{i1}}{y_{12}} & 0 \\ -\frac{1}{y_{12}} & 0 & \frac{1}{y_{12}} & 0 \\ -\frac{x_{12}}{y_{12}} & -1 & -\frac{x_{12}}{y_{12}} & 1 \end{bmatrix}$, to obtain

$$E_{i12}O_{i12}^p = \begin{bmatrix} \mathbf{I}_3 \\ \mathbf{0}_{1\times 3} \end{bmatrix}.$$

This implies that two different landmarks provides three independent rows to the observability matrix. Therefore, from Theorem 2.2.1 the single robot states are completely observable, i.e., $rank(O) = 3$. ■

From Lemma 2.3.6 we know that, if all of the n vehicles in the group are directly connected to two different landmark, then the system is completely observable ($rank(O) = 3n$). However, due to limited sensor range and bearing all of the vehicles in the group may not be able to see two landmarks. The following lemmas and theorem show how cooperative localization can overcome constraints posed by sensor limitations.

Lemma 2.3.7 *Given a 3-node RPMG G_2^1 with two robots and one landmark, if the graph G_2^1 is proper then the rows of the observability matrix of the two configurations of G_2^1 shown in Fig. 2.4(a) and Fig. 2.4(b) spans the same observable space.*

Proof: The two configurations shown in Fig. 2.4(a) and Fig. 2.4(b) only differs in the landmark l connection. In configuration (a) the landmark is directly connected to the i^{th} node whereas in

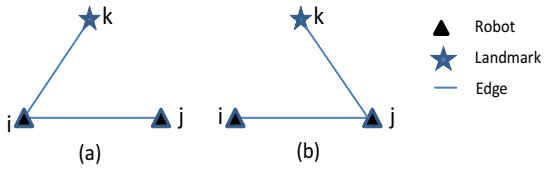


Figure 2.4: Two robots and one landmark RPMG G_2^1.

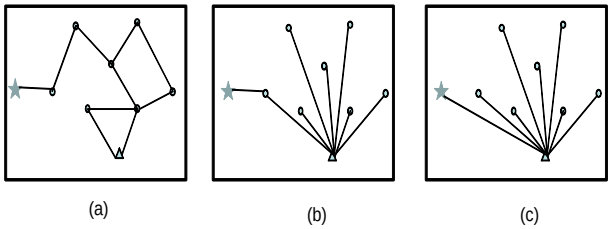


Figure 2.5: An example of converting an arbitrary connected RPMG with landmark to a two-level tree.

configuration (b) the landmark is directly connected to the j^{th} node. The observability matrix for the configurations shown in Fig. 2.4(a) and (b) can be written using (2.33) as

$$O_a^p = \begin{bmatrix} \mathbf{I}_3 & \bar{O}_{ij} \\ \bar{O}l_{ik} & 0_{2\times 3} \end{bmatrix}, \; O_b^p = \begin{bmatrix} \mathbf{I}_3 & \bar{O}_{ik} \\ 0_{2\times 3} & \bar{O}l_{jk} \end{bmatrix}. \tag{2.35}$$

We can perform elementary operation on O_a by multiplying by the elementary operation matrix $E_{ijk} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_{3\times 2} \\ -\bar{O}l_{ik} & \mathbf{I}_2 \end{bmatrix}$ to show that $E_{ijk}O_a^p = O_b^p$. Therefore, from Lemma 2.3.1 we can say that the observability matrix of both the configurations spans the same space. ■

Lemma 2.3.8 *Given the RPMG G_n^l , if it is proper and connected then the associated observability matrix is equivalent (observability matrix of both graphs span the same space) to the observability matrix of a 2-level tree.*

Proof: Consider a connected proper RPMG G_n^l(example for $l = 1$ is shown in Fig. 2.5(a)). We can write the observability matrix for the graph G_n^l using (2.33). To show that the observability matrix of a connected proper RPMG G_n^l is equivalent (observability matrix of both graphs span the same space) to the observability matrix of a 2-level tree we perform following steps.

- First we perform elementary operations only on the rows of the observability matrix of the edges between robots (subgraph G_n^0). From Theorem 2.3.1 we know that these operations leads to the observability matrix of a 2-level tree. Therefore, the resulting observability matrix of the graph G_n^l is equivalent to the observability of the 3-level tree (Example shown in Fig 2.5(b)) with landmarks on level three. In the 3-level tree the path between each landmark and the root consists of two edges, including edge η_{ij} between the root and the i^{th} node (landmark is directly connected to i^{th} node) and edge η_{ik} between k^{th} landmark and the i^{th} node. The path between each landmark and the root can be represented as a subgraph G_2^1 with two robots and one landmark. There are l such subgraphs in the graph G_n^l.

- Next we perform elementary operations on the rows of the observability matrix of each subgraph G_2^1 associated with each landmark. From Lemma 2.3.7 we know that the resulting observability matrix of the graph G_n^l is equivalent to the observability matrix of a two-level tree (see Fig. 2.5(c)) with $n-1$ robot-to-robot leafs and l leafs between root and landmarks.

■

Theorem 2.3.2 *Given a proper RPMG G_n^l, if for each robot there exists a path to at least two landmarks and the robot and two landmarks are not on the same line (i.e., $\eta_{i1} \neq \eta_{i2}, \forall\, i \in \mathcal{V}_n$) then the system is completely observable, i.e., the rank of the observability matrix is $3n$.*

Proof: There are two scenarios for paths between landmarks. (1) All the robots are directly connected to the landmarks. For this case, from Lemma 2.3.6 we know that the proper RPMG G_n^l is completely observable if $l \geq 2$ and $\eta_{i1} \neq \eta_{i2}, \forall\, i \in \mathcal{V}_n$. (2) Consider the general case where the proper RPMG G_n^l is connected and only a subset of nodes measure landmarks. In this case all of the robots are not directly connected to landmarks, but there exists a path between each robot and the landmarks. From Lemma 2.3.8 we know that the observability matrix of a connected proper RPMG G_n^l is equivalent to the observability matrix of a 2-level tree. Therefore connected proper RPMG G_n^l can be replaced by a two-level tree with $n-1$ robot to robot leafs and l leafs between the root and landmarks. Furthermore, from Lemma 2.3.5 we know that for a 2-level tree all of the $3(n-1)$ rows of the observability matrix of $n-1$ edges between all robots are linearly independent. Therefore, we only consider a subgraph (two-level) G_2^2 which consists of three edges, including an

edge η_{ij} between the root (j^{th} node) and i^{th} node and two edges η_{j1} and η_{j2} between the root and two landmarks. Using (2.33) the observability matrix for this sub-graph G_2^2 is given by

$$O^p = \begin{bmatrix} \mathbf{I}_3 & \bar{O}_{ij} \\ 0_{2\times 3} & \bar{O}l_{j1} \\ 0_{2\times 3} & \bar{O}l_{j2} \end{bmatrix}. \tag{2.36}$$

We perform elementary operation on (2.36) using

$$E_{ij12} = \begin{bmatrix} 1 & 0 & 0 & \frac{y_{i2}}{y_{12}} & 0 & -\frac{y_{i1}}{y_{12}} & 0 \\ 0 & 1 & 0 & -\frac{x_{i1}}{y_{12}} & 1 & \frac{x_{i1}}{y_{12}} & 0 \\ 0 & 0 & 1 & -\frac{1}{y_{12}} & 0 & \frac{1}{y_{12}} & 0 \\ 0 & 0 & 0 & \frac{y_{j2}}{y_{12}} & 0 & -\frac{y_{j1}}{y_{12}} & 0 \\ 0 & 0 & 0 & -\frac{x_{j1}}{y_{12}} & 1 & \frac{x_{j1}}{y_{12}} & 0 \\ 0 & 0 & 0 & -\frac{1}{y_{12}} & 0 & \frac{1}{y_{12}} & 0 \\ 0 & 0 & 0 & -\frac{x_{12}}{y_{12}} & -1 & \frac{x_{12}}{y_{12}} & 1 \end{bmatrix},$$

to obtain the reduced row echelon form

$$E_{ij12}O^p = \begin{bmatrix} \mathbf{I}_6 \\ \mathbf{0}_{1\times 6} \end{bmatrix}. \tag{2.37}$$

This implies that two landmarks add three linearly independent rows to the observability matrix of graph G_n^2, which are linearly independent to the $3(n-1)$ existing rows. Therefore, the rank of the observability matrix for the RPMG G_n^l with $l = 2$ is $3n$. ■

Corollary 2.3.1 *Given a RPMG G_n^0, if it is proper, connected, and one of the robot has its position and heading measurement from GPS then the system is completely observable, i.e., the rank of the observability matrix is $3n$.*

Proof: If RPMG G_n^0 is proper and connected then from Lemma 2.3.5 there are $3(n-1)$ linearly independent rows in the observability matrix and if one of the robot measures its position and heading directly from a GPS receiver then three linearly independent rows, which are linearly

independent to the existing $3(n-1)$ rows, are added. Therefore, the rank of the observability matrix becomes $3n$ and system becomes completely observable. ■

2.4 Results

In this section, we present simulation and hardware results to validate the observability conditions obtained earlier in this chapter.

2.4.1 Simulation results

The simulation environment consists of five homogenous ground robots and two landmarks. Each robot is equipped with an encoder, which provides the robot's linear and rotational velocities, and an omnidirectional camera, which measures bearing from other robots and landmarks in the sensor range R_{sensor}. We assume that the noise in encoder and camera measurement is Gaussian. Following are the different simulation parameters of a robot used in the simulation results presented in this chapter.

- Sensing range of the omni-directional camera ($R_{sensor} = 80m$).
- Linear velocity of the robot ($V = 5\ m/s$).
- Sampling time period $T_s = 1/30\ s$.
- Initial pose uncertainty ($\sigma_{x0}\ \sigma_{y0}\ \sigma_{\psi_0}] = [5m\ 5m\ 0.2rad]$).
- Standard deviations of process noise in encoder $[\sigma_v\ \sigma_\omega]^T = [0.2m/s\ 0.2rad/s]^T$.
- Standard deviation of measurement noise $\sigma_\eta = 0.2rad$.

In Fig. 2.6, we plot the smallest singular value σ_{min} of the observability matrix and $\beta(G_5^2)$, where $\beta(G_5^2) = 1$ if conditions of Theorem 2.3.2 are satisfied otherwise $\beta(G_5^2) = 0$. It can be seen that when $\beta(G_5^2) = 1$ then each robot node in RPMG has path to two landmarks and the system is completely observable. On the other hand, when $\beta(G_5^2) = 0$ then all of the robots in the do not have path to two landmarks and system is unobservable. This numerically proves the Theorem 2.3.2.

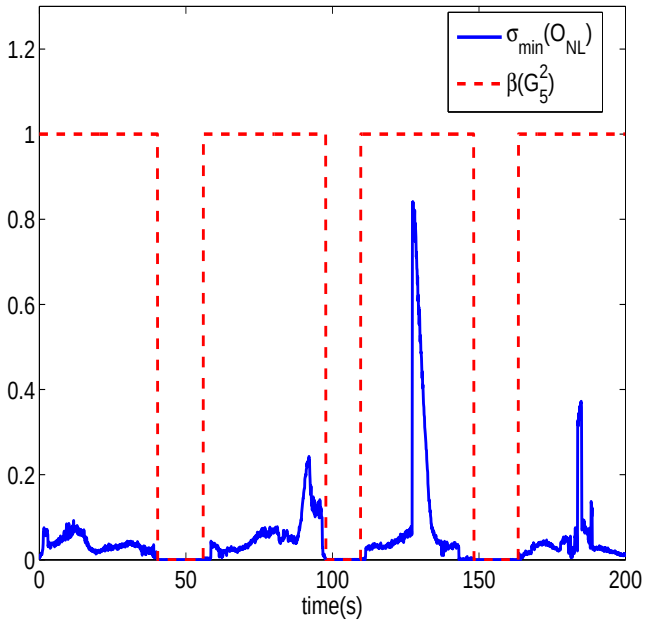


Figure 2.6: Connectivity vs Observability: The dashed red curve represents the connectivity, of a RPMG with five robots and two landmarks. The solid blue curve represents the smallest singular value of the nonlinear observability matrix as defined in (2.20). When $\beta(G_5^2) = 1$ then each robot node in RPMG has path to two landmarks and the system is completely observable. On the other hand, when $\beta(G_5^2) = 0$ then all of the robots in the do not have path to two landmarks and system is unobservable.

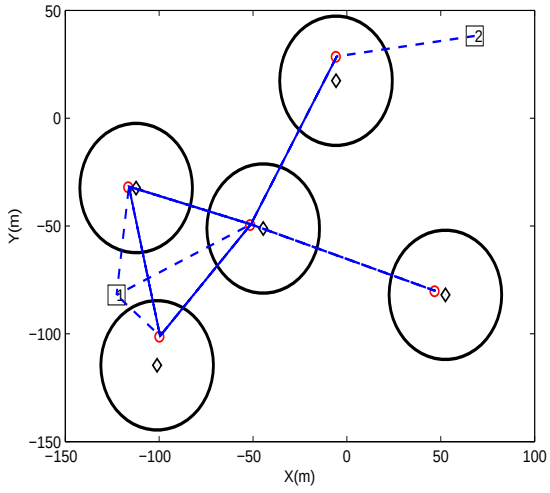


Figure 2.7: Initial topology for bearing-only cooperative localization. The black circles represent the initial position uncertainty (3σ). The black diamonds represent the initial position estimates of robots. The red circles are the true initial positions of the robots. The dashed blue line represents an edge (bearing measurement) between two nodes. Two numbered squares are the two known landmarks.

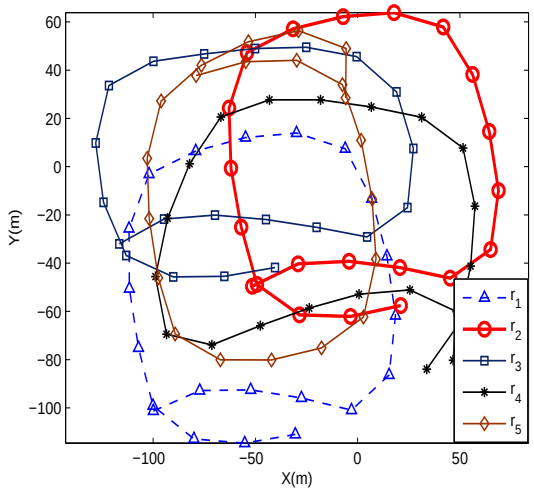


Figure 2.8: True trajectories of all of the five robots. The trajectories are generated by following way-points, and the way-points are chosen such that the RPMG G_5^2 remains connected.

We run the bearing-only cooperative localization algorithm for four different cases. In the first case the measurements from the landmarks are not used for estimation. In the second case bearing measurements from only one landmark are considered, and in the third case bearing measurements from two landmarks are considered. Finally, we do not consider any landmarks, however, the absolute position and heading of a robot from a GPS receiver are fused with inter robot bearing measurements to perform the cooperative localization. The Fig. 2.7 shows the initial topology of the RPMG with five robots and two landmarks. Initially, the RPMG is connected and the waypoints for all of the five robots are chosen such that the RPMG is remains connected. The actual trajectories of all of the five robots is shown in Fig. 2.8, and the Fig 2.9 shows the true and estimated trajectories of the second robot, for all of the four cases. It can be seen that estimated trajectories for CL with two landmarks and GPS (first robot) are closest to the true trajectory. The comparison for position error (second robot) for all the four cases is shown in Fig 2.10 and Fig. 2.11.

Fig 2.12, Fig. 2.13, and Fig. 2.14 show the plots for the second robot's estimated uncertainty (3σ) in x, y, and ψ respectively for all of the four cases. It can be seen that the uncertainty for two landmarks and GPS case is lower then the uncertainty related to no landmark and one landmark.

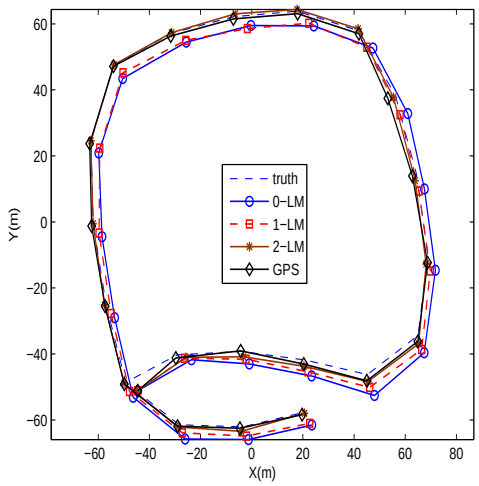


Figure 2.9: The second robot trajectories:(1) The true trajectory is represented by the dashed curve. (2) The estimated trajectory using CL with no landmarks is represented by solid curve with circles.(3) The estimated trajectory using CL with only one landmarks is represented by the solid curve with squares. (4)The estimated trajectory using CL with two landmarks is represented by the solid curve with stars. (5) The estimated trajectory using CL, when Robot 1's position and attitude is measured using a GPS, is represented by the solid curve with diamonds.

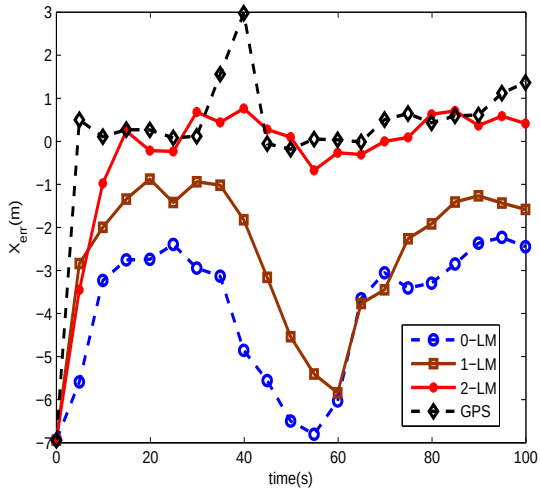


Figure 2.10: The comparison of second robot's error in x direction. This figure shows the plots of error in x direction with no landmark, one landmark, two landmark, and with GPS measurements of the Robot 1. The error with no landmark and one landmark is higher than error with two landmarks and GPS because in last two cases the system is observable.

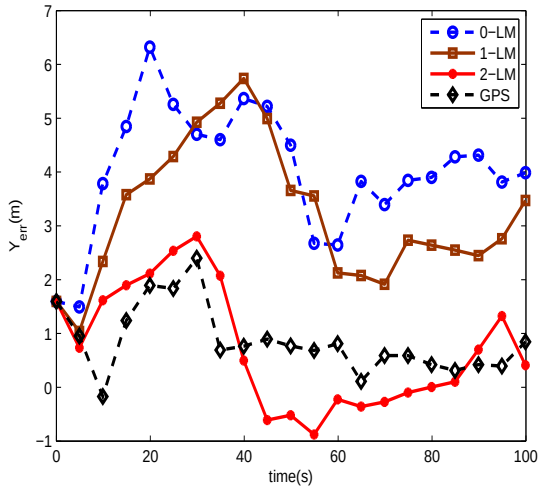


Figure 2.11: The comparison of second robot's error in y direction. This figure shows the plots of error in y direction with no landmark, one landmark, two landmark, and with GPS measurements of the Robot 1. The error with no landmark and one landmark is higher than error with two landmarks and GPS because in last two case the system is observable.

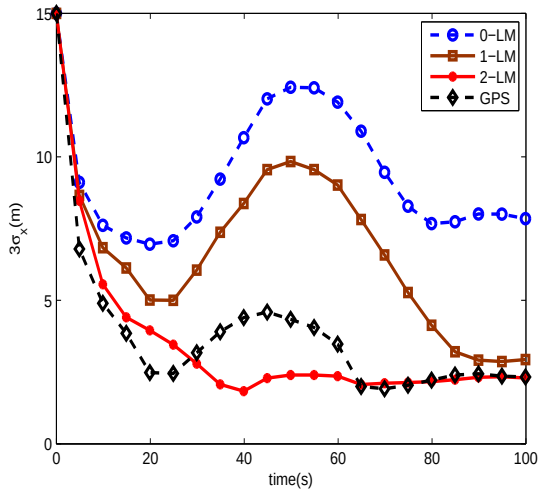


Figure 2.12: The comparison of second robot's estimation uncertainty($3\sigma_x$) in x direction. This figure shows the plots of error covariance in x direction with no landmark, one landmark, two landmark, and with GPS measurements of the Robot 1.

This is because with two landmarks and absolute position and heading measurement of a robot from GPS the system is observable.

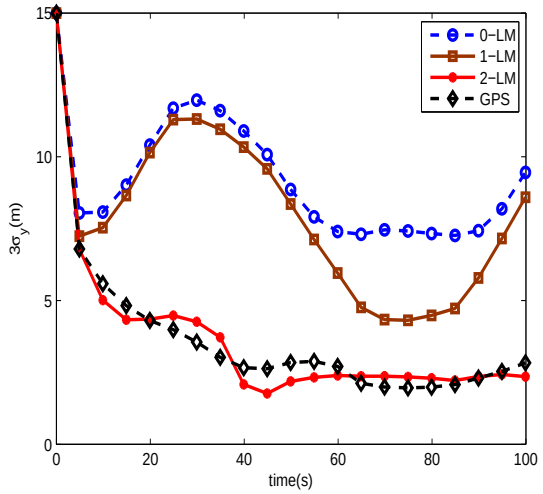


Figure 2.13: The comparison of second robot's estimation uncertainty ($3\sigma_y$) in y direction. This figure shows the plots of error covariance in y direction with no landmark, one landmark, two landmark, and with GPS measurements of the Robot 1.

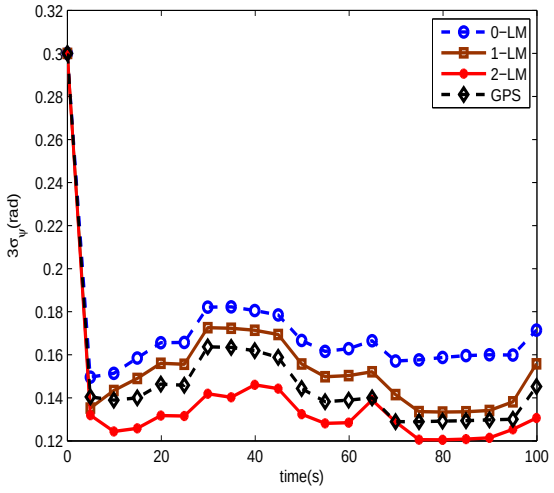


Figure 2.14: The comparison of second robot's estimation uncertainty ($3\sigma_\psi$) in heading. This figure shows the plots of error covariance in ψ direction with no landmark, one landmark, two landmark, and with GPS measurements of the Robot 1.

2.4.2 Experimental results

For experimental validation we use three stinger robots (see Fig. 2.15). For turn rate and velocity measurement wheel encoders are used. For bearing measurement we use an omnidirectional camera and an EEE pc for onboard processing and communication. The experimental setup is shown in Fig. 2.16. It consist of three robots with different color (green, blue, and orange) for their identification. The snapshot of the experimental area taken from the omnidirectional camera on the orange robot is shown in Fig. 2.17. The robots communicate with each other on a wireless network using a router. We use an overhead camera to obtain the true robot states and compare the estimated states. We use color segmentation to find the bearing of the robots and landmarks which are in the image plane of a omnidirectional camera.

To do cooperative localization, the robots needed a reliable method of exchanging information with one another. To meet this need we designed a software system that enabled multiple agents to discover and reliably communicate with one other over a small local area network. We called this system the Agent Management System. In the Agent Management System each robot was represented by a single software construct we called a Software Agent. The Software Agent was a single process, run on the robot's onboard computer that concurrently managed the robot's processing and communication. There were two core elements that enabled the Software Agents to communicate reliably. The first element was peer discovery. The second was message handling. Peer discovery enabled the Software Agents to find an active directory of all other agents that it could communicate with. As soon as a Software Agent process started, it began listening for and periodically broadcasting a short discovery message across the local area network. When another Software Agent heard one of these messages, it sent back a reply informing the sending agent its contact information (an IP address and port number). Within a short period of time all software agents had a list of every active agent on the network. Each agent would continue pinging each other (at a less frequent interval) if the neighboring robot failed to reply to a certain number of pings, it was removed from the list of available robot's to communicate with. With a reliable list of neighboring agents to communicate with, the agents were able to start sending information back and forth between each other. Each message was serialized and sent as a single UDP packet. When received, the Software Agent would put the messages in a single message queue that could be read from and processed. We elected to use the UDP protocol because we needed small rapid measure-

Figure 2.15: The Stinger robot. This robot is equipped with a controller, two wheel encoders, an omnidirectional camera, and Asus EEE computer.

ments from each robot in order to make good cooperative estimates. Using the Agent Management System, we were able to design, test, and record our experiment for cooperative navigation and control.

The following parameters are used for the experiment

- Linear velocity of the robot ($V = 0.2\ m/s$).
- Sampling time period $T_s = 1/30\ s$.
- Standard deviations of process noise in encoder $[\sigma_v\ \sigma_\omega]^T = [0.08m/s\ 0.12rad/s]^T$.
- Standard deviation of measurement noise $\sigma_{\eta_{ij}} = 0.13rad$.

Fig. 2.18 shows the trajectory of all of the three robots, which are computed (1) using only encoders, (2)using bearing-only cooperative localization, and (3)using the overhead camera. It can be seen that the trajectories computed using bearing only cooperative localization are closer to the trajectories computed using the overhead camera. The estimation error plots (x, y, and ψ) of the blue robot are shown in Fig. 2.19, 2.20, and 2.21. It can be seen that the estimation error of all of the states using bearing-only cooperative localization with two landmarks is bounded, however, the estimation error using only encoder measurement drifts.

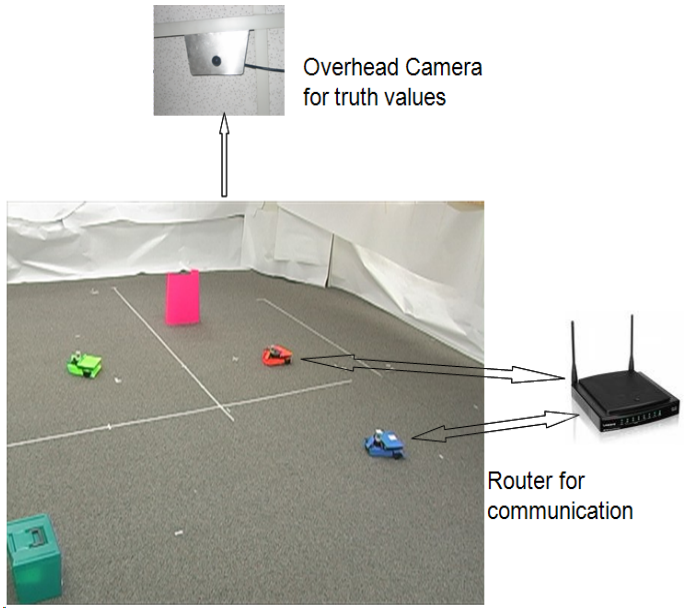


Figure 2.16: Experimental setup with three ground robots (orange, light green, and blue) and two landmarks (pink and dark green boxes). The robots and the landmarks are identified from their color. The robots communicate with each using the TCP-IP protocol over a communication router. Each robot shares its bearing measurements (measured using omni-directional camera) and linear and rotational velocities (measured by two wheel encoders) with the the other robots in the group. The truth position and heading of all of the robots computed using the overhead camera.

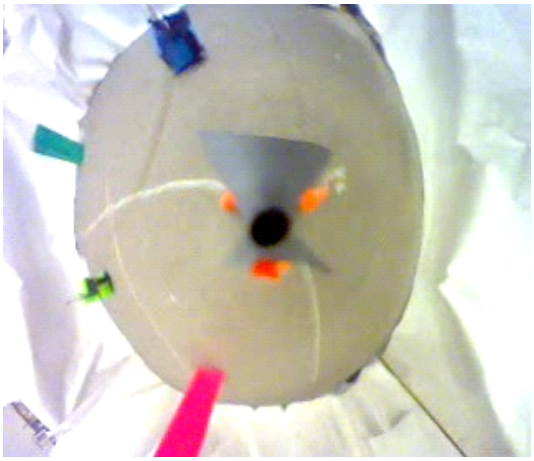


Figure 2.17: This figure shows the omnidirectional camera snap shot taken from the orange robot. The snap shot shows the 360^o view of the the experimental area. It can be seen that other two robots and two landmarks are clearly visible in the image plane. The bearing from the robots and landmarks in the image plane can computed using color segmentation.

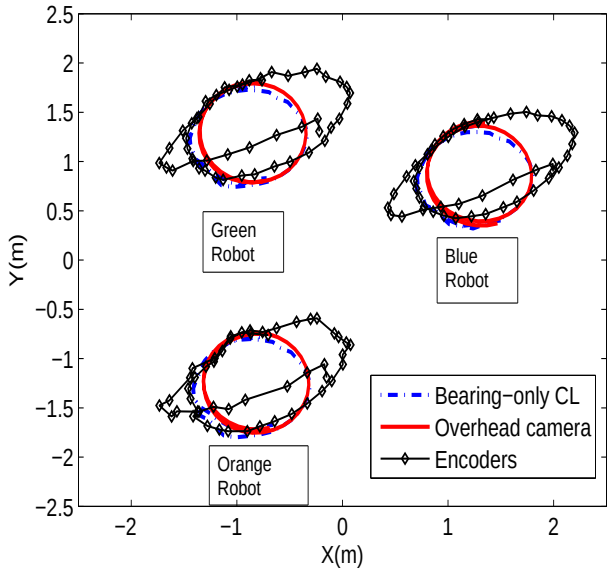


Figure 2.18: Trajectories of three robots. There are three different trajectories for each robot including trajectory computed from over head camera (red solid curve), estimated trajectory using encoder measurement only (black solid curve with diamonds), and estimated trajectory using bearing-only cooperative localization (blue dotted curve).

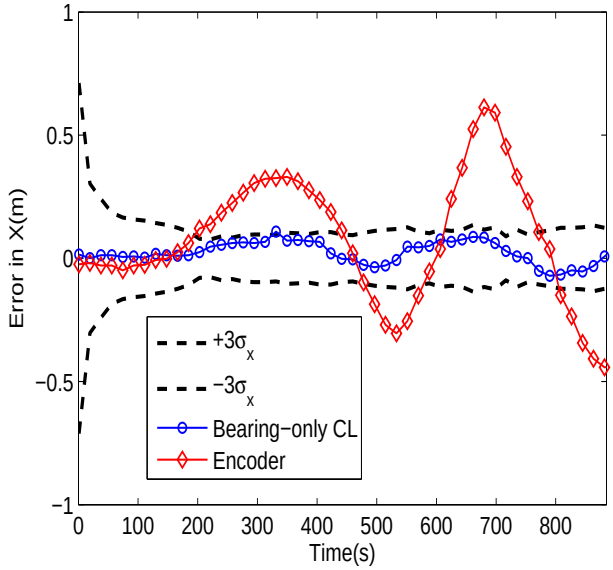


Figure 2.19: Error in x direction of the blue robot:The dashed black curve represents the $\pm 3\sigma_x$ uncertainty. The estimation error in x direction using encoder measurements is represented by the red solid curve with diamonds, and the estimation in x direction using bearing-only measurements is represented by the blue solid curve with circles.

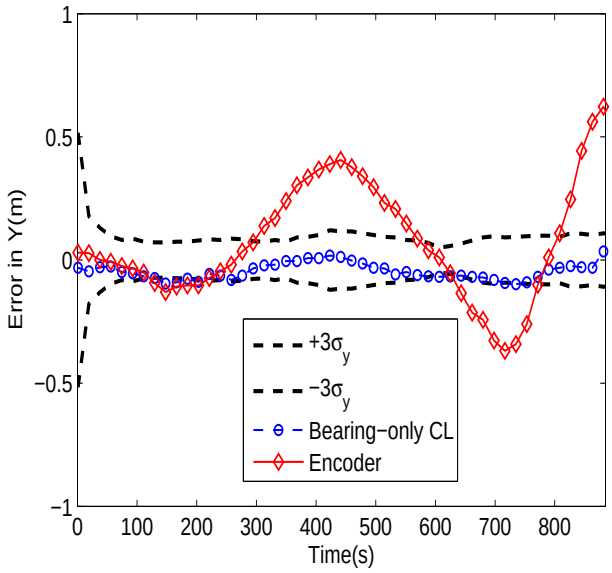


Figure 2.20: Error in y direction of the blue robot:The dashed black curve represents the $\pm 3\sigma_y$ uncertainty. The estimation error in y direction using encoder measurements is represented by the red solid curve with diamonds, and the estimation in y direction using bearing-only measurements is represented by the blue solid curve with circles.

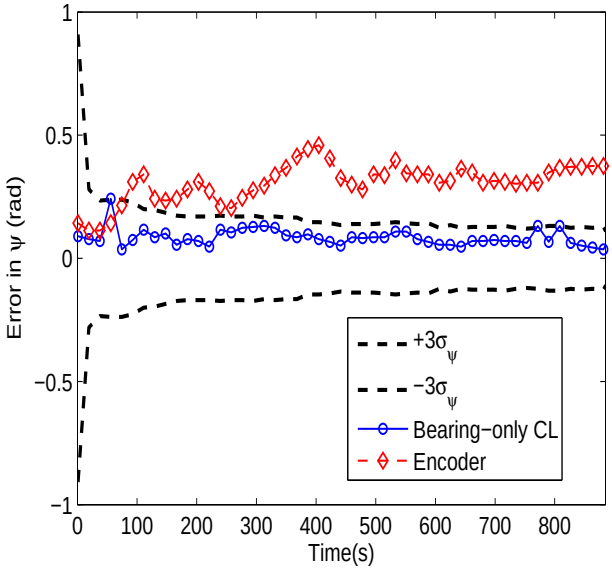


Figure 2.21: Error in heading of the blue robot:The dashed black curve represents the $\pm 3\sigma_\psi$ uncertainty. The estimation error in heading using encoder measurements is represented by the red solid curve with diamonds, and the estimation error in heading using bearing-only measurements is represented by the blue solid curve with circles.

Additional videos of simulation and experimental results related to bearing-only cooperative localization can be found in [78].

2.5 Conclusion

In this chapter we have shown that the observability properties of a system performing cooperative localization can be characterized by the properties of its relative position measurement graph (RPMG). Using graph theoretic properties and nonlinear observability theory, we have shown that for a connected proper RPMG G_n^0 without landmarks, the maximum rank of the observability matrix is $3(n-1)$. Furthermore, we have shown that to achieve full observability, all nodes in the graph must have a path to at least two different landmarks of known location. We have validated the theoretical observability condition by simulation and experimental results.

CHAPTER 3. BEARING-ONLY COOPERATIVE GEO-LOCALIZATION

In Chapter 2, we have derived the conditions for the complete observability of bearing-only cooperative localization for ground robots. In this chapter, we extend the results of Chapter 2 to obtain the observability conditions for bearing-only cooperative localization for Unmanned Aerial Vehicles (UAVs). Also, we use bearing-only localization to develop a cooperative technique to geo-localize a target in an urban terrain.

3.1 Introduction

Recently, there has been an increase in the use of Unmanned Aerial Vehicles (UAVs) in several military and civil application that are considered dangerous for human pilots. These applications include surveillance [23], reconnaissance [24], search [25], and fire monitoring [26, 27]. Among the suite of possible sensors, a video camera is inexpensive, lightweight, fits the physical requirements of small UAVs, and has a high information to weight ratio. One of the important applications of camera equipped UAVs is determining the location of a ground target when imaged from UAVs. The pixel location of the target in the image, the position and attitude of the air vehicles, the camera's pose angles, and knowledge of the terrain elevation are used to geo-localize the object. Previous target localization work using a camera equipped UAV is reported in [55–58] and references therein. Barber *et al.* [56] used a camera, mounted on a fixed-wing UAV, to geo-localize a stationary target. They discussed recursive least square (RLS) filtering, bais estimation, flight path selection, and wind estimation to reduce the localization errors. Pachter *et al.* [55] developed a vision-based target geo-location technique that uses camera equipped unmanned air vehicles. They jointly estimate the targets position and the vehicles's attitude errors using linear regression which results in improved target geo-localization. A salient feature of target geo-localization using bearing and range based sensors is the dependence of the measurement uncertainty on the position of the sensor relative to the target. Therefore, the influence of input parameters on nonlinear

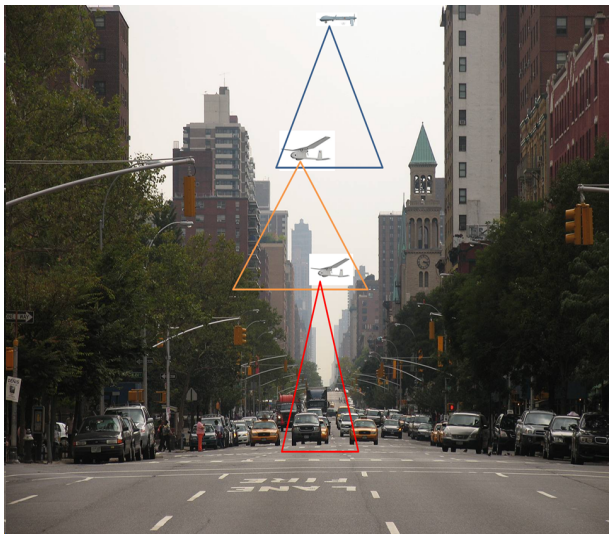


Figure 3.1: This figure shows the basic concept of bearing-only cooperative geo-localization. The objective is to geo-localize a ground moving target using a team of camera equipped UAVs by measuring bearing measures from each other and the the target. A larger air vehicle like predator, which flies at the higher altitude, measures it's position and heading using a GPS receiver. This larger vehicle shares its GPS and bearing measurement with other low flying smaller air vehicles to cooperatively geo-locate a ground target.

estimation problems, can be exploited to derive the optimal geometric configurations of a team of sensing platforms. However, maintenance of optimal configurations is not feasible given constraints on the kinematics of typical fixed wing aircraft. Frew [57] evaluated the sensitivity of target geo-localization to orbit coordination, which enables the design of cooperative line of sight controllers that are robust to variations in the sensor measurement uncertainty and the dynamics of the target tracked.

While the existing work on vision based geo-localization successfully demonstrates the target localization concept and provides several techniques to improve the accuracy of geo-localization, the limitations associated with geo-localizing a target in urban environments are not addressed. All of the existing methods require the vehicles's position and attitude to geo-localize a target. The standard method is to fuse measurements from a global positioning system (GPS) receiver and an inertial measurement unit (IMU) to estimate the position and attitude of a aircraft. However, in many environments of interest, the GPS signals are unavailable or unreliable, e.g., indoors, underwater, on other planets, in the urban canyon. Additionally, the accuracy of pose estimates based on GPS may be insufficient for accurate target localization. Therefore, it is important to develop

methods for localization in the absence of or in addition to GPS. Furthermore, a camera is a line of sight (LOS) sensor and there may exist many occlusions, e.g., buildings, trees, etc., in the urban environments which can lead to unreliable tracking of the target.

In this chapter, we address the aforementioned limitations for an urban canyon environment by using cooperative localization to jointly localize a team of air vehicles and geo-localize a ground target. In cooperative localization a group of vehicles exchange relative position measurements, e.g., range and bearing, from their exteroceptive sensors, e.g., camera, laser, etc., and their motion information, velocity and angular rate, from interoceptive sensors, e.g., IMU, encoders, etc., to collectively estimate their states. For ground robots, cooperative localization has been an active area of research (e.g., [15, 29–35]) because it provides several potential advantages, including increased localization accuracy, sensor coverage, robustness, efficiency, and flexibility. In our previous work [66], for bearing-only localization of robots, we used a graph based approach to show that if each robot has a path to two different known landmarks then the system is completely observable and the position and heading of all of the robots can be estimated with bounded uncertainty. This result can be extended and it can be shown that if the graph is connected and at least one of the robots measures its position and heading from a GPS receiver, than the system is completely observable.

In this work, we develop a bearing-only cooperative localization technique for UAVs to geo-localize a ground target. In bearing-only localization the vehicles measure bearing from each other and the targets that are in their sensor range. Each air vehicle flies at a constant altitude, as shown Figure. 3.1, such that the target is in the field-of-view of one of the agents and at least one vehicle, the larger vehicle which is flying at a higher altitude, receives signals from a sufficient number of GPS satellites to localize its position. By performing graph based nonlinear observability analysis, we demonstrate that this system is observable, meaning that the target can be geo-localized with only one vehicle receiving sufficient GPS satellite signals. This overcomes the limitation that requires a low-flying smaller UAVs to maintain line-of-sight while flying high enough to maintain GPS lock. Also, we design a distributed path planning algorithm using receding horizon control that improves the localization accuracy of the target and of all of the agents while satisfying the observability conditions.

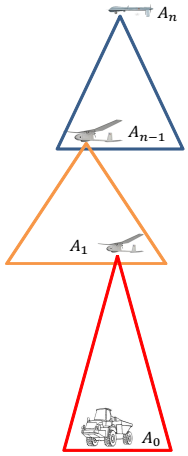


Figure 3.2: Bearing-only cooperative geo-localization. A_0 is the target that is to be cooperatively geo-localized by a team of n aerial vehicles $A_1, \cdots, A_n$ that are equipped with downward facing cameras. Each vehicle flies at a constant altitude in descending order and measure bearing from each other and the target. A_n flies at the highest altitude, and is assumed to have access to GPS while the smaller vehicles flying at lower altitude do not have GPS

The chapter is organized as follows. In the next section, we formulate the problem and discuss the bearing-only cooperative localization for unmanned aerial vehicles. In Section 3.3, we perform the graph based nonlinear observability analysis, demonstrating the conditions under which the system is observable. In Section 3.4, we develop a distributed path planner to improve the accuracy of geo-localization. In Section 3.5, we present simulation results. In Section 3.6, we give our conclusions.

3.2 Problem formulation

Consider a mobile ground target A_0 moving in an urban terrain as shown in Figure 3.2. Next, consider a team of n aerial vehicles $(A_1, \cdots, A_n)$ flying at different constant altitudes in descending order. A_n flies at the highest altitude where a sufficient number of GPS satellites are available for its localization and A_1 flies at the lowest altitude to keep the target in its field-of-view as shown in Figure 3.2. Each UAV is equipped with an inertial measurement unit (IMU) and a downward facing gimballed camera. The IMU measures angular rates and accelerations in the body frame and the camera of the i^{th} agent measure the bearing from the $(i-1)^{th}$ agent. The objective is to cooperatively geo-localize, i.e., find the global position and heading of the ground

target. For geo-localization, the vehicles must have accurate estimates of their position and attitude and the target must be in the sensor range of at least one of the vehicles. Since only A_n has the ability to sense its global position and heading through GPS, the global position and attitude of the other $n-1$ agents must be cooperatively estimated. This is accomplished by the agents exchanging information about their exteroceptive bearing measurements and interoceptive IMU measurements with other vehicles. This overcomes the limitation of requiring a low-flying smaller air vehicle to maintain line of sight while simultaneously flying high enough to maintain GPS lock.

There are two problems that need to be solved for cooperative target geo-localization. First, fuse the bearing-only measurements and IMU measurements from all of the agents and the GPS measurement from A_n to cooperative estimate position and heading of all of the air vehicles and the target. The second problem is to design a distributed path planner that improves the accuracy of localization and keeps the target in the field-of-view if at least one vehicle. In the next subsection, we discuss bearing-only cooperative localization for unmanned aerial vehicles.

3.2.1 Bearing-only cooperative localization

In this work, we use the following simplified guidance model for the i^{th} UAV which is given by

$$\begin{pmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{z}_i \\ \dot{\psi}_i \end{pmatrix} = f_i(X_i, u_i) = \begin{pmatrix} V_i \cos\theta_i \cos\psi_i \\ V_i \cos\theta_i \sin\psi_i \\ -V_i \sin\theta_i \\ \omega_i \end{pmatrix}, \tag{3.1}$$

where V_i is the airspeed, $[x_i, y_i, z_i]^\top$ is the 3-D position, $\omega_i = \frac{g}{V_i}\tan\phi_i$ is the turn rate, $u_i = [V_i, \omega_i]^\top$ is the control input, g is the gravitational acceleration constant, and $[\phi_i, \theta_i, \psi_i]^\top$ are roll, pitch, and yaw angles.

In this work we assume that there is no wind and the airspeed V_i is available, which can be estimated from the differential pressure sensor (see Chapter 8 in [79]). Also, we assume that roll ϕ_i and θ_i are available. For small UAVs the roll and pitch angles can be estimated using gyro and accelerometer measurements (see Chapter 8 in [79]).

The camera measures bearing from other UAVs, and landmarks that are in its sensor range. The bearing in azimuth and elevation measured from the i^{th} vehicle to the j^{th} vehicle are written as

$$\eta_{ij}^{a} = \tan^{-1}\left(\frac{y_j - y_i}{x_j - x_i}\right) - \psi_i + \upsilon^{a}, \tag{3.2}$$

$$\eta_{ij}^{e} = \tan^{-1}\left(\frac{z_j - z_i}{R_{ij}}\right) - \theta_i + \upsilon^{e}, \tag{3.3}$$

where $R_{ij} = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2}$ and $\upsilon^{a} \sim N(0, \sigma_\eta^2)$ and $\upsilon^{a} \sim N(0, \sigma_\eta^2)$ processes with zero mean and variance σ_η^2. In this work, we assume that each camera can be gimballed to keep the other UAVs, or the target, in it's field-of-view.

The GPS measurement for A_n is given as

$$z_{gps} = \begin{pmatrix} x_n \\ y_n \\ z_n \\ \psi_n \end{pmatrix} + \begin{pmatrix} \upsilon_x \\ \upsilon_y \\ \upsilon_z \\ \upsilon_\psi \end{pmatrix}, \tag{3.4}$$

where $\upsilon_x \sim N(0, \sigma_x^2)$, $\upsilon_y \sim N(0, \sigma_y^2)$, $\upsilon_z \sim N(0, \sigma_z^2)$, $\upsilon_\psi \sim N(0, \sigma_\psi^2)$ are zero mean Gaussian processes that model the measurement error in position and heading.

For cooperative localization, each vehicle exchanges its local sensor measurements, which include velocity, angular rates, and bearing measurements, with their neighbors. Let N_i^M be the set of neighbors for which i^{th} UAV can obtain bearing measurements, and let N_i^C be the set of neighbors with which i^{th} UAV can communicate. In this chapter, we assume that $N_i^M = N_i^C$ and we will therefore denote the set of neighbors as N_i. To represent the connection topology of the UAVs we use a relative position measurement graph (RPMG) [18] which is defined as follows.

Definition 3.2.1 *An RPMG for n nodes performing cooperative localization is a directed graph* $G_n \triangleq \{\mathscr{V}_n, \mathscr{E}_n\}$*, where* $\mathscr{V}_n = \{1, \cdots, n\}$ *is the node set consisting of n vehicle nodes and* $\mathscr{E}_n(t) \subset \{\mathscr{V}_n \times \mathscr{V}_n\} = \{\eta_{ij}^{a}, \eta_{ij}^{e}\}$*,* $i, j \in \mathscr{V}_n$ *is the edge set of m bearing measurements. Index* $p \in \mathscr{E}_n$ *represents the* p^{th} *measurement. An example RPMG* G_5 *is shown in Fig. 3.3.*

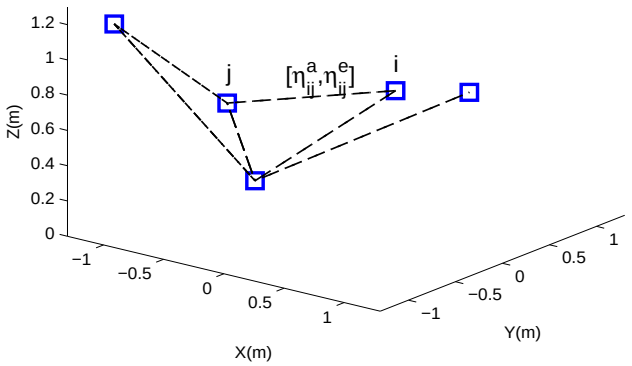


Figure 3.3: Relative position measurement graph (RPMG). The nodes of an RPMG (blue squares are UAVs) represent vehicle states and the edges represent bearing measurements between nodes.

The objective of cooperative localization is to estimate the combined state of all of the UAVs and the target. We use an extended information filter(EIF) [80] to implement bearing-only cooperative localization. The EIF is dual of the EKF and the EKF is a quasi-local asymptotic observer for nonlinear systems where a necessary condition for its convergence and boundedness are that the system is fully observable [51]. Therefore, a first step is to find the conditions under which the system is completely observable.

3.3 Graph-based observability analysis

To determine the observability of the entire system represented by the RPMG we use the nonlinear observability rank criteria developed by Hermann and Krener [75] which is summarized in the next paragraph.

Consider a system model with the following form

$$\Sigma: \begin{array}{c} \dot{X} = f(X,u) = [f_1^\top(X_1,u_1),\cdots,f_n^\top(X_n,u_n)]^\top, \\ Y = h(X) = [h_1(X),\cdots,h_m(X)]^\top, \end{array} \tag{3.5}$$

where $X = [X_1^\top X_2^\top \cdots X_n^\top]^\top \in \mathbb{R}^{4n}$ is the state of the system, $Y \in \mathbb{R}^m$ is the system output, $h_p : \mathbb{R}^{4n} \mapsto \mathbb{R}$ is the measurement model of the p^{th} sensor measurement, $u \in \Lambda \subseteq \mathbb{R}^{2n}$ is the control input vector, and $f : \mathbb{R}^{4n} \times \Lambda \mapsto \mathbb{R}^{4n}$.

We can write the process model in (3.5) can be written as,

$$\dot{X} = f(X(t), u(t)) = \sum_{i=1}^{n} f_{v_i}(X(t))V_i + \sum_{i=1}^{n} f_{\omega_i}(X(t))\omega_i, \tag{3.6}$$

where

$$f_{v_i} = [0_{4(i-1)\times 1},\ \cos\theta_i \cos\psi_i,\ \cos\theta_i \sin\psi_i,\ -\sin\theta_i,\ 0,\ 0_{4(n-i)\times 1}]^\top, \tag{3.7}$$

$$f_{\omega_i} = [0_{3(i-1)\times 1},\ 0,\ 0, 0, 0,\ -1,\ 0_{4(n-i)\times 1}]^\top, \tag{3.8}$$

are vector fields on $\mathbb{R}^{4n}$. Two features of this system can be observed. First, the system is time invariant, in that there is no explicit dependence on time. Second, the system above is linear in the control. In this dissertation, we use Lie derivatives, which comes under the field of differential geometry, to answer an important behavior of the system: Is it observable, in the sense that, for each state there exists at least one corresponding input which permits us to discriminate between this state all nearby states. We use Lie derivatives to obtain the observability matrix. A Lie derivative can be interpreted as the derivative of a scalar along integral curves of the vector field. Some basics of differential geometric methods and Lie derivatives are provided in Appendix A.

The zeroth-order Lie derivative of the p^{th} bearing measurement h_p is the function itself, i.e., $L^0 h_p = h_p$. Using the definition of Lie derivative in (A.4), the first-order Lie derivative of function h_p along with vector field f_{v_i} is given by

$$L^1_{f_{v_i}} h_p = \nabla L^0 h_p \cdot f_{v_i}, \tag{3.9}$$

where ∇ represents the gradient operator, and $\cdot$ denotes the vector inner product. Considering that $L^1_{f_{v_i}} h_p$ is a scalar function itself, the second-order Lie derivative of $L^1_{f_{v_i}} h_p$ along with vector field f_{v_i} is given by

$$L^2_{f_{v_i} f_{v_i}} h_p = \nabla L^1_{f_{v_i}} h_p \cdot f_{v_i}. \tag{3.10}$$

Higher order Lie derivatives are computed similarly. Additionally, it is possible to define mixed Lie derivatives, i.e., with respect to different functions of the process model. For example, the second-order Lie derivative of $L^1_{f_{v_i}} h_p$ along with vector field f_{v_j} is given by

$$L^2_{f_{v_i} f_{v_j}} h_p = \nabla L^1_{f_{v_i}} h_p \cdot f_{v_j}. \tag{3.11}$$

Based on the preceding expressions for the Lie derivatives the observability matrix is defined as the matrix with rows

$$O^p \triangleq \left\{ \nabla L^d_{f_{v_i}, \cdots, f_{v_j}, f_{\omega_i}, \cdots, f_{\omega_j}} h_p \right\}, \tag{3.12}$$

where $i, j \in \mathscr{V}_n$, $p \in \mathscr{E}_n$, and $d \in \mathbb{N}$. The important role of this matrix in the observability analysis of a nonlinear system is demonstrated by next theorem, which is proved in [76].

Theorem 3.3.1 ([76]) *A system is locally weakly observable if its observability matrix whose rows are given in* (3.12) *has full rank, e.g., in our case rank*$(O) = 4n$.

3.3.1 Rows in the observability matrix due to an edge

In this subsection, we derive the linearly independent rows in the observability sub-matrix of an edge between two nodes. The edge consists of two bearing measurements $h^a(X) = \eta^a_{ij}$, and $h^e(X) = \eta^e_{ij}$. The linearly independent rows of the observability sub-matrix of an edge serve as building block for the observability conditions for the graph G_n.

We rearrange the nonlinear kinematic equations in the following convenient form for computing Lie derivatives:

$$\dot{X} = \begin{bmatrix} \dot{X}_i \\ \dot{X}_j \end{bmatrix} = f_{v_i} V_i + f_{\omega_i} \omega_i + f_{v_j} V_j + f_{\omega_j} \omega_j, \tag{3.13}$$

where $f_{v_i} = [c_{\theta_i} c_{\psi_i} \; c_{\theta_i} s_{\psi_i} \; -s_{\theta_i} \; 0 \; 0 \; 0 \; 0 \; 0]^\top$, $f_{\omega_i} = [0 \; 0 \; 0 \; 1 \; 0 \; 0 \; 0 \; 0]^\top$, $f_{v_j} = [0, \; 0, \; 0, \; 0, \; c_{\theta_j} c_{\psi_j} \; c_{\theta_j} s_{\psi_j} \; -s_{\theta_j} \; 0]^\top$, $f_{\omega_j} = [0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 0 \; 1]^\top$, $c_{\psi_i} \triangleq \cos \psi_i$, $s_{\psi_i} \triangleq \sin \psi_i$, and $c_{\theta_i} \triangleq \cos \theta_i$.

Our objective is to find the number of linearly independent rows in a matrix. To do that, we transform a matrix into reduced row echelon form (RREF). RREF is the simplest possible form

of a matrix, which directly provides the number of linearly independent rows in the matrix. Since RREF is the backbone of the analysis presented in this chapter we state the next lemma, which explains the properties of a RREF matrix.

Lemma 3.3.1 ([77]) *A matrix $A \in \mathbb{R}^{m\times n}$, by means of a finite sequence of elementary row operations, can be transformed to a row reduced echelon form $U \in \mathbb{R}^{m\times n}$ such that*

$$EA = U, \tag{3.14}$$

where $E \in \mathbb{R}^{m\times m}$ is the elementary operation matrix. If the rank of A is r then

1. $$U = \begin{bmatrix} \mathbf{I}_r & B \\ \mathbf{0}_{(m-r)\times r} & \mathbf{0}_{(m-r)\times(n-r)} \end{bmatrix}, \tag{3.15}$$

   where $\mathbf{I}_r$ is the Identity matrix of size r and $B \in \mathbb{R}^{r\times(n-r)}$,

2. *the first r rows of matrix U are linearly independent,*

3. *the non zero rows of the matrix U spans the same row space spanned by A,*

4. *if A is an invertible matrix ($r = m = n$) then U is the Identity matrix.*

Lemma 3.3.2 *Let O_{ij} be the observability matrix associated with the bearing measurement η^a_{ij} and η^e_{ij} collected by vehicle i while observing vehicle j. The number of linearly independent rows of O_{ij} is four if*

1. $V_i > 0$,
2. $V_j > 0$,
3. $\theta_i \neq k\frac{\pi}{2}$, $\forall k = \pm1, \pm2, \cdots$,
4. $\theta_j \neq k\frac{\pi}{2}$, $\forall k = \pm1, \pm2, \cdots$,
5. *the i^{th} vehicle does not move directly along the projection of Euclidian line, connecting vehicle i and j, in the horizontal plane,*

6. *the j^{th} vehicle does not move directly perpendicular to the projection of Euclidian line, connecting vehicle i and j, in the horizontal plane.*

Proof: Let O_{ij} be the observability matrix associated with the bearing measurement η_{ij}^a and η_{ij}^e collected by vehicle i while observing vehicle j. If $V_i > 0$ and $V_j > 0$, then we can compute L^0h^a, L^0h^e, $L^1f_{v_i}h^a$, and $dL^1f_{v_j}h^a$. We can write the observability matrix O_{ij} as

$$O_{ij} = \begin{bmatrix} dL^0h^a \\ dL^0h^e \\ dL^1f_{v_i}h^a \\ dL^1f_{v_j}h^a \end{bmatrix}, \tag{3.16}$$

$$= \begin{bmatrix} -y_{ij} & x_{ij} & 0 & -R_{ij}^2 & y_{ij} & -x_{ij} & 0 & 0 \\ x_{ij}z_{ij} & y_{ij}z_{ij} & -R_{ij}^2 & 0 & x_{ij}z_{ij} & y_{ij}z_{ij} & -R_{ij}^2 & 0 \\ c_{\theta_i}s_{\psi_i} & -c_{\psi_i}c_{\theta_i} & 0 & c_{\theta_i}J_i^+ & -c_{\theta_i}s_{\psi_i} & c_{\psi_i}c_{\theta_i} & 0 & 0 \\ -c_{\theta_j}s_{\psi_j} & c_{\psi_j}c_{\theta_j} & 0 & 0 & c_{\theta_j}s_{\psi_j} & -c_{\psi_j}c_{\theta_j} & 0 & -c_{\theta_j}J_j^+ \end{bmatrix}, \tag{3.17}$$

where $y_{ij} \triangleq y_i - y_j$, $x_{ij} \triangleq x_i - x_j$, $z_{ij} \triangleq z_i - z_j$, $R_{ij}^2 = x_{ij}^2 + y_{ij}^2$, $c_\theta \triangleq \cos\theta$, $c_\psi \triangleq \cos\psi$, $s_\psi \triangleq \sin\psi$, $J_i^+ \triangleq x_{ij}c_{\psi_i} + y_{ij}s_{\psi_i}$, and $J_j^- \triangleq y_{ij}c_{\psi_j} - y_{ij}s_{\psi_j}$.

Also, we can write

$$J_i^- = v_1^\top v_i^{'} = y_{ij}\cos\psi_i - y_{ij}\sin\psi_i, \tag{3.18}$$

$$J_j^+ = v_1^\top v_j = x_{ij}\cos\psi_j + y_{ij}\sin\psi_j, \tag{3.19}$$

where $v_1 = [x_{ij}\ y_{ij}]^\top$ is a vector along the line between the two vehicles, $v_j = [\cos\psi_j\ \sin\psi_j]^\top$ is the heading vector of the j^{th} vehicle, and $v_i^{'} = [-\sin\psi_i\ \cos\psi_i]^\top$ is a vector perpendicular to the heading vector of the i^{th} vehicle.

From (3.18) and (3.19), we can verify that if the i^{th} vehicle, which is measuring the bearing, does not move directly along the projection of the Euclidian line connecting the two agents in the horizontal plane, then $J_i^- \neq 0$, and if the j^{th} vehicle does not move perpendicular to the projection of the Euclidian line connecting the two vehicles then $J_j^+ \neq 0$. Therefore, if $\theta_i \neq \pm\frac{\pi}{2}$ and $\theta_j \neq \pm\frac{\pi}{2}$

then the elementary operation matrix

$$E_{ij} = \begin{bmatrix} -\frac{c_{\psi_j} J_i^+}{J_i^- J_j^+} & 0 & -\frac{c_{\psi_j} R_{ij}^2}{c_{\theta_i} J_i^- J_j^+} & -\frac{y_{ij}}{c_{\theta_j} J_j^+} \\ -\frac{s_{\psi_j} J_i^+}{J_i^- J_j^+} & 0 & -\frac{s_{\psi_j} R_{ij}^2}{c_{\theta_i} J_i^- J_j^+} & \frac{x_{ij}}{c_{\theta_j} J_j^+} \\ \frac{z_{ij} J_i^+}{J_i^- R_{ij}^2} & -\frac{1}{R_{ij}^2} & -\frac{z_{ij}}{c_{\theta_i} J_i^-} & 0 \\ \frac{\sin(\psi_i - \psi_j)}{J_i^- J_j^+} & 0 & \frac{J_j^-}{c_{\theta_i} J_i^- J_j^+} & \frac{1}{c_{\theta_j} J_j^+} \end{bmatrix},$$

which transforms O_{ij} as

$$E_{ij} O_{ij} = U_{ij} = \begin{bmatrix} \mathbf{I}_4 & \bar{O}_{ij} \end{bmatrix}, \tag{3.20}$$

where

$$\bar{O}^{ij} = \begin{bmatrix} -1 & 0 & 0 & y_{ij} \\ 0 & -1 & 0 & -x_{ij} \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}. \tag{3.21}$$

From Lemma 3.3.1, since the row reduced matrix U_{ij} has four linearly independent rows and these rows span the same observability space spanned by rows of O_{ij}, the conditions of Lemma 3.3.2 are sufficient for $rank(O_{ij}) = 4$. ■

Definition 3.3.1 *An RPMG G_n (Definition 3.2.1) is called a proper RPMG if all of the edges between two nodes satisfy the conditions of Lemma 3.3.2.*

In a proper RPMG each edge contributes four linearly independent rows to the observability of the overall system. Using four linearly independent rows of U_{ij} in (3.20), we can write the observability matrix of a proper RPMG G_n as

$$O^p \triangleq \left\{ \; O_{ij}^p \; \right\}, \; i, j \in \mathcal{V}_n, \tag{3.22}$$

where $O_{ij}^p = \begin{bmatrix} 0_{4\times 4(i-1)} & \mathbf{I}_4 & 0_{4\times(4(j-1)-4i)} & \bar{O}_{ij} & 0_{4\times 4(n-j)} \end{bmatrix}$.

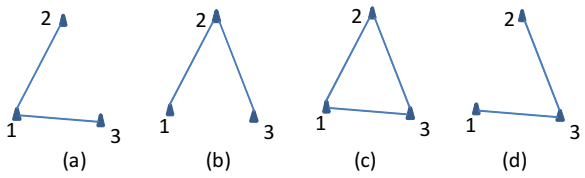


Figure 3.4: The observability conditions between these four possible configurations of a connected, 3-node RPMG are identical.

Remark 3.3.1 *The observability matrix O^p in* (3.22) *is not the original observability matrix of the graph G_n. Since the rows of* (3.22) *consist of the linearly independent rows after elementary row operations, from Lemma 3.3.1 we know that the rows of the observability matrix in* (3.22) *span the same observable space spanned by the original observability matrix.*

3.3.2 Rows in the observability matrix of a 3-node RPMG

Next, we discuss the observability properties for a 3-node graph G_3.

Lemma 3.3.3 *If a three node proper RPMG G_3 is connected, then the rank of the observability matrix is six.*

Proof: There are four possible configurations of a connected graph G_3, shown as sub-figures (a) through (d) in Fig. 3.4. We can write the transformed observability matrix for these configurations using (3.22) as

$$O_a^p = \begin{bmatrix} \mathbf{I}_4 & \bar{O}_{12} & \mathbf{0} \\ \mathbf{I}_4 & \mathbf{0} & \bar{O}_{13} \end{bmatrix}, \quad O_b^p = \begin{bmatrix} \mathbf{I}_4 & \bar{O}_{12} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_4 & \bar{O}_{23} \end{bmatrix},$$
$$O_c^p = \begin{bmatrix} \mathbf{I}_4 & \bar{O}_{12} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_4 & \bar{O}_{23} \\ \mathbf{I}_4 & \mathbf{0} & \bar{O}_{13} \end{bmatrix}, \quad O_d^p = \begin{bmatrix} \mathbf{I}_4 & \mathbf{0} & \bar{O}_{13} \\ \mathbf{0} & \mathbf{I}_4 & \bar{O}_{23} \end{bmatrix}.$$

We perform elementary operation on O_a^p, O_b^p, O_c^p, and O_d^p by multiplying them by elementary operation matrices $E_a = \begin{bmatrix} \mathbf{0}_4 & \mathbf{I}_4 \\ \bar{O}_{12} & -\bar{O}_{12} \end{bmatrix}$, $E_b = \begin{bmatrix} \mathbf{I}_4 & -\bar{O}_{12} \\ \mathbf{0}_4 & \mathbf{I}_4 \end{bmatrix}$, $E_c = \begin{bmatrix} \mathbf{I}_4 & -\bar{O}_{12} & \mathbf{0}_4 \\ \mathbf{0}_4 & \mathbf{I}_4 & \mathbf{0}_4 \\ -\mathbf{I}_4 & \bar{O}_{12} & \mathbf{I}_4 \end{bmatrix}$, and

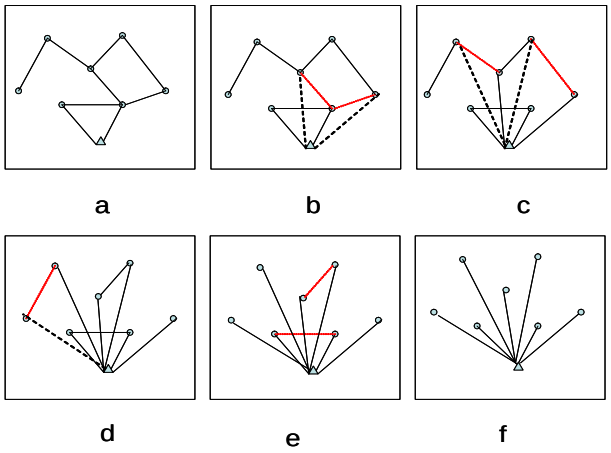


Figure 3.5: An example of converting an arbitrary connected RPMG to a 2-level tree.

$E_d = \mathbf{I}_8$ respectively to get

$$E_a O_a^p = E_b O_b^p = E_d O_d^p = \begin{bmatrix} \mathbf{I}_4 & \mathbf{0} & \bar{O}_{13} \\ \mathbf{0} & \mathbf{I}_4 & \bar{O}_{23} \end{bmatrix},$$

$$E_c O_c = \begin{bmatrix} \mathbf{I}_4 & \mathbf{0} & \bar{O}_{13} \\ \mathbf{0} & \mathbf{I}_4 & \bar{O}_{23} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

Therefore, Lemma 3.3.1 implies that the observability sub-matrix of all the four configurations have eight linearly independent rows and that these rows span the same observable space. ■

Remark 3.3.2 *The elementary operation matrix E_d for observability matrix O_d^p in Lemma 3.3.3 is Identity because O_d^p is already in a reduced row echelon form.*

From Lemma 3.3.3 we can say that the rows of two edges for a proper RPMG with a common node are independent. The following lemma is an extension of this idea.

Theorem 3.3.2 *If the graph G_n is a proper connected RPMG then the rank of the associated observability matrix is* $4(n-1)$.

Proof: Using Lemma 3.3.3, any connected 3-node subgraph in the larger graph can be replaced with any other connected 3-node subgraph, without affecting the rank of the system observability matrix because their associated observability sub-matrices span the same observable sub-space.

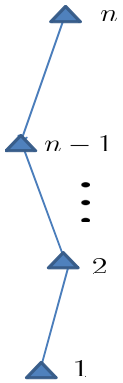


Figure 3.6: The cooperative geo-localization problem can be represented by a chain RPMG (G_n).

Therefore a connected graph G_n can be transformed to a 2-level tree (see Fig. 3.5 and the proof of Theorem 2.3.1 in Chapter 2).Without loss of generality, we assume that the root node of the 2-level tree is labeled as n. The system observability matrix can be transformed into the form

$$O^p_{2-level} = \begin{bmatrix} \mathbf{I}_4 & \mathbf{0} & \mathbf{0} & \cdots & \bar{O}_{1n} \\ \mathbf{0} & \mathbf{I}_4 & \mathbf{0} & \cdots & \bar{O}_{2n} \\ \vdots & & \ddots & & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{I}_4 & \bar{O}_{n-1,n} \end{bmatrix}. \tag{3.23}$$

Clearly, the rank of $O^p_{2-level}$ $4(n-1)$. ■

For example, the observability matrix for a chain RPMG G_n considered in this chapter, as shown in Fig.3.6, can be written using (3.22) as

$$O^p_{chain} = \begin{bmatrix} \mathbf{I}_4 & \bar{O}_{12} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_4 & \bar{O}_{23} & \cdots & \mathbf{0} \\ \vdots & & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{I}_4 & \bar{O}_{n-1,n} \end{bmatrix}. \tag{3.24}$$

We can find an elementary operation matrix

$$E_{chain} = \begin{bmatrix} \mathbf{I}_4 & -\bar{O}_{12} & -\bar{O}_{23} & \cdots & -\bar{O}_{1,n-1} \\ \mathbf{0} & \mathbf{I}_4 & -\bar{O}_{23} & \cdots & -\bar{O}_{2,n-1} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I}_4 & -\bar{O}_{n,n-1} \end{bmatrix}, \tag{3.25}$$

such that

$$E_{chain} O^p_{chain} = O^p_{2-level}, \tag{3.26}$$

which implies that $rank(O^p_{chain}) = 4(n-1)$.

Theorem 3.3.3 *Given an RPMG G_n, if it is proper, connected, and one of the vehicle has its position and heading measurement from GPS then the system is completely observable, i.e., the rank of the observability matrix is* $4n$.

Proof: If RPMG G_n is proper and connected then from Theorem 3.3.2 its observability matrix spans the same space as the observability matrix of 2-level tree ($O^p_{2-level}$). Let the n^{th} vehicle measure its position and heading directly from a GPS receiver and this results in the addition of four linearly independent rows to the observability matrix as

$$O^p_{gps} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I}_4 \end{bmatrix}. \tag{3.27}$$

The combined observability matrix of RPMG G_n with GPS measurement by the n^{th} vehicle can be written as

$$O^p = \begin{bmatrix} O^p_{2-level} \\ O_{gps} \end{bmatrix}, \tag{3.28}$$

$$= \begin{bmatrix} \mathbf{I}_4 & \mathbf{0} & \mathbf{0} & \cdots & \bar{O}_{1n} \\ \mathbf{0} & \mathbf{I}_4 & \mathbf{0} & \cdots & \bar{O}_{2n} \\ \vdots & & \ddots & & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{I}_4 & \bar{O}_{n-1,n} \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{I}_4 \end{bmatrix}. \tag{3.29}$$

Clearly, the rank of O^p is $4n$ and the system is completely observable. ■

The conditions of Theorem 3.3.3 explain the fact that cooperative localization creates a larger and flexible joint field-of-view and only one vehicle requires global position and heading measurement from a GPS receiver for the complete observability of position and heading of all of the agents. This fact motivates us to use cooperative bearing-only localization for target geo-localization in GPS denied areas.

3.4 Controller for geo-localization

Since in bearing-only cooperative localization both the motion model and measurement model are nonlinear, the uncertainty in estimates is path dependant. In this section, we develop a a distributed path planning algorithm that minimizes the uncertainty (maximizes the information) in the localization of all of the agents and the target while satisfying the observability constraints. For the target geo-localization problem, we restrict the motion of all of the vehicles in a horizontal plane at a constant altitude. Also, we assume that the vehicles and the target are moving with constant velocity, and that these velocities are in increasing order with the altitude i.e., $V_0 < V_1 < \cdots < V_n$. For developing a controller we use the equation of motion of an i^{th} UAV flying at constant

altitude, $\theta = 0$, and performing a coordinated turns conditions:

$$\begin{pmatrix} \dot{x}_i \\ \dot{y}_i \\ \dot{z}_i \\ \dot{\psi}_i \end{pmatrix} = \begin{pmatrix} V_i \cos\psi \\ V_i \sin\psi \\ 0 \\ \frac{g}{V_i}\tan\phi_i \end{pmatrix}. \tag{3.30}$$

Since target moves on ground, its equation of motion is similar to an UAV. For this work, we assume that the target moves in a straight line, i.e., $\dot{\psi}_1 = 0$, however, a maneuvering target can also be geo-localized using cooperative localization.

The information matrix is the inverse of the sensor measurement uncertainty and it contains all the information about the accuracy of the sensor measurement. The information matrix for bearing measurements in the horizontal plane can be written as

$$I(t) = H^{\top}(t) R_{meas}^{-1} H(t), \tag{3.31}$$

where H is the measurement Jacobian and R_{meas} is the covariance matrix of measurement noise. Since the bearing measurement is a nonlinear function of the states, the information matrix depends on the states of the vehicles. We can consider any number of cost functions derived from the information matrix, but in this chapter we will focus on the trace of $I(t)$, which is a scalar function and it captures the quality of the estimates obtained from the set of bearing measurements.

While designing the optimization algorithm we should keep in mind the constraints imposed by the observability of the system that are given in Theorem 3.3.3. The first constraint that global position and heading is measured by GPS to at least one of the vehicles is easily satisfied by flying a A_n at the high enough altitude. The second constraint requires that the RPMG, including target, be connected. To satisfy the connectivity constraint, we define a fixed RPMG topology in which agent A_i should measures the bearing to vehicle A_{i-1} at the next lower altitude. In other words, each vehicle should always keep the vehicle that is at the next lower altitude in its field-of-view. For the RPMG to be connected, the following condition should be satisfied.

$$R_i < R_{sensor}, \ \forall \ i \ \in \mathscr{V}_n, \tag{3.32}$$

where $R_i \triangleq R_{i,i-i} = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}$ is the horizontal separation between two nodes and R_{sensor} is the horizontal sensor range of the camera. In order to improve the localization accuracy consider the following T step receding horizon control problem:

$$\max_{\phi_1(t:t+T),\cdots,\phi_n(t:t+T)} \sum_{\tau=0:T} trace(I(t+\tau)), \tag{3.33}$$

subject to

$$R_i < R_{sensor},\ \forall i \in \mathscr{V}_n, \tag{3.34}$$

$$|\phi_i(t)| < \phi_{max}, \forall i \in \mathscr{V}_n. \tag{3.35}$$

The condition $|\phi_i(t)| < \phi_{max}$ imposes the physical control constrains on the air vehicle. To solve the problem (3.33)− (3.35), we use the distributed receding horizon control approach developed by Dunbar and Murray [81]. For the target geo-localization controller we only consider azimuth bearing measurement between two nodes and the edge set of the chain RPMG, as shown in Fig. 3.6, is given by

$$\mathscr{E}_n^o = \{\eta_{10}^a, \eta_{21}^a, \cdots, \eta_{n,n-1}^a\}. \tag{3.36}$$

For simplicity, we assume that $R_{meas} = \mathbf{I}$, which results in a simple cost function given by

$$trace(I(t)) = trace\left(\sum_{p\in\mathscr{E}_n^o} H_p^\top H_p\right), \tag{3.37}$$

$$= \frac{1}{R_{10}^2}(t) + \frac{1}{R_{21}^2(t)} + \cdots + \frac{1}{R_{n,n-1}^2(t)}, \tag{3.38}$$

where H_p is measurement Jacobian of the p^{th} bearing measurement. We can rewrite the distributed controller for the i^{th} UAV as

$$\max_{\phi_i(t:t+T)} \sum_{\tau=0:T} \frac{1}{R_{i,i-1}^2(t+\tau)}, \tag{3.39}$$

subject to

$$R_i < R_{sensor}, \tag{3.40}$$

$$|\phi_i(t)| < \phi_{max}. \tag{3.41}$$

We solve the above optimization problem using single agent dynamic programming [68].

Remark 3.4.1 *The maximization of the cost function in* (3.39) *will result in air vehicle trajectories that fly over the vehicle at the next lower altitude periodically because the uncertainty in the bearing measurement decreases when the distance between two vehicles decreases. In this approach the localization information flows from A_n to A_0, and the control information flows from A_0 to A_n.*

3.5 Results

In this section, we present the simulation results for the bearing-only cooperative geo-localization controller for stationary and mobile ground target. We consider four unmanned air vehicles that are flying at different constant altitudes and a ground mobile vehicle that needs to be geo-localized. The 4^{th} vehicle is flying at the highest altitude and receives its position and heading measurements from GPS receiver. The other three agents do not have a GPS receiver. The forward velocity of the target is $V_0 = 2\ m/s$ and the velocities of the air vehicles are $V_1 = 5m/s,\ V_2 = 8m/s,\ V_3 = 10m/s,\ V_4 = 12m/s$ and their respective altitudes are $z_1 = 20m,\ z_1 = 30_m,\ z_1 = 40m,\ z_4 = 50m$. The standard deviation for bearing measurement noise is $\sigma_n = 0.1\ rad$. The sampling time is $T_s = 0.1s$ and we use a 3-step receding horizon for the optimization.

First, we simulate the cooperative target geo-localization problem for a stationary target. Figure 3.7 shows the trajectories of all of the four air vehicles that are geo-localizing a stationary ground target. It can be seen that the estimated trajectories of all of the vehicles are close to their actual trajectories, demonstrating the effectiveness of this approach. The top view of thee trajectories is shown in Figure 3.8. It can be seen that these trajectories converge on circular orbits around the target on a line joining all of the air vehicles and the target. The trajectory of A_4, which is at the highest altitude and moving at fastest speed, converges to the outer most orbit and the trajectory of A_1, which is at the lowest altitude and moving at the slowest speed, converges to the inner most orbit. For clarity, we zoom in on Figure 3.8 to show the position estimates of the target

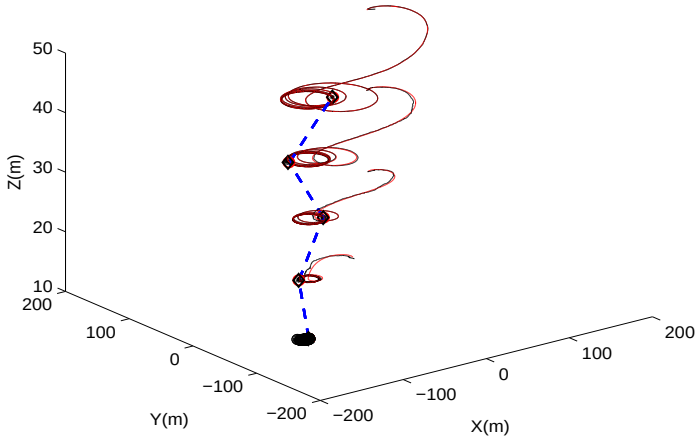


Figure 3.7: Bearing-only cooperative geo-localization for a stationary ground target. This figure shows the trajectories (solid red curve is actual, dashed black curve is estimated) and covariance (black ellipse) of all of the agents. The top most vehicle has its position and heading measurement from GPS. The blue dashed curve represents the bearing measurement between two nodes . The velocity and the turning radius of each vehicle increase with increase in the altitude.

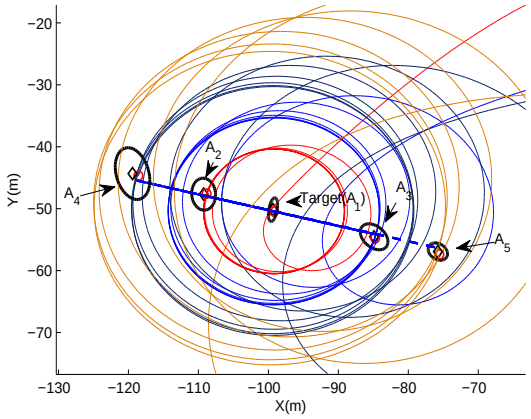


Figure 3.8: Bearing-only cooperative geo-localization for a stationary ground target. This figure shows the the top view of trajectories of all of the air vehicles. The black diamonds represents the estimated position and the red circles represents the true position. The black ellipses represents the 3σ uncertainties in the estimates. All of the vehicles settle on circular orbits around the target. Also, all of the air vehicles settle on a line joining them and the target.

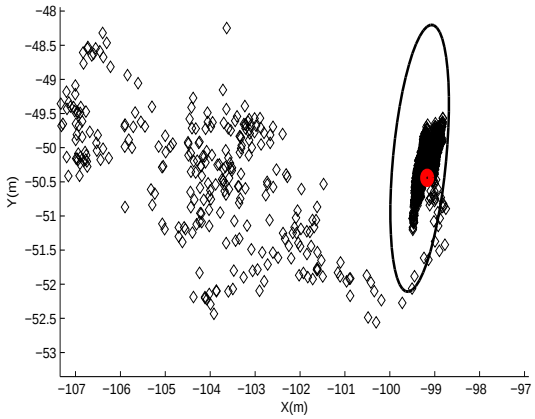


Figure 3.9: The zoomed view, from Figure 3.8, of the position estimates of the ground target. The position estimates of the stationary ground target is represented by black diamonds). The red circle represents the true position of the ground target, and the black ellipse represents the 3σ position uncertainty.

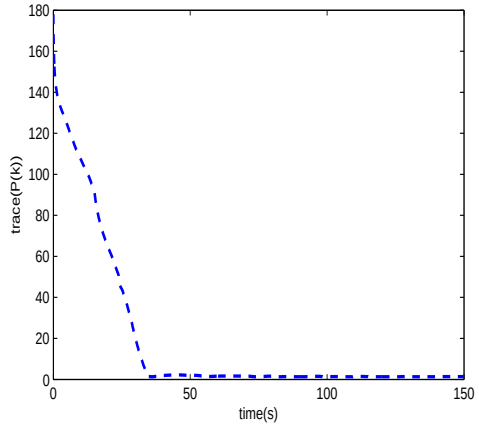
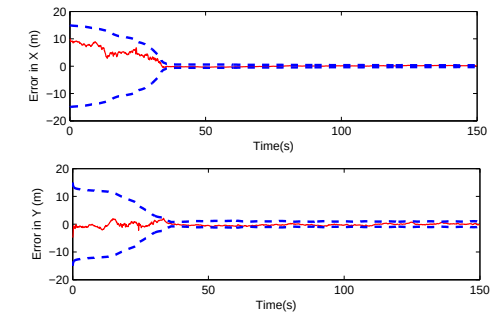
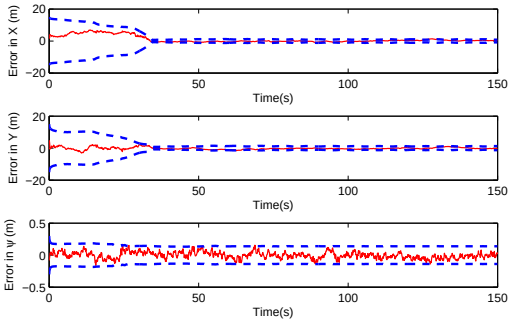


Figure 3.10: Bearing-only cooperative geo-localization of a stationary ground target. This figure shows the trace of the joint state error covariance matrix P. It can be seen that the algorithm minimizes the localization uncertainty in a distributed manner.

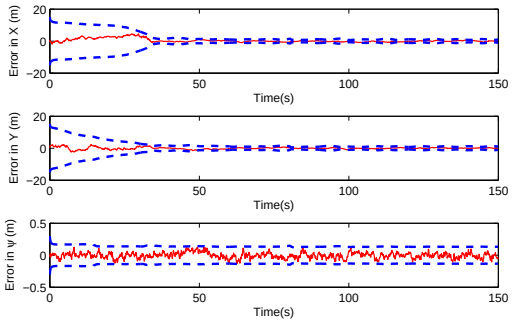
as shown in Figure 3.9. It can be seen that the position estimates converge to approximately the actual position and the uncertainty reduces to a small value. The trace of the covariance matrix of the joint states of all of the vehicles are shown in the Figure 3.10. It can be seen that the uncertainty (trace) is minimized using the distributed receding horizon controller. The Figure 3.11 shows the error plots of the position and heading of all of the vehicles that do not have GPS and the ground



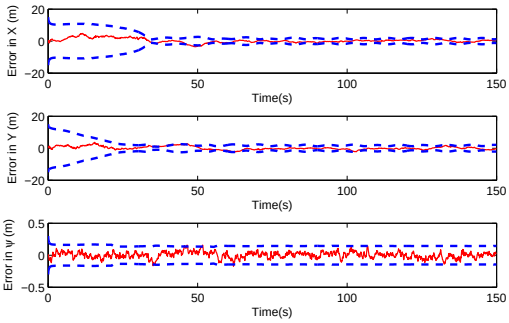
(a) Target



(b) UAV 2



(c) UAV 3



(d) UAV 4

Figure 3.11: Bearing-only cooperative geo-localization of a stationary ground target. This figure shows the error plots of position of the target and error plots of position and heading of the vehicles, which do not have GPS. The blue dashed curve is the 3σ error variance and red solid curve is the estimation error.

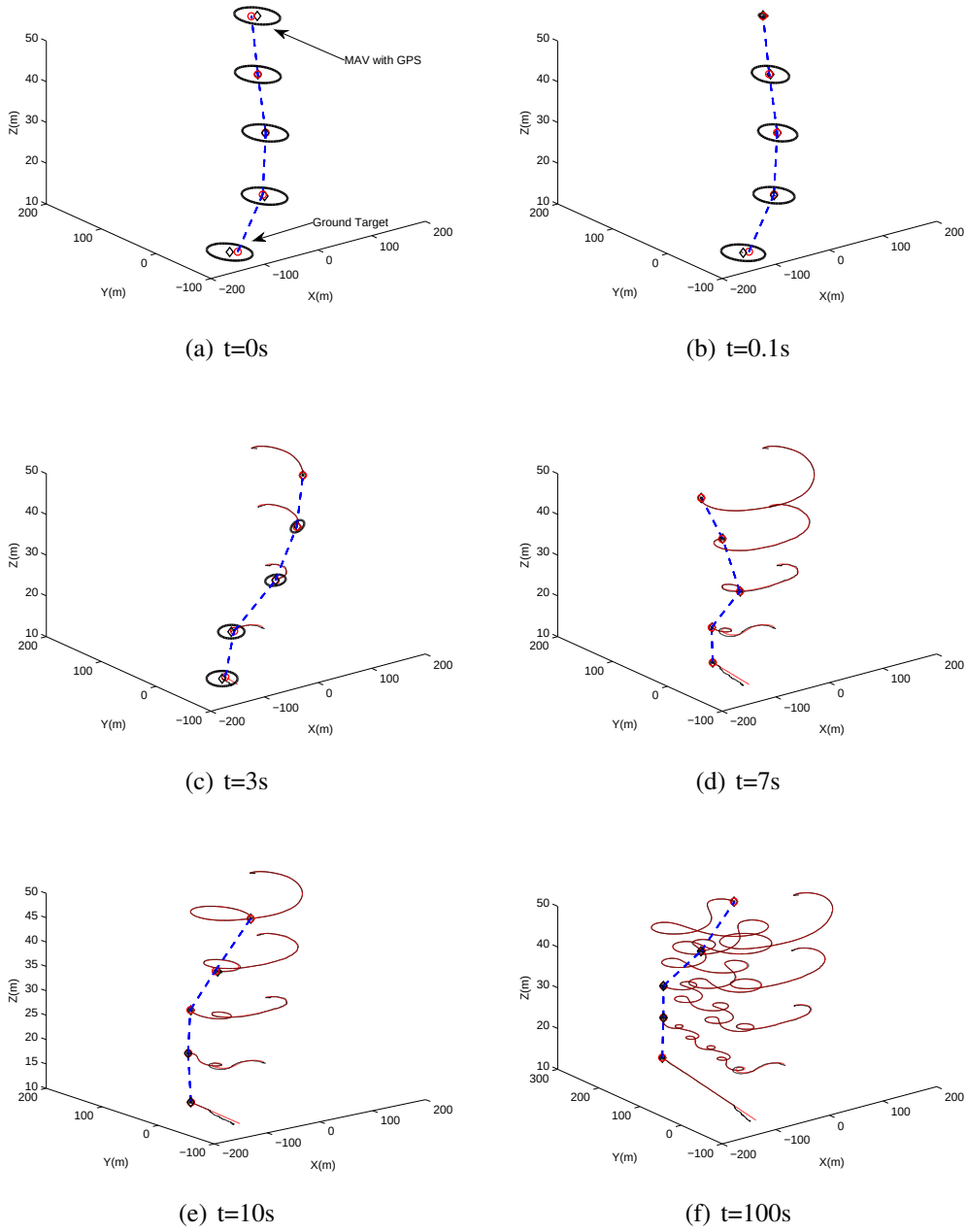
stationary target. It can be seen that uncertainties in the state estimation are minimized and they remain bounded.

Next, we simulate the bearing-only cooperative target geo-localization for a mobile ground target. Figure 3.12 shows the snapshots of the trajectories and uncertainty in 2-D position of all of the vehicles and the ground target taken at different time intervals. The initial RPMG is shown in Figure 3.12(a), and the Figure 3.12(b) shows the RPMG at $t = 0.1s$ when the 4^{th} UAV measures its position and heading for the first time from a GPS receiver. From Figure 3.12, we can say that the localization information flows from top to bottom and the control information flows from bottom to top. Also, the localization errors and the uncertainty in all of the air vehicles and the ground target decrease and remain bounded. The trace of the covariance matrix of the joint states of all of the vehicles and the mobile ground target is shown in the Figure 3.13. It can be seen that the

joint uncertainty is minimized using the distributed receding horizon controller. The Figure 3.14 shows the error plots of the position and heading of all of the vehicles that do not have GPS and the ground target. It can be seen that uncertainties in the state estimation are minimized and that remain bounded. The results presented in this chapter demonstrates that how a target can be geo-localized using multiple unmanned aerial vehicles in an urban environment where GPS signals are not directly available.

3.6 Conclusion

In this chapter, we develop a cooperative approach to geo-localize a ground moving target using bearing-only localization of UAVs. We perform a graph based nonlinear observability analysis of bearing-only localization to obtain the conditions for complete observability. We have shown that the system is completely observable if the graph is connected and at least one of the UAV measures its 3-D position and heading from a GPS receiver. We use this condition to develop a distributed path planning algorithm for UAVs to cooperative geo-localize the target using bearing-only measurements.



(a) t=0s (b) t=0.1s

(c) t=3s (d) t=7s

(e) t=10s (f) t=100s

Figure 3.12: Bearing-only cooperative geo-localization for mobile ground target. This figure shows the snap shots of the trajectories (solid red curve is actual, dashed black curve is estimated) and covariance (black ellipse) of all of the vehicles at different time intervals. The top most UAV has its position and heading measurement from the GPS. The blue dashed curve represent the bearing measurement between two nodes. The velocity of air vehicles decrease with the altitude. It can be seen that at the instant $t = 0.1s$ that the top most vehicle receives its global position from a GPS receiver and its uncertainty shrinks. Also, it can be seen that the localization information flows from top to bottom and that with time the uncertainty in the estimates is reduced

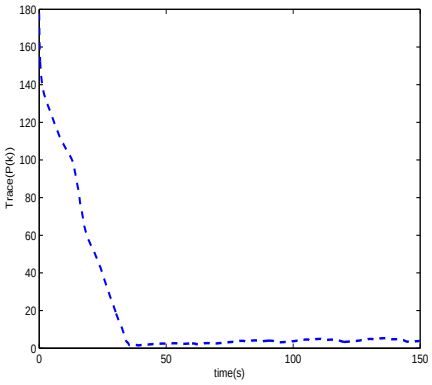
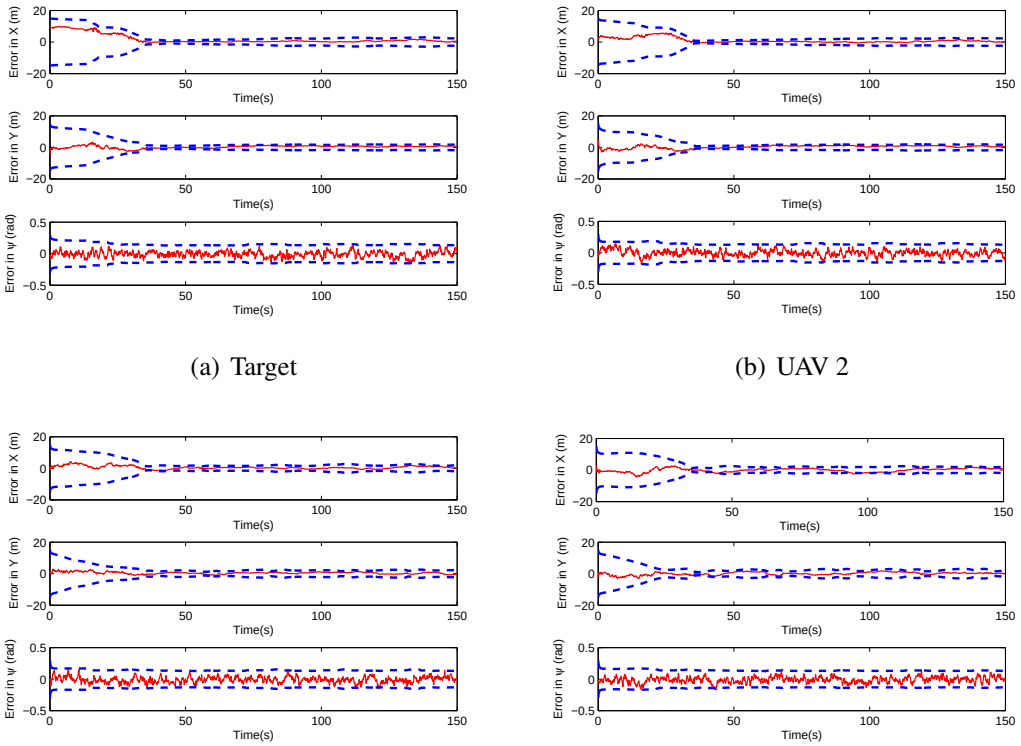


Figure 3.13: Bearing-only cooperative geo-localization of a mobile ground target. This figure shows the trace of the joint state error covariance matrix P. It can be seen that the algorithm minimizes the localization uncertainty in a distributed manner.



(a) Target

(b) UAV 2

(c) UAV 3

(d) UAV 4

Figure 3.14: Bearing-only cooperative geo-localization of a mobile ground target. This figure shows the estimation estimation error plots of the target states (position and heading) and error plots of position and heading of the vehicles, which do not have GPS. The blue dashed curve is the 3σ error variance and red solid curve is the estimation error.

CHAPTER 4. OBSERVABILITY BASED PATH PLANNING

In Chapter 2 and Chapter 3, we discussed how bearing measurement can be used for cooperative localization of ground robots and UAVs. In this and next chapter, we discuss how bearing measurement can be used for path planning of UAVs in urban terrain.

4.1 Introduction

Small and Unmanned Aerial Vehicles(UAVs) have the potential to perform tasks that are too difficult or dangerous for human pilots. For example, they can monitor critical infrastructure and real-time disasters, perform search and rescue, and measure weather in-storms [40]. For many of these applications, UAVs are required to navigate in urban or unknown terrain where obstacles of various types and sizes may hinder the success of the mission. UAVs must have the capability to autonomously plan paths that do not collide with buildings, trees or other obstacles. Therefore, the path planning and obstacle avoidance problems for UAVs have received significant attention [40–44].

The path planning problem can be grouped into global path planning and local path planning. Global path planning requires complete knowledge about the environment and a static terrain. In that setting a collision-free path from the start to the destination configuration is generated before the vehicle starts its motion [45]. The global path planning problem has been addressed by many researchers with common solutions being potential fields methods, roadmap methods and cell decomposition methods [82]. On the other hand, local path planning is executed in real-time during flight. The basic idea is to first sense the obstacles in the environment and then determine a collision-free path [40].

Local path planning algorithms require sensors to detect obstacles. Among the suite of possible sensors, a video camera is cheap and lightweight and fits the physical requirements for small UAVs [40]. However, because of projective geometry, a monocular camera really only measures

the bearing to the object. TTC can be estimated by considering the change in the size of the object in the image plane, but this estimate relies on accurately segmenting the image, which can be a noisy process. Therefore, it is a reasonable engineering choice to consider a monocular camera as a bearing-only measurement device and use the camera to estimate both TTC and bearing. We use an extended Kalman Filter (EKF) to extract TTC from bearing measurements.

The key idea presented in this chapter is to maneuver the UAV to minimize the state estimation uncertainty while simultaneously avoiding obstacles. We will show that these two tasks are complementary. We use the local mapping technique in our previous work [83–85], which builds a polar map in the local-level frame of the UAV using the camera measurements directly without transforming to the inertial frame. However, instead of using both TTC and bearing measurements as in [83–85], in this work we only use bearing measurements to estimate both the TTC and bearing to obstacles. For this purpose we will use the nonlinear observability theory developed by Hermann and Krener [75].

Observability is a measure of information available for state estimation. Song et al. [51] show that the EKF is a quasi-local asymptotic observer for discrete-time nonlinear systems, and that the convergence and boundedness of the filter are achieved when the system satisfies the nonlinear observability rank condition and when the states stay within a convex compact domain. Observability analysis has been studied extensively for the purpose of estimation [54, 86, 87]. While Bryson and Sukkarieh [88] perform the observability analysis of SLAM and develop an active control algorithm, the observability analysis is not used to develop active control. The contribution of this chapter is that we use the observability analysis to explicitly design the path planning algorithm. The main contributions of this chapter are as follows:

- We build polar maps using the TTC, which are independent of the ground or air speed of the UAV.
- We perform an observability analysis of the state estimation process from bearing-only measurements and find the necessary conditions for observability of the system.
- We design a path planning algorithm based only on the local map around the UAV in the local-level frame.

- The algorithm minimizes the uncertainties in the TTC and bearing estimates while simultaneously avoiding obstacles.
- We analyze the behavior of the path planning algorithm and determine the class of environments where the algorithm guarantees collision-free paths that maneuver the UAV to a goal configuration.

The chapter is organized as follows. Section 4.2 describes the model of the vehicle in the local-level frame and details a nonlinear observability analysis. In Section 4.3 we describe the observability-based path planning algorithm. Section 4.4 analyzes the behavior of the path planning algorithm. Numerical results are provided in Section 4.5, and our conclusions are in Section 4.6.

4.2 Observability analysis of state estimation

In this section we will build a local map using the TTC to obstacles in the local-level frame of the UAV. The map is constructed in polar coordinates by estimating the TTC and bearing to obstacles. We perform a nonlinear observability analysis of the state estimation problem using bearing-only measurements, find necessary conditions for complete observability of the system, and establish a link between estimation accuracy and collision avoidance.

We assume the UAV is flying at a constant height above ground level. Since the obstacle map is in the local-level frame of the UAV, the equations of motion of each obstacle relative to the UAV need to be derived. The origin of the local-level frame is the UAV's center of mass. The x-axis points out the nose of the airframe when the airframe is not pitching, the y-axis points out the right wing when the airframe is not rolling, and the z-axis points into the Earth. Let V represent the ground speed of the UAV and let ϕ and ψ represent the roll and heading angles, respectively. Figure 4.1 shows the motion of the i^{th} obstacle relative to the UAV in the local-level frame, where τ^i is the TTC, η^i is the bearing whose positive direction is defined as the right-handed rotation about the z-axis of the local-level frame, and O^i is the i^{th} obstacle. Based on Fig. 4.1, the equations

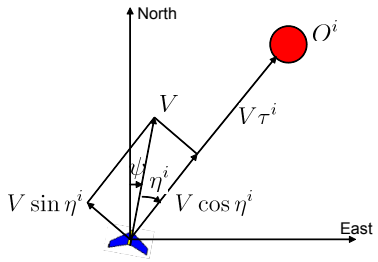


Figure 4.1: This figure shows the motion of the i^{th} obstacle relative to the UAV. The TTC and bearing to the obstacle are represented by τ^i and η^i. The ground speed is represented by V. The heading angle is represented by ψ. The i^{th} obstacle is represented by O^i.

of motion of the obstacle relative to the UAV in terms of TTC and bearing are given by

$$\dot{\tau}^i = -\cos\eta^i, \tag{4.1}$$

$$\dot{\eta}^i = \frac{\sin\eta^i}{\tau^i} - \dot{\psi}, \tag{4.2}$$

where, assuming coordinated turn conditions, $\dot{\psi} = \frac{g}{V}\tan\phi$ and where ϕ is the roll angle of the UAV, which we assume to be a control signal. Since we use the camera to measure the bearing only (which only requires data association), the measurement at time t is given by

$$z_t^i = \eta_t^i + v_t^i, \tag{4.3}$$

where v_t^i is the measurement noise that is assumed to be a zero-mean Gaussian random variable. Based on the state transition model expressed by Eqs. (4.1) and (4.2) and the observation model expressed by Eq. (4.3), we use the EKF to estimate the TTC and bearing and we build a TTC map in the local-level frame using polar coordinates, as shown in Fig. 4.2. The origin of the map is the current location of the UAV. The circles represent the obstacles and the ellipses around them represent the TTC and bearing uncertainties.

To decrease the uncertainties in the TTC and bearing estimates, we analyze the observability of the system given by Eqs. (4.1), (4.2), and (4.3). Let $\mathbf{x}^i = [\tau^i, \eta^i]^\top$ represent the state vector associated with the i^{th} obstacle and let $u = \phi$ represent the control input. Let $\dot{\mathbf{x}}^i = \mathbf{f}(\mathbf{x}^i, u)$ represent the state transition model given by Eqs. (4.1) and (4.2) and let $z_t^i = h(\mathbf{x}_t^i)$ represent the observation model given by Eq (4.3). The observability matrix is computed using Lie derivatives described by

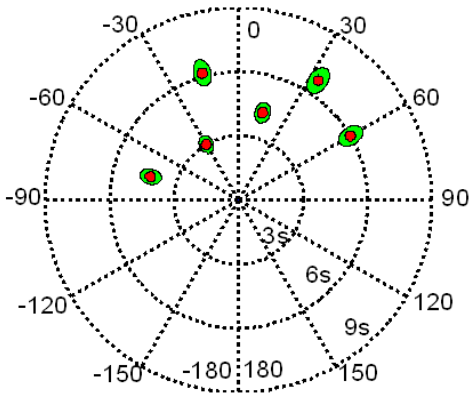


Figure 4.2: This figure shows the TTC map in the local-level frame of the UAV using polar coordinates. The origin of the map is the current location of the UAV. The circles represent the obstacles and the ellipses around them represent the TTC and bearing uncertainties. The radial direction is TTC and it is represented in seconds.

Hermann and Kerner [75]. The 0^{th} order Lie derivative is $L^0_{\mathbf{f}}(h) = \eta^i$ and the 1^{st} order Lie derivative is $L^1_{\mathbf{f}}(h) = \frac{\partial L^0_{\mathbf{f}}(h)}{\partial \mathbf{x}^i}\mathbf{f} = -\dot{\psi} + \frac{\sin\eta^i}{\tau^i}$. We define the vector of Lie derivatives $\Omega = [L^0_{\mathbf{f}}(h), L^1_{\mathbf{f}}(h)]^\top$. The observability matrix is computed as

$$\mathbf{O}^i = \frac{\partial \Omega}{\partial \mathbf{x}^i} = \begin{bmatrix} 0 & 1 \\ -\frac{\sin\eta^i}{(\tau^i)^2} & \frac{\cos\eta^i}{\tau^i} \end{bmatrix}. \tag{4.4}$$

The observability matrix has rank two if and only if $\tau^i \neq \infty$, $\eta^i \neq 2\pi p$ where $p \in \mathbb{Z}$. The EKF is a quasi-local asymptotic observer for nonlinear systems and its convergence and boundedness are achieved when the system is fully observable [51]. Bounds on the EKF error covariance $\mathbf{P}^i$ are related to the observability of the system given by Lemma 4.2.1 proved in [51].

Lemma 4.2.1 ([51]) *Suppose that there exist positive real scalars* $\alpha_1,\ \alpha_2,\ \beta_1,\ \beta_2$ *such that* $\beta_1\mathbf{I} \leq \mathbf{O}^{i\top}\mathbf{O}^i \leq \beta_2\mathbf{I}$ *and* $\alpha_1\mathbf{I} \geq \mathbf{C}^i\mathbf{C}^{i\top} \geq \alpha_2\mathbf{I}$ *then,*

$$\left(\frac{1}{\beta_2 + \frac{1}{\alpha_2}}\right)\mathbf{I} \leq \mathbf{P}^i \leq \left(\alpha_1 + \frac{1}{\beta_1}\right)\mathbf{I}, \tag{4.5}$$

where $\mathbf{I}$ *is the identity matrix and* $\mathbf{C}^i$ *is the controllability matrix.*

From Lemma 4.2.1, we can see that both the maximum and minimum singular values β_1 and β_2 of the observability matrix should be maximized in order to minimize both the upper and lower bounds of the error covariance matrix. For the problem in this chapter the order of the system is two, and therefore minimizing the inverse of the determinant of $\mathbf{O}^{i\top}\mathbf{O}^i$ will maximize the two eigenvalues of $\mathbf{O}^{i\top}\mathbf{O}^i$. The determinant of $\mathbf{O}^{i\top}\mathbf{O}^i$ related to the i^{th} obstacle is given by

$$\det(\mathbf{O}^{i\top}\mathbf{O}^i) = \frac{\sin^2\eta^i}{(\tau^i)^4}. \tag{4.6}$$

From Eq. (4.6), the inverse of determinant is given by $\frac{(\tau^i)^4}{\sin^2\eta^i}$. It can be seen that for large τ^i, the inverse is high, which means observability is less, because the change in the bearing measurement is very small with the large TTC (low parallax). It can also be seen that the inverse is minimum at $\eta^i = \pi/2$ and is maximum at $\eta^i = 0$, which means that the vehicle is moving directly towards the obstacle. Minimizing the inverse will ensure that $\eta^i \neq 2p\pi$ and will regulate $\eta^i \to \pi/2$. This implies that the minimization of the inverse of the determinant will minimize the lower and upper bounds of the error covariance matrix as well as steer the UAV away from the obstacle. Therefore the minimization of uncertainty and obstacle avoidance are complementary.

4.3 Observability-based path planning

Based on the observability analysis in the previous section, we design the observability-based path planning algorithm denoted by π^o such that (a) the uncertainties in the TTC and bearing estimates are minimized and (b) the UAV is maneuvered to the goal configuration. For the objective of goal reaching, the UAV requires knowledge of its own inertial position and the inertial position of the goal. Accordingly, the path planning algorithm π^o requires the use of GPS.

Let τ_t^g and η_t^g represent the TTC and bearing to the goal configuration at time t, and let $\mathbf{x}_t^g = [\tau_t^g, \eta_t^g]^\top$. Let τ_t^i and η_t^i represent the estimated TTC and bearing to the i^{th} obstacle and let $\mathbf{x}_t^i = [\tau_t^i, \eta_t^i]^\top$. Let $\mathbf{x}_t = [\mathbf{x}_t^{1\top}, \cdots, \mathbf{x}_t^{n\top}]^\top$. The determinant of the matrix $\mathbf{O}^{i\top}\mathbf{O}^i$ associated with the i^{th} obstacle is given by $\det(\mathbf{O}_t^{i\top}\mathbf{O}_t^i) = \frac{\sin^2\eta_t^i}{(\tau_t^i)^4}$. Let $\nu_t = [\mathbf{x}_t^{g\top}, \mathbf{x}_t{}^\top]^\top$. Let $\mathscr{I}_t$ represent the index set of all n obstacles and let τ^l represent the maximum TTC to obstacles that the planning algorithm π^o reacts to. Let $\mathscr{B}_t = \{i \in \mathscr{I}_t : \tau_t^i \leq \tau^l, \eta_t^i \leq \frac{\pi}{2}\}$ represent the index set of obstacles with the TTC no greater than τ^l and with the azimuth no greater than $\frac{\pi}{2}$. Define the utility function $S : \mathbb{R}^{2n+2} \to \mathbb{R}$

as

$$S(\nu_t) = a_1(\tau_t^g)^2 + a_2(\eta_t^g)^2 + \sum_{i=1}^{n} b_i I_{\mathscr{B}_t}(i) \frac{(\tau_t^i)^4}{\sin^2 \eta_t^i}, \tag{4.7}$$

where a_1, a_2, b_i, $i = 1, \cdots, n$ are non-negative weights, and $I_{\mathscr{B}_t}(i)$ is the indicator function of the index i, which zeros out the contribution of obstacles that are far away or that are passed by the UAV.

By minimizing the first two terms of Eq. (4.7), the algorithm drives the UAV towards the goal configuration. The third term penalizes the weighted sum of the inverse of the determinant of ${\mathbf{O}^i}^\top \mathbf{O}^i$ for all obstacles. By minimizing this term, the algorithm achieves two objectives simultaneously. First, it minimizes the uncertainties in the TTC and bearing estimates. Second, the UAV is steered around the obstacles. It is important to note that these two objectives are complementary to each other. We use a look-ahead policy over the horizon T to design the path planner π^o. The cost function to be minimized is given by

$$J = \int_t^{t+T} S(\nu_\rho) d\rho, \tag{4.8}$$

subject to the constraints

$$\begin{aligned} \dot{\mathbf{x}}_\rho^g &= \mathbf{f}(\mathbf{x}_\rho^g, u_\rho), \\ \dot{\mathbf{x}}_\rho^i &= \mathbf{f}(\mathbf{x}_\rho^i, u_\rho),\ i = 1, \cdots, n, \\ |u_\rho| &\leq \phi_{\max}. \end{aligned} \tag{4.9}$$

To solve the constrained optimization problem, we discretize the time horizon T as the m-step look-ahead horizon $\{t, t+\Delta t, \cdots, t+m\Delta t\}$, where $\Delta t = T/m$. Equation (4.8) then becomes

$$J = \sum_{j=1}^{m} S(\nu_{t+j\Delta t}). \tag{4.10}$$

The optimal path over the m-step look-ahead horizon is found using the nonlinear optimization function fmincon in MATLAB and is replanned once the UAV has followed the first portion of the m-step look-ahead path.

4.4 Analysis

The utility function given by Eq. (4.7) can be decomposed as the sum of

$$S_1(\mathbf{x}_t^g) = a_1(\tau_t^g)^2 + a_2(\eta_t^g)^2, \tag{4.11}$$

and

$$S_2(\mathbf{x}_t) = \sum_{i=1}^{n} b_i I_{\mathscr{B}_t}(i) \frac{(\tau_t^i)^4}{\sin^2 \eta_t^i}. \tag{4.12}$$

Accordingly, the observability-based path planner π^o that minimizes Eq. (4.8) can be decomposed into the goal reaching planner denoted by π^g that maneuvers the UAV to the goal by minimizing the cost function

$$J_1 = \int_t^{t+T} S_1(\mathbf{x}_\rho^g) d\rho, \tag{4.13}$$

and the collision avoidance planner denoted by π^c that maximizes the observability of the system by minimizing the cost function

$$J_2 = \int_t^{t+T} S_2(\mathbf{x}_\rho) d\rho. \tag{4.14}$$

Remark 4.4.1 *We decompose the observability-based path planner π^o into the collision avoidance planner π^c and the goal reaching planner π^g to simplify the analysis of collision avoidance and goal reaching behaviors.*

Accordingly, we analyze the obstacle avoidance behavior of the collision avoidance planner π^c that maximizes the observability of the system and describe under what environment π^c guarantees collision-free paths. We also describe under what environment the collision avoidance planner π^c is guaranteed to drive the UAV to the goal when it is combined with the goal reaching planner π^g.

4.4.1 Collision avoidance

We analyze the behavior of the collision avoidance planner π^c for avoiding circular obstacles. The collision avoidance planner π^c minimizes the cost function given by (4.14) subject to constraints

$$\dot{\mathbf{x}}_\rho^i = \mathbf{f}(\mathbf{x}_\rho^i, u_\rho),\ i = 1, \cdots, n, \tag{4.15}$$

$$V\tau_\rho^i \geq r^i, i = 1, \cdots, n, \tag{4.16}$$

where r^i is the radius of the i^{th} obstacle. To guarantee to avoid a single circular obstacle, it is necessary to establish a minimum turn away distance $d_{\min}$ from the obstacle. Let $\phi_{\max}$ represent the maximum roll angle of the UAV and let g represent the gravity constant. The minimum turning radius is then given by [79]

$$r_{mt} = \frac{V^2}{g\tan(\phi_{\max})}. \tag{4.17}$$

Theorem 4.4.1 shows the minimum turn away distance required to avoid a circular obstacle with the radius r using the collision avoidance planner π^c.

Theorem 4.4.1 *Using the collision avoidance planner π^c that minimizes the cost function* (4.14) *subject to the constraints* (4.15) *and* (4.16)*, collision avoidance with a circular obstacle with the radius r is guaranteed if the initial condition satisfies*

$$V\tau_0 > d_{\min} = \sqrt{(r + r_{mt})^2 - r_{mt}^2} - r, \tag{4.18}$$

where τ_0 represents the initial TTC to the circular obstacle. In addition, the UAV converges to a circle around the obstacle with the radius $\max\{r, r_{mt}\}$.

Proof: Consider the worst case scenario where the UAV is flying perpendicular to a circular obstacle O^i in the local-level frame map, as shown in Fig. 4.3. The minimum turn away distance $d_{\min}$ from the obstacle can be determined when the maximum roll angle $\phi_{\max}$ is applied and the generated circle with the minimum turning radius r_{mt} is tangent to the surface of the obstacle.

Based on the geometry, the planner π^c is guaranteed to avoid the obstacle if the initial condition satisfies $V\tau_0 > d_{\min} = \sqrt{(r+r_{mt})^2 - r_{mt}^2} - r$.

To show that the trajectory converges to an orbit around the obstacle, if $V\tau_0 > d_{\min}$, the collision avoidance planner π^c will cause the UAV to move in such a way that the TTC to the obstacle decreases and the bearing to the obstacle increases. The UAV will first reach a configuration at time t where the bearing to the obstacle $\eta_t = \frac{\pi}{2}$ and the range to the obstacle $V\tau_t > r$. Then the planner π^c will further cause the UAV to reach a configuration at time t' such that $\eta_{t'} = \frac{\pi}{2}$ and $V\tau_t > V\tau_{t'} > r$. This process is repeated such that the TTC decreases progressively. Because of the constraints (4.16) on the TTC and the minimum turning radius constraint, the UAV converges to $\max\{r, r_{mt}\}$. ■

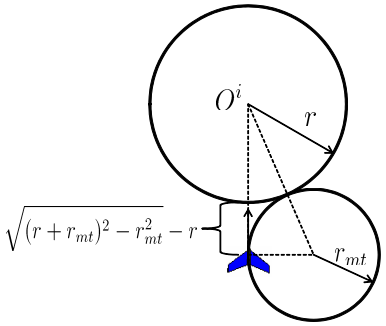


Figure 4.3: This figure shows the worst case scenario that the UAV is flying perpendicular to a circular obstacle.

To this point we have found the conditions under which a single circular obstacle can be successfully avoided using π^c. We extend the analysis to investigate the multiple obstacle avoidance problem. Our approach characterizes the environment with minimum separation between obstacles such that collisions are avoided with all the obstacles.

Let $\mathscr{C}$ represent the configuration space. For two configurations $q_1 = [q_{1n}, q_{1e}, q_{1\psi}]^\top \in \mathscr{C}$ and $q_2 = [q_{2n}, q_{2e}, q_{2\psi}]^\top \in \mathscr{C}$, where q_{in} and q_{ie}, $i = 1,2$, represent North and East coordinates, and $q_{i\psi}$, $i = 1,2$, represent the heading angle, define the distance between q_1 and q_2 as

$$\|q_1 - q_2\| \triangleq \sqrt{(q_{1n} - q_{2n})^2 + (q_{1e} - q_{2e})^2}. \tag{4.19}$$

For a configuration q and the i^{th} obstacle O^i, we define the distance between q and the boundary of O^i as

$$d_q^i \triangleq \min_{q' \in \partial O^i} \|q - q'\|. \tag{4.20}$$

Let $d_{\min}^i$ and $d_{\min}^j$ represent the minimum turn away distance for the i^{th} obstacle O^i and the j^{th} obstacles O^j given by Eq. (4.18). Let

$$d^{ij} \triangleq \min_{p_i \in \partial O^i, p_j \in \partial O^j} \|p_i - p_j\| \tag{4.21}$$

represent the shortest distance between the points along the boundaries of O^i and O^j. Let q_0 represent the initial UAV configuration. Theorem 4.4.2 describes the characteristics of the environment in which the collision avoidance planner π^c guarantees collision-free paths.

Theorem 4.4.2 *If the environment satisfies $d^{ij} > \max\{d_{\min}^i, d_{\min}^j\}$, $\forall i, j$ and the initial UAV configuration q_0 satisfies $d_{q_0}^i > d_{\min}^i$, $\forall i$, where d^{ij} is the distance between the i^{th} and the j^{th} obstacles given by Eq.* (4.21)*, $d_{\min}^i$ and $d_{\min}^j$ represent the minimum turn away distance for the i^{th} and the j^{th} obstacles given by Eq.* (4.18)*, and $d_{q_0}^i$ is the distance between q_0 and the i^{th} obstacle given by Eq.* (4.20)*, then the collision avoidance planner π^c, which minimizes the cost function* (4.14) *subject to constraints* (4.15) *and* (4.16)*, guarantees that the UAV will avoid all obstacles in the future.*

Proof: Consider that the UAV is initially located at q_0 with $d_{q_0}^i > d_{\min}^i$, $\forall i$, and that it will collide with an obstacle O^i if it flies along its initial heading, as shown in Fig. 4.4. Since $d_{q_0}^i > d_{\min}^i$ and $d^{ij} > \max\{d_{\min}^i, d_{\min}^j\}$, in the worse case scenario the planner π^c leads to a collision-free path from q_0 to q_A on the boundary of O^i with direction tangent to the boundary, where $d_{q_A}^j > d_{\min}^j$. This means that the UAV certainly has the capability to avoid the obstacle O^j when it reaches q_A. In addition, since $d^{jk} > \max\{d_{\min}^j, d_{\min}^k\}$, in the worse case scenario the planner π^c leads to a collision-free path from q_A to q_B on the boundary of O^j with direction tangent to the boundary, where $d_{q_B}^k > d_{\min}^k$. This process can be repeated so that the UAV does not collide with any obstacle using π^c for all time t. ■

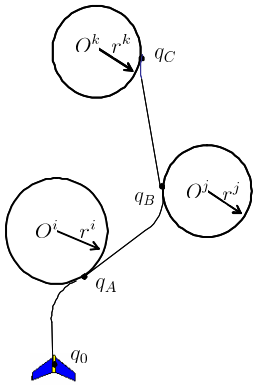


Figure 4.4: This figure shows the collision avoidance planner π^c maneuvers the UAV to avoid multiple obstacles.

4.4.2 Goal reaching

Besides the collision avoidance behavior of the planner π^c, we are also interested in its goal reaching behavior when it is combined with the planner π^g. In this section, we combine the two path planners using a switching algorithm that executes them alternately.

Remark 4.4.2 *We analyze the goal reaching behavior of the switching algorithm to simplify the determination of analytical conditions under which the collision avoidance planner π^c is guaranteed to drive the UAV to the goal when it is combined with the goal reaching planner π^g. However, in simulation we use the observability-based planning algorithm π^o that takes into account collision avoidance and goal reaching simultaneously.*

The switching algorithm is described as follows. The algorithm first executes the goal reaching planner π^g to maneuver the UAV toward the goal from an initial configuration. If there exist obstacles with the TTC no greater than τ^l that collide with the UAV, the algorithm executes the collision avoidance planner π^c to react to nearby obstacles. The collision avoidance planner π^c is executed until the UAV reaches a configuration such that the goal reaching planner π^g can generate a path from that configuration to the goal, which does not collide with any obstacle with TTC less than τ^l. The algorithm then executes the goal reaching planner π^g. This process is repeated until the UAV reaches the goal.

The switching algorithm needs conditions on the environment to ensure that it drives the UAV to the goal. The following theorem describes the conditions under which the UAV can reach

the goal from an initial configuration with the heading pointing to the goal using the switching algorithm.

Theorem 4.4.3 *If the environment satisfies $d^{ij} > 2V\tau^l$, $\forall i, j$ and the initial UAV configuration q_0 with the heading pointing to the goal satisfies $d^i_{q_0} > d^i_{\min}$, $\forall i$, then the UAV is guaranteed to be maneuvered to the goal using the switching algorithm.*

Proof: Let q_0 represent the initial UAV configuration with the heading pointing to the goal at time t_0 as shown in Fig. 4.5 and let $d_{t_0} = \|q_0 - q_f\|$ represent the distance between q_0 and q_f. Consider a scenario that there exists an obstacle O^i in the UAV's initial course towards the the goal q_f and that the TTC to the obstacle is no greater than τ^l. For this scenario, since $d^{ij} > 2V\tau^l$, $\forall i, j$, there are no other obstacles with TTC no greater than τ^l when the UAV is located at q_0. The switching algorithm executes the collision avoidance planner π^c to react to the obstacle O^i. Based on Theorem 4.4.1, the planner π^c will cause the distance between the UAV and the obstacle O^i to decrease until the UAV converges to a circle with the radius $\max\{r, r_{mt}\}$. This implies that the UAV will stay within the circle C^i centered at O^i with the radius $V\tau^l$, where the planner π^c only reacts to the obstacle O^i until the UAV converges to the circle with the radius $\max\{r, r_{mt}\}$. Accordingly, the switching algorithm executes the collision avoidance planner π^c until the UAV reaches a configuration q_1 at time t_1 such that the goal reaching planner π^g can generate a path from q_1 to q_f that does not collide with the obstacles O^i. While the UAV flies from q_0 to q_1 using π^c, the bearing to the obstacle O^i is no greater than $\frac{\pi}{2}$. Since the UAV inertial angle to the goal during its flight from q_0 to q_1 is less than the bearing to the obstacle, the inertial angle to the goal must be less than $\frac{\pi}{2}$. Therefore, it must be that $d_{t_1} = \|q_1 - q_f\| < d_{t_0}$.

Once the UAV reaches q_1, the algorithm executes the goal reaching planner π_g until the UAV reaches a configuration q_2 outside of the circle C^i, where another obstacle O^j with TTC no greater than τ^l exists in the UAV's course towards the goal. It is apparent that $d_{t_2} = \|q_2 - q_f\| < d_{t_1}$. Once the UAV reaches q_2, the switching algorithm executes the collision avoidance planner π^c to react to the obstacle O^j. As the process is repeated, the distance between the UAV and the goal decreases progressively and the UAV will be eventually maneuvered to the goal using the switching algorithm. ■

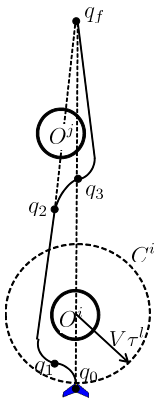


Figure 4.5: This figure shows the UAV can be maneuvered to q_f using the switching algorithm.

The conditions given by Theorem 4.4.3 require that each two obstacles in the environment are separated far enough so that the UAV reacts to and avoids obstacles one by one until it reaches the goal. We assume the environment satisfies these conditions in order to provide a theoretical guarantee for the goal reaching behavior of the collision avoidance planner π^c when it is combined with the goal reaching planner π^g. The conditions may not be necessary for the observability-based planning algorithm π^o to achieve goal reaching performance. This implies that there may exist environments that do not satisfy the conditions but where the UAV can still be maneuvered to the goal without causing collisions using π^o.

4.5 Numerical results

In this section, we tested the observability-based planning algorithm π^o that minimizes the cost function (4.8) and takes into account collision avoidance and goal reaching simultaneously using a simulation environment developed in MATLAB/SIMULINK. The simulator uses a six degree of freedom model of the aircraft. The coordinate system is represented by NED (north-east-down) system. The covariance matrices of the process and measurement noises were $\mathbf{Q}^i = \begin{pmatrix} 0.001 & 0 \\ 0 & 0.0076 \end{pmatrix}$ and $\mathbf{R}^i = 0.0012$. The weighting scalars a_1 and a_2 were 10 and 1. All the weighting scalars $b_i = 2$, $i = 1, \cdots, n$. A look-ahead policy over a horizon 3.6 seconds was used. The ground speed was $V = 13$ m/s. The maximum roll angle for the UAV was 30°. We tested the algorithm for both single and multiple obstacle avoidance scenarios. We also conducted Monte Carlo simulations to test the collision avoidance and goal reaching performance

of the observability-based planning algorithm π^o with varying measurement uncertainties in the environments with varying minimum distance between obstacles.

4.5.1 Single obstacle avoidance

In this scenario, the UAV was commanded to maneuver around an obstacle located at (150,250) between waypoint **S** (0,100,-40) and waypoint **E** (600,700,-40) represented by the box and plus signs shown in Fig. 4.6(a).

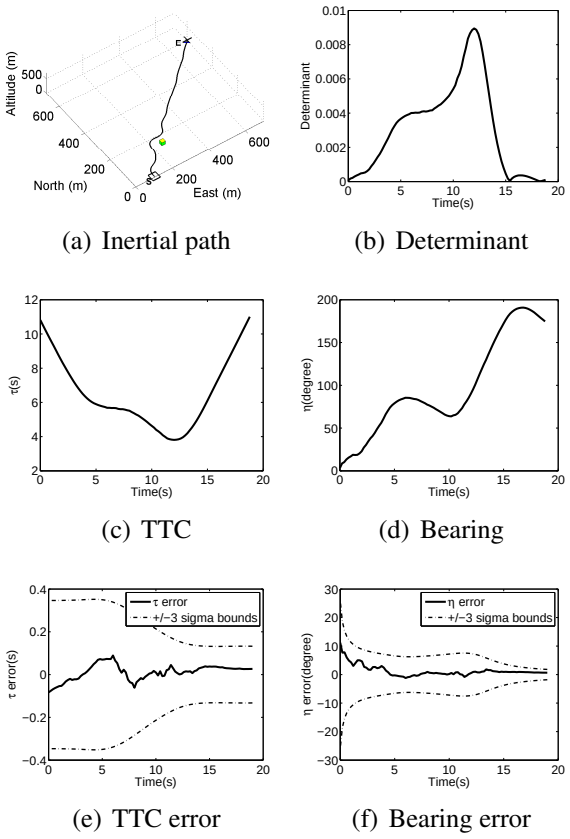
Figure 4.6 shows the path followed by the UAV for avoiding the obstacle using the planning algorithm π^o, the determinant of $\mathbf{O}^{i\top}\mathbf{O}^i$ for that obstacle, the TTC and bearing, and the TTC and bearing estimation error. It can be seen that when the determinant is maximum, then the bearing is $\eta = \frac{\pi}{2}$ and the TTC reaches its minimum value $\tau_{min} \approx 4$ s. At the same time, the bound on the error covariance for the TTC is minimum, which shows that the uncertainties in state estimates can be minimized while simultaneously avoiding collisions.

4.5.2 Multiple obstacle avoidance

In the multiple obstacle avoidance scenario, the UAV was commanded to maneuver through twenty-five obstacles between waypoint **S** (0,100,-40) and waypoint **E** (600,700,-40), as shown in the subfigures on the right of Fig. 4.7.

Figure 4.7 shows the evolution of the local map in the local-level frame and the update of the path in the inertial frame at different time. The dashed circles in the subfigures on the left represent the TTC at 3 s, 6 s, and 9 s for the inner, middle, and outer circles respectively. The plus sign in subfigure (d) on the left represents the waypoint **E** in the local-level frame. Red lines in the subfigures on the right represent the paths followed by the UAV and black lines represent the optimal look-ahead paths. Figure 4.8 shows the TTC and bearing to the obstacle located at (150,250), the TTC and bearing estimation error, and the determinant of $\mathbf{O}^{i\top}\mathbf{O}^i$ for that obstacle. We can see that minimizing the cost function for multiple obstacle avoidance gives the same behavior for the obstacle avoidance, observability and further estimation uncertainties.

Figure 4.9 shows how the value of the cost function changes as time progresses. Based on the figure, the cost function decreases initially when there are no obstacles in the local map.



(a) Inertial path (b) Determinant

(c) TTC (d) Bearing

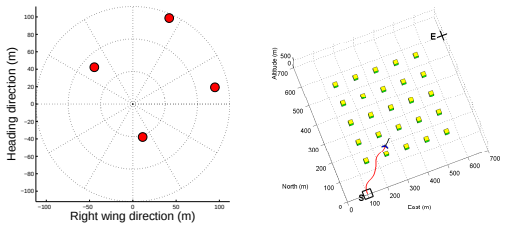
(e) TTC error (f) Bearing error

Figure 4.6: This figure shows the simulation results for single obstacle avoidance problem. Subfigure (a) shows the inertial path. Subfigure (b) shows the determinant of $\mathbf{O}^{i\top}\mathbf{O}^{i}$. Subfigures (c) and (d) show the TTC and bearing to the obstacle. Subfigure (e) and (f) show the estimation error and $\pm 3\sigma$ bounds of the error covariance for the TTC and bearing.

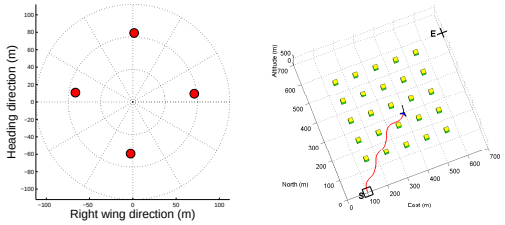
The cost function only consists of the first term. Once a new obstacle pops up, the cost function increases because the obstacle term is added to the cost function. The planning algorithm π^o then minimizes the second term, causing the cost function to decrease. Once the collision is avoided and the obstacle is passed, it does not add any cost to the cost function. The cost function then decreases based on the first term. Similar behavior occurs when multiple obstacles are observed.

4.5.3 Monte Carlo simulation

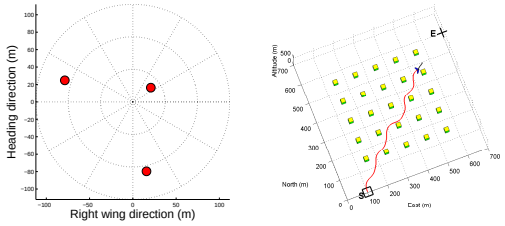
To simplify the analysis of collision avoidance and goal reaching performance and determine analytical conditions, in previous section we decompose the observability-based planning algorithm π^o into the collision avoidance planner π^c and the goal reaching planner π^g. We then



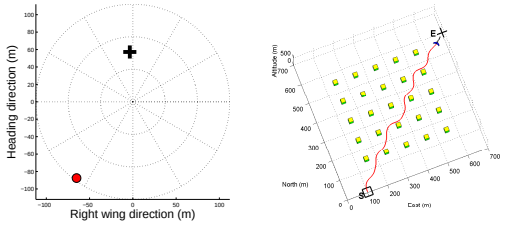
(a) t=20 s



(b) t=35 s



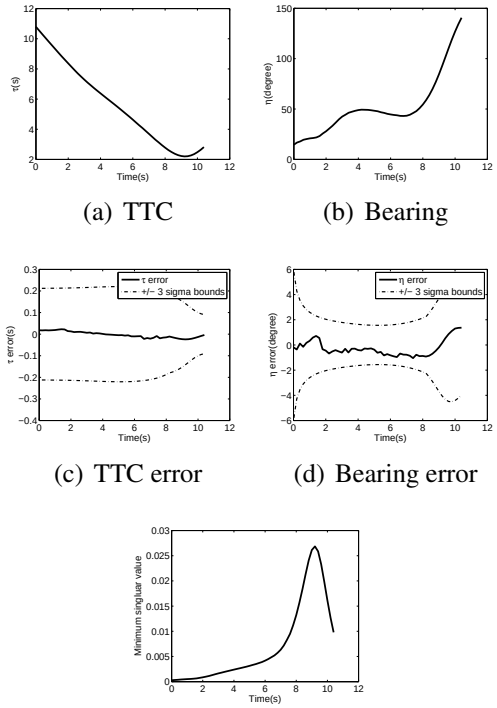
(c) t=53 s



(d) t=65 s

Figure 4.7: This figure shows the evolution of the local map and the update of the path at different times. Subfigures on the left show the evolution of the local map. The dashed circles represent the TTC at 3 s, 6 s, and 9 s for inner, middle, and outer circles respectively. Subfigures on the right show the path in the inertial frame. The black lines represent the three-step look-ahead paths and red lines represent the actual path followed by the UAV.

analyze of obstacle avoidance behavior of the planner π^c and the goal reaching behavior of the switching algorithm that executes the two planners alternately. Accordingly, the conditions for collision avoidance and goal reaching described in Theorem 4.4.2 and 4.4.3 may not be identical for the observability-based planner π^o to achieve collision avoidance and goal reaching performance.



(a) TTC (b) Bearing

(c) TTC error (d) Bearing error

(e) Determinant

Figure 4.8: This figure shows the TTC and bearing to the obstacle located at (150,250), the TTC and bearing tracking error and the determinant of $\mathbf{O}^{i\top}\mathbf{O}^{i}$. Subfigures (a) and (b) show the TTC and bearing. Subfigure (c) and (d) show the error and $\pm 3\sigma$ bounds of the error covariance. Subfigure (e) shows the determinant of $\mathbf{O}^{i\top}\mathbf{O}^{i}$.

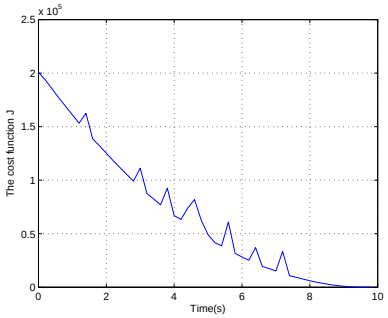


Figure 4.9: This picture shows the change in the value of the cost function.

Accordingly, we conducted Monte Carlo simulations to demonstrate the statistical performance of the observability-based planning algorithm π^{o}.

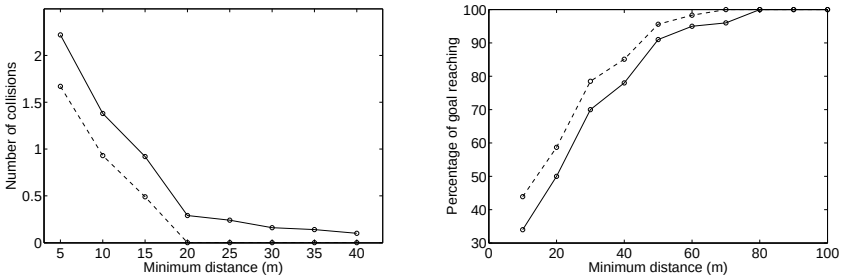
For each environment with a fixed minimum distance between obstacles, we executed 100 simulation runs. In each simulation run, the UAV was maneuvered from the initial position

(120,120,-40) to the end position (580,580,-40) through an environment using π^o. The environment was constructed so that each obstacle was added to the environment based on a uniform distribution over the square area with the South-West corner (100,100) and the North-East corner (600,600) until no more obstacles could be added. The height and radius for all obstacles were 100 meters and 20 meters, and the UAV was flying at a height of 40 meters. The observability-based planner π^o reacts to obstacles with TTC no greater than τ^l =4 s. We evaluate two criteria: the number of collisions and the percentage of runs where UAV reached the goal. We say that the UAV reaches the goal if it is maneuvered to the goal in $t < 100$ seconds without any collisions.

Figure 4.10 (a) plots the average number of collisions over 100 simulation runs versus the minimum distance between obstacles for the case where the standard deviation of the azimuth measurement noise is 2°, as shown by the solid line, and for the case where the locations of obstacles are perfectly known, as shown by the dashed line. The figure shows the average number of collisions decreases dramatically as the minimum distance between obstacles increases from 5 to 20 meters for both cases. After the minimum distance is greater than 20 meters, the average number of collisions decreases slowly for the case with measurement uncertainties and the average number of collisions is zero for the case where the locations of obstacles are perfectly known. The results match the obstacle avoidance behavior of the collision avoidance planner π^c. Given $V = 13$ m/s, $\phi_{\max} = 30^\circ$, and $R_i = 20$ m, $\forall i$, the minimum distance satisfying the obstacle avoidance conditions of Theorem 4.4.2 for the planner π^c is 19.9345 meters. When the minimum distance is less than 19.9345 meters, the number of collisions decreases quickly as the minimum distance increases. When the minimum distance is greater than 19.9345 meters, the conditions of Theorem 4.4.2 are satisfied. The collision avoidance planner π^c guarantees collision-free paths if the obstacle locations are perfectly known. For the case with the measurement uncertainties, the UAV still encounters a small number of collisions when the minimum distance is greater than 20 meters. In addition, when the minimum distance is 10 and 15 meters for the case with perfectly known obstacle locations, the average number of collisions is less than one, which implies that there exist environments that do not satisfy the conditions of Theorem 4.4.2 but where the observability-based planner π^o still generates collision-free paths.

Figure 4.10 (b) plots the percentage of runs where the UAV reached the goal versus the minimum distance for the two cases. The percentage increases as the minimum distance be-

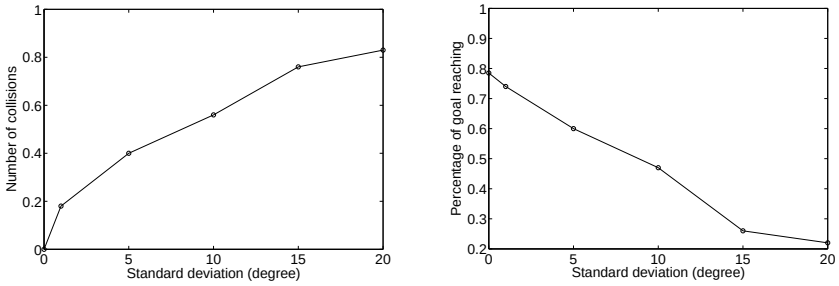
tween obstacles increases. When the minimum distance is greater than 80 meters for the case with measurement uncertainties or when the minimum distance is greater than 70 meters for the case with perfectly known obstacle locations, the UAV is always maneuvered to the goal using the observability-based planner. In addition, the percentage for all the environments with the minimum distance from 10 to 100 meters for both cases is nonzero. Accordingly, the minimum distance $2V\tau^l = 104$ m, which satisfies the goal reaching conditions of Theorem 4.4.3 for the switching algorithm, is not necessary for the observability-based planner π^o to achieve goal reaching performance.



(a) (b)

Figure 4.10: This figure shows the statistical performance of the observability-based planning algorithm implemented in the environments with varying minimum distance between obstacles for the cases with the standard deviation of the azimuth measurement noise 2° and with perfectly known obstacle locations. Subfigure (a) show the plots of the average number of collisions over 100 simulation runs versus the minimum distance between obstacles. Subfigure (b) show the plots of the percentage of runs when the UAV reached the goal versus the minimum distance.

To take into account the effect of measurement uncertainties, Monte Carlo simulations were also conducted to test the performance of the algorithm π^o with varying measurement uncertainties. Similarly, we evaluate the number of collisions and the percentage of runs for the UAV to reach the goal. Figure 4.11 plots the average number of collisions over 100 simulation runs and the percentage of runs where the UAV reached the goal versus the standard deviation of the azimuth measurement noise for the environment with the minimum distance 30 meters. Based on the figure, as the standard deviation of the measurement noise increases, the number of collisions increases and the percentage of runs where the UAV reached the goal decreases.



(a) (b)

Figure 4.11: This figure shows the statistical performance of the observability-based planning algorithm with varying measurement uncertainties in the environment with the minimum distance 30 meters. Subfigure (a) plots the average number of collisions over 100 simulation runs versus the standard deviation of the azimuth measurement noise. Subfigure (b) shows the percentage of runs when the UAV reached the goal versus the standard deviation.

4.6 Conclusions

This chapter presents an observability-based planning algorithm using bearing-only measurements. We perform a nonlinear observability analysis for state estimation and argue that collision avoidance and uncertainty minimization problems are complementary. Based on this analysis, we design a cost function that minimizes the estimation uncertainties while simultaneously avoiding obstacles. By minimizing the cost function, the path planning algorithm is developed directly in the local-level frame. We use a look-ahead policy to plan optimal paths over a finite time horizon. Numerical results show that the observability-based planning algorithm is successful in solving the single and multiple obstacle avoidance problems while improving the estimation accuracy.

CHAPTER 5. BEARING-ONLY REACTIVE COLLISION AVOIDANCE FOR UAVS

In Chapter 4, bearing measurement is used to estimate the time-to-collision TTC for an obstacle and the path planning required both bearing and TTC to avoid the obstacle. However, the estimation of TTC can slow down the path planning algorithm, which is not suitable for dynamic urban environments. In this chapter, we design a path planning algorithm for UAVs, which uses bearing-only measurement to avoid the obstacles.

5.1 Introduction

Recently, the use small Unmanned Aerial Vehicles (UAVs) in several civil and military applications have increased significantly. Smaller UAVs have different applications that require them to operate in urban terrains. Urban environments consist of trees, poles, buildings, walls, tunnels, canyons, etc. Since the urban environments are dynamic in nature, locations of some objects may not be known a priori, it is necessary to develop local path planning algorithms, which are fast enough to avoid pop up obstacles and guarantee obstacle avoidance.

In this chapter, we consider two types of object for collision avoidance including cylindrical obstacles and walls. Almost all the obstacles in an urban terrain can be modeled as cylindrical obstacles or walls. Our focus is to develop reactive guidance strategies for collision avoidance, which uses bearing-only measurements. There exists a suite of different sensors, which can provide bearing measurements. One of such sensors is camera, which fits in with size and weight requirements of a small UAV, and has high information to weight ratio.

Obstacle avoidance and path planning algorithms have been extensively studied in literature, especially for ground robots. For example, Probability Road Map (PRM) methods generate random points in the configuration space and connect the points to create a navigation map of the environment [46–48]. Another probabilistic planning technique is the Rapidly-Expanding Random Tree (RRT) [49, 50], which is often used in conjunction with fixed-wing air vehicles. RRTs

generate random points in the configuration space and connect them to a tree of other points such that the non-holonomic constraints of the vehicle are satisfied. Points unable to connect to the tree are removed. While these methods and their many variants (e.g. [46, 89–94]) have shown considerable success, they often require significant computation time to generate paths around obstacles. Therefore, in environments where pop-up threats are common, a reactive planner with fixed computational cost may be more appropriate. Reactive obstacle avoidance methods have been developed in previous work using dynamic replanning [59], potential fields [60], simulated annealing [61], and predefined maneuvers [62].

Potential fields is a common reactive obstacle avoidance technique [60, 62, 95], but unfortunately it, and many of its variants, only guarantee a high probability of avoiding obstacles. Since collisions in flight can be catastrophic, aircraft require guaranteed obstacle avoidance. In previous work, we develop a reactive method that generates a path to one side of an obstacle field, which is used in conjunction with waypoint paths [96]. While the method in [96] produced paths around obstacles, it was susceptible to oscillations in the presence of multiple obstacles. In order guarantee collision avoidance, we need a deterministic guidance strategy that reacts to obstacles within the sensor field-of-view and requires limited computational resources. A deterministic collision avoidance guidance strategy for stationery cylindrical objects is proposed in [63]. However, the guidance strategy in [63] requires both range from obstacle and bearing to obstacle to compute the control law. In this chapter, we extend the work done in [63] by re-deriving the guidance strategy using sliding mode control such that it does not require range from the obstacle, and uses only bearing to obstacle to avoid obstacles. Furthermore, we also extend the guidance strategy to avoid collision from straight and curved walls.

The basic concept of collision avoidance guidance strategy is to move the obstacle in the sensor plane and maintain the obstacle at a desired constant bearing angle. By keeping the obstacle at a desired bearing angle in the sensor plane causes the UAV to converge at a constant distance from the obstacle and guarantees collision avoidance. The reactive guidance strategy is fast, computationally inexpensive, and guarantee collision avoidance, which keeps the obstacle in the field-of-view and minimizes the chances of collision.

The chapter is organized as follows. In Section 5.2 we formulate detail the relative equations of motion and formulate the collision avoidance problem. In Section 5.3 we derive bearing-

only guidance laws for collision avoidance from cylindrical obstacle and following a wall. We provide simulation results in Section 5.4 and provide our conclusions in Section 5.5.

5.2 Model and problem formulation

In this chapter, we consider a fixed wing UAV with a strapdown camera mounted parallel to the x-axis of the body frame and is equipped with an onboard autopilot with inner loop control of roll, pitch, airspeed, and altitude. A forward looking camera allows objects to be viewed in the short reachable region of the UAV. Equation of motion of a UAV flying at constant altitude can be written as

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} V\cos\psi \\ V\sin\psi \\ \frac{g}{V}\tan\phi \end{pmatrix}, \tag{5.1}$$

where $[x,y]^\top$ is the position vector, ψ is the heading, V is the airspeed, ϕ is the commanded roll angle, g is the gravity constant, and we have assumed zero ambient wind.

Almost all the obstacles in urban environments can be modeled either as a cylindrical obstacle (e.g, trees, poles, small buildings, etc.) or a wall (large, buildings, tunnels, subways, canyons, etc). In next two subsections we discuss relative motions of equation for cylindrical obstacles and walls, which will be used to derive the guidance strategies.

5.2.1 Relative equation of motion between cylindrical obstacle and the UAV

The geometry of the cylindrical obstacle avoidance problem is shown in Figure 5.1 where ρ is the range from the UAV to the center of the obstacle and η is the bearing to the center of the obstacle. The relative equations of motion for the system are

$$\dot{\rho} = -V\cos\eta, \tag{5.2}$$

$$\dot{\eta} = \frac{V}{\rho}\sin\eta - \frac{g}{V}\tan\phi. \tag{5.3}$$

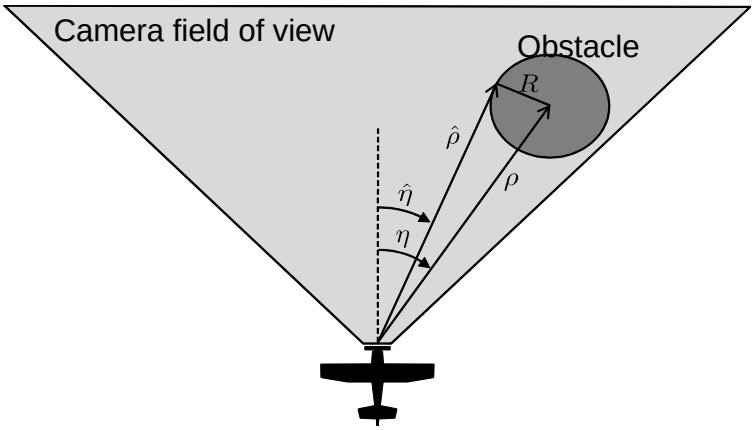


Figure 5.1: A conceptual view of the UAV approaching an obstacle. The guidance law moves the obstacle to the edge of the camera field-of- view to maintain safe distance between the UAV and the obstacle.

We will assume that the obstacle is a cylinder of radius R, and pose the guidance problem with respect to the range and bearing to the edge of the cylinder as shown in Figure 5.1 and from Figure 5.1 we have that

$$R = \rho \sin(\eta - \hat{\eta}), \tag{5.4}$$

$$\hat{\rho} = \rho \cos(\eta - \hat{\eta}). \tag{5.5}$$

Differentiating (5.4) and (5.5) with respect to time and using Equations (5.2) and (5.3) we get the modified relative equations of motion

$$\dot{\hat{\eta}} = -\frac{g}{V}\tan\phi + \frac{V}{\hat{\rho}}\sin\hat{\eta}, \tag{5.6}$$

$$\dot{\hat{\rho}} = -V\left[\cos\hat{\eta} - \frac{R}{\hat{\rho}}\sin\hat{\eta}\right]. \tag{5.7}$$

5.2.2 Relative equation of motion between wall and the UAV

The geometry between the UAV and a wall is shown in Figure 5.2. This geometry is similar to geometry for stationery obstacle, however in this case, the obstacle is moving on the wall. We

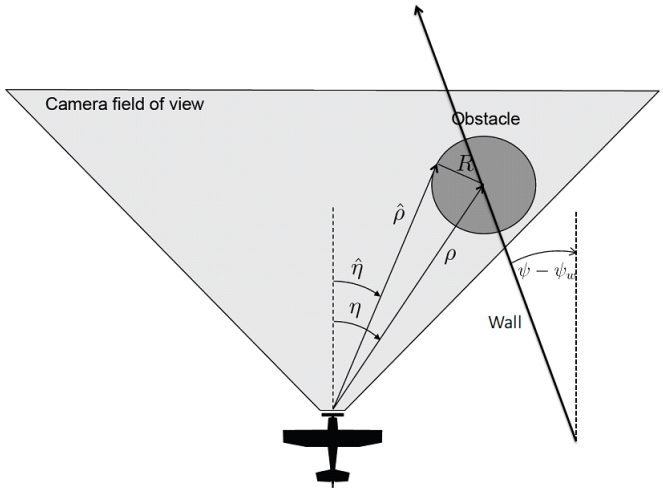


Figure 5.2: A conceptual view of the UAV approaching a wall. The guidance law keeps the wall at the edge of the camera field of view to avoid collision.

consider a virtual cylindrical obstacle of radius R. This obstacle moves on the wall with velocity

$$V_w = V\cos(\psi - \psi_w), \tag{5.8}$$

where $\psi - \psi_w$ is the relative orientation of wall with respect to UAV heading, V_w is the projection of UAV velocity on the wall. The relative orientation of the wall $\psi - \psi_w$ is directly computed from the image plane using the the image segmentation followed by orientation computation of the segment. By keeping the virtual obstacle at desired bearing angle, the portion of the wall always stays in the image plane and UAV can fly along the wall and maintain a safe distance from the wall. The minimum distance between wall and UAV is given as

$$R_{min} = \rho\sin(\psi - \psi_w + \hat{\eta}). \tag{5.9}$$

To avoid collision from the wall the UAV should maintain $R_{min} > 0$ for all time. From Figure 5.2

the equations of relative motion between UAV and the obstacle moving on wall can be written as

$$\dot{\rho} = V_w \cos(\psi - \psi_w - \eta) - V\cos(\eta), \tag{5.10}$$

$$\dot{\eta} = -\frac{V_w \sin(\psi - \psi + w + \eta) - V\sin(\eta)}{\rho} + \frac{g}{V}\tan(\phi). \tag{5.11}$$

Differentiating (5.4) and (5.5) with respect to time and using Equations (5.10) and (5.11) we get the modified relative equations of motion

$$\dot{\hat{\eta}} = \frac{V\sin\hat{\eta}}{\hat{\rho}} - \frac{V_w \sin(\psi_w - \psi + \hat{\eta})}{\hat{\rho}} + \frac{g}{V}\tan\phi, \tag{5.12}$$

$$\dot{\hat{\rho}} = \frac{R}{\hat{\rho}} V\sin\hat{\eta} + V_w\cos(\psi_w - \psi + \hat{\eta}) - \frac{R}{\hat{\rho}} V_w \sin(\psi_w - \psi + \hat{\eta}) - V\cos\hat{\eta}. \tag{5.13}$$

5.3 Bearing-only guidance strategies

Our approach is to push the edge of the obstacle to a specified angle η_d in the image plane. By keeping obstacle at a desired bearing angle the UAV maintains safe distance from the obstacle. Also, keeping the obstacle in the field-of-view is important because if the obstacle is pushed out of the field of view of the camera, then it can no longer be tracked by the guidance algorithm and a collision may occur. Following two subsections detail the collision avoidance guidance strategies for a cylindrical obstacle and a wall.

5.3.1 Guidance strategy for cylindrical obstacle

In this subsection we derive the control law for cylindrical obstacle. If $\bar{\eta}$ is the field of view of the camera, then we desire that $|\eta| \leq \bar{\eta}$. Let $|\eta_d| < \bar{\eta}$ be the desired position of the edge of the obstacle in the image plane. We use Lyapunov‘s stability theory to move the obstacle at a desired bearing angle in the image plane.

Control law for cylindrical obstacle using range and bearing measurement

For comparison purpose, we re-derive the control law [63] for cylindrical obstacle.

Consider the Lyapunov function candidate

$$W = \frac{1}{2}(\hat{\eta} - \eta_d)^2.$$

Differentiating with respect to time gives

$$\dot{W} = (\hat{\eta} - \eta_d)\left(-\frac{g}{V}\tan\phi + \frac{V}{\hat{\rho}}\sin\hat{\eta}\right).$$

Therefore, selecting the guidance law as

$$\phi = \tan^{-1}\left(\frac{V^2}{g\hat{\rho}}\sin\hat{\eta} + \frac{Vk}{g}(\hat{\eta} - \eta_d)\right), \tag{5.14}$$

gives

$$\dot{W}_1 = -k(\hat{\eta} - \eta_d)^2,$$

which implies that $\hat{\eta}(t) \to \eta_d$. The control law (5.14) is a function of both range and bearing measurement.

Control law for cylindrical obstacle using bearing-only measurement

To make control law in (5.14) independent of range from the obstacle we use sliding mode control to derive the control law. Suppose we can design a control law that constraints the motion of the system to the manifold $s = \hat{\eta} - \hat{\eta_d} = 0$, where $\hat{\eta} = \hat{\eta_d}$. The motion on this manifold is independent of $\frac{V\sin\hat{\eta}}{\hat{\rho}}$. The objective is to bring the states to this manifold. Consider Lyapunov function candidate $V = \frac{1}{2}s^2$. Differentiate V to obtain

$$\dot{V} = s\left(\frac{V\sin\hat{\eta}}{\hat{\rho}} - \frac{g}{V}\tan\phi\right), \tag{5.15}$$

$$\leq |s|\frac{V|\sin\hat{\eta}|}{\hat{\rho}_{min}} - s\frac{g}{V}\tan\phi, \tag{5.16}$$

where $\hat{\rho}_{min} > 0$ is the lower bound on $\hat{\rho}$. Choosing

$$\phi = \tan^{-1}\frac{V}{g}\left\{\left(\frac{|V\sin\hat{\eta}|}{\hat{\rho}_{min}} + \beta_0\right) sat(\frac{\hat{\eta} - \hat{\eta}_d}{\varepsilon})\right\}, \tag{5.17}$$

where constants $\beta_0 > 0$ and $\hat{\rho}_{min} > 0$ yields

$$\dot{V} \leq -\beta_0 |\hat{\eta} - \hat{\eta}_d| < 0, \; \forall \; |\hat{\eta} - \hat{\eta}_d| > \varepsilon. \tag{5.18}$$

Therefore, when $|\hat{\eta} - \hat{\eta}_d| > \varepsilon$, $|s(t)|$ is strictly decreasing, until it reaches the set $\{|s| \leq \varepsilon\}$ in finite time and remains inside thereafter and $|\hat{\eta} - \hat{\eta}_d| \leq \varepsilon$.

By holding the edge of a cylindrical object at angle η_d in the camera field of view the UAV will orbit the object, on a circular trajectory of radius $R\sqrt{1+\tan^2 \eta_d}$ [63], as shown in Figure 5.3. If the UAV is following a waypoint path and an obstacle appears in the image, the collision avoidance strategy is then to push the edge of the obstacle to an angle of η_d in the image plane using guidance strategy (5.14), until the UAV moves past the obstacle and can resume tracking its original path.

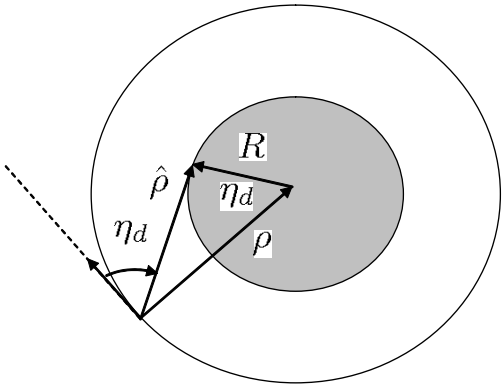


Figure 5.3: By keeping the object at angle η_d in the camera field of view, the UAV converges to an orbit of radius $R\sqrt{1+\tan^2 \eta_d}$ [63].

5.3.2 Guidance strategy for following a wall

If $\bar{\eta}$ is the field of view of the camera, then we desire that $|\eta| \leq \bar{\eta}$. Let $|\eta_d| < \bar{\eta}$ be the desired position of the edge of the virtual obstacle on the wall in the image plane.

Control law for following a wall using range and bearing measurement

To derive the control law for wall following, we choose a Layapunov candidate function as

$$W = \frac{1}{2}(\hat{\eta} - \eta_d)^2,$$

and differentiate to obtain

$$\dot{W} = (\hat{\eta} - \eta_d)(\frac{V \sin\hat{\eta}}{\hat{\rho}} + \frac{V_w \sin(\psi - \psi_w + \hat{\eta})}{\hat{\rho}} + \frac{g}{V}\tan\phi).$$

Therefore, picking the guidance law

$$\phi = \tan^{-1}(-\frac{V^2}{\hat{\rho} g}\sin\hat{\eta} + \frac{1}{\hat{\rho} g} V V_w \sin(\psi_w - \psi + \hat{\eta}) - \frac{Vk}{g}(\hat{\eta} - \eta_d)), \tag{5.19}$$

gives

$$\dot{W} = -k(\hat{\eta} - \eta_d)^2,$$

which implies that $\eta_1(t) \to \eta_d$.

Control law for following a wall using bearing-only measurement

To make control law in (5.19) independent of range from the obstacle we use sliding mode control to derive the control law. We choose the sliding manifold $s = \hat{\eta} - \hat{\eta_d}$ and consider Lyapunov function candidate $V = \frac{1}{2}s^2$. Differentiate V to obtain

$$\dot{V} = s\left(\frac{V \sin\hat{\eta} - V_w \sin(\psi - \psi_w + \hat{\eta})}{\hat{\rho}} - \frac{g}{V}\tan\phi\right), \tag{5.20}$$

$$\leq |s|\frac{V|\sin\hat{\eta}| + |V_w||\sin(\psi - \psi_w + \hat{\eta})|}{\hat{\rho}_{min}} - s\frac{g}{V}\tan\phi. \tag{5.21}$$

Choosing

$$\phi = \tan^{-1}\frac{V}{g}\left\{\left(\frac{|V\sin\hat{\eta}| + |V_w||\sin(\psi - \psi_w + \hat{\eta})|}{\hat{\rho}_{min}} + \beta_0\right) sat(\frac{\hat{\eta} - \hat{\eta}_d}{\varepsilon}),\right\}, \tag{5.22}$$

where constants $\beta_0 > 0$ and $\hat{\rho}_{min} > 0$ yields

$$\dot{V} \leq -\beta_0 |\hat{\eta} - \hat{\eta}_d| < 0, \; \forall \; |\hat{\eta} - \hat{\eta}_d| > \varepsilon. \tag{5.23}$$

Therefore, when $|\hat{\eta} - \hat{\eta}_d| > \varepsilon$, $|s(t)|$ is strictly decreasing, until it reaches the set $\{|s| \leq \varepsilon\}$ in finite time and remains inside thereafter and $|\hat{\eta} - \hat{\eta}_d| \leq \varepsilon$.

Remark 5.3.1 *It should be noted that if we replace $V_w = 0$ in controller for wall in* (5.19) *and (5.22) we get controller equation for a stationary obstacle in (5.14) and (5.18) respectively. Therefore, only one controller in (5.22) can be used for collision avoidance from both cylindrical obstacles and walls.*

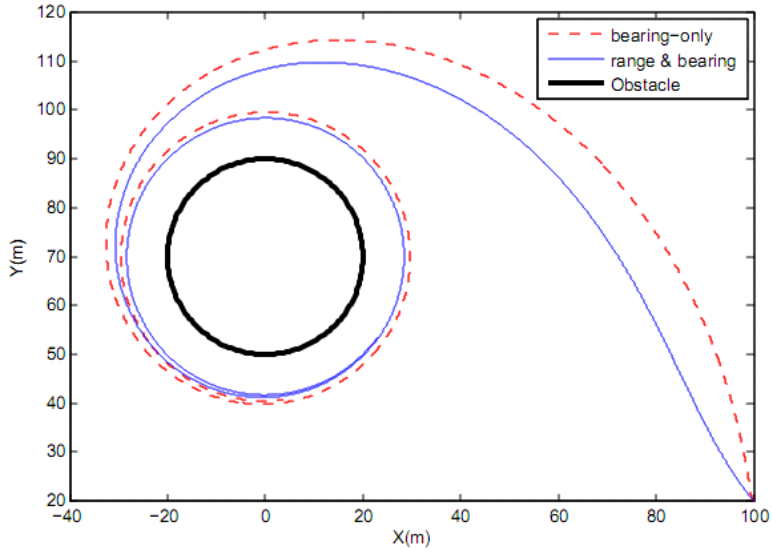
Guidance law in (5.19) enables the UAV to maintain the virtual obstacle at a constant bearing angle in the image plane. This indirectly helps the UAV to maintain safe distance from wall while following the wall.

5.4 Results

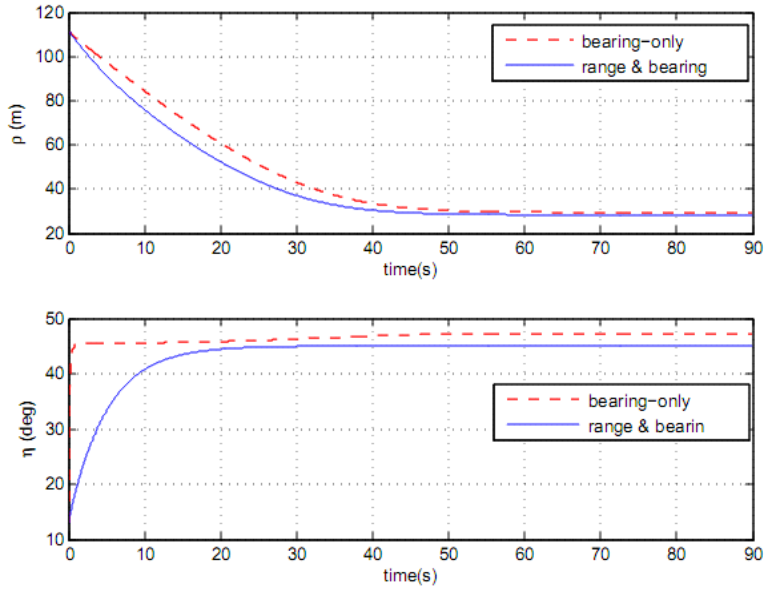
In this section we include simulation results to validate the collision avoidance algorithm. Some of the simulation parameters are; $V = 15m/s$,$k = 0.5$, $\beta_0 = 0.5$, and $\hat{\rho}_{min} = 1$.

We start with a stationary cylindrical obstacle of radius $R = 20m$. Figure 5.4(a) shows the trajectory of UAV around the obstacle using normal controller and sliding mode controller respectively. For both controllers, the UAV converges to a circular trajectory around obstacle of radius $R\sqrt{1 + \tan^2 \eta_d}$. In Figure 5.4(b) it can be seen that the states using sliding mode control are almost equivalent to the states using normal controller. Figure 5.5 shows the flight path of the UAV with multiple cylindrical obstacles in the initial flight path. It can be seen that UAV avoids all the obstacles successfully using the sliding mode controller.

Trajectories of UAV, close to a straight wall with orientation $\psi_w = 60^o$, using both normal and sliding mode controllers are shown in Figure 5.6(a). For both controllers the UAV converges on same trajectory parallel to the wall. Figure 5.6(b) shows that the $\psi \to \psi_w$, $\hat{\eta} \to \hat{\eta}_d$, and $R_{min} \to R_f > R \sin \hat{\eta}_d$. The guidance strategy can also follow and avoid collision from a curved wall. Figure 5.7(a) shows the trajectory of a UAV along a sinusoidal wall. It can be seen that the



(a) UAV trajectory around a cylindrical obstacle of radius 20m



(b) Range and bearing plots

Figure 5.4: Cylindrical obstacle avoidance

UAV also moves along a sinusoidal trajectory while maintains safe distance from the wall. The Figure 5.7(b) shows the plots of ψ, $\hat{\eta}$, ρ, and R_{min}, which are generated using both normal and sliding mode controller. It can be seen that $R_{min} > R\sin(\hat{\eta}_d)$, hence, collision is avoided successfully.

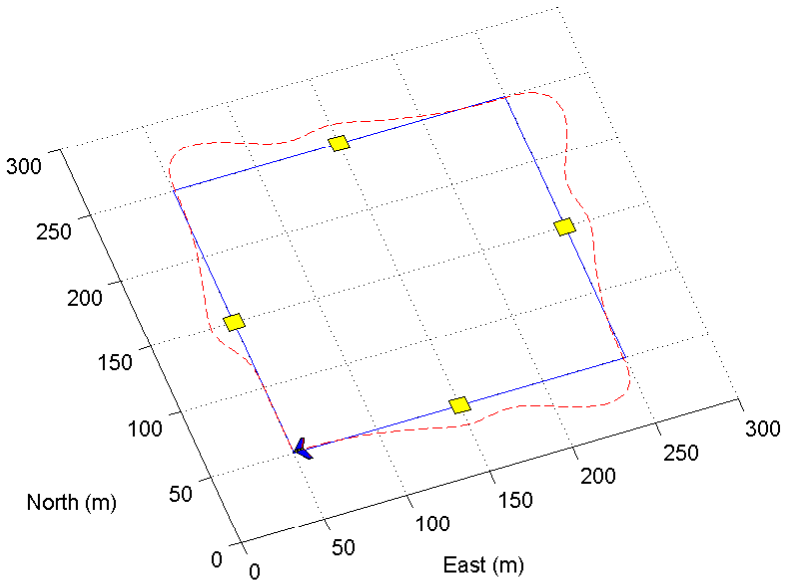
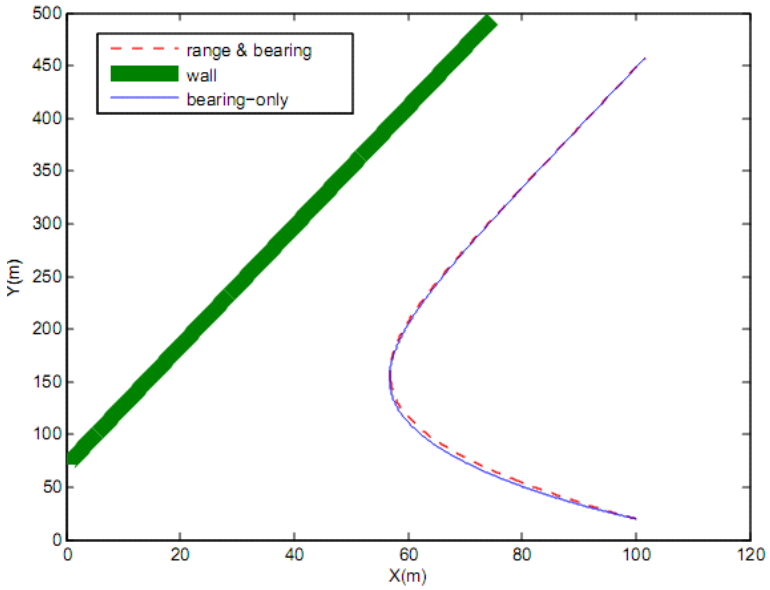


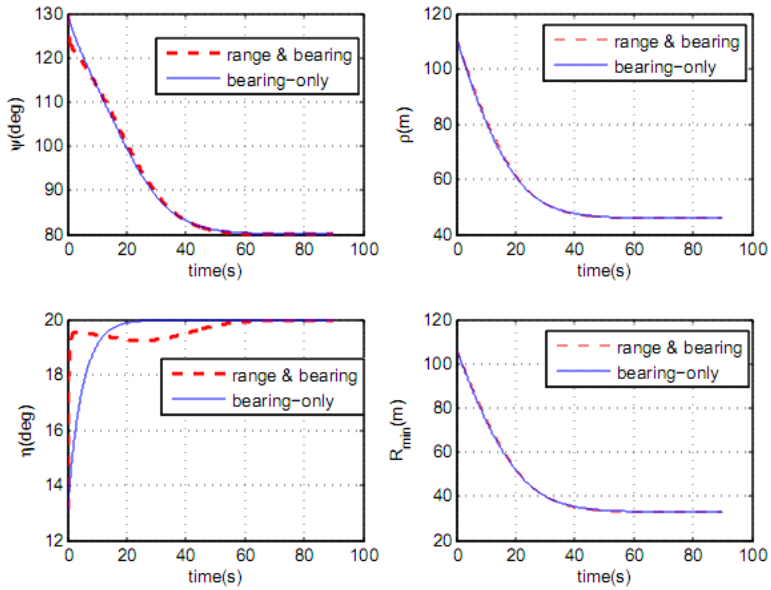
Figure 5.5: The results of a simulation are shown. The dashed line shows the path of the UAV in tracking the waypoint path and avoiding obstacles. The UAV successfully avoids the obstacles.

5.5 Conclusions

In this chapter, we develop vision based collision avoidance algorithms to avoid collision from different types of popup obstacles. The algorithm moves a obstacle in the image plane to a desired constant bearing angle. By keeping the obstacle at a constant bearing angle causes the UAV to maintain a constant distance from the obstacle. Since a camera only provides bearing measurement to obstacles, we modify the algorithm using sliding mode control such that that depth measurement is not required for the computation of the control input. The collision avoidance algorithms are fast, computationally inexpensive, and guarantee collision avoidance.

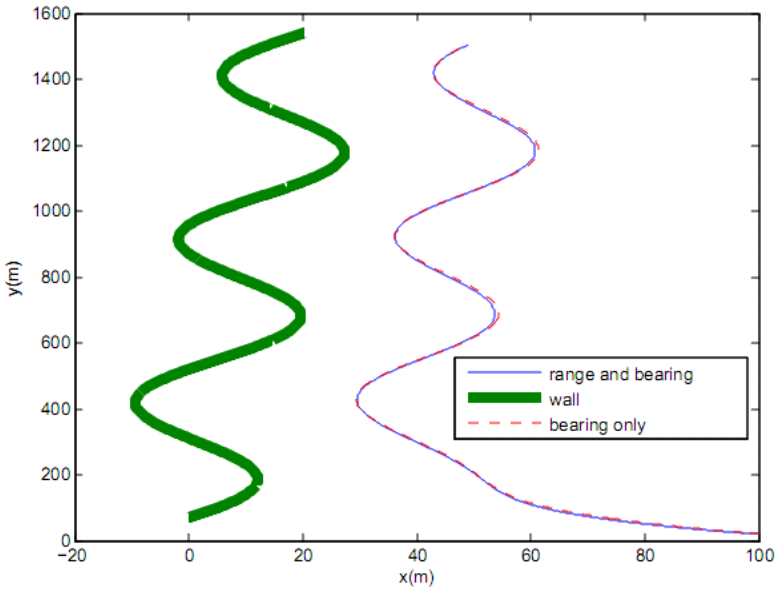


(a) UAV trajectory along a straight wall with orientation $\psi_w = 60^o$

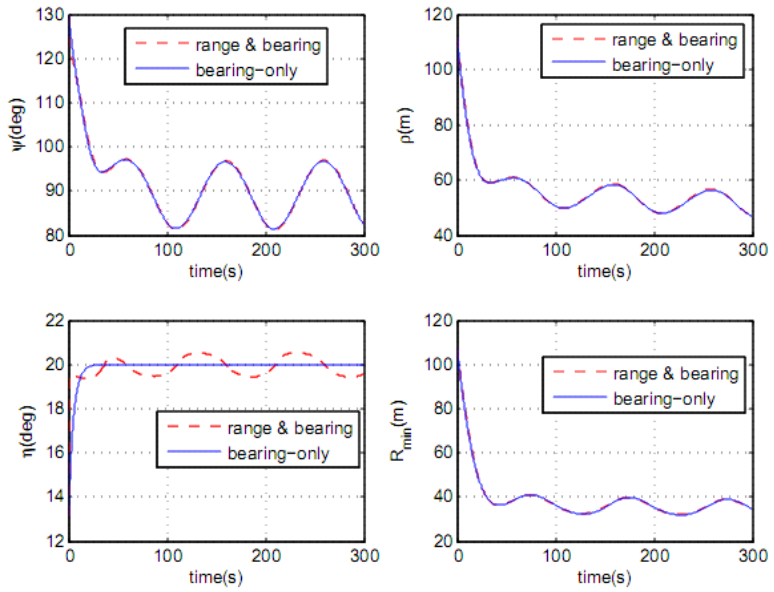


(b) UAV heading, bearing, range, and R_{min} plots

Figure 5.6: Straight wall following



(a) UAV trajectory along a sinusoidal wall



(b) UAV heading, bearing, range, and R_{min} plots

Figure 5.7: Sinusoidal wall following

CHAPTER 6. CONCLUSION AND FUTURE DIRECTIONS

6.1 Conclusion

In this dissertation, we focus on two fundamental problems related to the navigation of ground robots and unmanned aerial vehicle (UAVs): cooperative localization and path planning. The theme running through in all of the work is the use of bearing only sensors, with a focus on monocular video cameras mounted on ground robots and UAVs. The main contributions of this work are summarized as follows.

- In Chapter 2, we have derived conditions for complete observability of the bearing-only cooperative localization problem. The key element of this analysis is the Relative Position Measurement Graph (RPMG). The nodes of an RPMG represent vehicle states and the edges represent bearing measurements between nodes. We show that graph theoretic properties like the connectivity and the existence of a path between two nodes can be used to explain the observability of the system. We obtain the maximum rank of the observability matrix without global information and derive conditions under which the maximum rank can be achieved. Furthermore, we show that for the complete observability, all of the nodes in the graph must have a path to at least two different landmarks of known location. The complete observability can also be obtained without landmarks if the RPMG is connected and at least one of the robot has a sensor which can measure its global pose, for example a GPS receiver. We validate these conditions by simulation and experimental results.

  The theoretical conditions to attain complete observability in a localization system is an important step towards reliable and efficient design of localization and path planning algorithms. With such conditions, a designer does not need to resort to exhaustive simulations and/or experimentation to verify whether a given selection of a control strategy, topology of the sensor network, and sensor measurements meets the observability requirements of the

system. In turn, this leads to decreased requirements of time, cost, and effort for designing a localization algorithms.

- In Chapter 3, we have developed a cooperative approach to geo-localize a ground moving target using bearing-only localization of UAVs. We perform a graph based nonlinear observability analysis of bearing-only localization to obtain the conditions for complete observability. We have shown that the without any absolute information (landmarks or GPS) pitch and roll angles of all of the UAVs are observable. We prove that complete observability can be achieved under two conditions. First, the system is completely observable if each UAV in the group has path to two different known landmarks. Secondly, the system is completely observable if the graph is connected and at least one of the UAV measures its 3-D position and heading from a GPS receiver. We use the second condition to develop a distributed path planning algorithm for UAVs to cooperatively geo-localize the target using bearing-only measurements.

- In Chapter 4, we have developed an observability-based planning algorithm using bearing-only measurements. We perform a nonlinear observability analysis for state estimation and argue that collision avoidance and uncertainty minimization problems are complementary. Based on this analysis, we design a cost function that minimizes the estimation uncertainties while simultaneously avoiding obstacles. By minimizing the cost function, the path planning algorithm is developed directly in the local-level frame. We use a look-ahead policy to plan optimal paths over a finite time horizon. Numerical results show that the observability-based planning algorithm is successful in solving the single and multiple obstacle avoidance problems while improving the estimation accuracy.

- In Chapter 5, we have developed bearing-only collision avoidance algorithms to avoid collision from different types of popup obstacles using a sliding mode controller. The algorithm moves a obstacle in the image plane to a desired constant bearing angle. By keeping the obstacle at a constant bearing angle, a UAV maintains a constant distance from the obstacle. The collision avoidance algorithms are fast, computationally inexpensive, and guarantee collision avoidance.

6.2 Future directions

The graph based nonlinear observability analysis of cooperative localization problem in Chapter 2 has unveiled several interesting properties of RPMG which characterizes the observability of the robots in the group. However, several important questions remain open. Of particular interest is to study the effect of a robot's position in the RPMG on its state estimation quality. Also, we can characterize the quality of localization with the eigenvalues of the Laplacian of the RPMG.

In Chapter 2, we have shown that for complete observability of bearing-only localization each robot should always have a path to two different known landmarks. However, this is a very strict constraint to satisfy in real time implementation. Therefore, it is important to derive the conditions for periodic observability, where the system is observable when each robot has a path to two known landmarks over a time interval.

Furthermore, an important step forward is to design an efficient bearing-only cooperative localization system using a distributed path planning algorithm which satisfies the observability constraints and improves the localization accuracy while performing multiple tasks.

In Chapter 3, we have shown that bearing-only cooperative localization creates a larger and more flexible field-of-view, which overcomes the LOS limitation of a camera and can be used for applications such as geo-localization in urban environments where there are many occlusions and availability of GPS signals is questionable. The present distributed controller for geo-localization considers the UAVs moving in a horizontal plane, however, if the UAV's motion is planned in 3-D then it will be easier to maintain the connectivity of the RPMG and the localization accuracy in attitude and altitude can also be improved. Also the present geo-localization controller considers only non-maneuvering target. Therefore, an important future extension is to design a controller that can geo-localize a maneuvering target.

In Chapter 4, we analyzed the collision avoidance and the goal reaching behavior of the planning algorithm for cylindrical obstacles assuming the locations of obstacles are known. An important future work is to analyze the behavior of the algorithm for the obstacles with general shape taking into account the estimation uncertainties. Currently, we use uniform weights for all the obstacles, which may not result in the best possible behavior. However, the weights can be chosen based on the TTC and bearing to obstacles.

The bearing-only reactive collision avoidance algorithm for UAVs in Chapter 5, currently, takes into account one obstacle at a time. An important extension is to design a collision avoidance algorithm that can guarantee collision avoidance from multiple obstacles. Currently, the UAV is maneuvered in the horizontal plane to avoid collision from obstacles, however, maneuvering the UAV in 3-D will provide an extra degree of freedom for the collision avoidance, i.e., instead of moving around the obstacle the UAV can fly over it. Also, we consider only static obstacles, therefore, an another important future extension is to use bearing-only measurements to avoid collisions with moving objects.

REFERENCES

[1] B. J. Mouritsen, H. andFrost, "Virtual migration in tethered flying monarch butterflies reveals their orientation mechanisms," *Proceedings of the National Academy of Sciences*, vol. 99(15), pp. 10 162–10 166, July 2002. 1

[2] B. Ronacher and R. Wehner, "Desert ants cataglyphis fortis use self-induced optic flow to measure distances travelled." *Journal of Comparative Physiology A*, vol. 177(l), pp. 71–75, July 1995. 1

[3] P. Cain and S. Malwal, "Landmark use and development of navigation behaviour in the weakly electric fish gnathonemus petersii," *Journal of Experimental Biology*, vol. 205(24), pp. 3915–3923, Dec. 2002. 1

[4] B. Cochran, H. Mouritsen, and M. Wikelski, "Migrating songbirds recalibrate their magnetic compass daily from twilight cues," *Science*, vol. 304, pp. 405–408, Apr. 2004. 1

[5] J. A. Simmons, "Perception of echo phase information in bat sonar," *Science*, vol. 204(4399), pp. 1336–1338, 1979. 1

[6] A. Kelly, "Linearized error propagation in odometry," *International Journal of Robotics Research*, vol. 23(2), pp. 179–218, 2004. 1, 2

[7] N. Trawny, S. I. Roumeliotis, and G. B. Giannakis, "Cooperative multi-robot localization under communication constraints," in *Proc. IEEE Int. Conf. Robotics and Automation ICRA '09*, 2009, pp. 4394–4400. 1

[8] A. J. Davison and D. W. Murray, "Simultaneous localization and map-building using active vision," vol. 24, no. 7, pp. 865–880, 2002. 2

[9] J. Leonard, R. Rikoski, P. Newman, and B. M., "Mapping partially observable features from multiple uncertain vantage points," *The International Journal of Robotics Research*, vol. 21(10-11), pp. 943–975, 2002. 2

[10] F. Lu and E. Milios, "Globally consistent range scan alignment for environment mapping," *Autonomous Robots*, vol. 4(4), pp. 333–349, 1997. 2

[11] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann, K. Lau, C. Oakley, M. Palatucci, V. Pratt, P. Stang, S. Strohband, C. Dupont, L.-E. Jendrossek, C. Koelen, C. Markey, C. Rummel, J. van Niekerk, E. Jensen, P. Alessandrini, G. Bradski, B. Davies, S. Ettinger, A. Kaehler, A. Nefian, and P. Mahoney, "Stanley, the robot that won the darpa grand challenge," *Journal of Field Robotics*, vol. 23(9), pp. 661–692, June 2006. 2, 3

[12] W. B. S. Thrun and D. Fox., *Probabilistic robotics*. The MIT Press, Cambridge, MA, 2005. 2

[13] C. Harris and M. Stephens, "A combined corner and edge detector," in *4th Alvey Vision Conference, pages 147-151, Manchester, UK, Aug. 31 - Sep. 2*, 1998. 3

[14] D. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision,*, vol. 260(2), pp. 91–110, 2004. 3

[15] S. I. Roumeliotis and G. A. Bekey, "Segments: A layered, dual-kalman filter algorithm for indoor feature extraction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 454-461, Takamatsu, Japan, Oct. 30 - Nov. 5*, 2000. 3, 6, 11, 49

[16] P. Moutarlier and R. Chatila, "Stochastic multisensory data fusion for mobile robot location and environment modeling," in *Fifth International Symposium of Robotics Research ,pages 85-94, Tokyo, Japan, 28-31 Aug*, 1989. 3

[17] R. C. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Workshop on Uncertainty in Artificial Intelligence (AAAI), pages 1-21, Philadelphia, PA, Aug.*, 1986. 3

[18] A. I. Mourikis and S. I. Roumeliotis, "Analytical characterization of the accuracy of SLAM without absolute orientation measurements," in *Robotics: Science and Systems, pages 215-222, Philadelphia, PA, Aug*, 2006. 4, 12, 13, 52

[19] H. Durrant-Whyte and T. Bailey, "Simultaneous localisation and mapping (SLAM): Part i the essential algorithms," *Robotics and Automation Magazine*, vol. 13(2), pp. 108 – 117, 2006. 4

[20] A. Yamashita, M. Pukuchi, J. Ota, T. Arai, and H. Asama, "Motion planning for cooperative transportation of a large object by multiple mobile robots in a 3d environment," in *IEEE International Conference on Robotics and Automation, pages 3144-51, San Francisco, CA, Apr. 24-28*, 2000. 4

[21] C. Parker, H. Zhang, and C. R. Kube, "Multiple robot nest construction," in *IEEE/RSJ International Conference on Intelligent Robots and Systems, pages 2010-15, Las Vegas, NV, Oct. 27-31*, 2003. 4

[22] W. Burgard, D. Fox, M. Moors, R. Simmons, and S. Thrun, "Collaborative multirobot exploration," in *IEEE International Conference on Robotics and Automation, pages 476-481, San Francisco, CA, Apr. 24-28*, 2000. 4

[23] M. Quigley, M. A. Goodrich, S. Griffiths, A. Eldredge, and R. W. Beard, "Target acquisition, localization, and surveillance using a fixed-wing mini-UAV and gimbaled camera," in *Proc. IEEE Int. Conf. Robotics and Automation ICRA 2005*, 2005, pp. 2600–2605. 4, 47

[24] A. Ayyagari, J. P. Harrang, and S. Ray, "Airborne information and reconnaissance network," in *Proc. IEEE Military Communications Conf. MILCOM '96*, vol. 1, 1996, pp. 230–234. 4, 47

[25] R. W. Beard and T. W. McLain, "Multiple UAV cooperative search under collision avoidance and limited range communication constraints," in *Proc. 42nd IEEE Conference on Decision and Control*, vol. 1, 9–12 Dec. 2003, pp. 25–30. 4, 47

[26] D. W. Casbeer, R. W. Beard, T. W. McLain, S.-M. Li, and R. K. Mehra, "Forest fire monitoring with multiple small UAVs," in *Proc. American Control Conf the 2005*, 2005, pp. 3530–3535. 4, 47

[27] P. B. Sujit, D. Kingston, and R. Beard, "Cooperative forest fire monitoring using multiple UAVs," in *Proc. 46th IEEE Conf. Decision and Control*, 2007, pp. 4875–4880. 4, 47

[28] R. Kurazume, S. Nagata, and S. Hirose, "Cooperative positioning with multiple robots," in *IEEE International Conference in Robotics and Automation, pages 1250-1257, Los Alamitos, CA, May 8-13*, 1994. 4

[29] R. Kurazume and S. Hirose, "Study on cooperative positioning system: optimum moving strategies for CPS-III," in *Proc. IEEE International Conference on Robotics and Automation*, vol. 4, 16–20 May 1998, pp. 2896–2903. 6, 11, 49

[30] A. C. Sanderson, "A distributed algorithm for cooperative navigation among multiple mobile robots," *Advanced Robotics*, vol. 12, pp. 335–349, 1998. 6, 11, 49

[31] J. Spletzer, A. Das, R. Fierro, C. Taylor, V. Kumar, and J. Ostrowski, "Cooperative localization and control for multi-robot manipulation," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, 2001, pp. 631–636 vol.2. 6, 11, 49

[32] I. M. Rekleitis, G. Dudek, and E. E. Milios, "Multi-robot cooperative localization: a study of trade-offs between efficiency and accuracy," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and System*, vol. 3, 30 Sept.–5 Oct. 2002, pp. 2690–2695. 6, 11, 49

[33] P. Nebot, D. Gomez, and E. Cervera, "Agents for cooperative heterogeneous mobile robotics: a case study," in *Proc. IEEE International Conference on Systems, Man and Cybernetics*, vol. 1, 5–8 Oct. 2003, pp. 557–562. 6, 11, 49

[34] A. Bahr, J. J. Leonard, and M. F. Fallon, "Cooperative localization for autonomous underwater vehicles," *International Journal of Robotics Research*, vol. Volume 28 , Issue 6, pp. 714–728, 2009. 6, 11, 49

[35] A. Mourikis and S. Roumeliotis, "Analysis of positioning uncertainty in reconfigurable networks of heterogeneous mobile robots," in *Proc. IEEE International Conference on Robotics and Automation ICRA '04*, vol. 1, 2004, pp. 572–579 Vol.1. 6, 11, 49

[36] S. I. Roumeliotis and G. A. Bekey, "Distributed multirobot localization," *IEEE Transaction for Robotics and Autonomus Systems*, vol. 18, no. 5, pp. 781–795, 2002. 6, 11, 12

[37] A. Howard, M. J. Matark, and G. S. Sukhatme, "Localization for mobile robot teams using maximum likelihood estimation," in *Proc. IEEE/RSJ Int Intelligent Robots and Systems Conf*, vol. 1, 2002, pp. 434–439. 6, 11

[38] D. Fox, W. Burgard, H. Kruppa, and T. S, "A probabilistic approach to collaborative multi-robot localization," *Autonomous Robots*, vol. 8, pp. 325–344, 2000. 6, 11

[39] E. D. Nerurkar, S. I. Roumeliotis, and A. Martinelli, "Distributed maximum a posteriori estimation for multi-robot cooperative localization," in *Proc. IEEE Int. Conf. Robotics and Automation ICRA '09*, 2009, pp. 1402–1409. 6, 11

[40] B. Call, "Obstacle avoidance for unmanned air vehicle using computer vision," Master's thesis, Brigham Young University, December, 2006. 6, 73

[41] A. Curtis, "Path planning for unmanned air and ground vehicles in urban environments," Master's thesis, Brigham Young University, 2008. 6, 73

[42] E. Frazzoli, M. Dahleh, and E. Feron, "Real-time motion planning for agile autonomou vehicles," *Journal of Guidance, Control and Dynamics*, vol. 25, pp. 116–129, 2002. 6, 73

[43] Y. Watanabe, E. Johnson, and A. Calise, "Vision-based approach to obstacle avoidance," in *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, August 2005. 6, 73

[44] A. Pongpunwattana and R. Rysdyk, "Real-time planning for multiple autonomous vehicles in dynamics uncertain environments," *AIAA Journal of Aerospace Computing, Information, and Communication*, vol. 1, pp. 580–604, December 2004. 6, 73

[45] K. Sedighi, K. Ashenayi, R. Wainwright, and H. Tai, "Autonomous local path planning for a mobile robot using a genetic algorithm," *Congress on Evolutionary Computation*, vol. 2, pp. 1338–1345, June 2004. 6, 73

[46] M. Saha and J.-C. Latombe, "Finding narrow passages with probabilistic roadmaps: the small step retraction method." International Conference on Intelligent Robots and Systems, August 2005, pp. 622–627. 6, 95, 96

[47] N. M. Amato and Y. Wu, "A randomized roadmap method for path and manipulation planning," Minneapolis, MN, 1996, pp. 113–120. 6, 95

[48] L. E. Kavraki, P. Švestka, J. Latombe, and M. Overmars, "Probabilistic roadmaps for path planning in high-dimensional configuration spaces," vol. 12, no. 4, pp. 566–580, 1996. 6, 95

[49] P. Cheng, Z. Shen, and S. M. LaValle, "Using randomization to find and optimize trajectories for nonlinear systems," in *Proceedings of Annual Allerton Conference on Communications, Control, Computing*, 2000. 6, 95

[50] J. Kim and J. P. Ostrowski, "Motion planning of aerial robot using rapidly-exploring random trees with dynamic constraints," Taipei, Taiwan, September 2003, pp. 2200–2205. 6, 95

[51] Y. Song and J. W. Grizzle, "The extended kalman filter as a local asymptotic observer for descrete -time nonlinear systems," *Journal of Mathemetical Systems, Estimation, and Control*, vol. 5, pp. 59–78, 1995. 6, 16, 53, 74, 77

[52] M. A. Bicchi A., Pratichizzo D. and B. A, "On the observability of mobile vehicles localization," in *IEEE Mediterranean Conference on Control and Systems*, 1998. 6, 11

[53] G. Huang, A. Mourikis, and S. Roumeliotis, "Analysis and improvement of the consistency of extended kalman filter based SLAM," in *Proc. IEEE International Conference on Robotics and Automation ICRA 2008*, 2008, pp. 473–479. 6, 11

[54] A. Martinelli and R. Siegwart, "Observability analysis for mobile robot localization," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2005)*, 2–6 Aug. 2005, pp. 1471–1476. 6, 11, 12, 74

[55] M. Pachter, N. Ceccarelli, and P. R. Chandler, "Vision-based target geo-location using camera equipped MAVs," in *Proc. 46th IEEE Conf. Decision and Control*, 2007, pp. 2333–2338. 7, 47

[56] D. B. Barber, J. D. Redding, T. W. McLain, R. W. Beard, and C. N. Taylor, "Vision-based target geo-location using a fixed-wing miniature air vehicle," *Journal of Intelligent and Robotic Systems*, vol. vol 47, no. 4, December, 2006. 7, 47

[57] E. W. Frew, "Sensitivity of cooperative target geolocalization to orbit coordination," *Journal of Guidance, Control, and Dynamics*, vol. vol 31, no. 4, August, 2008. 7, 47, 48

[58] J. A. Ross, B. R. Geiger, G. L. Sinsley, J. F. Horn, L. N. Long, and A. F. Niessner, "Vision-based target geolocation and optimal surveillance on an unmanned aerial vehicle," in *AIAA, Guidance, Navigation, and Control Confrence, Honolulu, Hawaii*, Aug. 2008. 7, 47

[59] F. Schwarzer, M. Saha, and J.-C. Latombe, "Adaptive dynamic collision checking for single and multiple articulated robots in complex environments," *IEEE Transaction On Robotics*, vol. 21, no. 3, pp. 338–353, June 2005. 7, 96

[60] P. Vadakkepat, K. C. Tan, and W. Ming-Liang, "Evolutionary artificial potential fields and their application in real time robot path planning," *Proceedings of the 2000 Congress on Evolutionary Computation*, vol. 1, pp. 256–263, July 2000. 7, 96

[61] M. G. Park, J. H. Jeon, and M. C. Lee, "Obstacle avoidance for mobile robots using artificial potential field approach with simulated annealing," *IEEE International Symposium on Industrial Electronics*, vol. 3, pp. 1530–1535, June 2001. 7, 96

[62] R. Ghosh and C. Tomlin, "Maneuver design for multiple aircraft conflict resolution," *Proceedings of the American Control Conference*, June 2000. 7, 96

[63] J. Saunders and R. Beard, "Reactive vision based obstacle avoidance camera field of view constraints," in *Guidance Navigation and Control Confrence*, 2008. 7, 96, 100, 102

[64] R. Sharma and C. Taylor, "Cooperative navigation of MAVs in GPS denied areas," in *Proc. IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems MFI 2008*, 20–22 Aug. 2008, pp. 481–486. 8

[65] R. Sharma and C. N. Taylor, "Vision based distributed cooperative navigation for MAVs in GPS denied areas," in *AIAA Infotech@Aerospace Conference and AIAA Unmanned...Unlimited Conference, Seattle, Washington*, 6 - 9 April 2009. 8

[66] R. Sharma, R. Beard, C. N. Taylor, and S. Quebe, "Graph-based observability analysis of bearing-only cooperative localization," *IEEE Transaction On Robotics*, Accepted. 8, 49

[67] R. Sharma, R. W. Beard, and C. N. Taylor, "Bearing-only cooperative geo-localization using miniature aerial vehicles," american Control Confrence 2012 (in Review). 8

[68] H. Yu, R. Sharma, R. W. Beard, and C. N. Taylor, "Observability-based local path planning and collision avoidance for micro air vehicles using bearing-only measurements," in *Proc. American Control Conf. (ACC)*, 2011, pp. 4649–4654. 8, 66

[69] ——, "Observability-based local path planning and collision avoidance for micro air vehicles using bearing-only camera," *IEEE Transaction On Robotics*, In review. 8

[70] R. Sharma, J. Saunders, C. N. Taylor, and R. W. Beard, "Reactive collision avoidance for fixed-wing MAVs flying in urban terrain," in *AIAA Guidance, Navigation, and Control Conference, Chicago, Illinois*, 10 - 13 August 2009. 9

[71] R. Sharma, J. Saunders, and R. W. Beard, "Reactive path planning for micro air vehicles using bearing only measurements," *Journal of Intelligent and Robotic Systems*, vol. In press, June 2011. 9

[72] M. Betke and L. Gurvits, "Mobile robot localization using landmarks," *IEEE Transaction for Robotics and Autonomus Systems*, vol. 13, no. 2, pp. 251–263, 1997. 12

[73] L. D. L. Perera, A. Melkumyan, and E. Nettleton, "On the linear and nonlinear observability analysis of the SLAM problem," in *Proc. IEEE Int. Conf. Mechatronics ICM 2009*, 2009, pp. 1–6. 12

[74] K. W. Lee, W. S. Wijesoma, and I. G. Javier, "On the observability and observability analysis of SLAM," in *Proc. IEEE/RSJ Int Intelligent Robots and Systems Conf*, 2006, pp. 3569–3574. 12

[75] R. Hermann and A. Krener, "Nonlinear controllability and observability," *IEEE Transactions on Automatic Control*, vol. 22, no. 5, pp. 728–740, Oct 1977. 16, 53, 74, 77

[76] M. Vidyasagar, *Nonlinear systems analysis*. Prentice Hall, 1993. 18, 55, 125

[77] G. Strang, *Introduction to Linear Algebra, 4th Edition*. Wellesley-Cambridge Press, 2009. 22, 56

[78] *https://sites.google.com/site/rajnikantwiki/research/cooperative-localization*. 46

[79] R. Beard and T. W. McLain, *Small Unmanned Aircraft: Theory and practice*. Princeton university press, 2011. 51, 81

[80] D. W. Casbeer and R. Beard, "Distributed information filtering using consensus filters," in *Proc. ACC '09. American Control Conference*, June 10–12, 2009, pp. 1882–1887. 53

[81] W. B. Dunbar and R. M. Murray, "Receding horizon control of multi-vehicle formations: a distributed implementation," in *Proc. CDC Decision and Control 43rd IEEE Conf*, vol. 2, 2004, pp. 1995–2002. 65

[82] J. Latombe, *Robot Motion Planning*. Kluwer Academic Publishers, Boston, MA, 1991. 73

[83] H. Yu, R. Beard, and J. Byrne, "Vision-based local multi-resolution mapping and path planning for miniature air vehicles," in *Proceedings of American Control Conference*, June 10-12 2009. 74

[84] ——, "Vision-based local multi-resolution path planning and obstacle avoidance for micro air vehicles," in *Proceedings of the AIAA Guidance, Navigation and Control Conference*, August 10 2009. 74

[85] H. Yu, R. Sharma, R. Beard, and C. Taylor, "Observability-based local path planning and collision avoidance for micro air vehicles using bearing-only measurements," in *submitted to American Control Confrence*, 2010. 74

[86] I. Rhee, M. Abdel-Hafez, and J. Speyer, "Observability of an integrated GPS/INS during maneuvers," vol. 40, no. 2, pp. 526–535, April 2004. 74

[87] T. Vidal-Calleja, M. Bryson, S. Sukkarieh, A. Sanfeliu, and J. Andrade-Cetto, "On the observability of bearing-only SLAM," in *Proc. IEEE International Conference on Robotics and Automation*, 10–14 April 2007, pp. 4114–4119. 74

[88] M. Bryson and S. Sukkarieh, "Observability analysis and active control for airborne SLAM," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 1, pp. 261–280, January 2008. 74

[89] J. J. Kuffner and S. M. LaValle, "RRT-connect: An efficient approach to single-query path planning," San Francisco, CA, April 2000, pp. 995–1001. 96

[90] S. M. LaValle, "Rapidly-exploring random trees: A new tool for path planning," October 1998, tR 98-11, Computer Science Dept., Iowa State University. 96

[91] S. M. LaValle and J. J. Kuffner, "Rapidly-exploring random trees: Progress and prospects," in *Algorithmic and Computational Robotics: New Directions*, B. R. Donald, K. M. Lynch, and D. Rus, Eds. Wellesley, MA: A. K. Peters, 2001, pp. 293–308. 96

[92] ——, "Randomized kinodynamic planning," vol. 20, no. 5, pp. 378–400, May 2001. 96

[93] Z. Sun, D. Hsu, T. Jiang, H. Kurniawati, and J. H. Reif, "Narrow passage sampling for probabilistic roadmap planning," *IEEE Transactions on Robotics*, vol. 21, no. 6, pp. 1105–1115, December 2005. 96

[94] J. Kuffner, J.J. and S. LaValle, "RRT-connect: An efficient approach to single-query path planning," *IEEE International Conference on Robotics and Automation*, vol. 2, pp. 995–1001, 2000. 96

[95] J. Barraquand, B. Langlois, and J. C. Latombe, "Numerical potential field techniques for robot path planning," *IEEE Transaction On Systems, Man, and Cybernetics*, vol. 22, no. 2, pp. 224–241, March/April 1992. 96

[96] J. B. Saunders, B. Call, A. Curtis, R. W. Beard, and T. W. McLain, "Static and dynamic obstacle avoidance in miniature air vehicles," *AIAA Infotech@Aerospace*, September 2005. 96

APPENDIX A. DIFFERENTIAL GEOMETRY AND OBSERVABILITY

In this appendix we discuss basics of differential geometry, Lie derivatives, and observability of nonlinear systems.

A.1 Basics of differential geometry

The results presented in this dissertation pertain to nonlinear system described by a of equations of the form

$$\dot{x}(t) = f_0(x(t)) + \sum_{i=1}^{n} u_i(t) f_i(x(t)), \tag{A.1}$$

where $x(t) \in \mathbb{R}^n$, and f_i, $\forall\ \in \{0, 1, \cdots, n\}$ are vector fields on R^n. Two features of this system can be observed. First, the system is time invariant, in that there is no explicit dependence on time. Therefore we omit the time t term. Second, the system above is linear in the control.

The output of the nonlinear system is given in the form

$$y = h(x). \tag{A.2}$$

In this section, we present a well-known theorem from advanced calculus, namely the inverse function theorem, the notions of vector field, a form, and Lie derivative.

Suppose $f : \mathbb{R}^n \to \mathbb{R}^n$, and suppose each component of f is continuously differentiable with respect to each of its arguments. In other words, suppose that f is C^1. Then the $n \times n$ matrix whose ij_{th} entry is $\partial f_i / \partial x_j$ is called the Jacobian matrix of and is denoted by $\partial f / \partial x$. We say that f is smooth if every component of f has continuous derivatives of all orders with respect to all combinations of its arguments. Suppose VS_1, VS_2 are open subsets of R^n and that $f : VS_1 \to VS_2$ is C^1. Then we say that f is diffeomorphism of VS_1 onto VS_2 if (1) $f(VS_1) = VS_2$, (2)f is one-to-

one, and (3) the inverse function $f^{-1}: VS_2 \to VS_1$ is also C^1. f is called a smooth diffeomorphism if both f and f^{-1} are smooth functions.

Next theorem for inverse function can be found in Royden(1963)

Theorem A.1.1 (Inverse Function Theorem) *Suppose $f: \mathbb{R}^n \to \mathbb{R}^n$ is C^1 at $x_0 \in \mathbb{R}^n$, and let $y_0 = f(x_0)$. Suppose $[\partial f/\partial x]_{x=x_0}$ is non singular. Then there exists open sets $VS_1 \subseteq \mathbb{R}^n$ containing x_0 and $VS_2 \subseteq \mathbb{R}^n$ containing y_0 such that f is a diffeomorphism of VS_1 onto VS_1. If, in addition, f is also smooth, i.e., f is a smooth diffeomorphism.*

Throughout the remainder of this section, X denotes an open subset of $\mathbb{R}^n$, where n is a fixed integer.

Definition A.1.1 *A vector field on X is a smooth functions mapping X into $\mathbb{R}^n$. The set of all vector fields on X is denoted by $VS_1(X)$. The set of all smooth real-valued function mapping X into R is denoted by $S(X)$.*

For each $a, b \in S(X),\ f,\ g \in VS_1(X)$, we have

$$\begin{aligned} a(f+g) &= af + ag, \\ (a+b)f &= af + bf, \\ (a \cdot b)f &= a \cdot (bf). \end{aligned}$$

Definition A.1.2 *A form X is a smooth function mapping X into $(R^n)^*$, which is the set of $l \times n$ row vectors. The set of all forms on X is denoted by $F(X)$.*

Suppose $x_0 \in X$ is given. A curve in X passing through x_0 is a smooth function c mapping some open interval $(-\alpha, \beta)$ containing 0 into X, such that $c(0) = x_0$. Suppose f is a vector field on X and that $x_0 \in X$ is given. Then we know that there exists a unique solution of the differential equation

$$\frac{d}{dt}x(t) = f(x(t),\ x(0) = x_0, \tag{A.3}$$

for sufficiently small values of t. Viewed as a function of t, the solution $x(\cdot)$ defines a curve passing through x_0; it is called integral curve of f passing through x_0.

Suppose $h \in S(X)$, i.e., h is a smooth real-valued function on X. Then its gradient, denoted by ∇h, is defined as the row vector $[\partial h/\partial x_1, \cdots, \partial h/\partial x_n]$. Note that ∇h is a form on X. Now suppose $f \in VS_1(X)$. Then the map

$$x \mapsto \nabla h \cdot f(x) : X \to \mathbb{R}, \tag{A.4}$$

is smooth; it is called the Lie derivative of the function h with respect to the vector field f, and is denoted by $L_f h$. Note that $L_f h \in S(X)$. The Lie derivative $L_f h$ can be interpreted as the derivative of a scalar along integral curves of the vector field f.

A.2 Observability

Definition A.2.1 *Consider a system described described by (A.1) and (A.2). Two states x_o and x_1 are said to be distinguishable if there exists an input function $u(.)$ such that*

$$y(.,x_0,u) \neq y(.,x_1,u), \tag{A.5}$$

where $y(.,x_i,u),\ i = 1,2$ is the output function corresponding to the input function $u(.)$ and the initial condition $x(0) = x_i$. The system is said to be locally observable at $x_0 \in X$ if there exists a neighborhood Xn of x_0 such that every $x \in Xn$ other than x_0 is distinguishable from x_0. Finally, the system is said to be locally observable if it is locally observable at each $x_0 \in X$.

Before we find the sufficient conditions for the observability of a nonlinear system described by (A.1) and (A.2), we find the sufficient condition for the observability of a linear system given below.

$$\dot{x}(t) = Ax(t) + Bu(t), \tag{A.6}$$

$$y(t) = Cx(t). \tag{A.7}$$

where $A \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, m is the number of inputs. Suppose we know u and can measure y; assume for the sake of the convenience that $u(t)$ is a smooth function of t, i.e., has derivatives of all orders. Then successive differentiation of the output equation (A.7) gives

$$\begin{aligned} y(t) &= Cx(t), \\ \dot{y}(t) &= C\dot{x}(t) = CAx(t) + CBu(t), \\ \ddot{y}(t) &= CA^2x(t) + CABu(t) + CB\dot{u}(t). \end{aligned}$$

Hence, by successively differentiating y, we can infer the quantities

$$Cx(t),\ CAx(t),\ CA^2x(t), \cdots,\ CA^{n-1}x(t). \tag{A.8}$$

after subtracting the known quantities $CBu(t)$, $CABu(t)$, and $CB\dot{(u)}(t)$, etc, and we can write the observability matrix for a linear system (A.6) as

$$O_{lin} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{bmatrix}. \tag{A.9}$$

If O_{lin} has rank n, then it is possible to determine $x(t)$ uniquely. Of course there is no need to go beyond A^{n-1} because of the Cayley-Hamilton theorem.

For nonlinear systems the idea is pretty much the same. Let m denote the number of outputs, and let y_p, $h_p(x)$ denote respectively the p^{th} components of y and $h(x)$. Then

$$\begin{aligned} y_p &= h_p(x), \\ \dot{y}_p &= \nabla h_p \dot{x} = \nabla h^p f(x) = (L_f h_p)(x) = (L_{f_0} h_p)(x) + \sum_{i=1}^{n} u_i (L_{f_i} h_p)(x), \end{aligned}$$

where the Lie derivative $L_f h_p$, and $L_{f_i} h_p$ are defined in accordance with (A.4), and the explicit dependence on t is not displayed in the interest of clarity. Differentiating one more time gives

$$\ddot{y}_p = (L_{f_0}^2 h_p)(x) + \sum_{i=1}^{n} u_i (L_{f_i} L_{f_0} h_p)(x) + \sum_{i=1}^{n} \dot{u}_i (L_{f_0} h_p)(x),$$
$$\sum_{i=1}^{n} u_i (L_{f_0} L_{f_i} h_p)(x) + \sum_{j=1}^{n} \sum_{i=1}^{n} u_i u_j (L_{f_j} L_{f_i} h_p)(x).$$

Expressions for higher derivatives of y_p get progressively nastier, but pattern is clear enough. The quantity $y_p^{(d)}$ is a linear combination of terms of the form $(L_{z_s} L_{z_{s-1}} \cdots L_{z_m} h_p)(x)$, where $1 \leq s \leq d$, $d > 0$, and each of the vector fields $z_1, \cdots, z_s$ is from the set $\{f_0, f_1, \cdots, f_n\}$.

In view of the foregoing observation, next theorem below seems plausible which is proved in [76].

Theorem A.2.1 *Consider the system described by (A.1) and (A.2), and suppose $x_0 \in X$ is given. Consider the forms*

$$\nabla (L_{z_s} L_{z_{s-1}} \cdots L_{z_m} h_p)(x_0),\ s \geq 0,\ z_i \in \{f_0, f_1, \cdots, f_n\}, \tag{A.10}$$

evaluated at x_0. Suppose there are n linearly independent row vectors in this set. Then the system is locally observable around x_0.