Precision Navigation using Two-Way Ranging: Bounds and Performance

by

Samuel Welker

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Masters of Science

Approved March 2022 by the
Graduate Supervisory Committee:

Daniel Bliss, Chair
Andrew Herschfelt
Konstantinos Tsakalis

ARIZONA STATE UNIVERSITY

May 2022

ABSTRACT

Localization tasks using two-way ranging (TWR) are making headway in modern day navigation applications as an alternative to legacy global navigation satellite systems (GNSS) such as GPS. There is not currently literature that provides a closed-form expression for estimation performance bounds on position and attitude when a TWR system is employed. A Cramer-Rao Lower Bounds (CRLB) is derived for position and orientation estimation using both 2-D and 3-D geometries. A literature review is performed to give background and detail on the tools needed for a thorough analysis of this problem. Popular Least Squares techniques and solutions to Wahba's problem are compared to the derived bounds as proof of correctness using Monte Carlo simulations. A brief exploration on estimation performance using an Extended Kalman Filter for non-stationary users is also looked at as an introduction to future extensions to this work. The literature Applications like the CHP2 system are discussed as well to show how secure, inexpensive and robust implementation of TWR is highly feasible.

TABLE OF CONTENTS

Page

LIST OF TABLES iv

LIST OF FIGURES v

CHAPTER

1 INTRODUCTION 1

1.1 Background 3

1.1.1 Performance Bounds 3

1.1.2 Time-of-Flight Estimation 4

1.1.3 Position and Attitude Estimation 4

1.1.4 Tracking Filters 5

2 LITERATURE REVIEW 6

2.1 Introduction to Position and Attitude Estimation 6

2.1.1 Bounds 7

2.1.2 Estimators 9

2.1.3 Two-Way Ranging 12

2.1.4 Reference Frames and Attitude Representations 14

2.1.5 Geometric Dilution of Precision 18

2.1.6 Position and Orientation Estimation 21

2.1.7 Tracking 24

2.1.8 Applications 27

2.2 Commercial and Defense Applications 27

2.3 CHP2 Implementation 28

3 PROBLEM OVERVIEW 29

4 PERFORMANCE BOUNDS FOR POSITION AND ATTITUDE ESTIMATION 31

CHAPTER Page

4.1 Cramer-Rao Lower Bound on Position Estimation 31
4.1.1 2-D Position Bound 31
4.1.2 3-D Position Bound 34
4.2 Cramer-Rao Lower Bound on Attitude Estimation 38
4.2.1 Attitude Model 38
4.2.2 Attitude in 2D 39
4.2.3 Attitude in 3D 40
5 POSITION AND ORIENTATION ESTIMATION 44
5.1 Theory 44
5.2 Techniques 47
5.2.1 Position Estimation Techniques 47
5.2.2 Attitude Estimation Techniques 49
5.3 Simulations 57
6 POSITION AND ATTITUDE ESTIMATION OF A NON-STATIONARY TARGET 60
6.1 Theory 60
6.2 Techniques 62
6.3 Simulations 64
7 APPLICATION: CHP2 67
8 FUTURE WORK AND EXTENSIONS 70
9 CONCLUSION 72
REFERENCES 73

LIST OF TABLES

Table Page

2.1 Table Defining Numerical Values of GDOP 21

LIST OF FIGURES

Figure		Page
1.1	Image Showing How 3 Different GPS Satellites Can Be Used to Pinpoint Location As the Single Intersection of 3 Circles	2
2.1	Simplified Depiction of Two-Way Ranging Problem	6
2.2	Visualization of Statistical Measurements: Accuracy, Consistency and Bias	11
2.3	Example of Body Frame for an Airplane Adopted From [32]	15
2.4	Visualization of GDOP: A.) Depicts Two Nodes With Target Location Marked by the Intersection of the Red and Blue Circles, B Illustrates Quality Gdop Due to Adequate Relative Node Placement, C Shows Poor Gdop Where Relative Node Locations Increases the Measurement Error. Adapted From [37]	19
2.5	3-D Geometry for Performing Positioning With Reference Nodes $\overline{g}_i$ and Target Nodes $\overline{h}_i$ [31], [27]	21
2.6	Block Diagram of State Space Representation	26
4.1	2-D System Geometry for Position Estimation [31], [27]	32
4.2	CRLB for Position With 3-d Geometry Vs. Distance Estimate Variance	36
4.3	CRLB for Position Measured Against Distance Variation and GDOP	37
4.4	GDOP Cuts of CRLB for Position Measured Against Distance Variation	38
4.5	3-D CRLB for Attitude Over Varying Position Estimate Uncertainty	41
4.6	3-D CRLB for Attitude Over Varying Concentration Factor κ	42
4.7	Relationship Between Concentration Factor and Position Estimate Uncertainty	43
5.1	Performance of Position Estimators Vs. Their CRLB	58
5.2	Performance of Attitude Estimators Vs. Their CRLB	59

Figure Page

5.3 Performance of Attitude Estimators Vs. Their CRLB Plotted Over κ . . 59

6.1 Typical Flight Geometry for the Non-stationary PnO Estimation Problem [27] 61

6.2 Example Flight Path Showing the EKF's Ability to Estimate a Target . 64

6.3 Full State Position Vector Vs. Truth 65

6.4 Full State Attitude Vector Vs. Truth 66

7.1 CHP2 Waveform for a 4-antenna User. The Colored Navigation Sequences Are Each Associated With Their Own Antenna to Provide Spatial Distinction. The Communications Payload Drives the Tof Algorithm and Pre/Post Ambles Are Used for Frequency and Time Synchronization [7] 67

7.2 Experimental Results With Setup Similar to Figure (2.5). 4 Ground Reference Antennas Estimate the Range of a Target. 68

Chapter 1

INTRODUCTION

Highly accurate real time estimation of position and orientation (PnO) is a primary feature of interest in present-day navigation systems. Applications ranging from military aircraft guidance, commercial and military drone operation, smart agriculture and surveillance are just a few of the many systems where users desire high precision positioning capabilities. Two-way ranging (TWR) systems are setup to have 2 (or more) users who exchange a signal containing their own local clock information back and forth. A user can then post-process this waveform to extract the senders time and compare it to their local clock time. This allows a user to synchronize it's own clocks with the senders clocks and therefore determine how long it took for the waveform to travel, otherwise known as time of flight (ToF). Additionally, by multiplying the computed ToF by the speed the waveform traveled (speed of light for most applications), the distance between the two users can be resolved.

TWR systems function similarly to GPS in that they use time information to calculate relative location. In general, the family of global satellite navigation systems (GNSS) that includes the USA's NAVSTAR GPS and Europe's Galileo use ToF information in conjunction with a technique called trilateration to obtain geodetic coordinates. GPS for example relies on a constellation of 31 satellites to obtain 3 ToF measurements necessary to calculate 3-D position. An image showing the technique of trilateration is shown in figure (1.1). Using three different satellites or nodes, a single point at the center of all three circles resolves any ambiguities in position estimates.

Other algorithms are needed to find the orientation of one reference node with

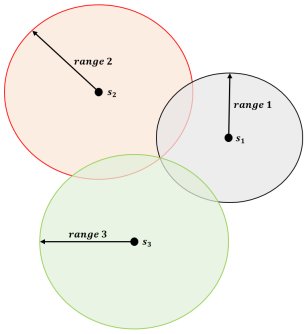

Figure 1.1: Image Showing How 3 Different GPS Satellites Can Be Used to Pinpoint Location As the Single Intersection of 3 Circles

respect to a target, which can be done by exploiting the geometry of multiple position estimates. Applications that implement navigators such as drone swarms, aircraft or automobiles will require real-time updates of position and orientation (PnO) to be available on demand. Tracking filters like the Extended Kalman Filter (EKF) or Particle Filter (PF) can be used to create physical predictions of the vehicles movement and can be easily implemented in real-time embedded software.

Although GNSS systems are very popular and provide reliable estimates, many users are looking for localization systems that have improved security, greater precision, simplified infrastructure and larger throughput. Due to the wide use of GPS and GNSS style systems, many "bad actors" have discovered methods to spoof and jam ranging receivers because the GPS waveform definition and its related post-processing algorithms are publicly available. Secondly, the GPS architecture relies on a constellation of satellites which introduces challenges to reconfigurability and maintenance. For custom applications that require a higher level of precision and specific size, weight and power requirements, a de-centralized architecture is preferred.

While there are apparent structural and security benefits to using a custom TWR system, one natural question that arises is how well can a TWR system perform?

Navigation engineers seek to gain an understanding of performance in a general way and one method for this is to derive performance bounds. In other words, given a TWR style architecture, how close can one come to finding the exact location and orientation of a target user? The purpose of this thesis is to try and answer this question through use of the Cramer-Rao Lower Bound derived for a stationary environment. On top of that, the work is extended to analyzing PnO estimation performance in a dynamic flight environment using a tracking filter with different rotational representations.

1.1 Background

Realizing a performance bounds for position and attitude determination using a TWR architecture requires an understanding of several different topics relevant to physics, statistics, signal processing and state estimation. The work discussed in [8, 25, 9] analyzes algorithms and implementations for a real-time TWR system. A network-timing protocol (NTP) style algorithm is used in conjunction with a tracking filter in [31, 30, 27] to show an example of how accurate ToF estimates can be made. In [27], Srinivas shows how position and attitude can be tracked using an EKF for various flight paths. A brief background on these topics will be introduced below and expanded upon in Chapter 2.

1.1.1 Performance Bounds

Bounds on performance are primarily used in the beginning stages of defining system requirements. It's important to find a lower-bound on performance metrics like the accuracy, speed and power of an algorithm to gain insight into what an architecture is capable of. Famous bounds like the Cramer-Rao Bound are popular

for initial analysis because they don't require algorithm details to be fully fleshed out. Tighter bounds have also appeared in the literature [20] such as the Weiss-Weinstein and Bobrovsky-Zakai bounds that impart more strict conditions in their calculations.

1.1.2 Time-of-Flight Estimation

ToF estimation has been another well-studied topic and found footing in many engineering applications like 2D/3D mapping, autonomous vehicles, robotics, object scanning, surveillance and more. As mentioned before, ToF estimation is a simple technique that utilizes separated clocks to capture the time it takes for light to travel between two users. Although the calculations are simple, the implementation of ToF measurement is complicated in the fact that real physics extend beyond Newtons Equations with factors like electronic noise and multipath that can delay and distort the signal. Additionally, the user clocks are not typically synchronized, so the offset between them must also be modeled. To combat these factors, a lot of research has been done to alleviate these errors such as the work in [8, 15, 2, 1].

1.1.3 Position and Attitude Estimation

GPS and an assortment of other PnO estimation algorithms use a combination of spatially diverse distance (and hence ToF) measurements from multiple nodes to extrapolate position and bearing. Least squares (LS) techniques using a known reference location and computed pseudo-ranges is the most common way of estimating position and used in GPS along with other time-based ranging systems. Extensions to LS have also been studied to improve the algorithms performance and include iteratively recursive least squares (IRLS) and nonlinear least squares (NLLS) [39].

Attitude estimation algorithms follow directly from positioning and are usually formulated through the solution to min/max optimization problems like Wahba [34, 17].

The idea of reference frames becomes important here due to the nature of orientation being a quantity that is *relative to* a known reference. Common measurement reference frames are the body, north-east-down (NED), wander-azimuth and geodetic (latitude, longitude and altitude) coordinates. The choice of frame-rotation representation is also important due to representations breaking down due to singularities. Euler angles are one example of this which suffer from "gimbal lock" which gives way to designers using quaternions or Rodrigues parameters as a substitute.

1.1.4 Tracking Filters

Sensors are commonly used for capturing real-time flight related information such as GPS signals, linear acceleration or angular velocity. Imperfections in the sensors, channel effects and nonlinear flight dynamics cause noisy measurements that make position and attitude estimates vary wildly. Tracking filters like the Kalman Filter and other variations [12, 10, 11, 38] improve these estimates by recursively finding the measurement, state and processes noise covariance matrices which are incorporated into the state variable prediction. Integrating the covariance matrices and sensor measurements into the prediction is equivalent to minimizing the mean-square estimation error in real-time.

Chapter 2

LITERATURE REVIEW

2.1 Introduction to Position and Attitude Estimation

Bounding and measuring performance on position and attitude estimation is crucial task in the development of many systems used in commercial flight, farming, land surveying and military defense. Although GPS is the most commonly used framework for performing this task, other two-way ranging architectures are becoming popularized to meet the security and performance needs of custom applications. In short, if the clocks on a two-way communications link can be synchronized, accurate measurements of the waveform propagation time can be made to extract the distance between the two nodes. A simplified picture describing the physics of this situation is shown below:

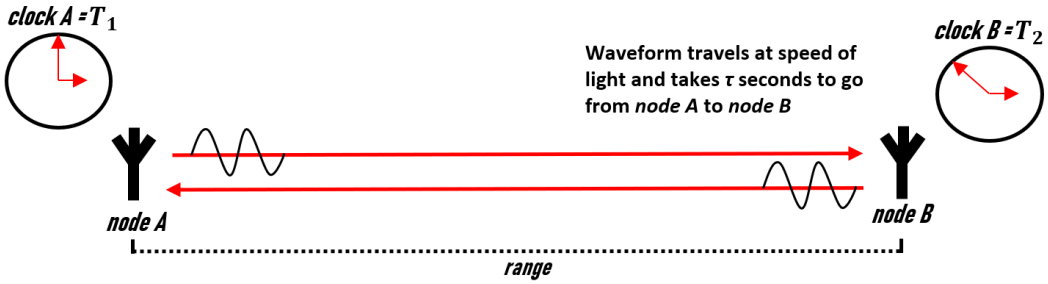

Figure 2.1: Simplified Depiction of Two-Way Ranging Problem

Deriving the bound on estimation performance for this situation is possible by forming a mathematical mode of the physical geometry in conjunction with the Cramer-Rao Lower Bound [19]. Once a bound has been found, real time estimation performance can be analyzed and simulated using known estimation techniques.

There are already many well known two-way ranging (TWR) algorithms that have been developed and tested that achieve high precision propagation delay estimates [31, 8, 15, 2]. Additionally, position and attitude estimation algorithms have been developed that are typically optimization solutions formed from functions of the prop-delay and a-priori node location knowledge [25, 39, 34, 17, 16, 23, 4, 22]. Attitude and position reference frames will be explored in detail and tracking performance will also be simulated using various tracking techniques to show a first-cut of real-time performance.

Throughout this report, the joint positioning-communications system called CHP2 from [9, 24] will be leveraged as an example platform that implements TWR. This system demonstrates a good example by which a joint waveform is designed for communications and positioning tasks. The TWR algorithm deployed here operates with only 10 MHz of spectral bandwidth and achieves less than 5 cm of ranging uncertainty.

2.1.1 Bounds

Bounds constantly appear in mathematics, and are defined for functions and more generally sets of numbers. For functions, an upper bound is a real number M such that a function $|f(x)| \leq M \ \forall_x$. Similarly, a lower bound is a real number M such that $|f(x)| \geq M \ \forall_x$. Generally speaking, if one has a set of numbers S, a bound is a constant C such that all numbers in S are either less than (upper bound) or greater than (lower bound) C. Formally, the infimum and supremum operators are used for defining bounds on sets. The infimum operator denoted $\inf_{x \in S}$ is called the greatest lower bound on a set S, and the supremum operator denoted by $\sup_{x \in S}$ is the least upper bound.

Bounds are important for analytic purposes because they allow for approximation, which means that the analysis doesn't require an exact solution to the problem.

Instead, a bound makes a universal statement about the maximum and minimum limits that a function or set of functions can take on.

One important bound for the purpose of this thesis is the Cramer-Rao Lower Bound (CRLB) which is rooted in probability and estimation theory. The CRLB is defined as a lower bound for the minimum variance, unbiased estimator. This means that instead of having to find the exact expression for an estimator, one can mathematically calculate the lower bound for an unbiased estimator and gain intuition into performance limitations. Having a closed-form expression for the CRLB is also beneficial as it will often provide insight into the individual physical parameters that control an estimators performance.

The most common way of computing the CRLB is by calculating the inverse of the Fisher Information usually denoted as $I(\theta)$ in the literature. Intuitively speaking, $I(\theta)$ measures the average value of the curvature of the log-likelihood function. Given a random variable X with density $p(x;\theta)$, the log-likelihood is simply the logarithm of the density function $ln\ p(x;\theta)$. Increasing curvature on the density function correlates to stronger likelihoods. For completeness, $I(\theta)$ is given as

$$I(\theta) = -E\left\{ \frac{\partial^2\ ln\ p(x;\theta)}{\partial\theta^2} \right\} \tag{2.1}$$

And the CRLB is given as

$$\sigma_\theta^2\ \geq\ \frac{1}{I(\theta)} \tag{2.2}$$

A lot of work has already been done to derive bounds for the relevant topics in this report. In [36] and [35], Weiss and Weinstein use the Ziv-Zakai lower bound

(ZZLB) as another tool to analyze the error performance for time delay estimation in both narrow and wide-band systems. In [41], Zeira looks at the Barankin Bounds for time delay estimation of narrow band signals in the low SNR regime. Bounds such as the ZZLB are sometimes more informative than the CRLB for low SNR regimes but more computationally difficult. Further details on these bounds are outside the scope of this report and won't be discussed in detail here.

2.1.2 Estimators

Given a set of measured data, is it possible to create an approximation for a desired feature of the data itself? For example, let's say that I've received a waveform $s(t)$ that I know is sinusoidal in nature. Can I use only the measured samples of this signal to determine it's amplitude, carrier frequency and relative phase? This is the key question that estimation theory attempts to answer. Estimators are mathematical guesstimates of key data metrics that are generated by functions who only depend on the data samples themselves.

The two most common types of estimators are point and interval estimates. A point estimate is a single quantity numerical representation of the thing being estimated. In comparison, interval estimates, also called confidence intervals, are a range of values that likely contain the true value. My thesis will focus mostly on measuring point estimates and comparing them to their relevant CRLBs.

There are three key metrics when it comes to studying the performance of different estimators; bias, consistency and accuracy. In reality, the accuracy is a derived metric based on the bias and consistency, but still important to separate from the other two for the purpose of analysis.

The bias of an estimate is a deterministic offset from it's true value. This can also be thought of as how the expected value of an estimator deviates from it's true value

[13]. Mathematically, for a given estimate $\hat{\theta}$, the bias is written as $Bias(\hat{\theta}) = E[\hat{\theta}] - \theta$, with $E[\cdot]$ being the expectation operator. Obtaining an unbiased estimator is a highly desired quality in estimation theory as it allows for comparison to the CRLB.

Another metric called the consistency of an estimate measures variation in the estimate with respect to truth [13]. The consistency is a random phenomenon and shows up in the limit of running larger and larger numbers of random trials. Given a set of N random variables (RV's) $\{\theta_1, \theta_2, ...\theta_N\}$, the consistency is a measure of how closely the statistics of a RV match the true value when $N \to \infty$. Another way this can be written is

$$var(\hat{\theta}) = E[(\hat{\theta} - \mu_\theta)^2] \tag{2.3}$$

where $Pr(\cdot)$ is the probability operator, and ϵ is a small constant. Both bias and consistency can be used to form another metric of interest, accuracy. Another common name for accuracy is the Mean Squared Error (MSE) and is defined as

$$E[|\hat{\theta} - \theta|^2] = var(\hat{\theta}) + bias^2(\hat{\theta}) \tag{2.4}$$

Equation (2.4) is an important statement in how the consistency (variance) and bias of an estimator contribute independently to the MSE. An ideal estimator has both bias and variance equal to zero, however there is usually a trade-off between bias and variance when it comes to development. Figure (2.2) below provides a visualization to better understand bias, consistency and accuracy.

Typical approaches for forming estimates include the Method of Moments, Maximum Likelihood Estimation (MLE), Bayesian Estimation and Least Squares [33].

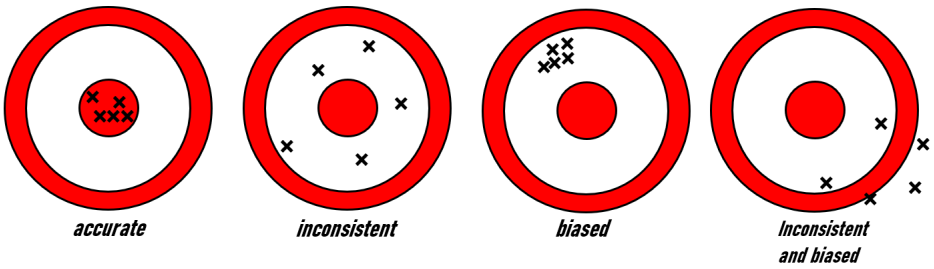

Figure 2.2: Visualization of Statistical Measurements: Accuracy, Consistency and Bias

Method of moments focuses on looking at first, second and higher order statistics formed using windows of incoming data samples. This is typically implemented in the form of sample mean and variance estimators. MLE focuses on optimizing over the likelihood function of a RV which guarantees the properties of asymptotically unbiased and efficiency [13], or that it's variance is equivalent to the CRLB. The downside of this method is that the likelihood function requires prior knowledge of a RV's Probability Mass Function which isn't always straight forward to obtain.

Least squares estimation is a well known method that assumes a linear model formed by minimizing the squared error of the measurements and model inputs. The well known solution of least squares coefficients is given by:

$$\overline{\hat{\theta}} = (X^T X)^{-1} X^T \, \overline{y} \tag{2.5}$$

Where in the above, $\mathbf{y}$ is a vector of observed samples and X is the known system input.

Finally, the Bayes Estimation leverages the Bayes Rule with prior-probabilities. Bayes Theorem is given by:

$$p(\theta|x) = \frac{p(x|\theta)\ p(\theta)}{p(x)} \tag{2.6}$$

Where $p(\theta|x)$ and $p(x|\theta)$ are the conditional densities. In order to use Bayes Estimation, the prior densities are required which are often difficult to obtain. A common implementation of the Bayes Estimator is by empirical construction where distributions for the priors/posterior are assumed and moments of these distribution are derived as new measurements are made. Once the posterior $p(\theta|x)$ is calculated, a loss function L is minimized to find the value of the estimate:

$$\hat{\theta} = \min_{\theta} \int L(\hat{\theta} - \theta) p(\theta|x) d\theta \tag{2.7}$$

2.1.3 Two-Way Ranging

Two-Way Ranging (TWR) is a method by which two or more communication links use a waveform to measure the propagation delay between each other. Assuming the signal travels at the speed of light, the two-way propagation delay is multiplied by the speed of light and divided by two to calculate the distance. This can be written as:

$$r = \frac{c\tau}{2} \tag{2.8}$$

As simple as this sounds, many problems exist in forming this measurement such

as asynchronous clocks between radios that also drift over time and effects from the channel. These impairments make it so that it takes multiple frames of data to synchronize system clocks and accurately measure the range.

Many two-way ranging estimators exist and are derived with varying levels of system dynamics. To keep things simple, I will focus mainly on a two node systems in the survey of estimators that follows. One typical TWR model from [8] is:

$$t_{B,Rx}^{(n-1)} - t_{A,Tx}^{(n-1)} = \tau^{(n-1)} - T^{(n-1)} \tag{2.9}$$

Where $t_{B,Rx}^{n}$ is the clock time at node B on the n^{th} frame, $t_{A,Tx}^{n}$ is the transmitted clock time on node A, τ is the one-way propagation delay and T is the clock offset between nodes A and B. The minus sign on the $T^{(n)}$ term is a chosen orientation that assumes clock A is ahead of clock B. Higher order models include other dynamics such as clock drift $\dot{T}$, frequency drift $\ddot{T}$, radial velocity $\dot{\tau}$ and radial acceleration $\ddot{\tau}$.

Several methods have been developed to both synchronize the clocks between nodes by estimating the clock offset, and obtain accurate time-of-flight estimates. Srinivas [26, 29, 28] address this problem by creating a first and second order model for the propagation delay and time offsets, and then formulating estimates for $\hat{\tau}$ and $\hat{T}$ on the current frame n as solutions to a system of linear equations. This method requires that at least 3 frames worth of data has been processed.

Another solution proposed by Bidigare [2] models the synchronization and TWR problem using a state space representation and then tracks estimates of prop-delay, radial velocity, clock offset and frequency drift $\hat{\tau}$, $\hat{\dot{\tau}}$, $\hat{T}$ and $\hat{\dot{T}}$ respectively using an Extended Kalman Filter (EKF). The Wiess-Weinstein bound [20] is also used as a way of initializing the measurement covariance matrix in the EKF formulation.

Bidigare [1] also shows a maximum-likelihood (ML) delay estimator technique using optimization and root finding. Here, a three-step process by which 2 cost functions are minimized and newtons method is used to iterate over ambiguity and obtain a high-precision phase accurate delay estimate that approaches the CRLB.

A formulation created by Lee [15] proposes a model that includes multi-path effects and $\hat{\tau}$ is estimated by solving the nonlinear optimization problem:

$$\hat{\tau} = \min_{\tau} \left[\min_{p_d, M, \underline{\alpha}, \underline{\beta}} \left|\left| \underline{r} - p_d \underline{s}_{\tau} - \sum_{k=1}^{M} \alpha_k \underline{s}_{\beta_k} \right|\right|^2 \right] \tag{2.10}$$

Where $\alpha_k' s$ are the signal strength of the multi-path components that arrived before the main component, $\beta_k' s$ are the time delays associated with those components, p_d is the signal strength of the direct path component, M is the number of multipath signal components that arrived earlier than the direct signal and $\underline{r}$ is the measured signal at the receiver. The form of $\underline{r}$ is:

$$\underline{r} = a_d s(t - \tau_d) + \sum_{n=1}^{L} a_n s(t - \tau_n) + n_m(t) \tag{2.11}$$

In this formulation, p_d, M, α_k and β_k are treated as nuisance parameters and an iterative nonlinear-programming method is used to solve the optimization problem for the desired $\hat{\tau}$.

2.1.4 Reference Frames and Attitude Representations

Before talking about position and attitude estimation, I want to spend time introducing the concept of reference frames and attitude representations, as that will better aid in the formulation of the estimator discussion. A reference frame can be thought of as a coordinate system that has a known reference. For the sake of sim-

plicity, I will only consider inertial reference frames here which are frames that don't experience acceleration. This will need to be extended to non-inertial reference frames in future work for more accurate physical modeling.

Typically when talking about position and attitude in space, this is the position or orientation of an object *with respect to* another object. For example, the WGS84 earth model [21] defines the origin of it's reference frame at the center of the earth, and has corresponding equations that model the earths shape and distances from core to the surface. Another common frame is the body frame that attaches a coordinate system to a central location on the object of interest (such as the center of gravity), and consecutive position and orientations are measured with respect to the objects initial conditions. Below is an example image that shows the body frame and how it could be used.

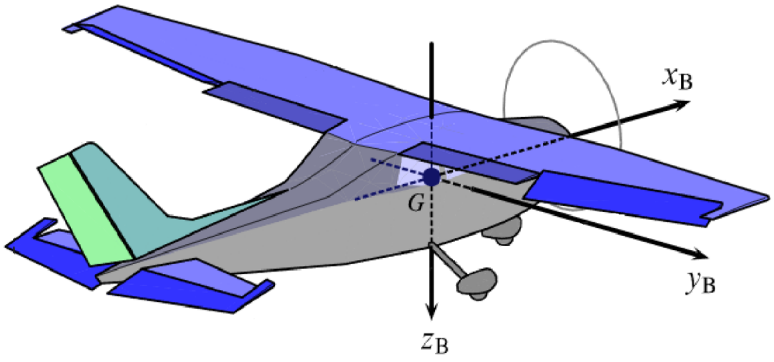

Figure 2.3: Example of Body Frame for an Airplane Adopted From [32]

Other common frames used in various problems are the north-east-down (NED) frame, the Wander-Azimuth frame, and the local Geodetic frame which represents position using latitude, longitude and altitude [5]. The body frame will be used for deriving bounds and simulations in the following chapters, so I will go into more detail about it here. In the body frame, angles are called roll α, pitch β and yaw γ. There

are many different conventions for defining these angles but in this paper I will stick to the convention that a positive rotation corresponds to the direction the fingers curl when the right-hand thumb is pointing in the positive body-axis direction.

When representing a rotation in 3-D space, both a Direction-Cosine Matrix (DCM) and quaternion derivation will be considered for reasons that will soon be explained. A DCM is a matrix of sines and cosines that performs a 3-d rotation of the reference frame to a desired roll, pitch and yaw. This matrix has many variations, but for the sake of having a consistent representation I define mine as:

$$\mathbf{C}_{\alpha,\beta,\gamma} = \begin{bmatrix} c(\alpha)c(\beta) & c(\alpha)s(\beta)s(\gamma) - s(\alpha)c(\gamma) & c(\alpha)s(\beta)c(\gamma) + s(\alpha)s(\gamma) \\ s(\alpha)c(\beta) & s(\alpha)s(\beta)s(\gamma) + c(\alpha)c(\gamma) & s(\alpha)s(\beta)c(\gamma) - c(\alpha)s(\gamma) \\ -s(\beta) & c(\beta)s(\gamma) & c(\beta)c(\gamma) \end{bmatrix} \tag{2.12}$$

where $c(x)$ and $s(x)$ have been used as a shorthand for $cos(x)$ and $sin(x)$ respectively. With this definition, the rotation of a vector in the body frame from one orientation to the next can be calculated as the linear transformation of a vector. Namely, if my initial position and attitude is at $\overline{v_0}$, I can rotate to a new orientation $\overline{v_1}$ using the transformation:

$$\overline{v_1} = \mathbf{C}_{\alpha,\beta,\gamma}\overline{v_0} \tag{2.13}$$

Another important quality of the DCM is that it's a linear transformation, so it can be decomposed into sub-rotation matrices that represent individual rotations about the x-y-z body axis. This can be written as:

$$\mathbf{C}_{\alpha,\beta,\gamma} = \mathbf{C}_{\alpha}\mathbf{C}_{\beta}\mathbf{C}_{\gamma} \tag{2.14}$$

Where:

$$\mathbf{C}_\alpha = \begin{bmatrix} c(\alpha) & -s(\alpha) & 0 \\ s(\alpha) & c(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix}, \ \mathbf{C}_\beta = \begin{bmatrix} c(\beta) & 0 & s(\beta) \\ 0 & 1 & 0 \\ -s(\beta) & 0 & c(\beta) \end{bmatrix},$$

$$\mathbf{C}_\gamma = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\gamma) & -s(\gamma) \\ 0 & s(\gamma) & c(\gamma) \end{bmatrix} \tag{2.15}$$

One problem with using DCM representations is that they suffer from singularities that are within 90 degrees. When $\beta = \frac{\pi}{2}$ for example, the rotation matrix $\mathbf{C}_{\alpha,\frac{\pi}{2},\gamma}$ gets locked into rotations about a single axis regardless of what α and γ are. This is known as "Gimbal Lock" and motivates the need for other types of representations.

Another type of attitude representation that doesn't suffer from singularities (i.e. Gimbal Lock) are called Euler Parameters or Quaternions. Quaternions can be thought of as a complex representation of angles and are given by:

$$\overline{q} = q_0 + q_1\hat{i} + q_2\hat{j} + q_3\hat{k} \tag{2.16}$$

Using the Euler Identity, this can also be written as

$$\overline{q} = cos(\frac{\theta}{2}) + \hat{n}sin(\frac{\theta}{2}) \tag{2.17}$$

Where $\hat{n}$ is the unit vector $[\hat{i}, \hat{j}, \hat{k}]^T$, and θ is the angle of rotation about the axis

$\hat{n}$. Rotation of vectors using quaternions is nearly as simple as using DCMs, namely, given a rotation described by quaternion $\overline{q}$, the quaternion rotates a vector $\overline{v_0}$ by:

$$\overline{v_1} = \overline{q v_0 q}^{-1} \tag{2.18}$$

where the Hamilton product is used to calculate the solution to equation (2.18). Prior to transformation, it is common to normalize quaternions so as to not add gain by using the fact that $||\overline{q}|| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}$, and the normalized quaternion is given by $\overline{q}_{norm} = \frac{\overline{q}}{||\overline{q}||}$.

Lastly, it is very easy to transform from quaternions back to a DCM representation using the following relationship [14] where:

$$\mathbf{C}_{\alpha,\beta,\gamma} = \begin{bmatrix} ||\overline{q}||^2 & 2(q_1q_2 - q_0q_3) & 2(q_0q_2 + q_1q_3) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_0q_1 + q_2q_3) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \tag{2.19}$$

Other common attitude representations that won't be used in this report but are worth mentioning are Rodrigues Parameters (classical and modified) which have a larger linear region than DCM's (180^o for classical and a full 360^o for modified) and Principle Rotation Parameters.

2.1.5 Geometric Dilution of Precision

An important metric for position accuracy in systems that use trilateration techniques is the Geometric Dilution of Precision or GDOP for short. GPS operates by making sure at least 4 of the 24 operational GPS satellites have a line of sight to any position on earth at all times which ensures unambiguous pseudo-range and time

estimation. The always changing locations of the 4 satellites with respect to each other affects the accuracy of the GPS measurement. Figure (2.4) visually describes what good GDOP looks like in comparison to poor GDOP:

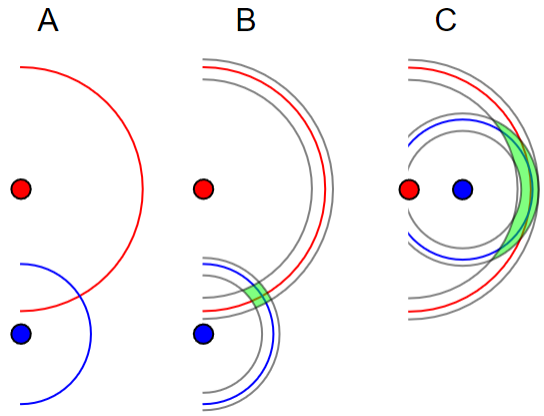

Figure 2.4: Visualization of GDOP: A.) Depicts Two Nodes With Target Location Marked by the Intersection of the Red and Blue Circles, B Illustrates Quality Gdop Due to Adequate Relative Node Placement, C Shows Poor Gdop Where Relative Node Locations Increases the Measurement Error. Adapted From [37]

The equation for GDOP [40] is defined as:

$$GDOP = \frac{\Delta(Output\ Location)}{\Delta(Measured\ Data)} \tag{2.20}$$

or in other words, how do variations in the locations of the target nodes (satellites in the case of GPS) affect the measurement error? In the case of a TWR system, a matrix $\mathbf{A}$ of unit distance vectors is formed and a covariance matrix $\mathbf{Q}$ is calculated to measure GDOP. For the purpose of this report, I will ignore clock bias, as that will be assumed 0 once the time-alignment algorithms are performed. Assuming the positions of N nodes are being estimated, the matrix $\mathbf{A}$ is:

$$A = \begin{bmatrix} \frac{(x_1-x)}{R_1} & \frac{(y_1-y)}{R_1} & \frac{(z_1-z)}{R_1} \\ \frac{(x_2-x)}{R_2} & \frac{(y_2-y)}{R_2} & \frac{(z_2-z)}{R_2} \\ \vdots & \vdots & \vdots \\ \frac{(x_N-x)}{R_N} & \frac{(y_N-y)}{R_N} & \frac{(z_N-z)}{R_N} \end{bmatrix} \tag{2.21}$$

where $R_i = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}$ is the 3-D distance of the known ground node to the i^{th} target position of interest. Forming the covariance matrix:

$$Q = (A^T A)^{-1} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy} & \sigma_{xz} \\ \sigma_{xy} & \sigma_y^2 & \sigma_{yz} \\ \sigma_{xz} & \sigma_{yz} & \sigma_z^2 \end{bmatrix} \tag{2.22}$$

and GDOP is:

$$GDOP = \sqrt{Tr(Q)} = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2} \tag{2.23}$$

with $Tr(\cdot)$ being the trace operator which is the sum of the diagonal elements for a square matrix.

Other similar metrics of interest include time dilution of precision (TDOP) that measures how clock bias affects the position accuracy and and horizontal/vertical dilution of precision (HDOP/VDOP) which is equivalent to GDOP but in a localized NED frame. For completeness, a table that numerically describes GDOP is included below:

DOP VALUE	1	1-2	2-5	5-10	10-20	>20
Rating	Ideal	Excellent	Good	Moderate	Fair	Poor

Table 2.1: Table Defining Numerical Values of GDOP

2.1.6 Position and Orientation Estimation

Having defined the necessary estimation and reference frame tools in the previous sections, we can now move on to discussion of position and orientation (PnO) estimation. Position and attitude can be estimated by leveraging the previously discussed TWR algorithms in conjunction with the geometry of the communication nodes. A typical 3-d geometry of a system whose goal is to estimate PnO is shown in figure (2.5).

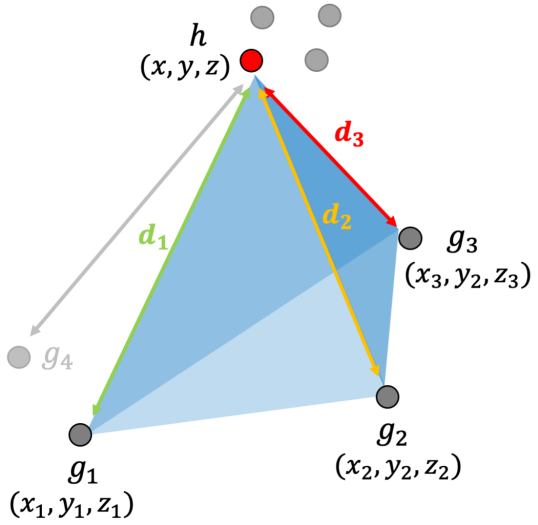

Figure 2.5: 3-D Geometry for Performing Positioning With Reference Nodes $\overline{g}_i$ and Target Nodes $\overline{h}_i$ [31], [27]

We first consider position estimation with a two-antennae system where each radio can both transmit and receive. This can then be expanded to an $n \times n$ multiple-input multiple-output (MIMO) system that estimates the distance between each antennae

pair. Assuming that accurate time-of-flight (ToF) estimates $\hat{\tau}$ have been obtained, the distance between nodes $\overline{g}$ and $\overline{b}$ is given by $d = ||\overline{g} - \tau c||^2$, where g is a known location vector of a ground node and $||\cdot||$ is the 2-norm operator [25]. Additionally, if there are N ground nodes $[g_1, g_2, \ldots, g_N]$ as shown in figure (2.5) who each receive a ToF estimate $[\tau_1, \tau_2, \ldots, \tau_N]$, then the distance from each ground node to node B is given by $d_i = ||g_i - \tau_i c||^2$.

Given the distance measurement described above, various estimators can then be used to estimate position. The typical LS solution shown in [39] is a good first choice with closed form shown in (2.5). Other solutions are found using an iterative version of LS called iterative recursive least squares (IRLS) [39] that produces an enhanced estimate by recursively updating the measurement covariance matrix and using the modified optimal LS solution:

$$\overline{\hat{\theta}} = (X^T \Sigma^{-1} X)^{-1} X^T \Sigma^{-1} \overline{y} \tag{2.24}$$

where Σ is covariance matrix of the measured distances from node G to B. A third version of LS called Non-Linear Least Squares (NLLS) [39] also constructs an accurate position estimator by solving the following nonlinear optimization problem:

$$\hat{\theta} = \min_{\theta} \sum_{i=1}^{n} [r_i - d_i r(\theta)]^2 \tag{2.25}$$

With r_i being the noisy distance measurement and $d_i r(\theta)$ the distance between the known location of G and a statistical reference point. Manolakis in [16] forms a similar solution to Navidi by cleverly transforming the distance equation into a root finding problem.

Attitude estimation has also been explored by many, and in terms of DCM's, the fundamentel problem can be stated as finding the rotation matrix $\mathbf{R}$ that is a solution to the equation:

$$\overline{b}_i = \mathbf{R}\overline{v}_i \tag{2.26}$$

which is the rotation matrix that aligns the reference vectors $\overline{v}_i$ with observation vectors $\overline{b}_i$. A typical starting point for this was first stated by Wahba [34] who observed that a rotation matrix $\mathbf{R}$ that minimizes the loss function:

$$L(\mathbf{R}) = \frac{1}{2}\sum_{j=1}^{n} w_j ||\overline{v_j^*} - \mathbf{R}\overline{v_j}||^2, \ \forall \ n \geq 2 \tag{2.27}$$

would create a least squares attitude estimate, where the set $\{v_j\}$ are vectors in a known reference (body) frame, and $\{v_j^*\}$ are the set of vectors in the observation frame. Typically these measurements are captured by integrating angular rates using a gyroscope sensor.

Algorithms such as TRIAD [23], Davenport Q's Method [4], QUEST [22] and OLAE [17] are just a few of the numerous solutions developed for this problem. TRIAD solves (2.27) as a system of overdetermined equations by creating triads of the reference and observation vectors $\{v_j\}$ and $\{v_j^*\}$ respectively.

Davenport Q's Method solves (2.27) by maximizing the gain function (equivalent to minimizing the loss function $L(M)$ stated in (2.27)):

$$G(\mathbf{R}) = 1 - L(\mathbf{R}) \tag{2.28}$$

$G(\mathbf{R})$ is maximized by forming a gain matrix represented with quaternions and finding the eigenvector corresponding to the max eigenvalue of this matrix as the best estimate. QUEST addresses (2.27) in a similar way to Davenport except solves the optimization problem using a mixture of Lagrange Multipliers and the Newton-Raphson root finding method.

Lastly, OLAE finds a solution to the problem of attitude estimation by forming a separate cost function:

$$L_{OLAE}(\mathbf{R}) = \frac{1}{2}\sum_{i=1}^{n} \xi_i([\tilde{s}_i\times]g - \tilde{d}_i)^T([\tilde{s}_i\times]g - \tilde{d}_i) \;\forall\; n \geq 2 \tag{2.29}$$

which is an unconstrained optimization problem that seeks to minimize L_{OLAE} with respect to Rodrigues parameter vector g.

2.1.7 Tracking

Once TWR (and hence position and attitude) estimates have been obtained, it's important to maintain a certain level of estimation accuracy over time as the vehicle traverses space. Typically in inertial navigation systems (INS), sensors like gyroscopes and accelerometers aid in navigation by providing attitude offsets, velocity and position of a moving body by integrating angular rate and acceleration measurements. This is a sound idea in theory, however, real sensors introduce noise into the integrators, which accumulate errors without bound over time. Additionally, real-time systems require estimates that are available on-demand. Regular first and second order statistics fail to deliver here due to requiring a sufficiently large amount of data before reliable predictions are available.

Popular solutions to these problems have been termed "tracking" filters because of their ability to provide accurate estimates in real-time along with statistical uncertainty's of these measurements. Common algorithms that will be simulated and analyzed in this thesis are the Kalman Filter (KF) [12] and Extended Kalman Filter (EKF) [10]. Other variants of the KF will also be mentioned like the Unscented Kalman Filter (UKF) [11] and Particle Filter [38] and may be useful in future extensions to this work.

The Kalman Filter [12] is a recursive algorithm that seeks to estimate the unmeasured states of a physical model that have direct mathematical relationships to the measurements. Given a nonlinear model or Plant represented by:

$$\begin{aligned} \dot{x} &= f(x, u, w) \\ y &= g(x, u, v) \end{aligned} \tag{2.30}$$

It can be linearized and represented by:

$$\begin{aligned} \dot{x} &= \mathbf{A}x + \mathbf{B}u + w \\ y &= \mathbf{C}x + \mathbf{D}u + v \end{aligned} \tag{2.31}$$

Where $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ are the matrices that represent the dynamic system, x is the states being estimated, w and v are both process and measurement noise respectively (typically Gaussian), u is the system input and y is the measurement. Graphically, the model can be represented as in figure (2.6) below.

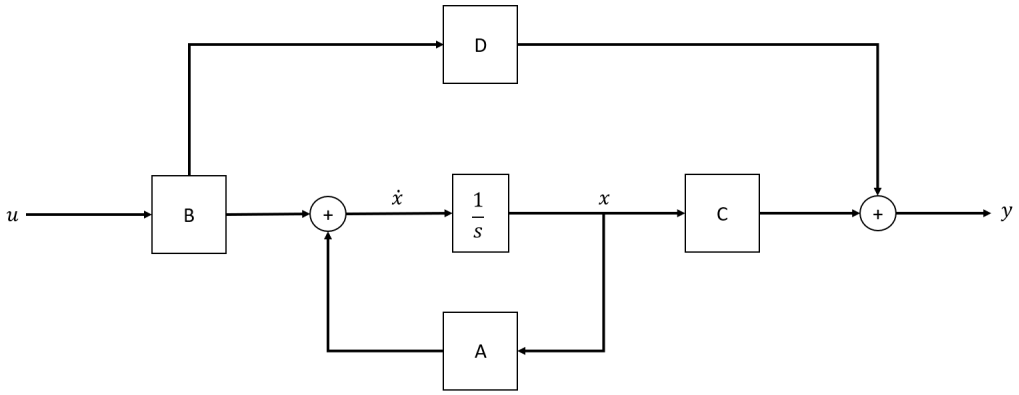

Figure 2.6: Block Diagram of State Space Representation

The Kalman Filter works by using a copy of the model ($\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$) derived from physics or techniques like system identification, and updating the state estimates according to the following algorithm:

Algorithm 1 Kalman Filter

1: $\hat{x}_k^- = \mathbf{F}\hat{x}_{k-1} + \mathbf{B}u_{k-1} + w$
2: $\mathbf{P}_k^- = \mathbf{F}\mathbf{P}_{k-1}^-\mathbf{F}^T + \mathbf{Q}_k$
3: $\mathbf{K}_k = \mathbf{P}_k^-\mathbf{H}_k^T(\mathbf{H}_k\mathbf{P}_k^-\mathbf{H}_k^T + \mathbf{R}_k)^{-1}$
4: $\hat{x}_k^+ = \hat{x}_k^- + \mathbf{K}_k(z_k + \mathbf{H}_k\hat{x}_k^-)$
5: $\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_k^-$

In algorithm 1, $\mathbf{F}$ is the state transition matrix where $\mathbf{F} = e^{(\mathbf{A}\Delta t)}$, $\mathbf{Q}_k$ and $\mathbf{R}_k$ represent the process and measurement noise covariance matrices at time k, $\mathbf{H}_k$ is the measurement matrix which relates predictions to measurements, $\mathbf{P}_k$ is the state covariance matrix also assumed Gaussian, and $\mathbf{K}_k$ is the famous Kalman Gain term that represents filters confidence in its ability to produce accurate estimates.

Many offshoots of the Kalman Filter have been produced as people have realized the limitations of this algorithm. The most common issue is that of representing nonlinear dynamics. The EKF handles this issues by linearizing the state and measurements models by taking the Jacobian of the nonlinear f and g with respect to

each state variable.

The UKF is another variation to the Kalman Filter that estimates the statistics of the states in the nonlinear system by applying a technique called the "Unscented Transform". A similar and more powerful algorithm called the Particle Filter estimates the mean and covariance for the probability density function of the states. Without going into a detailed derivation, it does this by initializing a set of "particles" or distribution points that are updated with each incoming measurement. This method has the added benefit of being able to gain accurate estimates while breaking the Gaussian assumption.

2.1.8 Applications

2.2 Commercial and Defense Applications

Navigation systems find homes in numerous applications ranging from and not limited to military, space, agriculture, supply chain, mining and weather prediction. Many of these applications would benefit from migrating their current GPS infrastructure to a TWR one that provides improved security and configurability. Military assets such as missiles, jets and satellites all use GPS along with other localization algorithms to find the position and orientation of a vehicle as it traverses space. Sensor fusion algorithms integrate together multiple different PnO estimates taken from GPS, local INS, radar amongst other things to produce the best solution possible. For example, an aircraft might have a GPS receiver and inertial-measurement unit (IMU) on-board to capture information from both sources and post-process the incoming data in a way that enhances its accuracy.

Business owners in the agricultural complex desires GPS-like capabilities for their farms and crops. Being able to make correlations between the land type, water use

age and yield helps optimize yearly profits. Marking exact areas where bug and weed infestations occur helps farms plan which areas to avoid when growing new crops. The technology for self-driving tractors has also made a good deal of recent advances [6] and relies on several sensors and GPS.

The supply chain industries such as brokerages also greatly benefit from localization. Tasks such as automating fleet maintenance, scheduling and more control over inventory are all handled with GPS. Mining applications also benefit by gaining increased accuracy in mapping, land surveying and improved safety. Once an open pit mine has been created, engineers can use TWR or GPS to aid in guiding drills, bull-dozing and managing other vehicles necessary to architect a closed mine.

2.3 CHP2 Implementation

One specific system that implements TWR is the CHP2 system described in [9, 7]. CHP2 is unique in that it was designed to jointly handle positioning and communications tasks which optimizes its spectral usage. Users who are on the CHP2 network will be able to synchronize their own local clocks to a global clock and obtain phase-accurate ToA estimates [7].

CHP 2 has exceptional performance as well achieving ranging estimates with $\sigma < 5$ cm, a limited 10 MHz bandwidth and a condensed acquisition time that is less than 3 seconds. These aspects allow the system to host a large number of users with no performance loss. The CHP2 processing was realized on COTS hardware which is inexpensive and small and proves ease-of-installation for a wide range of legacy systems.

Chapter 3

PROBLEM OVERVIEW

TWR PnO estimation techniques provide a method by which users can upgrade GPS dependent systems with a more flexible, secure, accurate and robust technology. While GPS carries the advantage of being low-cost to implement, many applications like secure network communications and air traffic management desire a localized custom navigation solution [7]. There are also many parts of the world where GNSS satellites are not available for use so having a local solution is necessary.

The desire to have other options for localization motivates the remainder of this report where I seek to quantify the level of position and attitude prediction accuracy a TWR architecture can provide. To do this, I will show how to derive the CRLB for position and attitude in a stationary environment for the cases of 2D and 3D geometry. After that, I will compare the derived CRLB to the performance of well-known PnO estimation algorithms. Additionally, I will show how tracking filters perform in comparison to TWR least squares and optimization methods for non-stationary systems.

Several factors like GDOP and ToF play a roll in the level of uncertainty in the derived CRLB. Therefore, Monte Carlo simulations will be performed to eliminate the individual effects of those factors and provide an "average" bound. Factors such as multipath and sensor noise add additional delay and distortion to the TWR signal. Clock jitter is also troublesome because it creates a time varying synchronization bias between the reference and target users. The individual effects of multipath, sensor noise and clock jitter will not be explicitly modeled but will be tied into the simulated ToF uncertainty. Nonlinearities are apparent in non-stationary systems

due to acceleration but will not be modeled for the sake of simplicity.

My contribution to the previously discussed body of work was to simulate the CRLB for 2-d and 3-d position and orientation estimation, and compare those bounds to well-known estimation algorithms. I believe this is novel because there are currently no existing closed form derivations for performance bounds with a TWR architecture. Furthermore, having a comparison between TWR and GPS provides designers with more options during initial system requirement derivation and planning stages. This makes it both interesting and challenging to formulate bounds and study the potential benefits and drawbacks of TWR.

Chapter 4

PERFORMANCE BOUNDS FOR POSITION AND ATTITUDE ESTIMATION

In this section, 2-D and 3-D bounds on the performance of PnO estimation will be derived in detail. The geometry of the problem will be defined graphically and mathematically and model assumptions made will be explained. Simulations of the derived bounds will also be shown by varying important parameter that appear in the calculations such as GDOP and concentration factor κ which will be explicitly defined in the following sections.

4.1 Cramer-Rao Lower Bound on Position Estimation

4.1.1 2-D Position Bound

We start out by rigorously defining the geometry of the 2-D positioning model. Let the N ground node antennae $[\overline{g_1}, \overline{g_2}, \ldots, \overline{g_N}]^T$ be 2-D Cartesian coordinates w.r.t the origin O. For each ground node g_i, let there be N prop-delay estimates $[\tau_1, \tau_2, \ldots, \tau_N]^T$ from the i^{th} ground node to the target location $\overline{h}$ as shown in figure (4.1) below.

For simplicity, it's assumed that the i^{th} time-delay estimate τ_i is distributed normally according to $\tau_i \sim N(\frac{d_i}{c}, \sigma^2_{\tau_i})$, with d_i being the measured distance, c being the speed of light in meters per second and $\sigma^2_{\tau_i}$ being the variance of the estimate. Mathematically, $d_i = ||g_i - h||^{1/2}$ which is just the 2-norm between the ground node and target point. The variance σ_{τ_i} of the ToF estimate heavily depends on the signals signal-to-noise ratio (SNR) as well as other factors like multipath and distance. Obviously, lower SNRs, larger distances, and difficult channels are going to impart greater negative effects on the estimates and increase their variance. Additionally, assuming that the distance between g_i and h is large in comparison to the size of the ground

platform, the ToF variance will be nearly equal for each ground node. In other words, $\sigma^2_{\tau_i} = \sigma^2_{\tau},\ \forall\ i$.

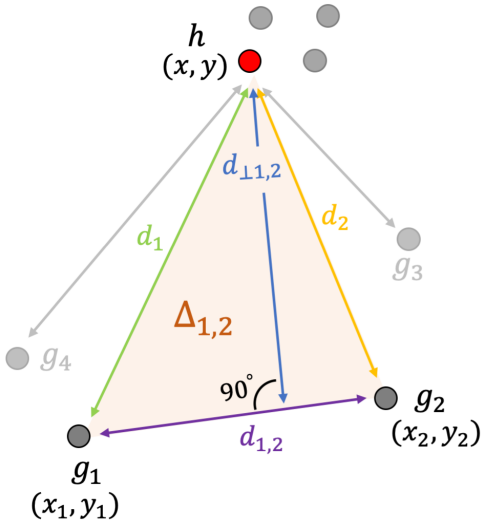

Figure 4.1: 2-D System Geometry for Position Estimation [31], [27]

Recalling the definition of the CRLB from equation 2.2, we know that the variance of our position estimate must be greater than or equal to the inverse Fisher Information $I(\theta)$. The position estimate is a vector, so the Fisher Information becomes a matrix $[I(\theta)]_{ij}$. We can calculate the Fisher Information Matrix as:

$$[I(\theta)]_{ij} = -E\Big[\frac{\partial^2}{\partial\theta_i \partial\theta_j} ln\ p(z;\theta)\Big] \tag{4.1}$$

and given that the distribution is Gaussian, its density can be written as:

$$p_\tau(t;h) = \frac{1}{\sqrt{2\pi\sigma^2_\tau}}\ exp\Big\{-\frac{1}{2\sigma^2_\tau}\sum_{i=1}^{N}\Big(t - \frac{d_i}{c}\Big)^2\Big\} \tag{4.2}$$

Using the definition of the 2-norm for $d_i = \sqrt{(x-x_i)^2 + (y-y_i)^2}$ which is the

distance between $\overline{g_i}$ and $\overline{h}$, we can calculate entries in $[I(\theta)]_{ij}$ as:

$$\begin{aligned}
[I(h)]_{11} &= \frac{1}{c^2\sigma_\tau^2}\sum_{i=1}^{N}\frac{(x-x_i)^2}{d_i^2} \\
[I(h)]_{12} = [I(h)]_{21} &= \frac{1}{c^2\sigma_\tau^2}\sum_{i=1}^{N}\frac{(x-x_i)(y-y_i)}{d_i^2} \\
[I(h)]_{22} &= \frac{1}{c^2\sigma_\tau^2}\sum_{i=1}^{N}\frac{(y-y_i)^2}{d_i^2}
\end{aligned} \tag{4.3}$$

Inverting the matrix using the formula that $I^{-1}(h) = adj\{I(h)\}/|I(h)|$, where the $adj\{\}$ is the ad-jugate operator and $||$ is the determinant, we obtain:

$$\sigma_{\mathbf{p}}^{\mathbf{2}} \geq \frac{c^2\sigma_\tau^2}{2\sum_{i=1}^{N}\sum_{i=1}^{N}\left(\frac{D_2}{2d_id_j}\right)^2}\begin{bmatrix} \sum_{i=1}^{N}\frac{(y-y_i)^2}{d_i^2} & -\sum_{i=1}^{N}\frac{(x-x_i)(y-y_i)}{d_i^2} \\ -\sum_{i=1}^{N}\frac{(x-x_i)(y-y_i)}{d_i^2} & \sum_{i=1}^{N}\frac{(x-x_i)^2}{d_i^2} \end{bmatrix} \tag{4.4}$$

The above matrix has the diagonal terms as the variance in each of the (x, y) component directions and the off-diagonal terms represent the cross-correlations. Assuming that most of the variation is represented in the σ_x^2 and σ_y^2 components, we can represent the 2-d position CRLB as a scalar value by taking the trace of equation (4.4). Doing so gives:

$$\sigma_p^2 \geq \frac{c^2\sigma_\tau^2 N}{2}\Big[\sum_{i=1}^{N}\sum_{j=1}^{N}\Big(\frac{D_2}{2d_id_j}\Big)^2\Big]^{-1} \tag{4.5}$$

Where:

$$D_2 = \begin{vmatrix} x - x_i & y - y_i \\ x - x_j & y - y_j \end{vmatrix} \tag{4.6}$$

The trace metric was chosen as one option but other options could include looking at the error ellipsoids spanned by the eigenvalues of the FIM to potentially gain additional insight.

4.1.2 3-D Position Bound

A 3-D extension to the position CRLB can be made by simply increasing the dimension of the vector space from $\mathbb{R}^2$ to $\mathbb{R}^3$. This can be visualized by refering back to figure (2.5) which shows the reference and target nodes defined in a 3-dimensional frame. Recall the term d_i in equation (4.2) and redefine the distance to be $d_i = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2}$, or the 2-norm in R^3. Once again, the i^{th} ground and target nodes have components $g_i = [x_i, y_i, z_i]^T$ and $h = [x, y, z]^T$ respectively. The FIM in 3-D is written as:

$$\mathbf{I}(h) = \frac{1}{c^2\sigma\tau^2} \begin{bmatrix} \sum_{i=1}^{N} \frac{(x-x_i)^2}{d_i^2} & \sum_{i=1}^{N} \frac{(x-x_i)(y-y_i)}{d_i^2} & \sum_{i=1}^{N} \frac{(x-x_i)(z-z_i)}{d_i^2} \\ \sum_{i=1}^{N} \frac{(x-x_i)(y-y_i)}{d_i^2} & \sum_{i=1}^{N} \frac{(y-y_i)^2}{d_i^2} & \sum_{i=1}^{N} \frac{(y-y_i)(z-z_i)}{d_i^2} \\ \sum_{i=1}^{N} \frac{(x-x_i)(z-z_i)}{d_i^2} & \sum_{i=1}^{N} \frac{(y-y_i)(z-z_i)}{d_i^2} & \sum_{i=1}^{N} \frac{(z-z_i)^2}{d_i^2} \end{bmatrix} \tag{4.7}$$

And as before, the CRLB is:

$$\sigma_h^2 \geq \mathbf{I}(h)^{-1} = \frac{adj\{\mathbf{I}(h)\}}{|\mathbf{I}(h)|} \tag{4.8}$$

Using equation (4.6) and defining the volume of 2-simplex $V_{i,j}(a,b) = D_2(a,b)/2!$, where a and b are the components of a vector in $\mathbb{R}^2$ and:

$$D_2(a,b) = \begin{vmatrix} a - a_i & b - b_i \\ a - a_j & b - b_j \end{vmatrix} \tag{4.9}$$

We can calculate the $adj\{\mathbf{I}(h)\} = \frac{2}{(c^2\sigma_\tau^2)^2}\mathbf{A}$ where $\mathbf{A}$ is defined as:

$$\mathbf{A} = \begin{bmatrix} \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{V_{i,j}(y,z)^2}{(d_i d_j)^2} & \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{V_{i,j}(x,z)V_{i,j}(z,y)}{(d_i d_j)^2} & \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{V_{i,j}(x,y)V_{i,j}(y,z)}{(d_i d_j)^2} \\ \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{V_{i,j}(x,z)V_{i,j}(z,y)}{(d_i d_j)^2} & \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{V_{i,j}(x,z)^2}{(d_i d_j)^2} & \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{V_{i,j}(y,x)V_{i,j}(x,z)}{(d_i d_j)^2} \\ \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{V_{i,j}(x,y)V_{i,j}(y,z)}{(d_i d_j)^2} & \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{V_{i,j}(y,x)V_{i,j}(x,z)}{(d_i d_j)^2} & \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{V_{i,j}(x,y)^2}{(d_i d_j)^2} \end{bmatrix} \tag{4.10}$$

Additionally, the determinant of the FIM is:

$$|\mathbf{I}(h)| = \frac{6}{(c^2\sigma_\tau^2)^3} \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N} \Big(\frac{V_{i,j,k}}{d_i d_j d_k}\Big)^2 \tag{4.11}$$

where geometrically, $V_{i,j,k}$ is the volume of a tetrahedron with vertices $h,\ g_i,\ g_j$ and g_k. This volume can be calculated using the Cayley-Menger [3] determinant which is:

$$V_{i,j,k} = \frac{1}{6}\begin{vmatrix} x - x_i & y - y_i & z - z_i \\ x - x_j & y - y_j & z - z_j \\ x - x_k & y - y_k & z - z_k \end{vmatrix} \tag{4.12}$$

Therefore, the inverse FIM is:

$$\mathbf{I}(h)^{-1} = \frac{c^2\sigma_\tau^2}{3}\mathbf{A}\left[\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}\left(\frac{V_{i,j,k}}{d_i d_j d_k}\right)^2\right]^{-1} \tag{4.13}$$

Once again the trace of the inverse FIM is used to easily make sense of the CRLB performance but other metrics like singular values could also be for analysis. Using the trace, the scalar 3-dimensional CRLB for position is:

$$\sigma_p^2 = Tr\{\mathbf{I}(h)^{-1}\} \geq \frac{c^2\sigma_\tau^2}{3}\frac{\left[\sum_{i=1}^{N}\sum_{j=1}^{N}\frac{V_{i,j}(x,y)^2+V_{i,j}(y,z)^2+V_{i,j}(x,z)^2}{(d_i d_j)^2}\right]}{\left[\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{N}\left(\frac{V_{i,j,k}}{d_i d_j d_k}\right)^2\right]} \tag{4.14}$$

Notice that the scalar quantity outside of the summation in the numerator of (4.14) is the distance variance because $\sigma_d^2 = c^2\sigma_\tau^2$. In figure (4.2) below we show an image of the 3-d bound using the trace metric and as expected, we see the linear relationship between σ_d and σ_p.

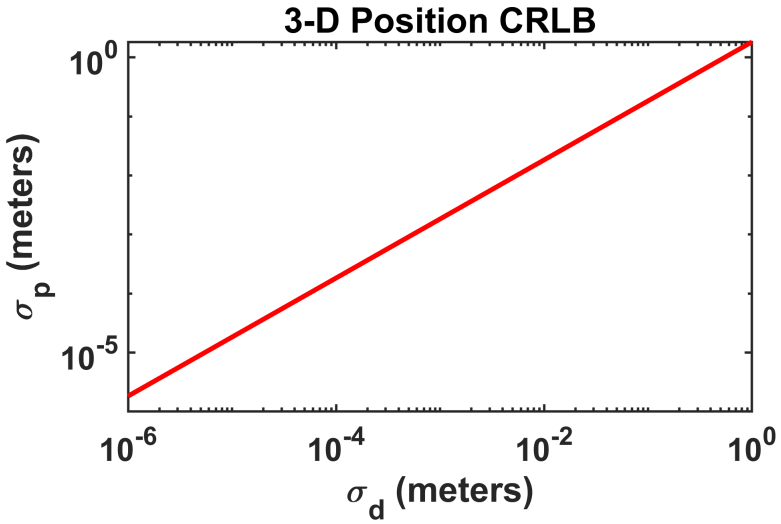

Figure 4.2: CRLB for Position With 3-d Geometry Vs. Distance Estimate Variance

The bound is also heavily dependent on the geometry of the ground and target nodes. Using the metric of GDOP described in section 2.1.5, we can look at how the bound is affected by relative locations of the target with respect to the ground nodes. Recall that the GDOP is a measurement of how the physical relationship of the geometry between the ground and target nodes affects position error. For the scenario being studied, the GDOP can be written as:

$$GDOP = \sqrt{\frac{N}{2\sum_{i=1}^{N}\sum_{\substack{j=1 \\ i\neq j}}^{N}\Delta_{i,j}^2}} = \sqrt{\frac{2N}{\sum_{k=1}^{M} sin^2(\gamma_k)}} \tag{4.15}$$

where $\Delta_{i,j} = \frac{1}{2}sin(\gamma_{i,j})$ is the area of the triangle formed by the unit vectors $\overline{hg_i}$, $\overline{hg_j}$, $\gamma_{i,j}$ is the angle between them, and $M = \binom{N}{2}$ is the total number of unique triangles formed from each ground and target node unit vector when $i \neq j$ [25]. A plot measuring how the CRLB is affected by GDOP is shown below:

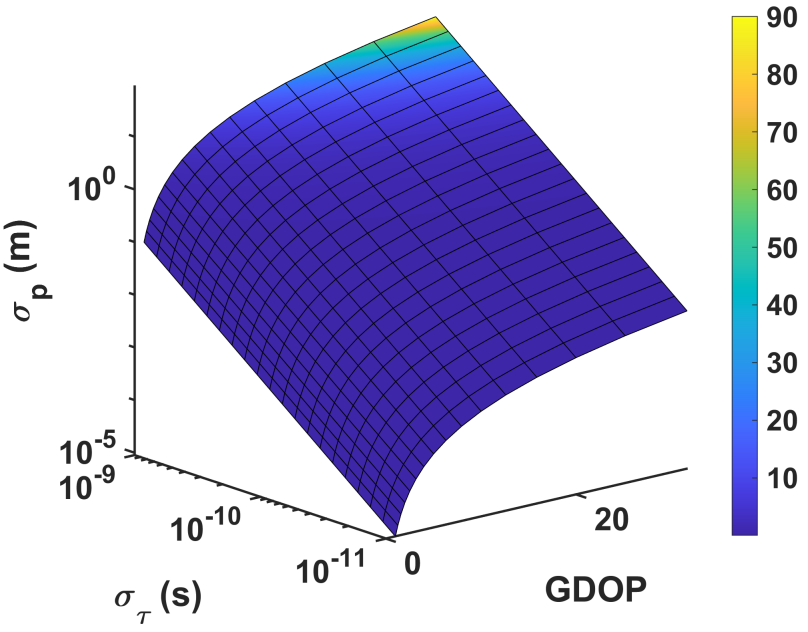

Figure 4.3: CRLB for Position Measured Against Distance Variation and GDOP

Here we see that lower GDOP improves the bound, however there are diminishing losses when the GDOP goes above approximately 10. The bound also shows as expected that lower GDOP is better, and GDOP ¡ 1 is ideal. Another perspective of the GDOP, σ_d and σ_p relation is shown below by taking cuts of figure 4.3 and is more easily comparable with figure 4.2.

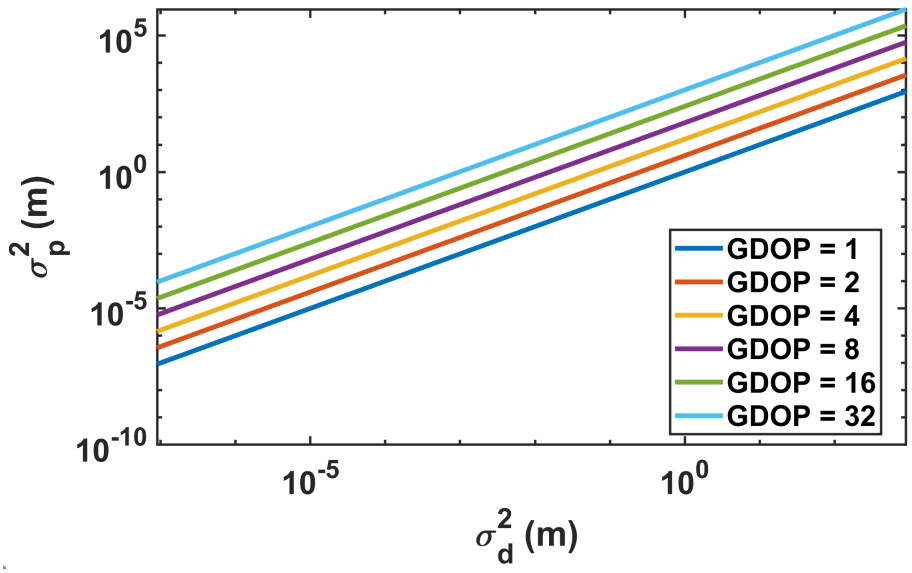

Figure 4.4: GDOP Cuts of CRLB for Position Measured Against Distance Variation

4.2 Cramer-Rao Lower Bound on Attitude Estimation

4.2.1 Attitude Model

In section 4.1.1, a model for the estimation of N target nodes $\{\overline{h_1}, \overline{h_2}, \ldots, \overline{h_N}\}$ using N ground nodes $\{\overline{g_1}, \overline{g_2}, \ldots, \overline{g_N}\}$ was shown. By defining the local distance vectors $\overline{h_i h_j}$ and $\overline{g_i g_j}$ with $i \neq j$, we can describe the attitude estimation problem as one of finding the rotation matrix between $\overline{h_i h_j}$ and $\overline{g_i g_j}$. Here, only the angles

between these vectors matter, so we can define normalized local distance vectors $\overline{b}_k = \overline{h_i h_j}/||\overline{h_i h_j}||^{1/2}$ and $\overline{v}_k = \overline{g_i g_j}/||\overline{g_i g_j}||^{1/2}$ for $k \in \{1, 2, \ldots, M\}$.

Unit vector $\overline{b}_k$ is assumed to be distributed with respect to a von Mises distribution [25], parameterized by mean direction $\overline{\mu}_k$ and concentration factor κ_k as:

$$\overline{b}_k \sim \mathcal{M}(\overline{\mu}_k, \kappa_k) \quad ; \quad \overline{\mu}_k = \mathbf{R}\overline{v}_k \tag{4.16}$$

with $\kappa > 0$, μ having a unity 2-norm, and rotation matrix $\mathbf{R}$ having orthogonal columns and the property that $|\mathbf{R}| = 1$. We note that when $\kappa << 1$ the von Mises distribution reduces to a uniform distribution on a circle and doesn't depend on $\mathbf{R}$. For other scenarios, we define the circular variance of $\overline{b}_k$ as $\sigma_{\overline{b}_k} = \sigma_{h_i} + \sigma_{h_j}$, where σ_{h_i} and σ_{h_j} are the deviation of the position estimate corresponding to nodes h_i and h_j respectively. Using this information, the concentration κ_k can be calculated using the modified Bessel functions $\mathbf{A}(\cdot)$ as $\kappa_k = \mathbf{A}^{-1}(\rho_k)$ where mean resultant length $\rho_k = e^{-\sigma^2_{\overline{h}_i h_j}/2}$.

4.2.2 Attitude in 2D

Using the definition of the von Mises distribution function with concentration parameter κ_k, relative rotation $\overline{\theta}$, rotation matrix $\mathbf{R}(\theta)$ and M unity length ground node vectors $\overline{v} = \{\overline{v}_1, \overline{v}_2, \ldots, \overline{v}_M\}$, we can write the density of $\overline{b}_k$ as:

$$p_{\overline{b}_k}(\overline{\nu}, \overline{v}, \theta, \kappa) = \left(\prod_{k=1}^{M} \frac{1}{2\pi I_0(\kappa_k)} \right) \exp\left\{ \sum_{k=1}^{M} \kappa_k \overline{\nu}_k^T \mathbf{R}(\theta) \overline{v}_k \right\} \tag{4.17}$$

with the term $I_0(\cdot)$ in the denominator being a modified Bessel function of the first kind at order 0. The second derivative of the rotation matrix $\mathbf{R}''(\theta) = -\mathbf{R}(\theta)$ and the FIM can be computed as:

$$\mathbf{I}(\theta) = -E\left[\frac{\partial^2}{\partial\theta^2} ln f(\overline{\nu}, \overline{v}, \theta, \kappa)\right] = \sum_{k=1}^{M} \kappa_k \overline{b}_k^T \mathbf{R}(\theta) \overline{v}_k \tag{4.18}$$

and therefore the CRLB is:

$$\sigma_\theta^2 \geq \mathbf{I}^{-1}(\theta) = \frac{1}{\sum_{k=1}^{M} \kappa_k \overline{b}_k^T \mathbf{R}(\theta) \overline{v}_k} \tag{4.19}$$

4.2.3 Attitude in 3D

Extending the derivation of the CRLB from the previous section into $\mathbb{R}^3$, we can write the modified von Mises density function as:

$$p_{\overline{b}_k}(\overline{\nu}, \overline{v}, \theta, \kappa) = \left(\prod_{k=1}^{M} C_3(\kappa_k)\right) \exp\left\{\sum_{k=1}^{M} \kappa_k \overline{\nu}_k^T \mathbf{R}(\theta) \overline{v}_k\right\} \tag{4.20}$$

where as before, κ is positive or zero, the 2-norm of the mean direction vector is unity, and $C_3(\kappa)$ is a normalization coefficient given by:

$$C_3(\kappa) = \frac{\kappa}{4\pi \ \sinh \ \kappa} = \frac{\kappa}{4\pi(e^{\kappa} - e^{-\kappa})} \tag{4.21}$$

Recall from section 2.1.4 that in contrast to the 2-dimensional rotation matrix that's represented by only one angle θ, the rotation matrix in 3-dimensional space when attached to the body frame is represented by 3 angles, namely roll (α), pitch (β) and yaw (γ). For our derivation, the rotation matrix being estimated represents a rotation between a body frame attached to the ground nodes and another body

frame attached to the target nodes. Using equation (2.14), the rotation matrix can be decomposed into its individual component rotations about the principle axis as $\mathbf{R}(\overline{\theta}) = \mathbf{R}(\alpha)\mathbf{R}(\beta)\mathbf{R}(\gamma)$, with $\overline{\theta} = [\alpha, \beta, \gamma]^T$. The FIM for 3-D attitude estimation is therefore given as:

$$\mathbf{I}(\theta) = \begin{bmatrix} -\sum_{i=1}^{N} \eta_i \frac{\partial^2}{\partial \alpha^2}\mathbf{R}(\overline{\theta})\overline{v}_i & -\sum_{i=1}^{N} \eta_i \frac{\partial^2}{\partial \alpha \partial \beta}\mathbf{R}(\overline{\theta})\overline{v}_i & -\sum_{i=1}^{N} \eta_i \frac{\partial^2}{\partial \alpha \partial \gamma}\mathbf{R}(\overline{\theta})\overline{v}_i \\ -\sum_{i=1}^{N} \eta_i \frac{\partial^2}{\partial \alpha \partial \beta}\mathbf{R}(\overline{\theta})\overline{v}_i & -\sum_{i=1}^{N} \eta_i \frac{\partial^2}{\partial \beta^2}\mathbf{R}(\overline{\theta})\overline{v}_i & -\sum_{i=1}^{N} \eta_i \frac{\partial^2}{\partial \beta \partial \gamma}\mathbf{R}(\overline{\theta})\overline{v}_i \\ -\sum_{i=1}^{N} \eta_i \frac{\partial^2}{\partial \alpha \partial \gamma}\mathbf{R}(\overline{\theta})\overline{v}_i & -\sum_{i=1}^{N} \eta_i \frac{\partial^2}{\partial \beta \partial \gamma}\mathbf{R}(\overline{\theta})\overline{v}_i & -\sum_{i=1}^{N} \eta_i \frac{\partial^2}{\partial \gamma^2}\mathbf{R}(\overline{\theta})\overline{v}_i \end{bmatrix} \tag{4.22}$$

with $\eta_i = \kappa_i b_i^T$.

Using a numerical differentiation library such as the one in MATLAB in conjunction with equation (4.8), the CRLB can be calculated for different body angles α, β and γ. The 3-D orientation CRLB was simulated over varying position uncertainty σ_p and concentration factor κ. These plots are shown below for $\overline{\theta} = [\pi/6, \pi/6, \pi/6]^T$.

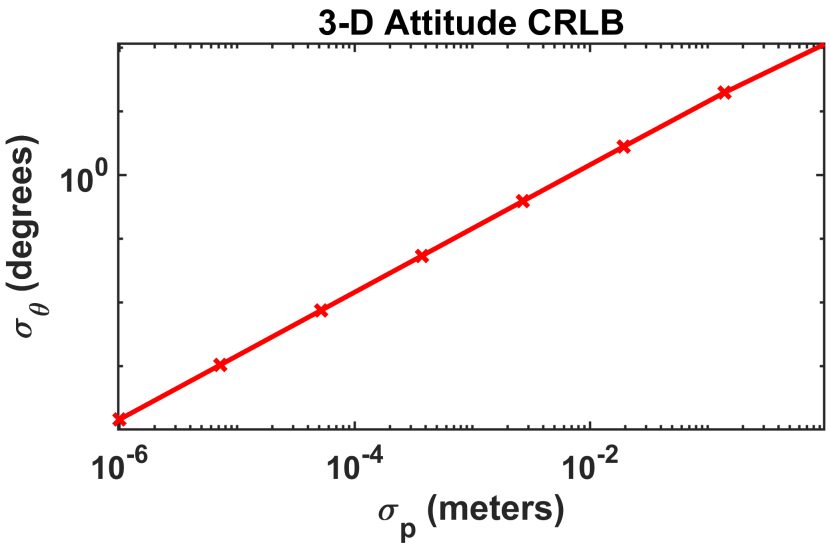

Figure 4.5: 3-D CRLB for Attitude Over Varying Position Estimate Uncertainty

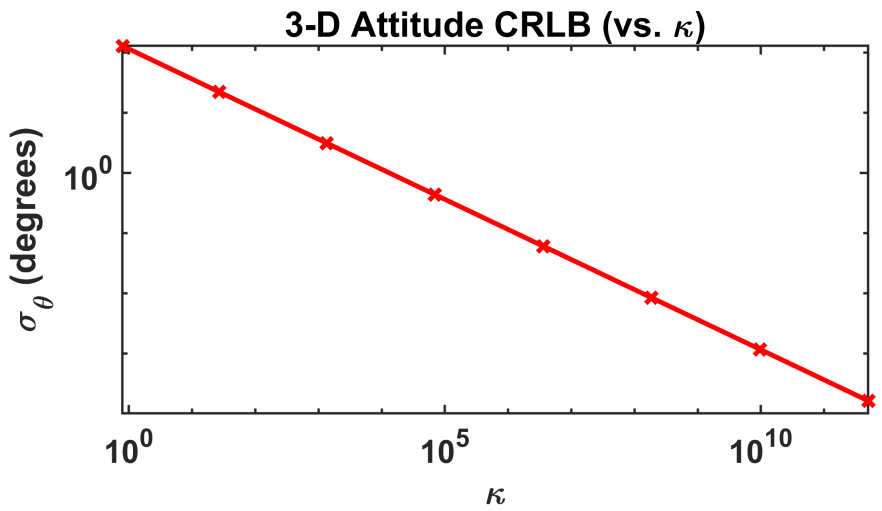

Figure 4.6: 3-D CRLB for Attitude Over Varying Concentration Factor κ

Once again, we see that the attitude CRLB linearly increases with increasing position estimation uncertainty. Figure 4.6 on the other hand shows how σ_θ decreases with increasing concentration, which is intuitive because normally concentration is the inverse of variance. The exact relationship between concentration and position variance is complicated however because of the Bessel relationship given by $\kappa = A^{-1}(\rho)$ with $A(\cdot) = I_1(\cdot)/I_0(\cdot)$ where I_1 and I_0 are first and zeroth order Bessel functions of the first kind respectively. An approximation to $A^{-1}(\cdot)$ can be made however and concentration can be approximated as:

$$\hat{\kappa} = \frac{\overline{\rho}(2-\overline{\rho}^2)}{1-\overline{\rho}^2} \quad ; \quad \overline{\rho} = e^{-\sigma_p^2} \tag{4.23}$$

where $0 \leq \overline{\rho} \leq 1$ is the mean resultant length and for completeness a graph of (4.23) is shown below:

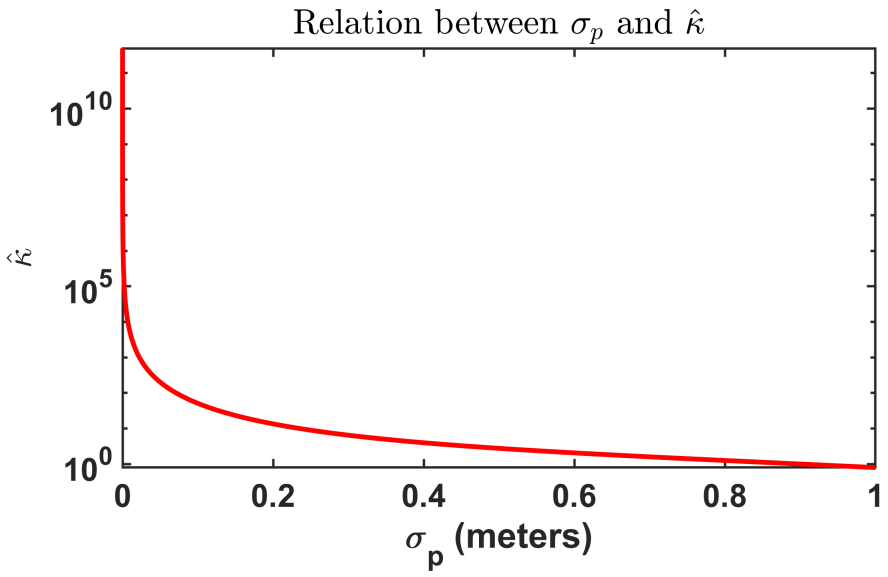

Figure 4.7: Relationship Between Concentration Factor and Position Estimate Uncertainty

Chapter 5

POSITION AND ORIENTATION ESTIMATION

Various known position and attitude estimators are studied in detail in this section and compared to the CRLB derived in chapter 4 using simulations. Least squares approaches mentioned in section 2.1.6 such as IRLS and NLLS will be used to obtain position estimates. Similarly, methods like TRIAD, QUEST, Davenport Q and OLAE will be looked at for orientation. The outcome of this section is to provide evidence that our derived CRLB is valid and give more confidence in using this bound for deriving system requirements.

5.1 Theory

Let there be N ground nodes $\{\overline{g}_1, \overline{g}_2, \ldots, \overline{g}_N\}$ with known locations and N target nodes $\{\overline{h}_1, \overline{h}_2, \ldots, \overline{h}_N\}$ we seek to estimate the position and attitude of with respect to a local reference frame for our ground stations. Each ground node $\overline{g}_n$ calculates N ToF estimates that it uses to find the distance between itself and each of the target locations $\overline{h}_n$.

Given these ToF estimates, many techniques have been developed to extract position and attitude using only known geometry and distance measurements. Most PnO estimation solution in the literature come in the form of Least Squares (LS) approaches or min/max optimization problems, therefore, more background on these topics will be briefly discussed here before going into specific estimator details.

For the problem of obtaining position [39], lets first define a distance metric

$$d_i(\overline{h}) = \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2} \tag{5.1}$$

for the distance between exact target coordinate $\overline{h}_i = [x_i, y_i, z_i]^T$ and ground coordinate $\overline{g} = [x, y, z]^T$. The measurements are noisy so the i^{th} ground node sees distance $r_i = d_i(\overline{h}) + n_i$, where n_i is noise caused by imperfections in the sensors, channel effects, signal power and distance amongst other things. The noise terms are considered independent with 0 mean and variance $Var(n_i) = \sigma_d^2$.

The measured distance model creates a nonlinear regression, so a linear model can be formed by creating a common reference point $[x_r, y_r, z_r]^T$ and defining

$$d_{ir} = \sqrt{(x_i - x_r)^2 + (y_i - y_r)^2 + (z_i - z_r)^2} \tag{5.2}$$

as the distance from reference point to the i^{th} ground node and

$$d_r(\overline{h}) = \sqrt{(x - x_r)^2 + (y - y_r)^2 + (z - z_r)^2} \tag{5.3}$$

as the distance from reference point to target. By writing the original distance equation and including the reference points, it can be shown that:

$$\begin{gathered} 2[(x - x_r)(x_i - x_r) + (y - y_r)(y_i - y_r) + (z - z_r)(z_i - z_r)] = \\ d_r(\overline{h})^2 + d_{ir}^2 - d_i(\overline{h})^2 \end{gathered} \tag{5.4}$$

and using this relation, a linear model can be formed by defining:

$$\mathbf{X} = \begin{bmatrix} 1 & 2(x_1 - x_r) & 2(y_1 - y_r) & 2(z_1 - z_r) \\ 1 & 2(x_2 - x_r) & 2(y_2 - y_r) & 2(z_2 - z_r) \\ \vdots & \vdots & \vdots & \vdots \\ 1 & 2(x_N - x_r) & 2(y_N - y_r) & 2(z_N - z_r) \end{bmatrix} \quad , \quad \overline{\beta} = \begin{bmatrix} -d_r(\overline{h})^2 - \sigma_d^2 \\ (x - x_r) \\ (y - y_r) \\ (z - z_r) \end{bmatrix} \tag{5.5}$$

and,

$$E(\overline{Y}) = \mathbf{X}\overline{\beta} \tag{5.6}$$

where $E(Y_i) = d_{ir}^2 - d_i(h)^2 - \sigma_d^2$. Equation (5.6) is the basis for the following position estimator derivations and will be frequently referred to in the next section.

Another important preliminary to the study of attitude estimators is that of stating Wahba's problem, as many of the popular methods are varying solutions to this. As mentioned in section 4.2.1, we can define the normalized local distance vectors for ground and target nodes as $\overline{b}_k = \overline{h_i h_j}/||\overline{h_i h_j}||^{1/2}$ and $\overline{v}_k = \overline{g_i g_j}/||\overline{g_i g_j}||^{1/2}$ for $k \in \{1, 2, \ldots, M\}$. We seek to find the rotation matrix $\mathbf{R}(\overline{\theta})$ that aligns $\overline{b_k}$ with $\overline{v_k}$, with $\overline{\theta} = [\alpha, \beta, \gamma]^T$. More specifically, Wahba states the problem as finding the $\mathbf{R}(\overline{\theta})$ that aligns the two vectors in the least squares sense [34], and thus can be written as:

$$\hat{\mathbf{R}} = \min_{\mathbf{R}} \frac{1}{2} \sum_{i=1}^{N} w_i ||\overline{b}_i^* - \mathbf{R}\overline{v}_i||^2 \tag{5.7}$$

where w_i is a weight applied to each term in the summation.

To wrap up this introduction, the distance estimates obtained from TWR are noisy and that noise propagates through the position and attitude acquisition algorithms. Therefore, estimation solutions like LS or others produces a type of "average" position and attitude that will always have some error.

5.2 Techniques

5.2.1 Position Estimation Techniques

We start this section by showing the ordinary least-squares (OLS) estimator from [39], because it is the most simple. We first form the matrix $\mathbf{X}_*$ by taking the right 3 columns of the matrix $\mathbf{X}$ from equation (5.5). An average reference point $\overline{\Delta}$ is then formed by taking the average x,y and z components from each of the ground nodes. We also set $\overline{Y}_* = \overline{Y} + \mathbf{X}_*\overline{\Delta}$, and then form the linear model:

$$E(\overline{Y}_*) = \mathbf{X}_*\overline{h} \tag{5.8}$$

which allows us to use the typical least squares solution of:

$$\hat{\overline{h}} = (\mathbf{X}_*^{\mathbf{T}}\mathbf{X}_*)^{-1}\mathbf{X}_*^{\mathbf{T}}\overline{Y}_* \tag{5.9}$$

The transformation in equation (5.8) makes it so the position vector estimate is unbiased with covariance matrix

$$Cov(\hat{\overline{h}}) = (\mathbf{X}_*^{\mathbf{T}}\mathbf{X}_*)^{-1}\mathbf{X}_*^{\mathbf{T}}\mathbf{\Sigma}\mathbf{X}_*(\mathbf{X}_*^{\mathbf{T}}\mathbf{X}_*)^{-1} \tag{5.10}$$

with the matrix $\mathbf{\Sigma}$ having the variance of the i^{th} row of $\overline{Y}$ along it's diagonal. Additionaly, it can be shown that the MSE for the OLS method is given by:

$$\mathbf{MSE}(\hat{\overline{h}}) = Tr[s^2({\mathbf{X}_*}^T\mathbf{X}_*)^{-1}] \tag{5.11}$$

with

$$s^2 = \frac{1}{n-4}\sum_{i=1}^{N}(\overline{Y}_i - \overline{\Delta} - \hat{\overline{Y}}_i)^2 \quad , \quad \hat{\overline{Y}}_i = \mathbf{X}_* \hat{\overline{h}} \tag{5.12}$$

The iteratively reweighted least squares (IRLS) [39] and its relevant moments follows directly from the previous OLS derivation. The uncertainty in each row of $\overline{Y}$ is different because the estimates are independently distributed, and therefore if covariance matrix $\mathbf{\Sigma}$ were known, the optimal linear estimation $\overline{\beta}$ could be found by:

$$\hat{\overline{\beta}}_{opt} = (\mathbf{X}^T\mathbf{\Sigma}^{-1}\mathbf{X})^{-1}\mathbf{X}^T\mathbf{\Sigma}^{-1}\overline{Y} \tag{5.13}$$

This is made possible by initializing $\mathbf{\Sigma}$ to the identity matrix and as the name says, iteratively updating $\hat{\overline{\beta}}$ and $\mathbf{\Sigma}$ until the estimate converges to within some tolerance. The covariance matrix for the optimal estimate $\hat{\overline{\beta}}_{opt}$ is given by:

$$Cov(\hat{\overline{\beta}}_{opt}) = (\mathbf{X}^T\mathbf{\Sigma}^{-1}\mathbf{X})^{-1} \tag{5.14}$$

Similarly to the OLS estimator, it can be shown that an MSE metric for the IRLS can be calculated with the equation:

$$\mathbf{MSE}(\hat{\overline{h}}) = Tr[{\mathbf{X}_*}^T\mathbf{\Sigma}^{-1}\mathbf{X}_*] \tag{5.15}$$

We lastly talk about the NLLS estimator by using nonlinear sum of squares equation as a cost function that compares the distance measurements with a prediction using $d_i(\overline{h})$. The cost function is:

$$F(\overline{h}) = \sum_{i=1}^{N}\left(r_i - \sqrt{(x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2}\right)^2 \tag{5.16}$$

To solve the minimization problem using the cost function above, a nonlinear least squares solver can be used using a method such as Levenberg-Marquardt.

Furthermore, [18] mentions that the covariance of the NLLS estimator can be found by forming the Jacobian $\mathbf{J}(\overline{h})$ as:

$$\mathbf{J}(\overline{h}) = \begin{bmatrix} \frac{\partial d_1(\overline{h})}{\partial x} & \frac{\partial d_1(\overline{h})}{\partial y} & \frac{\partial d_1(\overline{h})}{\partial z} \\ \frac{\partial d_2(\overline{h})}{\partial x} & \frac{\partial d_2(\overline{h})}{\partial y} & \frac{\partial d_2(\overline{h})}{\partial z} \\ \vdots & \vdots & \vdots \\ \frac{\partial d_N(\overline{h})}{\partial x} & \frac{\partial d_N(\overline{h})}{\partial y} & \frac{\partial d_N(\overline{h})}{\partial z} \end{bmatrix} \tag{5.17}$$

and then the covariance of the estimatate is $Cov(\hat{\overline{h}}) = \sigma^2[\mathbf{J}(\overline{h})^T\mathbf{J}(\overline{h})]^{-1}$ with variance estimated using:

$$\sigma^2 = \frac{1}{n-3}\sum_{i=1}^{N}[r_i - d_i(\hat{\overline{h}})]^2 \tag{5.18}$$

One final note is that taking the trace of the covariance matrix will provide a measure of the NLLS MSE [39].

5.2.2 Attitude Estimation Techniques

Details of four different attitude estimation schemes will be shown here in order to provide more insight into their ability to approach the CRLB, the first of which is called three-axis attitude determination or TRIAD [23]. Let there be 2 sets of local direction vectors, $\{\overline{b_1}, \overline{b_2}\}$ and $\{\overline{v_1}, \overline{v_2}\}$. An overdetermined system of equations can be written for these as:

$$\overline{b_1} = \mathbf{R}\overline{v_1} \quad \overline{b_2} = \mathbf{R}\overline{v_2} \tag{5.19}$$

Two triads can be constructed from these sets to create a singular equation in $\mathbf{R}$ by letting:

$$\begin{aligned} \overline{r}_1 = \overline{b}_1 \quad \overline{r}_2 = \frac{\overline{b}_1 \times \overline{b}_2}{|\overline{b}_1 \times \overline{b}_2|} \\ \overline{r}_3 = \frac{\overline{b}_1 \times (\overline{b}_1 \times \overline{b}_2)}{|\overline{b}_1 \times \overline{b}_2|} \end{aligned} \tag{5.20}$$

$$\begin{aligned} \overline{s}_1 = \overline{v}_1 \quad \overline{s}_2 = \frac{\overline{v}_1 \times \overline{v}_2}{|\overline{v}_1 \times \overline{v}_2|} \\ \overline{s}_3 = \frac{\overline{v}_1 \times (\overline{v}_1 \times \overline{v}_2)}{|\overline{v}_1 \times \overline{v}_2|} \end{aligned} \tag{5.21}$$

and from this, we can say that there exists a rotation matrix R that satisfies:

$$\overline{r}_i = \mathbf{R}\overline{s}_i \tag{5.22}$$

whose solution is:

$$\mathbf{R} = \sum_{i=1}^{3} \overline{r}_i \overline{s}_i^T \tag{5.23}$$

Another solution to the attitude estimation problem is written about in [4] and called the Davenport's Q method. We start with Wahba from equation (5.7) and form the gain function as:

$$\begin{aligned} g(\mathbf{R}) = 1 - L(\mathbf{R}) &= 1 - \frac{1}{2}\sum_{i=1}^{N} w_i ||\overline{b}_i - \mathbf{R}\overline{v}_i|| \\ &= \sum_{i=1}^{N} w_i \mathbf{B}^T \mathbf{R} \mathbf{V} \end{aligned} \tag{5.24}$$

where the best estimate of $\mathbf{R}$ is the matrix that maximizes equation (2.28). By defining the attitude profile matrix $\mathbf{A}$ as:

$$A = \sum_{i=1}^{N} w_i \mathbf{B}\mathbf{V} \tag{5.25}$$

we can simplify the gain function to $g(\mathbf{R}) = Tr(\mathbf{R}\mathbf{A}^T)$, with $Tr(\cdot)$ being the trace operator. Using the trace and quaternions, the gain function and rotation matrix are transformed into a different representation as:

$$g(\overline{q}) = \overline{q}^T \mathbf{K} \overline{q} \tag{5.26}$$

where $\overline{q}$ is the quaternion representation of $\mathbf{R}$. The matrix $\mathbf{K}$ is constructed as:

$$\mathbf{K} = \begin{bmatrix} \sigma & \overline{z}^T \\ \overline{z} & \mathbf{S} - \sigma \mathbf{I}_3 \end{bmatrix} \tag{5.27}$$

with substitution variables defined as:

$$\begin{aligned} \sigma &= Tr(\mathbf{A}) \\ \mathbf{S} &= \mathbf{A} + \mathbf{A}^T \\ \overline{z} &= \begin{bmatrix} A_{23} - A_{32} \\ A_{31} - A_{13} \\ A_{12} - A_{21} \end{bmatrix} \end{aligned} \tag{5.28}$$

and I_3 being the 3 identity matrix. Once $\mathbf{K}$ has been found, the optimal quaternion corresponds to solving the eigenvalue problem for:

$$\mathbf{K}\overline{q} = \lambda\overline{q} \tag{5.29}$$

and finding the eigenvector that corresponds to $\mathbf{K}'s$ largest eigenvalue.

Next we discuss the quaternion estimator algorithm or QUEST shown in [22]. Similarly to Davenport-Q's method, we form a cost function $g(\mathbf{R})$ to be maximized and make the same substitutions as equations (5.24) and (5.25). As before. the orientation matrix $\mathbf{R}$ can be represented using quaternions:

$$\mathbf{R}(\overline{q}) = (q^2 + \mathbf{Q}\cdot\mathbf{Q})\mathbf{I}_2\mathbf{Q}\mathbf{Q}^T + 2q\lfloor\mathbf{Q}\rfloor_x \tag{5.30}$$

where $\overline{q} = [q, \mathbf{Q}]^T$ is the scalar and vector parts of the quaternion respectively and the $\lfloor\cdot\rfloor_x$ operation is the skew-symmetric matrix operator that transforms a vector $\overline{v} = [v_x, v_y, v_z]^T$ into the matrix:

$$\lfloor\mathbf{Q}\rfloor_x = \begin{bmatrix} 0 & v_3 & -v_2 \\ -v_3 & 0 & v_1 \\ v_2 & -v_1 & 0 \end{bmatrix} \tag{5.31}$$

The gain function can be re-written using the quaternion rotation matrix representation and simplified into the compact form shown in equation (5.26). This time, $\mathbf{K}$ is represented slightly differently from equation (5.27) and given by:

$$\mathbf{K} = \begin{bmatrix} \mathbf{S} - \sigma \mathbf{I}_3 & \overline{z} \\ \overline{z}^T & \sigma \end{bmatrix} \tag{5.32}$$

with the transformation of variables to $\mathbf{S}$, $\overline{z}$ and σ being the same as equation (5.28).

The fundamental difference between the Davenport-Q and QUEST algorithms is the approach used to solve the quaternion parameterized optimization problem. Contrary to Davenport-Q, QUEST uses the approach of Lagrange Multipliers, and defines:

$$g'(\overline{q}) = \overline{q}^T \mathbf{K} \overline{q} - \lambda \overline{q}^T \overline{q} \tag{5.33}$$

with λ being the lagrange multiplier. Given that the largest eigenvalue (lagrange multiplier) in $\mathbf{K}$ and it's corresponding eigenvector give the optimal quaternion estimate $\overline{q}_{opt}$, the eigenvalue equation $\mathbf{K}\overline{q}_{opt} = \lambda_{max}\overline{q}_{opt}$ can be re-formulated as:

$$\lambda = \sigma + \overline{z}\ \overline{y} \tag{5.34}$$

where $\overline{y}$ is the Rodrigues vector defined as

$$\overline{y} = \frac{\mathbf{Q}}{q} = \hat{\mathbf{X}} tan \frac{\theta}{2} \tag{5.35}$$

and after a few more mathematical manipulations, a 4^{th} order expression in the eigenvalues can be written as:

$$\lambda^4 - (a+b)\lambda^2 - c\lambda + (ab + c\sigma - d) = 0 \tag{5.36}$$

with substitutions:

$$\begin{aligned} a &= \sigma^2 - \kappa \\ b &= \sigma^2 + \overline{z}^T\overline{z} \\ c &= \Delta + \overline{z}^T\mathbf{S}\overline{z} \\ d &= \overline{z}^T\mathbf{S}^2\overline{z} \\ \sigma &= \frac{1}{2}Tr(\mathbf{S}) \\ \kappa &= Tr(adj(\mathbf{S})) \\ \Delta &= |\mathbf{S}| \end{aligned} \tag{5.37}$$

A Newton Raphson root finding method is implemented to solve the polynomial in λ and the optimal quaternion can be written in closed form as:

$$\overline{q}_{opt} = \frac{1}{\sqrt{\gamma^2 + |\mathbf{X}|^2}} \begin{bmatrix} \mathbf{X} \\ \gamma \end{bmatrix} \tag{5.38}$$

with:

$$\begin{aligned} \mathbf{X} &= (\alpha\mathbf{I} + \beta\mathbf{S} + \mathbf{S}^2)\overline{z} \\ \gamma &= (\lambda + \sigma)\alpha - \Delta \\ \alpha &= \lambda^2 - \sigma^2 + \kappa \\ \beta &= \lambda - \sigma \end{aligned} \tag{5.39}$$

The last attitude estimation algorithm we will explore in this report is the optimal linear attitude estimator or OLAE [17]. We start with the typical problem statement from equation (2.26) and using the Cayley Transformation [17], we convert the DCM $\mathbf{R}$ into its equivalent Rodrigues matrix $\mathbf{G}$ through the relation:

$$\mathbf{G} = (\mathbf{I} - \mathbf{R})(\mathbf{I} + \mathbf{C})^{-1} = (\mathbf{I} + \mathbf{C})^{-1}(\mathbf{I} - \mathbf{R}) \tag{5.40}$$

where $\mathbf{G}$ is a 3×3 skew-symmetric matrix formed using the components of Rodriguez vector $\overline{g}$ defined in equation (5.35). By substituting equation (5.40) into equation (2.26), we obtain the sum and difference relationships:

$$[\overline{g}]_x(\overline{b}_i + \overline{v}_i) = -(\overline{b}_i - \overline{v}_i) \tag{5.41}$$

where we make a change of variables for the sum and difference components where $\overline{s}_i \triangleq \overline{b}_i + \overline{v}_i$ and $\overline{d}_i \triangleq \overline{b}_i - \overline{v}_i$. Using properties of the skew symmetric matrix, and substituting $\overline{s}_i$ and $\overline{d}_i$ in equation (5.41), we get a similarly looking relationship to equation (2.26) but in terms of the Rodriguez Vector:

$$[\overline{s}_i]_x\overline{g} = \overline{d}_i \tag{5.42}$$

The true observation vectors $\overline{b}_i$ will be noisy due to the sensor and other factors, so we define the true measurement vector as:

$$\tilde{\overline{b}}_i = \overline{b}_i + \overline{w}_i \tag{5.43}$$

with the term $\overline{w}$ being zero mean white noise. Using equations (5.42) and (5.43) we form the measurement error in terms of Rodriguez Vectors given by:

$$e_i = \left|\left| [\overline{s}_i]_x \overline{g} - \tilde{\overline{d}}_i \right|\right| \tag{5.44}$$

and from this form a new optimization problem separate from Wahba as:

$$L_m = \frac{1}{2} \sum_{i=1}^{N} \xi_i ([\overline{s}_i]_x \overline{g} - \tilde{\overline{d}}_i)^T ([\overline{s}_i]_x \overline{g} - \tilde{\overline{d}}_i) \tag{5.45}$$

Where ξ are relative weights for each estimate that have the property $\sum_i \xi_i = 1$. Expanding equation (5.45) as shown in [17], we get:

$$\begin{gathered} 2L_m = \sum_i \xi_i ([\overline{s}_i]_x \overline{g} - \tilde{\overline{d}}_i)^T ([\overline{s}_i]_x \overline{g} - \tilde{\overline{d}}_i) = \\ \sum_i \xi_i (-\overline{g}^T [\overline{s}_i]_x [\overline{s}_i]_x \overline{g} + g^T [\overline{s}_i]_x \tilde{\overline{d}}_i - \tilde{\overline{d}}_i^T [\overline{s}_i]_x \overline{g} + \tilde{\overline{d}}_i^T \tilde{\overline{d}}_i) = \\ \sum_i \xi_i \tilde{\overline{d}}_i^T \tilde{\overline{d}}_i - \overline{g}^T \big[\sum_i \xi_i [\overline{s}_i]_x [\overline{s}_i]_x \big] \overline{g} - 2 \big[\sum_i \xi \overline{d}_i^T [\overline{s}_i]_x \big] \overline{g} \end{gathered} \tag{5.46}$$

and therefore:

$$L_m = c - \overline{g}^T \tilde{\mathbf{M}}_m \overline{g} + 2 \tilde{\overline{v}}^T \overline{g} \tag{5.47}$$

where:

$$c = \frac{1}{2}\sum_{i=1}^{N} \xi \tilde{\overline{d}}_i^T \tilde{\overline{d}} \quad \tilde{\overline{v}} = \frac{1}{2}\sum_{i=1}^{N} \xi [\overline{s}_i]_x \tilde{\overline{d}}_i \quad \tilde{\mathbf{M}}_m = \frac{1}{2}\sum_{i=1}^{N} \xi [\overline{s}_i]_x [\overline{s}_i]_x \tag{5.48}$$

The optimal Rodrigues matrix is found by differentiating equation (5.47) with respect to $\overline{g}$ and setting equal to 0:

$$\frac{dL_m}{dg} = 2(\tilde{\overline{v}} - \tilde{\mathbf{M}}_m \overline{g}) = 0 \tag{5.49}$$

with solution found by solving the rigorously linear normal equation [17]:

$$\tilde{\mathbf{M}}_m \hat{\overline{g}} = \tilde{\overline{v}} \tag{5.50}$$

Finally, using the relationship that relates quaternions and Rodrigues vectors, the optimal quaternion representation of the attitude estimate is:

$$\hat{\overline{q}}_{opt} = \frac{\hat{\overline{q}}}{\sqrt{\hat{\overline{q}}^T \hat{\overline{q}}}} \tag{5.51}$$

5.3 Simulations

The PnO estimators described in the previous section were simulated against their respective CRLBs derived in chapter 4. Figure (5.1) shows variations of least squares

estimators plotted on top of the CRLB over varying levels of uncertainty in the TWR estimate. The estimators and CRLB were run for 20,000 trials with random noise seeded differently on each iteration. We see from the plot below that the NLLS estimator meets the CRLB, while the IRLS and OLS don't quite converge. The IRLS and OLS estimators actually have a similar level of estimation accuracy, although the IRLS does slightly better by incorporating the adapted covariance matrix.

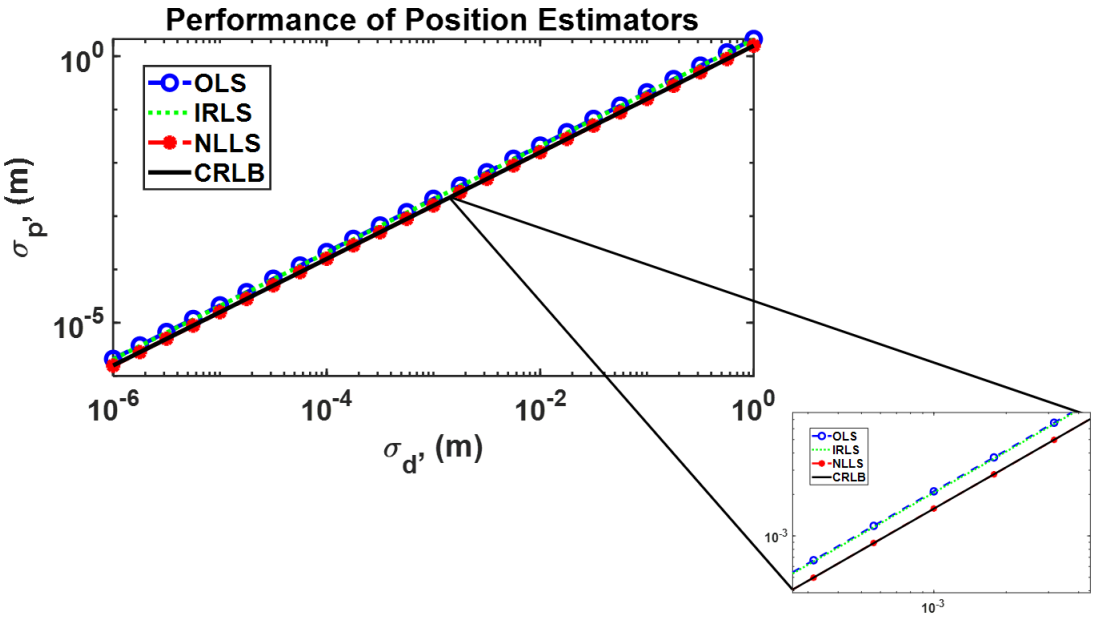

Figure 5.1: Performance of Position Estimators Vs. Their CRLB

Figure (5.2) shows the QUEST, TRIAD, Davenport-Q and OLAE algorithms comparing their performance to the CRLB over varying levels of position estimate uncertainty. The estimators were run over 100,000 trials with random measurement noise being seeded differently for each Monte-Carlo (MC) trial. We see that OLAE performs the best, followed by a tie between QUEST and Q-Method and lastly TRIAD.

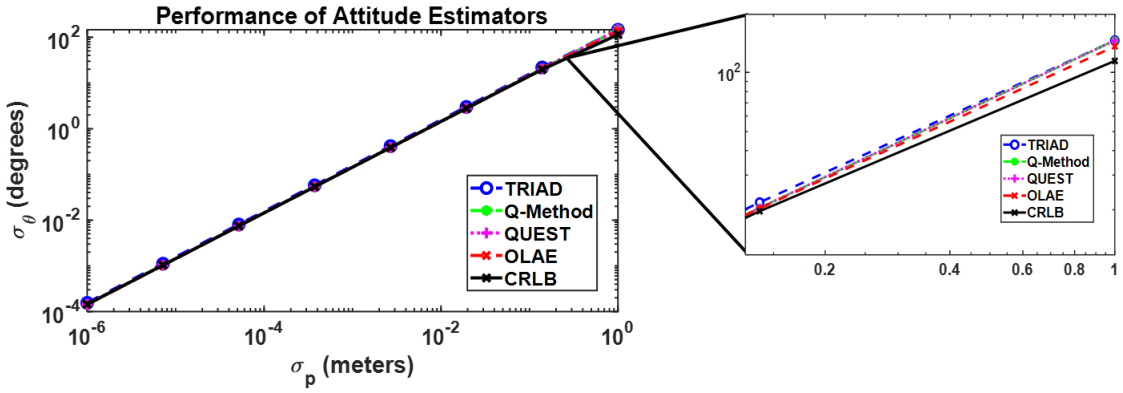

Figure 5.2: Performance of Attitude Estimators Vs. Their CRLB

For completeness, the attitude estimators vs their CRLB were also simulated over κ to show a different perspective on performance seen below in figure (5.3). This figure was also simulated and averaged over 100,000 MC trials.

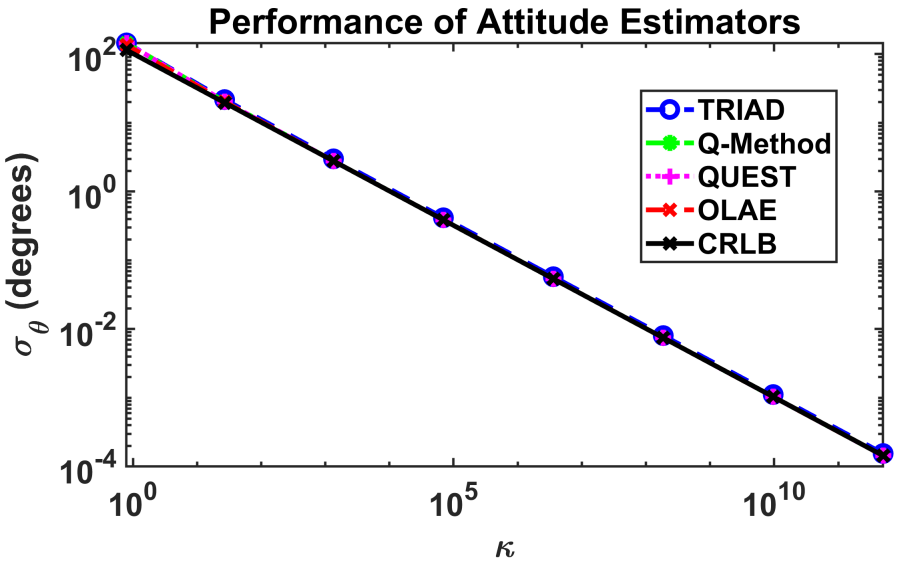

Figure 5.3: Performance of Attitude Estimators Vs. Their CRLB Plotted Over κ

Chapter 6

POSITION AND ATTITUDE ESTIMATION OF A NON-STATIONARY TARGET

In this section, a brief amount of theory and simulation is shown for PnO estimation using TWR when the target is non-stationary. The dynamics are modeled using a simple application of Newtons Equations with 6 degrees of freedom. An Kalman Filter (KF) is applied to perform state estimation, with the states being 3-D position, velocity, acceleration, orientation and angular rate. State space matrices like the state-transition matrix $\mathbf{F}$ and measurement matrix $\mathbf{H}$ will be derived in detail. A MATLAB simulation was also created to show the performance for different target trajectories. Attitude estimation will be simulated first using Euler Angles and then quaternions which are more robust to ambiguities.

6.1 Theory

We start by deriving the non-stationary target model that will be used to explore estimation. Figure (6.1) shows a typical setup with the set of reference nodes modeled as known 3-D position vectors $\{\overline{g_1}, \overline{g_2}, \ldots, \overline{g_N}\}$ and desired target node position vectors $\{\overline{p_1}, \overline{p_2}, \ldots, \overline{p_N}\}$. Let $\mathbf{G}$ and $\mathbf{P}$ represent two users, with index i representing the i^{th} node for user $\mathbf{G}$ and j representing the j^{th} node for user $\mathbf{P}$.

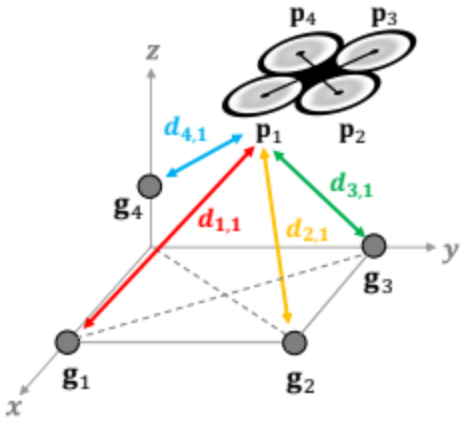

Figure 6.1: Typical Flight Geometry for the Non-stationary PnO Estimation Problem [27]

By employing a TWR protocol like the one modeled in chapter 4, each reference node i calculates a ToF estimate τ for each target node j and therefore has an estimate of the distance from i to j called $d_{i,j}$. This distance is also represented in figure (6.1) with the colored arrows and can be calculated as:

$$d_{i,j} = \sqrt{\sum_{k=1}^{3}(g_i[k] - p_j[k])^2} = c\tau_{i,j} \tag{6.1}$$

We can also define attitude in relation to an initial position in the targets body frame by creating a reference vector π_j and central reference position vector $\overline{p}$ such that:

$$\overline{p}_j = \overline{p} + \mathbf{R}(\overline{\theta})\pi_j \tag{6.2}$$

where $\mathbf{R}(\overline{\theta})$ is a direction-cosine matrix, and $\overline{\theta} = [\alpha, \beta, \gamma]^T$ are the body frame angles roll, pitch and yaw. This rotation can be easily extended to quaternions by using the mapping:

$$q_0 = \frac{1}{2}\sqrt{C_{11} + C_{22} + C_{33} + 1}$$

$$\begin{pmatrix} q_1 \\ q_2 \\ q_3 \end{pmatrix} = \frac{1}{4q_0} \begin{pmatrix} C_{23} - C_{32} \\ C_{31} - C_{13} \\ C_{12} - C_{21} \end{pmatrix} \tag{6.3}$$

and then replacing the rotation matrix $\mathbf{R}(\overline{\theta})$ in equation (6.2) with the 3×3 matrix given in equation (2.19).

6.2 Techniques

We start by showing the regular KF derivation for moving target estimation under the discussed geometry and then mention the linearized EKF extension. We define the Kalman Filter Markov model [27], with the state progression at time n represented as:

$$\hat{x}_n^- = \mathbf{f}(\hat{x}_{n-1}) + w_n \tag{6.4}$$

and measurement model:

$$\hat{z}_n^- = \mathbf{h}(\hat{x}_n^-) + v_n \tag{6.5}$$

Here, $\mathbf{f}$ and $\mathbf{g}$ are the nonlinear representations of the process and observation equations. We can leave the $\mathbf{B}$ matrix out of equation (6.4) assuming that there are no inputs to our system. The noise components w_n and v_n are the process and measurement noise respectively modeled using Gaussian distributions with $w \sim N(0, \Sigma_w)$ and $v \sim N(0, \Sigma_v)$. Given the geometry and applying Newtons Equations of Motion,

$\mathbf{f}$ and $\mathbf{h}$ can be defined like:

$$\mathbf{f} = \begin{cases} \overline{p}_n = \overline{p}_{n-1} + \overline{v}_{n-1}\Delta t + \frac{\overline{a}_{n-1}}{2}\Delta t^2 \\ \overline{v}_n = \overline{v}_{n-1} + \overline{a}\Delta t \\ \overline{a}_n = \overline{a}_{n-1} \\ \overline{\theta}_n = \overline{\theta}_{n-1} + \overline{\omega}_{n-1}\Delta t \\ \overline{\omega}_n = \overline{\omega}_{n-1} \end{cases} \tag{6.6}$$

and

$$\mathbf{h} = \sqrt{\sum_{x,y,z} \left[\overline{g}_i^n - \overline{\hat{p}}^n - \mathbf{R}(\overline{\hat{\theta}})\overline{\pi}_j\right]^2} \tag{6.7}$$

where the relative distance metric has been used for the state-to-measurement relation. The state vector $\overline{x}$ is:

$$\overline{x} = [\overline{x}, \overline{v}, \overline{a}, \overline{\theta}, \overline{\omega}]^T \tag{6.8}$$

with p position, v velocity, a acceleration, θ orientation and ω angular rate.

The measurement covariance matrix is initialized as a diagonal matrix using the known TWR (distance) uncertainty. The process covariance model was derived and initialized with small uncertainty factors in order to maintain a level of controllability throughout the filters evolution. By employing algorithm (1), over time the states and covariance matrices converge to the true trajectory.

The above equations implemented in the Kalman Filter framework function well for linear flight paths, but an extension is needed for nonlinear flight scenarios. An EKF is algorithmically similar to the regular KF, however the nonlinear functions $\mathbf{f}$ and $\mathbf{h}$ are linearized using the Jacobian. The new state transition and measurement

matrices become:

$$\mathbf{F} = \frac{\partial \mathbf{f}}{\partial \overline{x}}\Big|_{\hat{x}_{k-1|k-1}}, \quad \mathbf{H} = \frac{\partial \mathbf{h}}{\partial \overline{x}}\Big|_{\hat{x}_{k|k-1}} \tag{6.9}$$

The EKF assumes that the functions $\mathbf{f}$ and $\mathbf{h}$ are differentiable but all other aspects of the KF recursion (shown in algorithm (1)) remain the same.

6.3 Simulations

An example flight path shown in figure (6.2) was simulated and the EKF was used to estimate and track a user with 4 nodes. The full state position vector is plotted in dashed red to show the filters ability to accurately track the vehicles movement. Antenna for TWR were placed 1 meter from the center of the target along the x and y axis. We also show the ground (reference) nodes marked as 'x' on the plot and their geometry relative to the flight path.

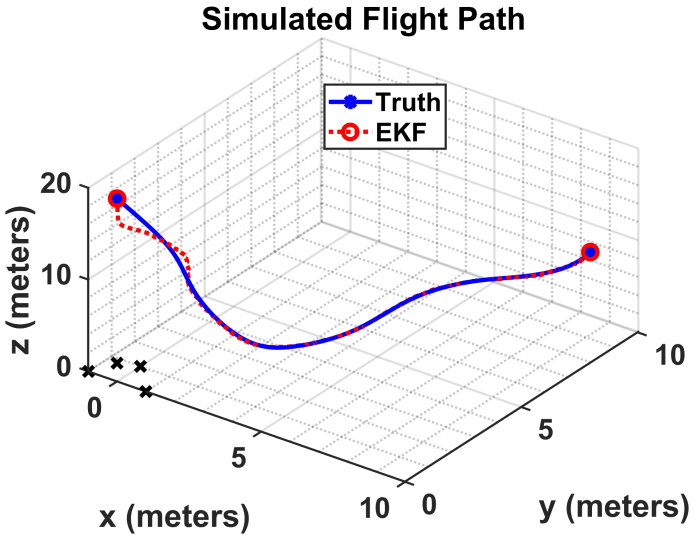

Figure 6.2: Example Flight Path Showing the EKF's Ability to Estimate a Target

Plots of the position and attitude full state components were adopted from [27] and plotted to show their deviation from truth. We see in figures (6.3) and (6.4) that the filter estimates position with about 1.5 centimeters of uncertainty and attitude with about 1.5 degrees of uncertainty. The simulated results were derived with attitude represented as Euler Angles. As mentioned in Chapter 2, for certain orientations and flight trajectories, Euler Angles suffer from ambiguities and Gimbal Lock that can lead to incorrect estimates. I extended the results from [27] by implementing the EKF using quaternions.

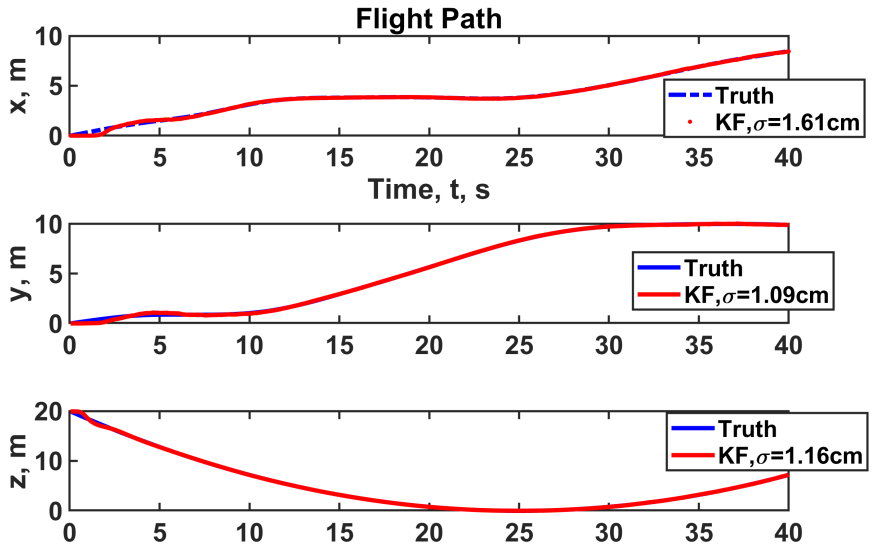

Figure 6.3: Full State Position Vector Vs. Truth

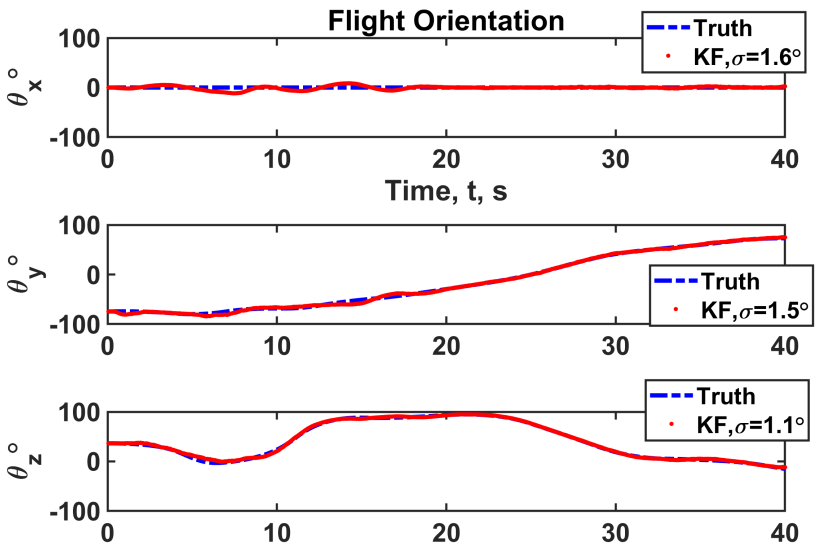

Figure 6.4: Full State Attitude Vector Vs. Truth

Chapter 7

APPLICATION: CHP2

This report wouldn't be complete without discussing at least one implementation of a TWR based navigation system. The CHP2 architecture presented in [7] and [8] will be discussed here with commentary on how the derived PnO bounds and estimator analysis could enhance this system.

The CHP2 architecture is considered a "joint system" because it employs a waveform that was designed for simultaneously performing both positioning and communications tasks. This methodology was created in response to applications that required both comms and localization but are limited in spectral resources. A state-of-the-art coherent ToA estimation technique is used on CHP2 that provides ranging data with as little as 5 cm of uncertainty.

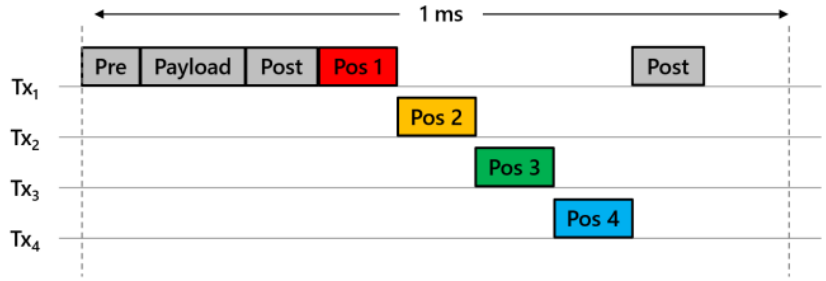

Figure 7.1: CHP2 Waveform for a 4-antenna User. The Colored Navigation Sequences Are Each Associated With Their Own Antenna to Provide Spatial Distinction. The Communications Payload Drives the Tof Algorithm and Pre/Post Ambles Are Used for Frequency and Time Synchronization [7]

A drawing of the joint waveform is shown in figure (7.1) where we can see the preamble, payload, 2 postambles and positioning sequences. The preamble and

postamble are used to perform frequency offset correction. The communications payload feeds the ToF algorithm on the receiver and positioning sequences provide the necessary spatial diversity for localization. In total the waveform is approximately 1 ms long, and transmit waveforms from each user area created and sent every 50 ms.

Experimental results from a flight test performed using CHP2 are shown in figure (7.2) take from [7]. A drone with the CHP2 system installed was flown around outside and 4 ground antennas employed the TWR algorithm to estimate ToF and therefore range. This is one huge step in the direction of getting precise localization. The estimates have a small amount of noise but the average estimate of each antenna appears to achieve the expected uncertainty.

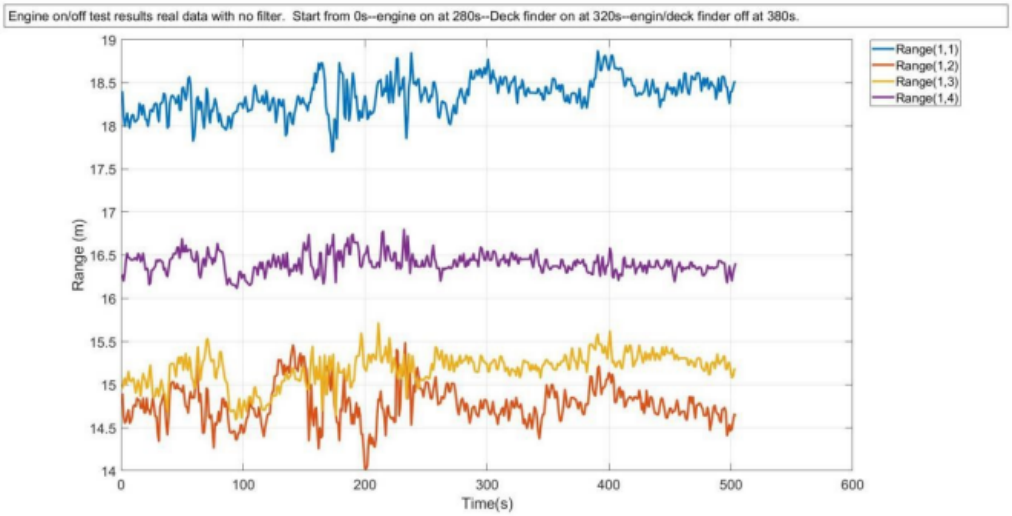

Figure 7.2: Experimental Results With Setup Similar to Figure (2.5). 4 Ground Reference Antennas Estimate the Range of a Target.

The system performance analysis done for CHP2 could benefit greatly by using the derive CRLB when deciding which kinds of PnO estimation algorithms to consider. My simulation results allow extrapolation of CHP2's 5 cm ranging uncertainty (0.33 ns of timing uncertainty) to show the accuracy of PnO estimation. System

trade-offs can be easily determined and requirements derived for an enhanced CHP2 that includes navigation.

My simulated CRLB could also be used to enhance decision making when setting up TWR applications in farming, surveying, mining, road transportation and others. By knowing the limitations on timing (and hence position and orientation) beforehand, one can plan ahead for things like land coverage capabilities, define network hand-offs and hardware and software cost budgeting.

Chapter 8

FUTURE WORK AND EXTENSIONS

I believe that several important extensions can be made to the body of work presented here which are highlighted below.

- Extending the work on bounds to deriving bounds for a non-stationary target. This would allow for a direct comparison between the bounds derived in Chapter 4, so that dynamic performance can better be assessed. This would also allow for comparison between different linear and nonlinear state estimation methods like the EKF, UKF, Particle Filter and MHE and give designers more insight into algorithm trade-offs.

- Implementation of real-time position and attitude estimation processing on the CHP2 platform. This would require both firmware and software updates to the current architecture but would enhance the overall abilities of the system. Some first steps in doing this would be to analyze the computational complexity of different algorithms. Being able to map the computations to hardware operations would allow one to figure out the amount of resources and expected latency on hardware.

- Studying how using a nonlinear state estimator would enhance the system. Various nonlinearities could be introduced into a more realistic motion model such as incorporating forces and moments. Studying the impacts of those on the EKF and then seeing how an algorithm like the Particle Filter could handle it could enhance the usefulness of the system in different applications.

- Reformulating the problem of non-stationary PnO estimation using a more robust attitude state representation such as quaternions. This could aid in making non-stationary TWR system more robust under certain trajectories.

Chapter 9

CONCLUSION

In summary, the work shown in this Thesis provides insight into bounds for the 3-D position and attitude estimation problem, how known estimators perform against those bounds, important analysis tools for deriving bounds and the beginning of an extension to non-stationary PnO estimation bounds and performance. I believe this work is important and can provide engineers who design position and navigation systems with more tools to make sound trade-offs. I also hoped to shed light on how TWR PnO estimation is a feasible and in some applications better alternative to legacy system like GPS due to it's enhanced security, more configurable architecture and decreased spectral resource needs.

REFERENCES

[1] Bidigare, P., U. Madhow, R. Mudumbai and D. Scherber, “”Attaining Fundamental Bounds on Timing Synchronization””, pp. 5229–5232 (2012).

[2] Bidigare, P., S. Pruessing, D. Raeman, D. Scherber, U. Madhow and R. Mudumbai, “”Initial Over-The-Air Performance Assessment of Ranging and Clock Synchronization Using Radio Frequency Signal Exchange””, pp. 273–276 (2012).

[3] Colins, K. D., “Cayley-menger determinant”, URL `https://mathworld.wolfram.com/Cayley-MengerDeterminant.html`, from MathWorld–A Wolfram Web Resource, created by Eric W. Weisstein (????).

[4] Davenport, P. B., “”A Vector Approach to the Algebra of Rotations with Applications””, NASA Technical Notes **vol. D-4696** (1968).

[5] Farrell, J. A., *Aided Navigation* (The McGraw-Hill Companies, 2008).

[6] Furioli, S., M. Corno, P. Cesana and S. M. Savaresi, “Automatic steering control for agricultural tractors in vineyards”, pp. 271–276 (2021).

[7] Herschfelt, A., *Simultaneous Positioning and Communications: Hybrid Radio Architecture, Estimation Techniques, and Experimental Validation*, Ph.D. thesis, Arizona State University (2019).

[8] Herschfelt, A., H. Yu, S. Wu, H. Lee and D. W. Bliss, “”Joint Positioning-Communications System Design: Leveraging Phase-Accurate Time-of-Flight Estimation and Distributed Coherence””, pp. 433–437 (2018).

[9] Herschfelt, A., H. Yu, S. Wu, S. Srinivas, Y. Li, N. Sciammetta, L. Smith, K. Rueger, H. Lee, C. Chakrabarti and D. W. Bliss, “Joint Positioning-Communications System Design and Experimental Demonstration”, pp. 1–6 (2019).

[10] Jazwinski, A. H., “”Stochastic Process and Filtering Theory””, Mathematics in Science and Engineering, Academic Press Inc **vol. 64** (1970).

[11] Julier, S. and J. Uhlmann, “”Unscented Filtering and Nonlinear Estimation””, Proceedings of the IEEE **92**, 3, 401–422 (2004).

[12] Kalman, R. E., “”A New Approach to Linear Filtering and Prediction Problems””, Transactions of the ASME–Journal of Basic Engineering **vol. 82**, Series D, 35–45 (1960).

[13] Kay, S. M., *Fundamentals of Statistical Signal Processing* (Prentice Hall, 1993).

[14] Kuipers, J. B., *Quaternions and Rotation Sequences* (Princeton University Press, 1999).

[15] Lee, J.-Y. and R. Scholtz, ""Ranging in a Dense Multipath Environment Using an UWB Radio Link"", IEEE Journal on Selected Areas in Communications **vol. 20**, 9, 1677–1683 (2002).

[16] Manolakis, D., ""Efficient Solution and Performance Analysis of 3-D Position Estimation by Trilateration"", IEEE Transactions on Aerospace and Electronic Systems **vol. 32**, 4, 1239–1248 (1996).

[17] Mortari, D., L. Markley and P. Singla, ""Optimal Linear Attitude Estimator"", Journal of Guidance Control and Dynamics **vol. 30**, 1619–1627 (2007).

[18] P, M. and N. J., ""Generalized Linear Models, 2nd Edition"", (1989).

[19] Rao, C. R., ""Information and the Accuracy Attainable in the Estimation of Statistical Parameters"", Bulletin of the Calcutta Mathematical Society **37**, 81–89 (????).

[20] Reece, S. and D. Nicholson, "Tighter Alternatives to the Cramer-Rao Lower Bound for Discrete-Time Filtering", 2005 7th International Conference on Information Fusion, FUSION **1**, 6 pp. (2005).

[21] Rogers, R. M., *Applied Mathematics in Integrated Navigation Systems 3rd. Edition* (American Institute of Aeronautics and Astronautics, 2007).

[22] Shuster, M. D., ""Kalman Filtering of Spacecraft Attitude and the QUEST Model"", The Journal of Astronautical Sciences **vol. 39**, 377–393 (1990).

[23] SHUSTER, M. D. and S. D. OH, ""Three-Axis Attitude Determination from Vector Observations"", Journal of Guidance and Control **vol. 4**, 1, 70–77, URL `https://doi.org/10.2514/3.19717` (1981).

[24] Srinivas, S., *Communications and high-precision positioning (chp2) system: Enabling distributed coherence and precise positioning for resource-limited air transport systems*, Ph.D. thesis, Arizona State University (2020).

[25] Srinivas, S., A. Herschfelt and D. W. Bliss, "Joint Positioning-Communications System: Optimal Distributed Coherence and Positioning Estimators", pp. 317–321 (2019).

[26] Srinivas, S., A. Herschfelt and D. W. Bliss, "Joint positioning-communications system: Optimal distributed coherence and positioning estimators", in "2019 53rd Asilomar Conference on Signals, Systems, and Computers", pp. 317–321 (IEEE, 2019).

[27] Srinivas, S., A. Herschfelt and D. W. Bliss, "Communications and high-precision positioning (chp2): Joint position and orientation tracking", in "2021 IEEE Aerospace Conference (50100)", pp. 1–6 (IEEE, 2021).

[28] Srinivas, S., A. Herschfelt and D. W. Bliss, "Estimation and tracking of position information in a distributed radio frequency (rf) communications system", US Patent App. 17/089,086 (2021).

[29] Srinivas, S., A. Herschfelt and D. W. Bliss, "Position information estimation in a distributed radio frequency (rf) communications system", US Patent App. 17/089,074 (2021).

[30] Srinivas, S., A. Herschfelt, A. Chiriyath and D. W. Bliss, "Joint positioning-communications: Constant-information ranging for dynamic spectrum access", in "2020 IEEE 92nd Vehicular Technology Conference (VTC2020-Fall)", pp. 1–5 (IEEE, 2020).

[31] Srinivas, S., A. Herschfelt, H. Yu, S. Wu, Y. Li, H. Lee, C. Chakrabarti and D. W. Bliss, "Communications and high-precision positioning (chp2): Enabling secure cns and apnt for safety-critical air transport systems", in "2020 AIAA/IEEE 39th Digital Avionics Systems Conference (DASC)", pp. 1–6 (IEEE, 2020).

[32] Team, J., "Standard aircraft body axis frame", URL `https://jsbsim-team.github.io/jsbsim-reference-manual`, [Online; accessed February 25, 2022] (????).

[33] Van Trees, H. L., K. L. Bell and Z. Tian, *Detection, Estimation, and Modulation Theory* (Wiley, 2013).

[34] Wahba, G., ""A Least Squares Estimate of Satellite Attitude"", Siam Review **vol. 7**, 409–409 (1965).

[35] Weinstein, E. and A. Weiss, "Fundamental Limitations in Passive Time-Delay Estimation–Part II: Wide-Band Systems", IEEE Transactions on Acoustics, Speech, and Signal Processing **32**, 5, 1064–1078 (1984).

[36] Weiss, A. and E. Weinstein, ""Fundamental Limitations in Passive Time Delay Estimation–Part I: Narrow-Band Systems"", IEEE Transactions on Acoustics, Speech, and Signal Processing **vol. 31**, 2, 472–486 (1983).

[37] Wikipedia, "Understanding geometric dilution of precision", URL `https://en.wikipedia.org/wiki/Dilution-of-precision-(navigation)`, [Online; accessed February 25, 2022] (2013).

[38] Xu, Y., K. Xu, J. Wan, Z. Xiong and Y. Li, "Research on Particle Filter Tracking Method Based on Kalman Filter", pp. 1564–1568 (2018).

[39] Y, W. N., W. S. M. J. Z and W. H. Y, ""Statistical Methods in Surveying by Trilateration"", (1999).

[40] Yongcai, A., Z. Bo, Z. Baozhuo and W. Shili, ""Change of Geometric Dilution of Precision (GDOP) for Integrated System"", pp. 660–662 (2016).

[41] Zeira, A. and P. Schultheiss, "Realizable lower bounds for time delay estimation", IEEE Transactions on Signal Processing **41**, 11, 3102–3113 (1993).