# ABSTRACT

Outlier Selection Methods for Improved Bearings-only Geolocation

Trevor J. Barnett, M.S.E.C.E

Mentor: Michael Thompson, Ph.D.

Bearings-only location estimation is a problem that has widespread applications. While the concept of bearings-only location estimation is not new, there are still many problems inherent to the process. There is high demand for a process that can reduce bias, remove outliers, and more accurately estimate emitter location using bearing data, exclusive of range. This thesis applies outlier removal methods, the parameterization of which are characterized within, to create more reliable data sets from noisy data which likely contain a significant percentage of outliers. Secondly, it uses statistical estimation and resampling to create a small, plausible ellipse representing the location of the emitter being tracked. In this thesis simulation studies show these methods to be a significant improvement over the standard implementation of Cartesian Pseudo-Linear Estimation with the presence of outlier data.

Outlier Selection Methods for Improved Bearings-Only Geolocation

by

Trevor J. Barnett, B.S.

A Thesis

Approved by the Department of Electrical & Computer Engineering

---

Kwang Y. Lee, Ph.D., Chairperson

Submitted to the Graduate Faculty of
Baylor University in Partial Fulfillment of the
Requirements for the Degree
of

Master of Science

Approved by the Thesis Committee

---

Michael Thompson, Ph.D., Chairperson

---

Yang Li, Ph.D.

---

Stephen McClain, Ph.D.

Accepted by the Graduate School

December 2022

---

J. Larry Lyon, Ph.D., Dean

*Page bearing signatures is kept on file in the Graduate School.*

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

# ACKNOWLEDGMENTS

I would, firstly, like to thank Dr. Thompson, my advisor, for the time and effort he's put into my academic career over the past years. Without his help I would not have reached the point I'm at today. I'd also like to thank the other faculty and staff who I've interacted with during my time at Baylor. The knowledge and skills I've gained through their classes have been invaluable in my career. Also, the members of my thesis committee for the provision of their time and effort in finalizing this process. Lastly, my family and friends whose support, financial and otherwise, allowed me to succeed.

DEDICATION

*To my wife Emma, thank you for pushing me to finish what I start*

# CHAPTER ONE

## Introduction

Emitter localization (or emitter tracking, in the more general case of a nonstationary emitter), is the task of estimating the location of a given radio frequency (RF) emitter. Upon the invention of radio-based applications there has been considerable interest in using the physical properties of a received signal to estimate the location from which the signal was transmitted. Emitter "localization" can be described as the process of using the information embedded in the signal received from the transmitter to estimate the location of the transmitter. Often the problem is generalized to emitter "tracking" for the case where the emitter is moving, although for this work, we consider the emitter to be at a fixed location.

Emitter localization has several applications. The military, for example, would be interested in unraveling the location of an adversarial radar transmitter signal that is "received" by one of its aircraft. A lost hiker carrying an emergency location transponder would be hopeful that a search and rescue team would be able to accurately decode the information from the transponder to so that they would be found quickly. And one can imagine navigation applications where certain points of interest could be marked "electronically" for either guidance or avoiding obstacles. In some of these cases (for example, search and rescue) both parties are cooperating in the sense that the lost hiker wants to be found and the rescuers want to find the lost hiker. However, for the military application mentioned previously, the adversary typically does not want to be detected and will take measures to limit their exposure.

Emitter location can be accomplished with a variety of approaches which depend on the sensors and application at hand. In this thesis, we focus on bearings-only localization. In many cases, range data is considered to be very inaccurate and therefore including range information is often a detriment to geolocation performance; hence the restriction to bearings-only. Furthermore, in many cases the altitude component of the data is also inaccurate relative to the azimuth. Removing the altitude component reduces the problem to two dimensions, rather than three. Bearings-only localization, the process of locating an RF emitter using only angle of arrival data, is an approach born out of research done by Stansfield and published in 1947 [4], which details a method for using separate unrelated observations of a signal to determine its origin. Angle of arrival (AoA) data can be calculated based on the time-difference of arrival between elements of a sensor array. There are two primary sources of error that are inherent to AoA data. The phenomenon associated with the physics of waveform propagation whereby the signal propagates along more than one path is one significant source of error. A second source of error arises from the signal processing algorithms used to extract phase and time differences from the received signals [9]. It should be noted that both of these types of error can result in significantly poor AoA estimates. When these types of errors occur, we consider the data to by outlier data in need of removal.

A focus of this thesis is on practical techniques for outlier removal. We also attempt to characterize the overall system performance that can be expected with outlier removal. The frame work for this study begins with the assumption of a signal, fixed-location emitter in cartesian space. A moving platform passes through that space with sensors capable of receiving AoA data.   The receiver gathers data about the signal at specified intervals, and records that data. Specifically, the receiver knows only the angle between itself and the source of the signal and stores it relative to its own position in space. This data, known as Angle of Arrival (AoA) data, is then used to generate a position estimate for the emitter. Figure 1.1 shows this common scenario.

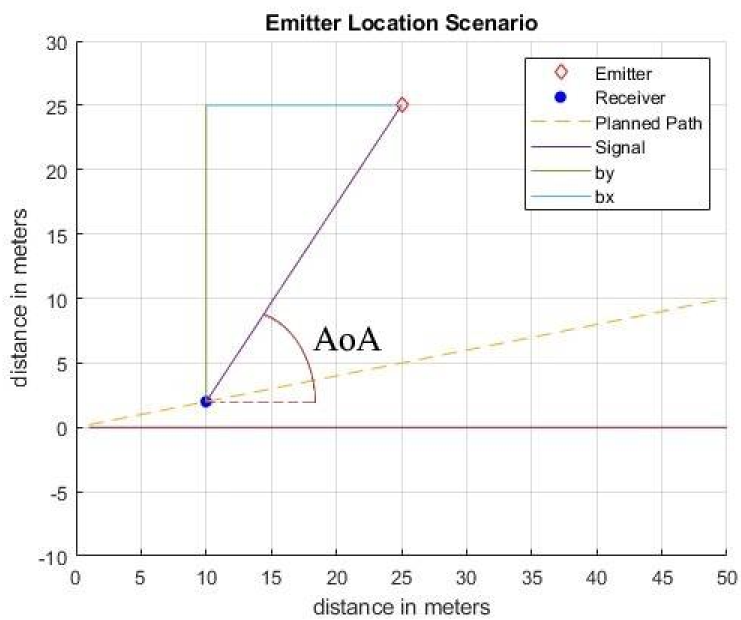


Figure 1.1: Emitter Location Scenario

For practical applications, single point estimates formed this way are not particularly useful without also specifying some measure expectation regarding the accuracy of the estimate. An elliptical bound on the point estimate which encompasses a small area rather than a single point does have merit. As mentioned by Koks [2] and Stone [5], no well-established and tractable closed form mathematical model for the creation of such an ellipse is readily available. These efforts did, however, establish a few pivotal concepts for this thesis. First, they established the idea that simulation-based approaches are a very promising tool for the development of location strategies. The ability to generate, alter, and observe data in simulation allows for a much wider range of testing with only computational costs. Second, prior work borrowed concepts from statistical bootstrapping which, when combined with numerical methods for processing the data, allows for exploration of plausible estimation of a bounding ellipse.

Another important point to make is that the simulation studies done by Koks and Stone did not include a model for outlier data. As this student finds, data that departs significantly from an assumed nominal can have a significant impact on location estimate reliability. In addition to outlier data, a location estimate based on AoA data come with two inherent issues. Firstly, angle measurements are assumed to contain no reliable information related to signal magnitude. This means that a single AoA estimation, even in a noiseless world, could only tell us about the line on which the emitter exists, and not the specific location. Secondly, angle measurements, like all measurements in a physical system, are susceptible to noise. This means that any given AoA measurement has a degree of uncertainty which results in uncertainty about the emitter's true location.

The first problem, a lack of magnitude data, is handled through the process of Cartesian Pseudo-Linear Estimation (CPLE) as detailed below. This is a well-documented and reviewed process [2,3] which removes the need for magnitude data in estimation. It utilizes angle data and position in space and computes an estimate through the clever use of matrix operations.

The second problem, outlier data caused by noise in the systems measurements, can lead to wildly inaccurate estimations. It is imperative that a solution to this issue is found. Application of different outlier mitigation methods can reduce the errant data in a system and improve the results of the bearings only location estimation process. In this thesis we explore three different outlier mitigation methods and compare their merits via simulation studies.

## *1.1 Literature Review*

In this thesis, we draw on work done previously in the area of bearings only localization and statistical resampling, with the purpose of applying previously unstudied methods to well established processes. The concept of Cartesian Pseudo-Linear Estimation (CPLE) was born out of the work described in a paper [3] published in 1984. This concept of an estimator brought the field of bearings only localization forward to the point of needing only angle measurement data. This is the point where estimation tools truly enter relevance for our area of research. The research in the field since has been extensive and has worked toward reduction of error and bias, improvement of the process, and generation of repeatable results [1,2]. This prior work gives us a sound position from which to begin our study of geolocation scenarios.

The area of statistical resampling, and specifically the area of bootstrapping, has been heavily researched. Due to the extensive work done by Efron [7] in establishing the concept, and the further study of Singh [6] the efficacy of bootstrapping has been well established. Even further, the application of bootstrapping toward bearings only localization has been explored in a paper [5] published by Stone and Thompson. This prior work allows us to accept the merits of Bootstrapping as applied to CPLE and gives us the starting point for research.

## *1.2 Outline*

Chapter Two gives a foundation of the necessary knowledge for understanding the concepts explored in this thesis. Cartesian Pseudo-Linear Estimation, Elliptical Estimation, Plausibility Ellipses, bootstrapping, and Outlier Identification and Selection Methods are all addressed.

Chapter Three details the methodology used to apply Outlier Mitigation methods and elliptical estimation to the bearings-only localization problem detailed in this thesis. Three methods are compared, Distance from the Mean, Rate of Change Score, and Best Fit.

Chapter Four covers the results of our experimentation and simulation. It discusses comparison of each method and the merits of each.

Chapter Five contains a summation of our conclusions, discusses necessary assumptions made throughout, and details the possibilities for future work in continuation of this study.

CHAPTER TWO

Background

## *2.1 Cartesian Pseudo-Linear Estimation*

### *2.1.1 Emitter Location Problem*

A normal emitter location scenario involves attempting to locate an emitter from a receiver site. The standard scenario described in this section (shown in figure 1.1), provides a common framework for evaluating outlier mitigation. With this goal in mind, we will be focusing on the situation by which we can most realistically model real-world conditions without adding unnecessary complexity. We begin this scenario description by placing the emitter at a fixed location in cartesian space $(e_x, e_y)$. The receiver is placed on a moving platform (i.e., an airplane) and is assumed to be traveling with constant velocity on a known flight path past the emitter. The position of the receiver at any given time can be referred to as $(f_x, f_y)$. As it travels past the emitter and is within signal range, the receiver is gathering bearing information from the emitter. It uses the bearing data to compute an AoA for each data point gathered. For the idealized (noiseless) case, a straight line drawn from the receiver at a specified angle of arrival it would eventually intersect with the position of the emitter (represented by the signal line in figure 1.1). If this line is drawn for each AoA, the point where any two lines (and consequently all lines) intersect will show the emitters location (shown in figure 2.1). Note that this scenario description represents the problem in two dimensions. It is possible to also have the elevation angle as a part of AoA data; however, the noisy nature of elevation data has resulted in the practice of discarding elevation data for many applications.

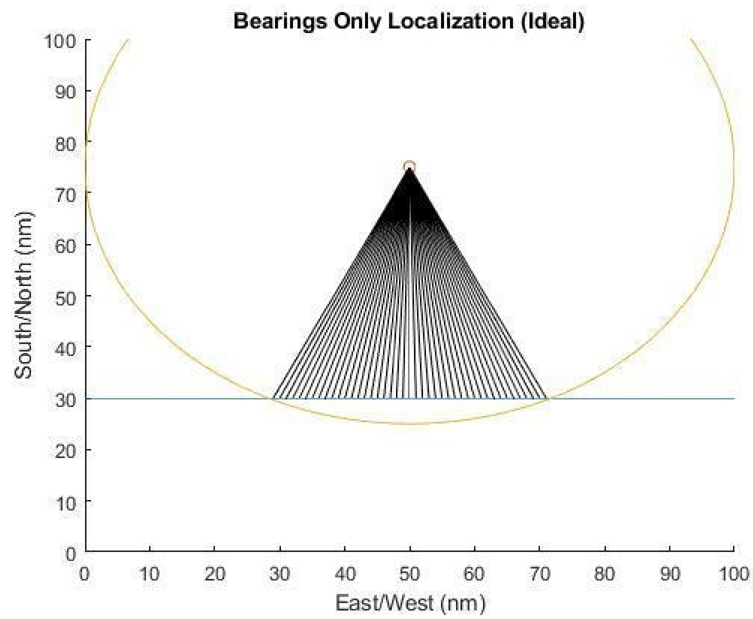


Figure 2.1: Ideal Bearings Only Localization

*2.1.2 Cartesian Pseudo-Linear Estimation (CPLE)*

The intersecting lines method described above, even in a noiseless world, can be a resource intensive process. In the real world where noise affects all measurements, we need a method by which to find an estimate. The CPLE algorithm is a popular approach by which bearing data can be used to generate a two dimensional estimated location $(x)$ of the source of an emission. The process requires only knowledge of the angle of arrival, $\theta_a$, the position in space where the measurement was taken, represented by vector (p), and the heading of the measurement platform at the time the data was collected, $\theta_h$. Using geometry, both angles are adjusted to be represented between -180 degrees and 180 degrees as measured from the x axis. Using equations 2.1 and 2.2 we find the angle of arrival and convert it into bearing data($b$), taking into account its relation to the x axis to determine whether the bearings are positive or negative. Figure 2.2 shows this scenario and its variables.

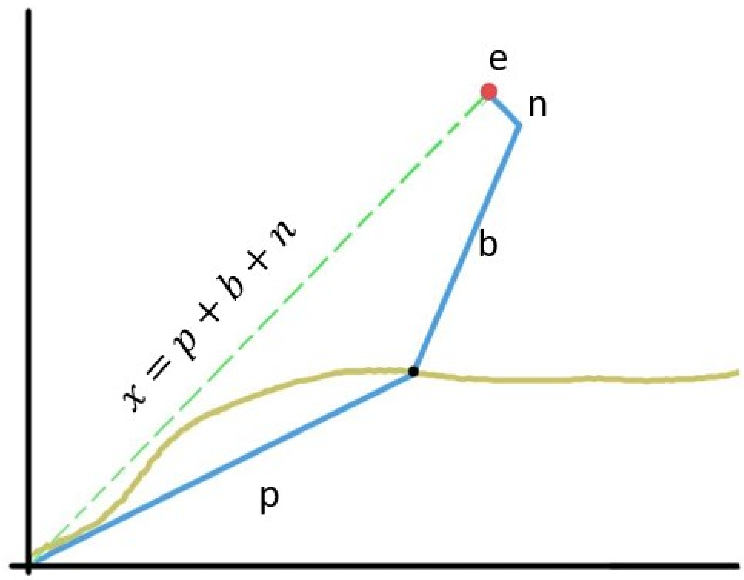


Figure 2.2: CPLE Scenario

$$\theta_n = \theta_h - \theta_a \tag{2.1}$$

$$b_x = \cos(\theta_n)\,; b_y = \sin(\theta_n) \tag{2.2}$$

The bearing data is then converted to unit length and perpendicular bearing vectors $b^{\perp}$ are generated from those unit vectors. This process is repeated for every angle of arrival recorded, and the perpendicular bearing vectors are used to create a matrix, H, where

$$H = \begin{bmatrix} {b_1}^{\perp} \\ \vdots \\ {b_n}^{\perp} \end{bmatrix} \tag{2.3}$$

and matrix z where

$$z = \begin{bmatrix} {b_1}^{\perp} \\ \vdots \\ {b_n}^{\perp} \end{bmatrix} \cdot \begin{bmatrix} p_1 \\ \vdots \\ p_n \end{bmatrix} \tag{2.4}$$

Lastly, we utilize the following derivation, provided by Koks [2] to yield equation 2.7, which is used to generate our location estimate.

$$x = p + b + noise \tag{2.5}$$

$$b^{\perp} \bullet x = b^{\perp} \bullet p + b^{\perp} \bullet b + b^{\perp} \bullet noise \tag{2.6}$$

$$x = H^{\#} \bullet z + noise \tag{2.7}$$

$$H^{\#} = H^{T}(HH^{T})^{-1} \tag{2.8}$$

In the CPLE process, in order to "divide" both sides of equation 2.6 by the matrix H, we are using the Moore-Penrose pseudo-inverse of the matrix and multiplying it by both sides of the equation. The Moore-Penrose pseudo-inverse matrix is also called the general inverse and provides a least-squares solution to a set of linear equations in cases where a unique inverse does not exist. The Moore-Penrose pseudo-inverse is a well known result from linear algebra and in implementation of an algorithm for computing the inverse is included with a standard installation of Matlab. The result means our location estimate is formed by multiplying the pseudo inverse with the z matrix and then accounting for noise. The pseudo inverse, $H^{\#}$, can be represented by equation 2.8.

### *2.2 Elliptical Estimation and Plausibility*

#### *2.2.1 Elliptical Estimation*

An ellipse (figure 2.3) is a set of all points **x** = (x,y) in a plane, the sum of whose distances from two distinct fixed points is a constant. It can be fully parameterized by its center point $(c_x, c_y)$ and a defining matrix, A, containing its orientation and radii information. The ellipse is then defined as the collection of all points, $t_p$, where equation 2.9 holds true.

$$\left(t_p - c\right)' * A * \left(t_p - c\right) \leq 1 \tag{2.9}$$

Elliptical estimation is the process of finding an ellipse, in cartesian space, which contains all chosen (x,y) coordinate pairs and using the parameters of that ellipse to determine the accuracy of a system.

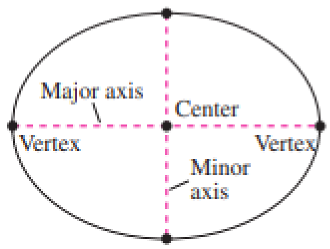


Figure 2.3: Ellipse

When estimating on a set of data, metrics must be selected that provide meaningful insight into the system. The elliptical parameters that represent meaningful metrics in a two-dimensional cartesian coordinate system are major and minor axis length, area inside the ellipse, and center point of the ellipse. The axis length provides some insight into the way the data is distributed, and which dimension has larger variance. The area inside the ellipse tells us how spread out our data is, with a smaller area representing a tighter distribution of points. The center of the ellipse represents the point of estimation.

Ellipses, due to these characteristics, are a good estimator for locations in a real-world space. Finding the minimum size ellipse that contains all data points tells us a lot about the system we're working with. A smaller ellipse means we have a more reliable estimate. In this CPLE based system, major and minor axis lengths do not provide us with a single figure of merit to evaluate the size of a bounding ellipse. Instead, we use the total area contained within the bounding ellipse tells us much more about the accuracy of our system. Furthermore, the ellipse area is easily calculated from the determinate of the matrix A from the ellipse equation given by 2.9 Our goal with an elliptical estimator is to reduce the total area of our bounding ellipse without losing the accuracy provided by the system.

### *2.2.2 Confidence Ellipse, A posteriori Bounding Ellipse, and Plausibility Ellipse*

A confidence interval refers to the statistical likelihood that a data point falls within a specified boundary. Confidence intervals are defined by the percentage chance that represents that likelihood. For instance, referring to a set of values as a 95% confidence interval would mean that there is a 95% probability (calculated from an assumed distribution model) that a given piece of data falls within that value set. In the described geolocation scenario, if we want a confidence interval of a certain percentage, what we're looking for is a system that will yield a bounding ellipse containing the emitter that percentage of the time.

In traditional statistics, confidence intervals can be computed using well-known methods and are also traditional applied to unbiased estimators. In our work, we do not want to mislead the reader by describing the bounding ellipses that use as a performance measure as "confidence" ellipses. One well-known problem with CPLE estimation is that it does not result in an unbiased estimate. The simulations we undertake in this study are more accurately described as "ground-truthing". In a ground-truth study, algorithms can be evaluated by comparing location estimates with ground-truthed (known) locations. We can account for a small degree of bias in our estimator by simply enlarging the bounding ellipse until the known location of the emitter is encapsulated. However, the resulting bounding ellipse cannot be considered a confidence ellipse because the location of the emitter was used in the calculation. We refer to a bounding ellipse found in this way as an a posteriori bounding ellipse to make it clear that knowledge of the emitter location is required for calculating this metric.

Our work offers the possibility of developing rules of thumb for determining bounding ellipses that are in some sense "likely" to cover the emitter location. For example, simulation runs could justify choosing a bounding ellipse scale factor (which will be discussed in greater detail in Chapter 3) without knowledge of the emitter location. This is closer to traditional concept of a confidence interval, however, given that the CPLE is biased, and we do not have a statistical model to accurately account for the estimator bias – we are reluctant to use the term "confidence" ellipse and instead use the word "plausibility" with the goal of correctly representing this uncertainty.

## *2.3 Resampling*

### *2.3.1 Bootstrapping*

"Bootstrapping is a statistical procedure that resamples a single dataset to create many simulated samples. This process allows you to calculate standard errors, construct confidence intervals, and perform hypothesis testing" [8]. Put simply, it's using random sampling, with replacement, to select data points from a given set. These data points are placed into a new set with a predetermined population size. Due to the sampling being random and done with replacement, the new data set does not require knowledge of the statistical parameters of the original set. [6]

In the described geolocation scenario, gathering of data is very expensive in terms of resources and time. An airplane flying past an emitter for the sake of data gathering introduces real world barriers into the equation. Bootstrapping allows us to generate a collection of point estimates that help us establish the variation we can expect for the location estimate. For instance, if our moving platform passes through the range of an emitter long enough to gather N data samples, we could generate a single location estimate from that data. Each repetition of the process yields a new estimated location and we're left with a cloud of estimates to work with.

### *2.4 Outlier Identification Methods*

When observing a set of data, a point that differs widely from the rest of the set can be considered an outlier. In real world conditions, without a sound, well fitted model for noise, outliers are always going to exist. These outliers, if left unaddressed, lead to inaccuracy in modeling and understanding of systems. An exploration of different outlier identification methods, and subsequent removal of those outliers, can lead to stronger correlation in post-process data and better models of real-world systems. Outliers are often obvious to the human eye when visualized along with the whole of a data set. It's much more difficult to automate a process which recognizes outliers and removes them with consistent results.

In our case, the data we're examining is received in the form of angles. These angles represent the angle between the x axis and a line drawn between two points A and B. A represents a stationary point, and B represents a point that is moving through space along a known path at a known speed. These angles are listed in chronological order and are obtained at uniform intervals. Given all that we know about the system, the rate at which the angle data changes should follow a trend. When noise is introduced to the system, the deterministic relationships in our data are broken. By observing which pieces of data stray the furthest from that deterministic relationship, we can identify outliers from the set.

### *2.4.1 Distance from the Mean*

The first such method for outlier identification that we'll explore is a Distance from the Mean (DftM) approach, as shown in figure 2.4.

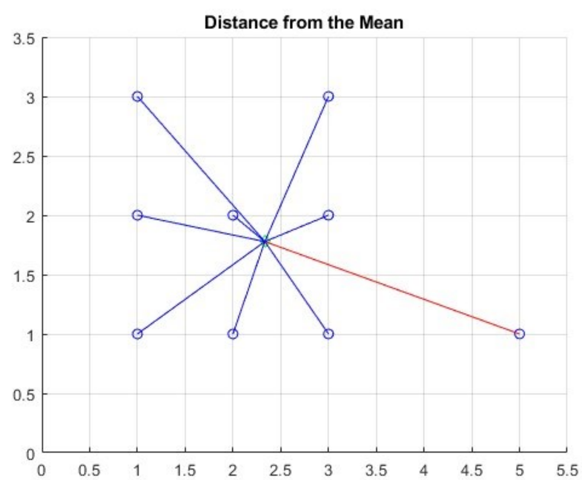


Figure 2.4: Distance from the Mean Example

If we take our given set of data and remove our prior knowledge of how it was obtained, this method can help to remove data points that stand out from the set. In our scenario, the data is a set of cartesian (x,y) pairs representing location estimates. Taking the mean of the x values in the set, and the mean of the y values in the set provides us with a cartesian pair $(x_m, y_m)$. We can then calculate the distance between each individual (x,y) pair, and our mean. After doing this, we can sort the (x,y) pairs based on their distance from the mean, and ultimately remove a predetermined percentage of them with the furthest distance from the set's mean. This process is shown in figure 2.4 with each open circle representing a point, the blue lines representing the distance between each point and the mean, and the red line representing the furthest from the mean.

### *2.4.2 Rate of Change Score*

Another method for outlier identification which we'll explore is the Rate of Change Score (RoCS) comparison approach. This method requires some knowledge about the data set under observation. Specifically, we must know that the data being received is provided in chronological order and at uniform intervals. If the data being received meets these criteria, then we can use a few known quantities from experimentation to remove outliers in our system.

First, we move through the entire set, looking at each individual angle. We compare it to the data point before it (accounting for geometry) and establish the rate of change between them. We repeat this process by comparing the data point to the point after it. We average these two values and assign a Rate of Change Score to each individual data point. We can sort based on this score and remove a percentage of the worst scoring data.

### *2.4.3 Best Fit Comparison*

The final method we'll explore pays attention to trends in data as a whole. The Best Fit Comparison, much like the rate of change method, takes into account flight path, data polling intervals, and changing patterns in angle of arrival.  Using this data, we can fit a curve that represents the trend of the angles as we move through time. Next, we can calculate which data points are the furthest from the established trend and assign those data points a score based on their distance. We can then sort the set by this score and remove a predetermined percentage of them with the highest score. In doing this, we remove the data points furthest from the expectation, which should remove the points that have been affected most heavily by noise.

### *2.5 Noise Modeling inspired by Epsilon Contamination*

As mentioned in Poor et al. [10], the Epsilon Contamination noise model is often used for robustness studies. The model assumes a pdf of the form

$$f(x) = (1 - \varepsilon) * f_0(x) + \varepsilon * h(x) \tag{2.10}$$

where $f_0$ represents the nominal pdf and h denotes an arbitrary pdf. This model is popular because the parameter epsilon allows one to model a given percentage of unmodeled data. For example, let epsilon=0.1 could be reasonably interpreted as having 10% of the data be a poor fit with the nominal. It should be noted that for simulation studies, we are not implementing a true epsilon contamination model because of the eimpracticality of representing an arbitrary pdf for h(x). We approximate this model by using a normal distribution for h with a much larger standard deviation with respect to the nominal noise model.

### *2.6 Example*

To illustrate the best-fit model consider the following example. Consider a person walking past a clock tower. They will travel along a known path and calculate the angle between themself and the tower at pre planned intervals. We place the clock tower at a specific cartesian coordinate $(C_x, C_y)$ and treat it as a single point. The person's location at any time can be denoted as $(P_x, P_y)$. The angle between the person and the tower can be calculated using equation 2.11 and using simple geometry we can adjust the angles, so they're all represented as values between +/- 180 degrees from the x axis.

$$atan\left(abs\left(\frac{C_y - P_y}{C_x - P_x}\right)\right) \tag{2.11}$$

We set the person's starting location as the origin and have them walk in a straight-line due east along the x axis for 50 meters and place the clock tower at the location 25 meters north, and 25 meters east of the person's start. If they stop to calculate the angle between themselves and the tower every 5 meters, they should get the results shown in Table 2.1

Table 2.1: Ideal Angles of Arrival

| (x,y) | 0,0 | 5,0 | 10,0 | 15,0 | 20,0* | 25,0 | 30,0 | 35,0* | 40,0 | 45,0 | 50,0 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\theta$ | 45 | 51.34 | 59.03 | 68.19 | 78.69 | 90 | 101.30 | 111.80 | 120.96 | 128.65 | 135 |

This data, due to its noiseless nature, is easy to apply a line of best fit to. The ideal best fit line is represented by equation 2.12.

$$2.6954e^{-20}x^4 - 6.9288e^{-4}x^3 + .052x^2 + .9272x + 45.1686 \tag{2.12}$$

In the real world, however, data can't be measured noiselessly. If we apply our epsilon-contamination-based noise model to these 11 measurements, we can arrive at a more realistic picture of these measurements. We'll assume that 20 percent of these measurements are subjected to uniformly distributed noise with zero mean and a standard deviation of 30. The other 80 percent are subject to uniformly distributed zero mean noise with a standard deviation of .06. Applying these constraints, the person's data could be represented by the row denoted as "Angle" in table 2.2**,** and a 4th order fitting of that data would yield equation 2.13.

$$9.5064e^{-5}x^4 - .01x^3 + .2977x^2 - .0561x + 44.0892 \tag{2.13}$$

In order to utilize the best fit outliers removal method, we now must calculate what the best fit value is at each measurement point and compare it to the measured data. The values which differ by the largest amount are then selected as outliers and removed. Knowing our noise model, we've chosen to remove 25 percent of received data. Table 2.2 contains the data from this process.

Table 2.2: Outlier Selection for Clocktower Example

| (x,y) | 0,0 | 5,0 | 10,0 | 15,0 | 20,0* | 25,0 | 30,0 | 35,0* | 40,0 | 45,0 | 50,0 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| $\theta$ | 44.9 | 51.42 | 59.13 | 68.22 | 137.22 | 89.94 | 101.30 | 140.73 | 120.85 | 128.63 | 134.95 |
| Best Fit | 44.0 | 50.06 | 64.29 | 81.45 | 97.64 | 110.39 | 118.65 | 122.81 | 124.67 | 127.46 | 135.84 |
| Diff | .85 | 1.35 | 5.16 | 13.23 | 39.58** | 20.45** | 17.35 | 17.92** | 3.81 | 1.17 | .88 |

* Known Outlier
**Selected Outlier

After removing our selected outliers we are left with a data set much closer to the ideal scenario. Utilizing CPLE on this data set will yield an estimate closer to the tower's true position. If we utilize the process detailed by equation 2.7 and apply it to the noisy angle bearings, we get an estimate of (22.6371,18.3731). The distance from the true clock tower location is 4.264m. If we apply the same equations to the data after removing the selected outliers, we get an estimate of (25.0035,25.0359). The distance from the true clock tower location to our estimate is .0324m, a marked improvement.

This illustration, while effective, fails to show the usefulness of two of the presented concepts. As such, we need to expand our scope. To continue, we change our clock tower's location to be (50,50). In addition, we'll have the person walk 100 meters due east and take measurements every meter. This will yield 101 data points, a much larger sample size. Assuming that we have a target sample size of 300 data points, we're still 199 short. Repeating this process would be very time and effort intensive. Instead, we can apply the bootstrap resampling methodology detailed above. We'll use the same epsilon contamination based model as described above, and subject 20 percent of our data to a large error. That means that 20 of our 101 data points can be assumed to be outliers. We'll select and remove the worst 25 percent of the data through application of a best fit line as shown in figure 2.5, leaving us with 76 samples.

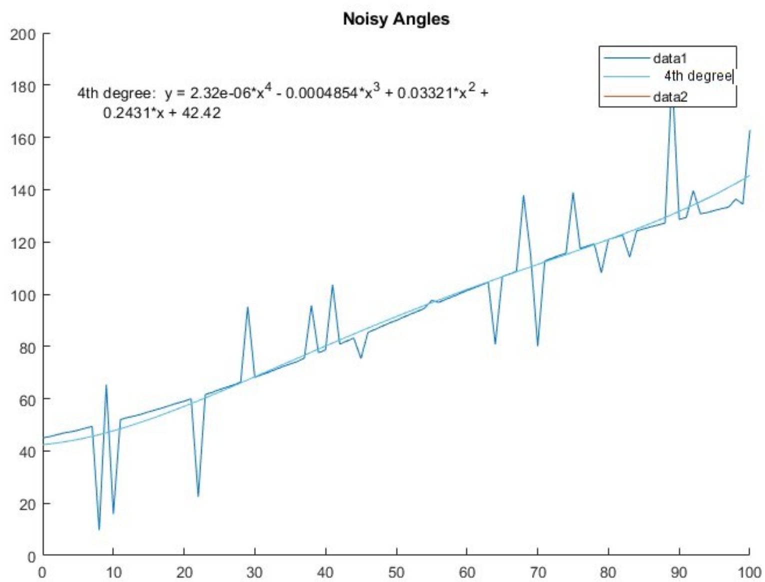


Figure 2.5: Noisy Angle Example

We then choose 300 samples, with replacement, from our smaller, less error prone, pool. We can apply CPLE to these resampled points and get an estimated location. Rather than assuming our new distribution of points is perfect, we can repeat the bootstrapping process a thousand times with each bootstrap yielding an estimated value. After doing this, we have a very large sample size of location estimates and can apply elliptical estimation as shown in figure 2.6.

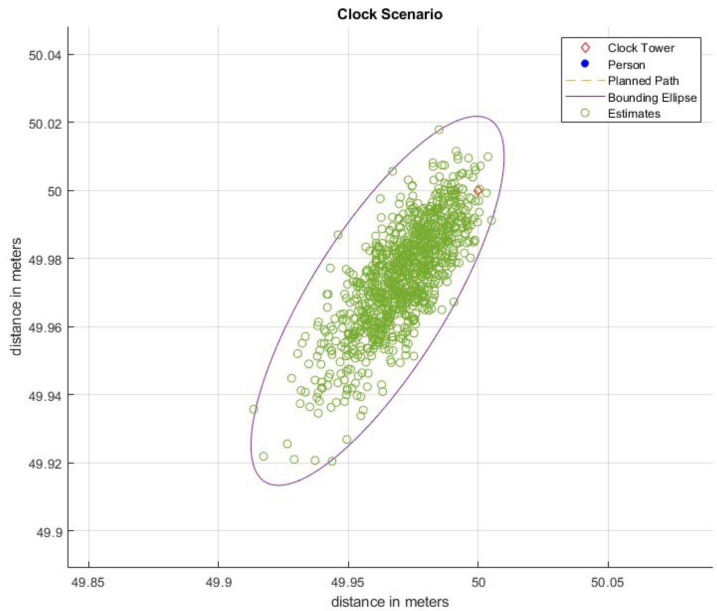


Figure 2.6 Clocktower Example Ellipse

It's clear from our results that any single estimate would have been inadequate. However, if we find the smallest bounding ellipse that contains all the location estimates we can know with some degree of confidence that the true location of the clock tower is inside of that ellipse.

# CHAPTER THREE

## Outlier Method Comparison via CPLE

### *3.1 Design of Experiments and Explanation of Assumptions*

As mentioned above, the process of measuring angle of arrival data using real emitters, receivers, and an airplane across even one flight is expensive and therefore it is impractical to wholly rely on ground truthing to assess system performance. Furthermore, accurate simulation models can be used in conjunction with operation data to produce more accurate assessments of target locations. The simulations detailed below are designed to emulate scenarios and provide benefits beyond simply cost, as we describe below.

First, the emitter can be placed at any point within cartesian space. In real data measurements, the ability to poll data and the effect of noise on that data are dictated by the emitter's signal power, and the ratio of that power to the strength of the noise. As the receiver is moved further from the emitter the reliability of its measurements lowers significantly. With a simulation, we can control the Signal-to-Noise Ratio (SNR) and generate useful data at any range. This allows us to test our system based on any flightpath, any angle between that path and emitter location, and any noise model.

Second, the characteristics of the receiver and the flight platform can be altered with high fidelity. In a real-world scenario, the physical properties of a piece of hardware are set. The number of measurements taken in one flight are governed by the speed of the flight platform, the polling speed of the receiver, and the range of the emitter. In our simulation the polling speed of the receiver can be controlled by changing the resolution of the flight path function. We can effectively control any change in flight speed, any number of measurements, and any receiver characteristics. This allows us to verify the robustness of the outlier mitigation while remaining agnostic to real-world systems.

Third, we can control the flight path of the platform without placing any of the physical constraints of an airplane against it. The physical properties of flight are removed as constraints which allows us to focus on generation of meaningful and usable data as we test various operational hypotheses. In other words, we can generate data representative of multiple flights, multiple planes, or multiple receivers without worrying about what it would require to do that in the real world.

It's important to note that the true emitter location is used in two places in our calculations. First, we use it in generation of the noiseless AoA and bearing data. Drawing a straight line between the receiver and the emitter requires knowledge of the emitter's location. After doing this, noise (which is emitter location agnostic) is applied to the system and true emitter location is no longer known by the system. Second, the emitter's true location is used for analysis of results. After our system makes an elliptical estimate, we use the true emitter location to test whether it is inside of the drawn ellipse.

It's also important to note that our noise modeling and outlier selection methods are treated as separate entities throughout the simulation. When using the Epsilon Contamination based model described in section 2.5, we assume that a certain percentage of our data will be subject to much stronger noise. In practical systems, outlier dta occurs due to factors such as multipath and numerical sensitivity of the angle estimates. In our system, the AoAs being selected this way are chosen at random. The system does store which points are chosen, but only for diagnostic purposes. These values are never used in the identification of outliers, but rather in analysis of the efficacy of our methodology. With these assumptions in place, we can understand why and how the experiments detailed below were designed.

### *3.2 Distance from the Mean*

The first outlier removal method tested was the Distance from the Mean method as described in section 2.4.1. This method was initially chosen due to its simplicity, computationally and logically. The flow of the process is shown in figure 3.10.

We illustrate this approach with an example that defines a "standard scenario". First, we must define relevant variables for our system. We'll represent our flight path ($fp$) as a matrix containing the cartesian (x,y) pairs of the flight path as it moves through space. We'll choose a horizontal flightpath 100 nautical miles (nm) long, with data taken every nm. We'll place our emitter at a location ($e_x, e_y$) central to the path and located above it. All data is represented as distance in nm from the start of the flightpath at position (0,0).

$$fp = \begin{bmatrix} fp_x \\ fp_y \end{bmatrix} = \begin{bmatrix} 0 & 1 & \cdots & 100 \\ 0 & 0 & \dots & 0 \end{bmatrix} \tag{3.1}$$

$$(e_x, e_y) = (50,50)$$

We also need to define our noise parameters. In this scenario, we will use a contamination percentage ($\varepsilon$) of 30. This means that 30 percent of our data will be subjected to strong noise, and the other 70 percent will be subjected to noise with much lower standard deviation. This scenario is shown in figure 3.1 with lines selected as outliers plotted in red, and the emitter location shown as a green diamond. We'll refer to this set of flight path, emitter location, and noise parameters as the standard scenario.

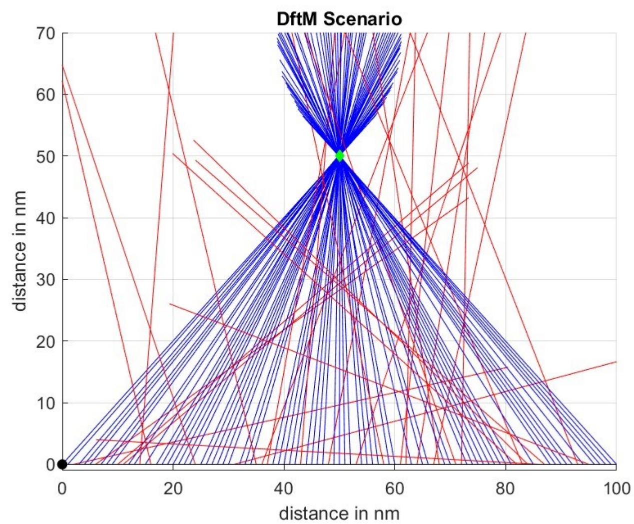


Figure 3.1: Distance from the Mean Scenario

This process yields 101 AoA measurements. We must then bootstrap across this data to fill out a set of data with our desired population size. In this scenario, our desired population size is 300. This value was chosen as it would be representative of 3 receivers flying through the space along the same path. Doing this yields a set of 300 AoA measurements. If we use CPLE on this bearing set, we get an estimated location of (50.2504, 38.6026).

We then repeat this process 1000 times. 1000 runs means 1000 sets of resampling on this same noisy data to generate a cloud of location estimates. This cloud and the ellipse shown in figure 3.2 are the result of using all the gathered data, good and bad, to estimate.

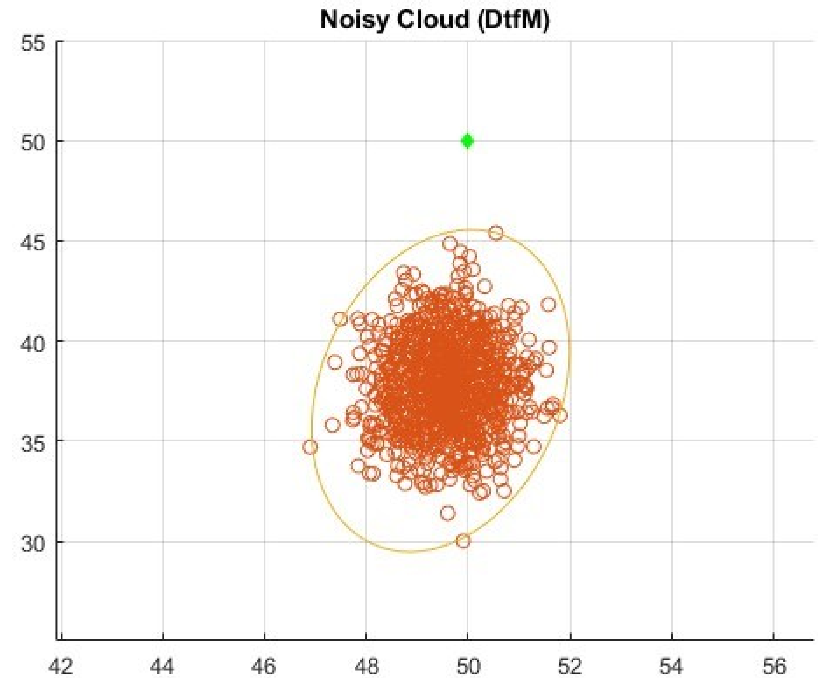


Figure 3.2: Noisy Cloud (DftM)

Now, we can utilize our Distance from the Mean method to try to reduce the size of our ellipse. First, we find the mean of the estimate set $(x_m, y_m)$ to be (49.5812, 37.8537). We then calculate the cartesian distance between every (x,y) pair and $(x_m, y_m)$. Since we do have knowledge of our noise model, we're going to remove a percentage of our data equal to our chosen $\varepsilon$. We sort across this set of distances to find the points furthest from the mean and remove them from the elliptical estimation. Figure 3.3 shows the result of that process.

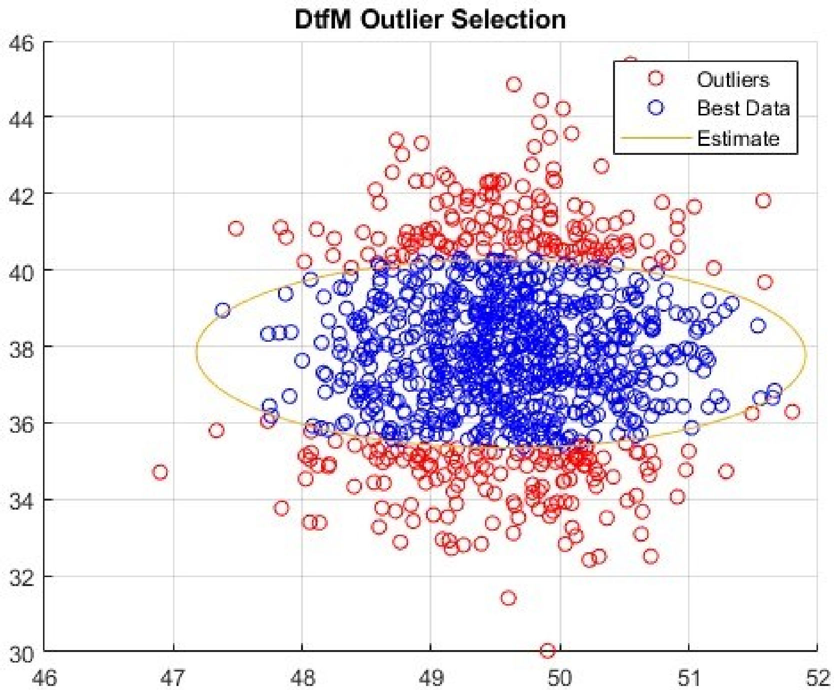


Figure 3.3: DftM Outlier Selection

We have clearly reduced the area of our ellipse giving us a smaller target area, but we haven't improved the precision of our system.

There is, however, a scenario where this method can be made to be effective. To show this must we reduce the distances we're working with and our chosen $\varepsilon$ down so that the SNR is much higher. We set our $(e_x, e_y)$ to be (50,30) and choose an $\varepsilon$ value of 5. We repeat the rest of the process detailed above. Figure 3.4 shows the resulting estimate cloud and ellipse.

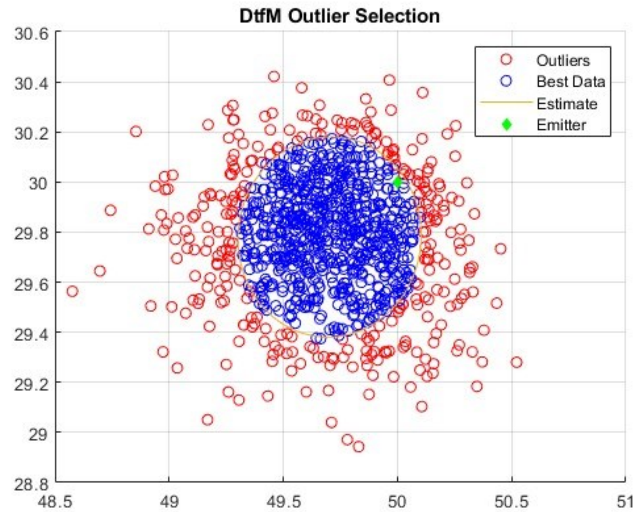


Figure 3.4: Ideal DftM Outlier Selection

*3.3 Rate of Change Score*

The second outlier removal method tested was the Rate of Change Score method as described in section 2.4.2. This method was chosen after the Distance from the Mean method, as the relationships between data points became obviously relevant. The flow of the process is shown in figure 3.11.

*3.3.1 Standard Scenario*

We will maintain most of the variables from the last experiment as listed here. The flight path matrix ($fp$) will remain unchanged, as it allows us to compare the performance of this method to the one before it more directly. Our polling frequency will remain at one sample per nm. We'll leave our emitter at position (50,50). All data is represented as distance in nm from the start of the flightpath at position (0,0). We'll leave our noise parameters unchanged as well, so $\varepsilon = 30$. With these parameters in place, we can refer to figure 3.1 to see the layout of the scenario.

This, however, is where our method departs from the Distance from the mean method. Given the lessons we learned from the first method we know that we need to perform analysis on the data before it enters our CPLE calculations. We're starting with 101 pieces of angle of arrival data. We know from our noise model that 30 of them have been subjected to very strong noise, and that the rest have been subjected to noise of a lower power level. With our receiver moving at a constant speed along the path we would expect a smooth transition from angle to angle. This means we should be able to identify which measurements have been subjected to the worst noise data by looking at the measurements as they compare to the ones around them. First, we'll follow the process defined in section 2.4.2. We subtract the AoA at each position from the AoA at the next position. We then subtract each AoA from the AoA at the position before it. We take the average of the values and then compute the absolute value. This new value, computed via equation 3.2, is what we refer to as the Rate of Change Score (RoCS).

$$RoCS_i = abs\left(\frac{(AoA_i - AoA_{i+1}) + (AoA_i - AoA_{i-1})}{2}\right) \tag{3.2}$$

We then sort our data based on RoCS. As mentioned above, we know we want to remove the worst 30 percent of our data. For a set of 101 data points, that means were remove the 30 data points with the highest RoCS. After doing this, we're left with 71 data points which have been tagged as usable.

We have the same desired population size of 300, so we bootstrap across the data to determine the population we will be taking a location estimate from. Doing this will yield a single estimate point in space. We can now repeat the process 1000 times and generate an estimate cloud. We apply elliptical estimation to this cloud and note the resulting ellipse. In order to known how well our algorithm works, we must repeat these same steps on data that has not been treated by outlier removal. We follow the same steps in this case of calculating the location estimates and corresponding ellipse, and then storing the statistics for that ellipse for later comparison to our treated results.

### *3.3.2 Curved Flight Path*

In order to analyze the efficacy of an outlier selection method. It's unrealistic to believe that our receiver would always follow a straight-line flight path through the emitters range. In this example, we will maintain our emitter location and noise modeling from the prior experiments but change our flight path. We'll repeat the same tests across a new flight path ($fp_2$) and note the results. The new flightpath can be parameterized by equation 3.3 and 3.4 and is shown in figure 3.5 where the red lines represent outlier data points, and blue lines represent useable data.

$$fp_{x2} = [0 \quad 1 \quad \dots \quad 100] \tag{3.3}$$

$$fp_{y2} = -.01(fp_{x2} - 50)^2 + 25 \tag{3.4}$$

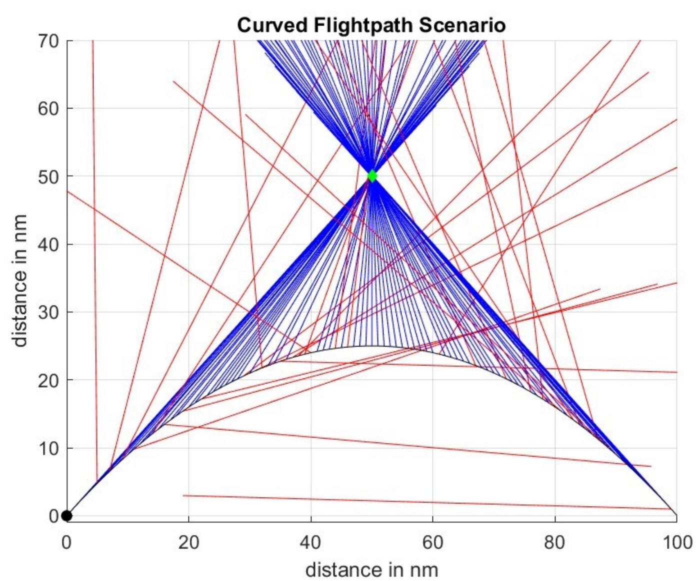


Figure 3.5: Curved Flightpath Scenario

*3.3.3 Emitter Proximity*

Lastly, we utilize our initial flightpath but change the emitter location 3 times so it's nearer to the flightpath, further from the flightpath, and off center. We'll utilize the locations $(e_x, e_y) = (20,35), (50,10), and\ (50, 75)$ as shown in figure 3.6 and repeat our same set of tests. In doing this we can test our algorithm's efficacy as it relates to the emitter's proximity to the flight path. This experiment, in tandem with the two before, provides a good picture of the typical performance of our outlier removal method.

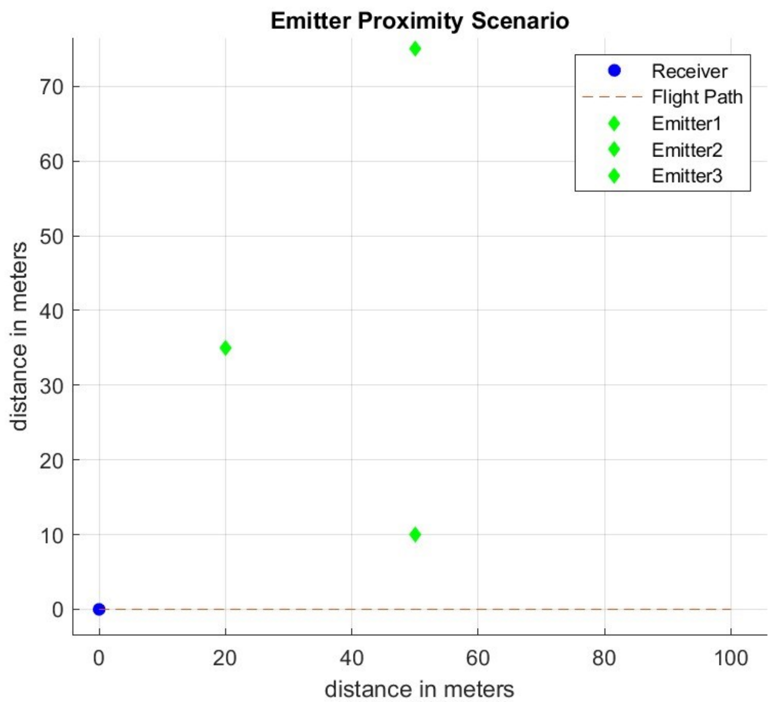


Figure 3.6: Emitter Proximity Scenario

*3.4 Best Fit Comparison*

The final outlier removal method tested was the Best Fit Comparison method as described in section 2.4.3. The inception of the idea came from the shortcomings of the Rate of Change score method. Non-linear flight paths mean that the relationship between data points may not be as easily defined. Local peaks, multiple outliers in a row, and other characteristics of a more general system necessitated advancement of the selection method. The novelty of this method is that it pays attention to trends in the data. The flow of the process is shown in figure 3.12. For the same reasons listed above, we'll be performing analysis on the data before it enters our CPLE calculations. Our goal with this method is to improve the outlier selection model being used in the process.

### *3.4.1 Standard Scenario*

For the first tests we will again maintain most of the variables from the first experiment as listed here. The flight path matrix ($fp$) will remain a constant speed straight line, with fy = 0 at every position. Our polling frequency will remain at one sample per nm. We'll leave our emitter at position (50,50). All data is represented as distance in nm from the start of the flightpath at position (0,0). We'll leave our noise parameters unchanged as well, so $\varepsilon$ = 30. With these parameters in place, we can again refer to figure 3.1 to see the layout of the scenario.

After generating a set of ideal angles of arrival, we utilize the same noise model described in section 3.2 to apply noise to them. Figure 3.7 shows the resultant noisy angles plotted in chronological order. It also shows the ideal angles plotted in chronological order. Reviewing this noisy plot, it is usually clear to the eye where outlier data lies, but calculating this automatically can be challenging. This is where we can utilize the Best Fit Comparison Algorithm.

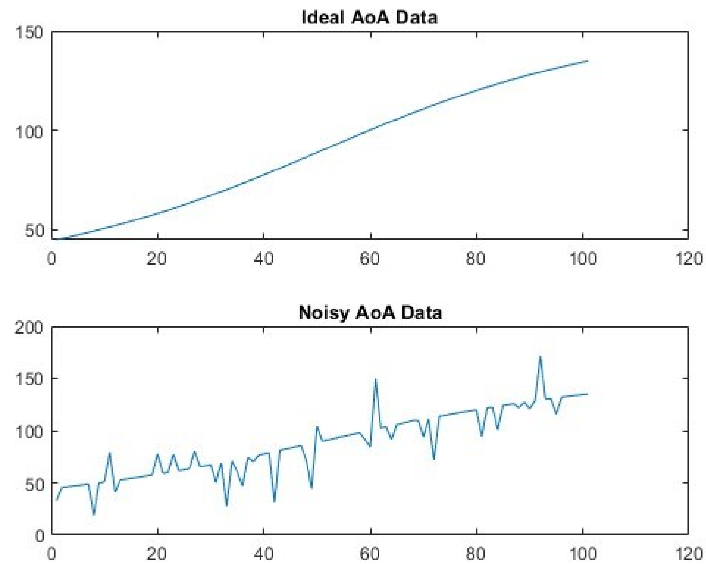


Figure 3.7: Noisy vs Ideal Angles

We find a line of best fit going through our noisy AoA data as shown in figure 3.8. Preliminary testing led to using a 4th order polynomial fit as it accounts for curved and straight flightpaths without being overly sensitive to outlier data.

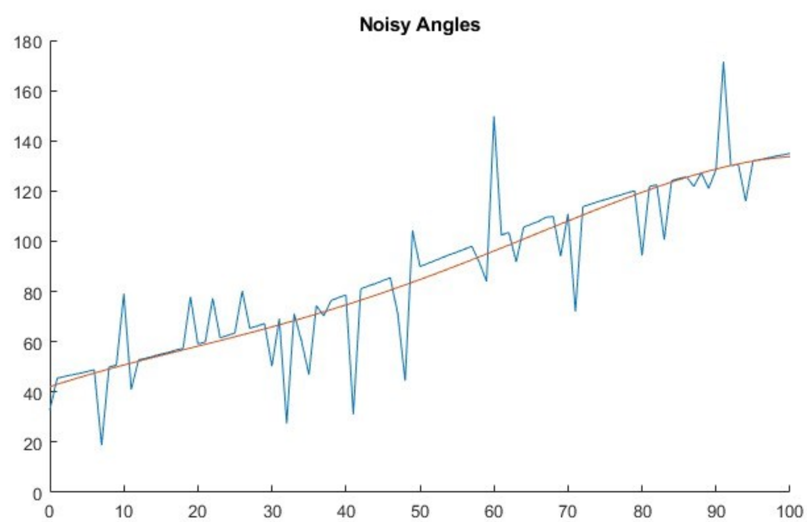


Figure 3.8: Noisy Line of Best Fit

Using this polynomial fit we can then calculate what the “best fit value” at each position along the flightpath would be and compare it to the measured AoA value by subtracting the measured AoA from the best fit AoA and taking the absolute value of the result. We then sort the set of results and remove a determined percentage of the top values. The results of that process are shown in figure 3.9.

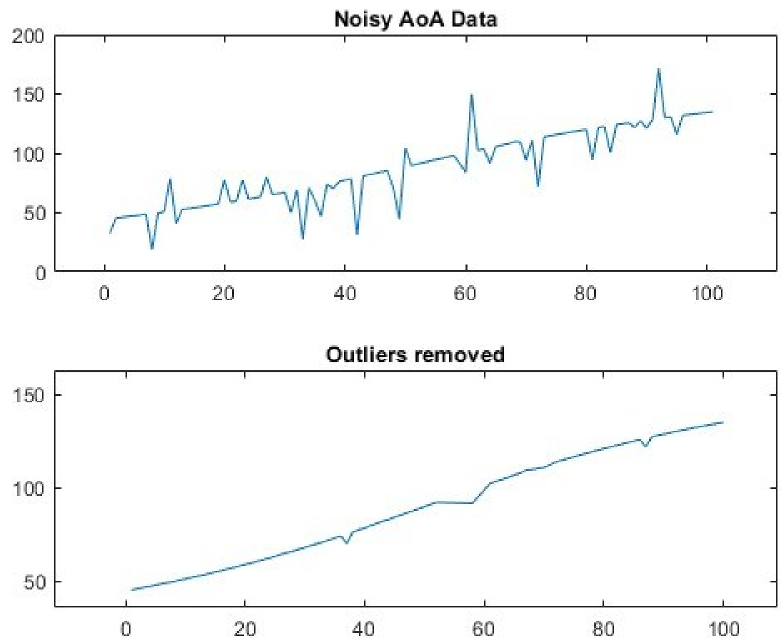


Figure 3.9: Noisy AoA Data vs Mitigated Data

This process leaves us with 71 good data points. We bootstrap across those data points until we have reached our desired population. The resulting AoA data can then be used to calculate a location estimate via CPLE. We repeat this bootstrap 1000 times, to generate 1000 unique location estimates and draw an ellipse around those points.

In order to known how well our algorithm works, we must repeat these same steps on data that has not been treated. We calculate location estimates, draw an ellipse, and store the statistics on that ellipse for later comparison to our treated results.

We then repeat this entire process 1000 times. We generate new noise values, a new line of best fit, 1000 new location estimates and a new ellipse each time and we store the results of each run for analysis.

### *3.4.2 Curved Flightpath*

Our second experiment will be done with the exact same process described above, with one change. We will utilize a curved flight path with the same parameters as the curved flightpath experiment from section 3.3. The scenario is shown in figure 3.5. In repeating the steps of our prior experiment, we're able to draw lines of direct comparison between the methods being tested. We will utilize our Best Fit Comparison algorithm to remove outliers before calculating location estimates, draw an ellipse around those estimates and store the relevant data that results from this process.

### *3.4.3 Emitter Proximity*

The third experiment performed will be done with the same parameters as the emitter proximity experiment from section 3.3.3. The scenario is shown in figure 3.6 above. This will allow us to look at the effect of emitter location on the effectiveness of our algorithm, and then compare the results of the different trials to gain insight into the overall performance of the Best Fit Comparison method.

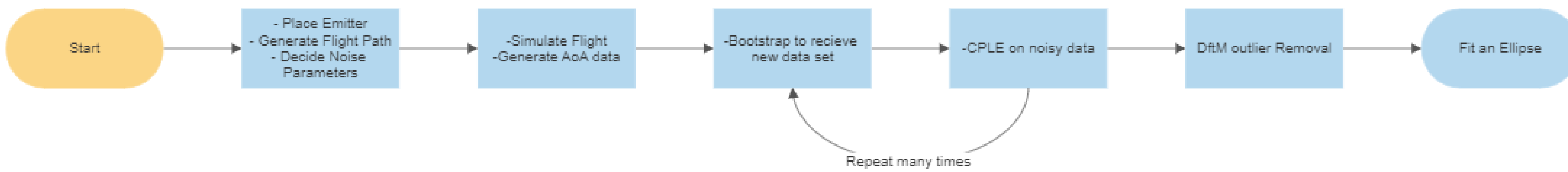


Figure 3.10: DftM Process

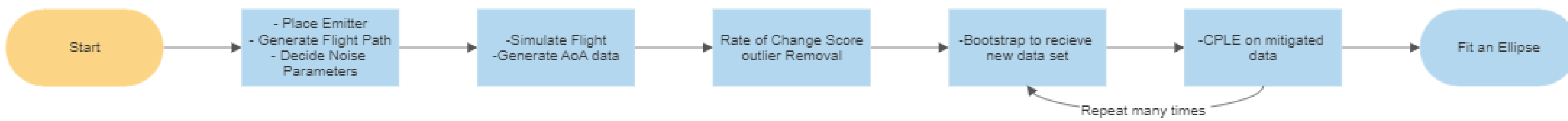


Figure 3.11: RoCS Process


Figure 3.12 Best Fit Process

# CHAPTER FOUR

## Results

### *4.1 Discussion of Metrics*

To gather meaningful results, we need to repeat the processes described above many times. Each experiment in this trial was run 1000 times to verify that results could be counted as typical. Across each run, important metric data was stored which allows us to characterize the performance of each outlier mitigation method.

The first metric worth discussing is the total area in our bounding ellipse. For each run we can calculate the area inside the ellipse formed before outlier mitigations are done and then calculate the area of the new ellipse formed around the mitigated data. The percent reduction as well as the actual magnitude of the reduction provides a useful comparison of methods. When working in a scale of nautical miles, even a small percentage reduction can represent a very large area in space. It is important to note that this doesn't tell us much about the precision (proximity of estimate to target) of our system but does give us a view into improving the estimate bounds.

The second and possibly most interesting metric we tracked is the scaling factor. After completing our mitigation methods and finding a new bounding ellipse, we want to use a "ground truth" type of approach to assess how often the emitter location is inside of the plausibility ellipse. To do that, we need to consider the geometry of an ellipse. For all (x,y) pairs inside an ellipse with center c and defining matrix A we know that the solution to equation 4.1 will be less than or equal to one. If we set $t_p = (e_x, e_y)$ then we can tell just from the solution of the equation where the emitter lies.

$$SF = \left(t_p - c\right)' * A * \left(t_p - c\right) \tag{4.1}$$

Because of this geometric principle, SF is the value we would have to scale the area of the ellipse by to place the emitter directly on the bounding line. Tracking this across the different trials allows us to judge the precision of our system. When discussing scale factor, we reference a metric we call SF95. SF95, in this case, is the scale factor required to include the emitter inside the estimate ellipse 95% of the time. If you were to scale the result of the CPLE process by this factor you should include the emitters true location in 95% of cases. Figure 4.1 shows an example of this concept.

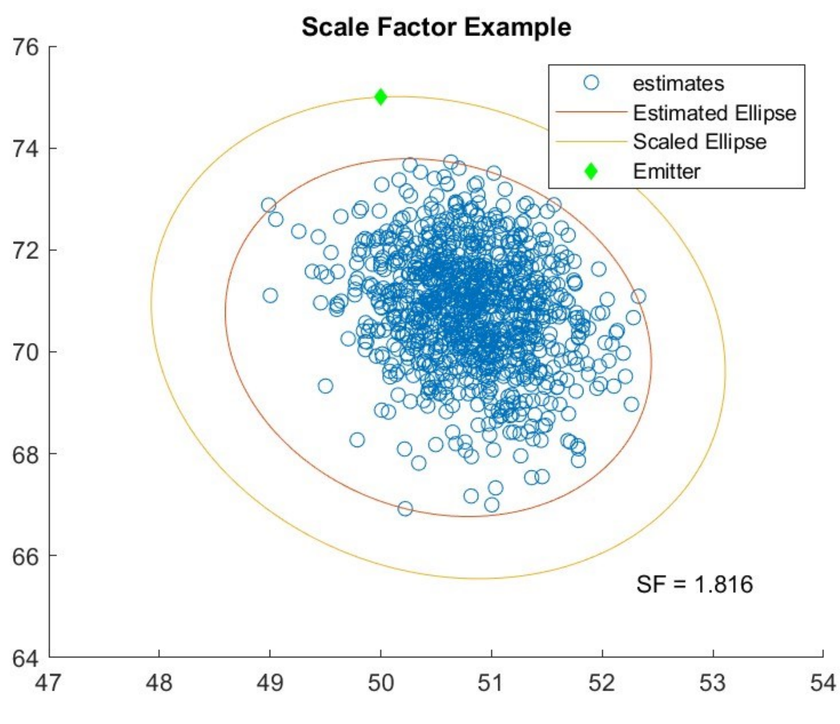


Figure 4.1: Scale Factor Example

The third metric we tracked is distance from the center of our estimating ellipse to the true location of the emitter. While this doesn't tell us whether our estimating ellipse bounds the emitter, it does give us meaningful insight into the precision of our system. This, in tandem with our other metrics, paints a strong picture of the performance of our designed systems.

### *4.2 Results of CPLE with No Outlier Mitigation*

In order to give the results of our trials meaning, we need to establish the baseline performance of the CPLE process without implementation of an outlier mitigation method. The three scenarios described above, Standard scenario, curved flightpath, and emitter proximity, were all tested with raw, untreated data and the results are presented below.

#### *4.2.1 Standard Scenario*

When working with a straight-line flight path located along the line fy = 0 and an emitter location of (50,50) the results of the untreated data across 1000 runs leave much to be desired. The mean area of the resulting ellipse was 50.126 nm^2. The mean scale factor required to include the emitter in the estimate was 4.554, with a max of 13.232 and a minimum of 1.039. The mean distance from the ellipse center to emitter the emitter's true location was 12.150 nm. Figure 4.2 shows the results of a typical run through the standard scenario without outlier mitigation.

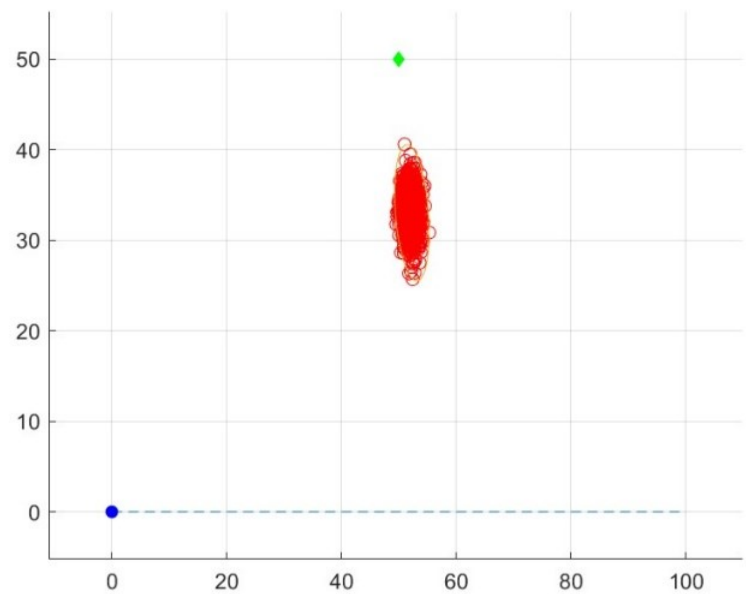


Figure 4.2: Typical Result for Standard Scenario with No Mitigation

*4.2.2 Curved Flightpath*

When working with a curved flight path located along the line $fy = -.01 * (fx - 50)^2 + 25$, the results of the untreated data across 1000 runs are lacking. The mean area of the resulting ellipse was 26.928 nm^2. The mean scale factor required to include the emitter in the estimate was 3.0335, with a max of 10.189 and a minimum of .0931. The mean distance from the ellipse center to emitter the emitter's true location was 5.943 nm. Figure 4.3 shows the results of a typical unmitigated run with a curved flightpath.

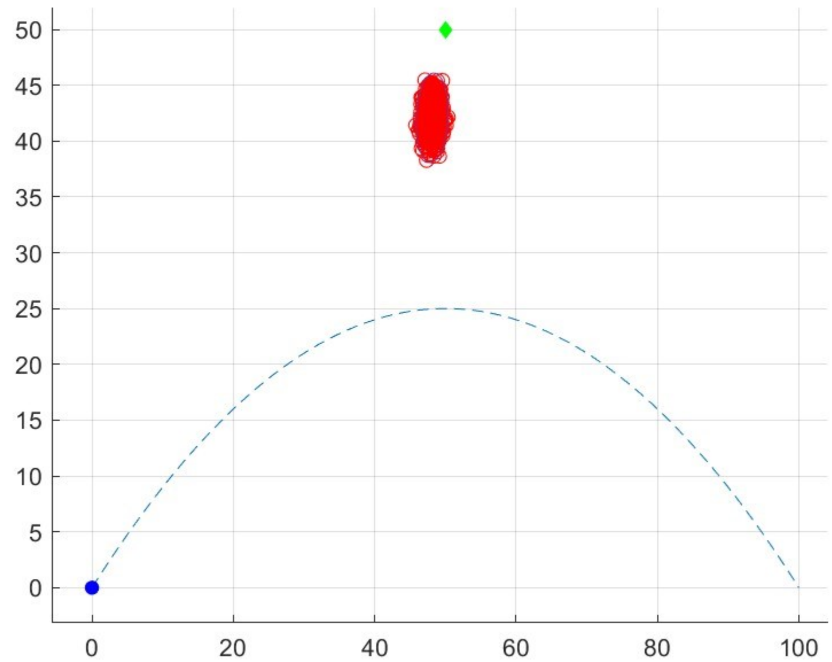


Figure 4.3: Typical Results for Curved Flightpath with No Mitigation

*4.2.3 Emitter Proximity*

As described above, we wanted to investigate the relationship between emitter proximity to the flight path and accuracy of results. We returned to using a straight flight path represented by fy=0. The three emitter locations we used were (20,35), (50,10) and (50,75). We did this test first with unmitigated data. The results of a typical run of this experiment are shown in figure 4.4.

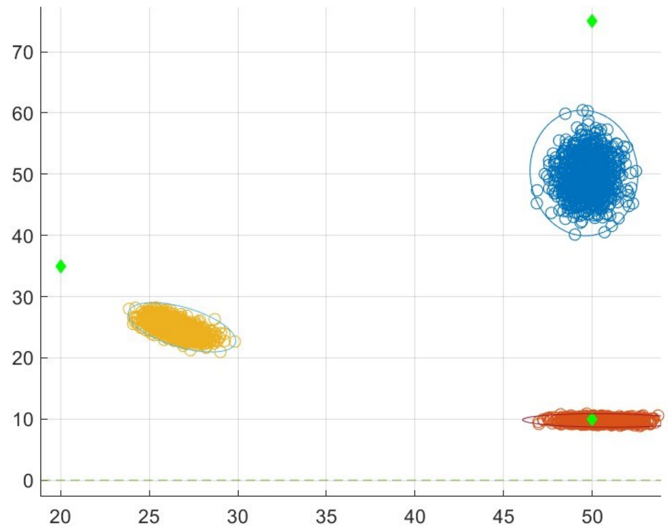


Figure 4.4: Typical Results for Emitter Proximity with No Mitigation

With the emitter at position (20,35) the results of the untreated data across 1000 runs were as follows. The mean area of the resulting ellipse was 45.7328 nm^2. The mean scale factor required to include the emitter in the estimate was 3.9379, with a max of 10.4371 and a minimum of .3672. The mean distance from the ellipse center to emitter the emitter's true location was 11.3293 nm.

With the emitter moved to (50,10) the results of the untreated data across 1000 runs were as follows. The mean area of the resulting ellipse was 12.6045 nm^2. The mean scale factor required to include the emitter in the estimate was .6563, with a max of 4.338 and a minimum of .0023. The mean distance from the ellipse center to emitter the emitter's true location was 1.6425 nm.

Lastly, we moved the emitter to (50,75) and the results of the untreated data across 1000 runs were as follows. The mean area of the resulting ellipse was 108.112

nm^2. The mean scale factor required to include the emitter in the estimate was 7.142, with a max of 19.1497 and a minimum of 2.071. The mean distance from the ellipse center to emitter the emitter's true location was 25.508 nm. The results of this experiment are summarized in table 4.1.

Table 4.1: Results of Emitter Proximity with No Mitigation

| Position | Emitter Distance (nm) | Mean Area ($nm^2$) | Error Distance (nm) | Mean SF |
|---|---|---|---|---|
| 20, 35 | 51.995 | 45.7328 | 11.3293 | 3.9379 |
| 50, 10 | 28.039 | 12.6045 | 1.6425 | .6563 |
| 50, 75 | 80.333 | 108.112 | 25.508 | 7.142 |

*4.2.4 Conclusions*

Unsurprisingly, the results of unmitigated CPLE are filled with inaccuracy. The results of this early experimentation reaffirmed the purpose of our research and our chosen methodology. We saw clear effects on results from changes in flightpath shape and changes in emitter location. If this is true of unmitigated data, then it's worth exploring these same concepts in mitigated data.

The data being collected in a geolocation scenario is likely to contain outliers. If those outliers remain unmitigated, the results of CPLE serve no useful function. Worse, in scenarios where accurate location results mean quicker rescues errant data can lead to an inability to locate a person in need. As previously stated, we need to find a method for cleaning up data collected in these scenarios. The results of the mitigation efforts shown below serve that purpose.

*4.3 Results of Distance from the Mean Method*

Our implementation of Distance from the Mean mitigation yielded promising results when considering area as a metric. As shown in figure 3.3, removing the points furthest from the mean of the estimate will always yield a reduced area of the final estimate. As described in section 3.2, we utilized an $\varepsilon$ value of 30. Because of this, we removed the worst 30 percent of our final data. Across 1000 runs, the mean area of the elliptical estimate before mitigation was $74.88\ nm^2$. After Distance from the mean mitigation, the mean of the area of the new ellipse was $2.87\ nm^2$. This represents a mean percent area reduction of 78.2%, and a mean area reduction of $72.01\ nm^2$. This is a notable reduction of estimate area.

Unfortunately, the issues with this mitigation method begin to show themselves when we consider scale factor. Across 1000 runs, we measure the scale factor required to include the emitter in the original ellipse and the post mitigation ellipse. The mean scale factor required to include the emitter in our original ellipse was 148.87. The minimum scale factor was 5.09, and the maximum was 330.44. The mean scale factor required to include the emitter in our post mitigation ellipse was 390.7. The minimum scale factor was 51.28, and the maximum was 563.34. The reduction in area that we saw as a result of using this mitigation method made our scale factors worse due to the error present in the system.

The issues continue to show themselves when we consider distance from the center of our estimate to the true location of the emitter. Across 1000 runs we measured the distance from the center of the original ellipse to the emitter and the distance from the center of the new ellipse to the emitter. We compared those two numbers and got a mean distance increase of .016 nm. Using distance from the mean estimation moved the center of our estimate further from the emitter. It's notable here that the data fluctuated between low positive and negative values. The mean of these 1000 runs, and thousands of others, shows us that the change in distance is nearly negligible.

### *4.3.1 Conclusions*

All this points us to one conclusion: Distance from the Mean mitigation is only useful if our system is already a sound estimator. It reduces the area of our estimate but outside of convenient edge cases, like the one described in section 3.2, it does not improve the precision of our system. This results from the fact that the outlier mitigation happens too late in the process. We're doing the entirety of our location estimate calculations with noisy data. Even with resampling and large population numbers we're filling our CPLE matrices with noisy biased data. The takeaway from this experiment should be twofold. First, the results of this standard scenario experiment tell us that we should not spend more time researching this method but rather should iterate and improve on it. Second, and more importantly, it presented the idea that a reduction in estimate area (an increase in accuracy) is not meaningful if it's not coupled with a move toward the target (an increase in precision).

## *4.4 Results of Rate of Change Score Method*

Learning from the results of the DftM method has led us to a method where we process the data before making location estimates. Doing this gives us promising results which are a marked improvement over the DftM method. Figure 4.5 shows the results of a typical run using RoCS outlier mitigation. It's clear that the area of our estimate shrinks, and in most cases, it moves toward the emitter.

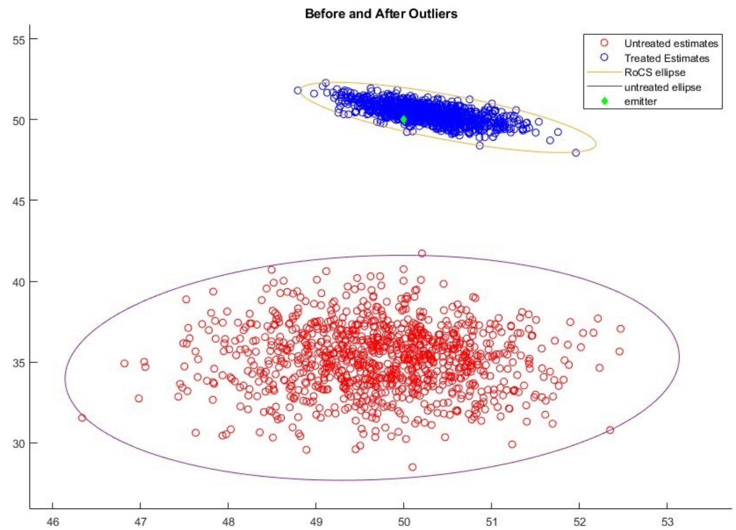


Figure 4.5: Typical RoCS Results

*4.4.1 Standard Scenario*

The results on the treated data for the same flightpath were much better. The mean area after applying the RoCS algorithm was 7.973 nm^2, a mean reduction of 84.1 % or 42.153 nm^2. The mean scale factor required to include the emitter in the estimate was 1.459, with a max of 6.16 and a minimum of .005. The mean distance from the ellipse center to emitter the emitter's true location was 2.458 nm, an improvement of 9.692 nm. The SF95 for this scenario is 7.973.

*4.4.2 Curved Flightpath*

The results on the treated data for the curved flightpath were much better. The mean area after applying the RoCS algorithm was 4.1517 nm^2, a mean reduction of 84.58 % or 22.77 nm^2. The mean scale factor required to include the emitter in the estimate was 1.2688 with a max of 5.388 and a minimum of .0012. The mean distance from the ellipse center to the emitter's true location was 1.395 nm, an improvement of 4.548 nm. The SF95 for this scenario is 2.895.

*4.4.3 Emitter Proximity*

At location (20,35), the mean area after applying the RoCS algorithm was 6.386 nm^2, a mean reduction of 86.03 % or 39.347 nm^2. The mean scale factor required to include the emitter in the estimate was 1.6106 with a max of 6.611 and a minimum of .0035. The mean distance from the ellipse center to the emitter's true location was 2.254 nm, an improvement of 9.075 nm. The SF95 for this scenario is 3.338.

At location (50, 10), the mean area after applying the RoCS algorithm was 1.5443 nm^2, a mean reduction of 87.75 % or 11.0605 nm^2. The mean scale factor required to include the emitter in the estimate was 1.122 with a max of 5.358 and a minimum of .00006 The mean distance from the ellipse center to the emitter's true location was .7155 nm, an improvement of .927 nm. The SF95 for this scenario is 2.765.

At location (50,75), the mean area after applying the RoCS algorithm was 21.1594 nm^2, a mean reduction of 80.42% or 86.952 nm^2. The mean scale factor required to include the emitter in the estimate was 1.802 with a max of 7.451 and a minimum of .0067. The mean distance from the ellipse center to the emitter's true location was 5.7237 nm, an improvement of 19.784 nm. The SF95 for this scenario is 3.741. The results for this experiment are summarized in table 4.2.

Table 4.2: RoCS Method Emitter Proximity Results

| Position | Emitter Distance (nm) | Mean Area ($nm^2$) | Area Δ ($nm^2$) | Error Distance (nm) | Dist Δ (nm) | Mean SF | SF95 |
|---|---|---|---|---|---|---|---|
| 20, 35 | 51.995 | 6.386 | 39.3468 | 2.254 | 9.0753 | 1.6106 | 3.338 |
| 50, 10 | 28.039 | 1.544 | 11.0605 | .7155 | .927 | 1.122 | 2.765 |
| 50, 75 | 80.333 | 21.1594 | 86.9526 | 5.7237 | 19.7843 | 1.802 | 3.741 |

*4.4.4 Conclusions*

As stated in section 4.2, a decrease in area of estimate isn't worth much if it isn't coupled with movement toward the emitter's true location. With this method, we've managed to find a way to reduce the area of the estimate and the distance from the center of our estimate to the true emitter location in all tested scenarios.

To understand this result, it's important to look at the effectiveness of the algorithm that is selecting the outliers. To do that, we need to consider analytically where this algorithm has flaws. The algorithm looks for the areas in the AoA data where the rate of change is greatest, regardless of surrounding data points. If each outlier point is surrounded on both sides by good data, the algorithm will correctly identify it. Problems start to arise when outlier data is not isolated. If two or more outlier data points are side by side and they are all subjected to similar magnitude noise, we start to see issues with the selection process. The RoCS of those multiple outliers would be low in the same way multiple good data points in a row would have a low RoCS. For similar reasons, the algorithm does not perform as well when used on data collected during a curved flight path. While the results of the overall tests show that this CPLE calculated with data treated by this algorithm does outperform the same calculations done with untreated data by a large margin, there is a better way to find outliers.

### *4.5 Results of Best Fit Comparison Method*

Given the results shown in section 4.3, we can assume that we're on the right track for finding a solid outlier identification method. We've learned from the shortcomings of the RoCS method and implemented the Best Fit Comparison method. Figure 4.6 shows the results of a typical Best Fit Comparison run.

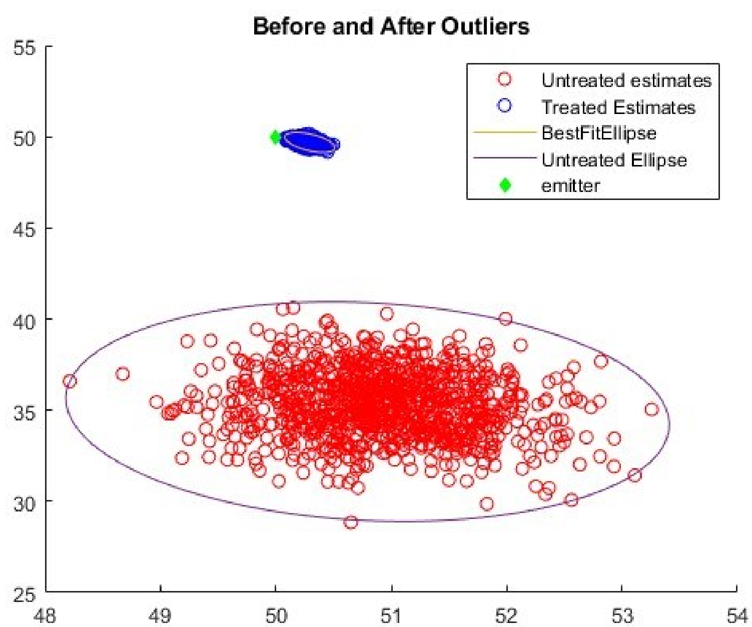


Figure 4.6: Typical Best Fit Results

#### *4.5.1 Standard Scenario*

The results on the treated data for the same flightpath were much better. The mean area after applying the Best Fit Comparison algorithm was .495 nm^2, a mean reduction of 98.99 % or 49.632 nm^2. The mean scale factor required to include the emitter in the estimate was 1.233, with a max of 3.718 and a minimum of .0187. The mean distance from the ellipse center to emitter the emitter's true location was .528 nm, an improvement of 11.6215 nm. The SF95 for this scenario is 2.232.

### *4.5.2 Curved Flightpath*

The results on the treated data for the curved flightpath were again better than the prior methods. The mean area after applying the Best Fit Comparison algorithm was .3685 nm^2, a mean reduction of 98.55 % or 26.559 nm^2. The mean scale factor required to include the emitter in the estimate was 1.234 with a max of 4.844 and a minimum of .0012. The mean distance from the ellipse center to the emitter's true location was .415 nm, an improvement of 5.528 nm. The SF95 for this scenario is 5.287.

### *4.5.3 Emitter Proximity*

At location (20,35), the mean area after applying the Best Fit Comparison algorithm was .3847 nm^2, a mean reduction of 99.15 % or 45.348 nm^2. The mean scale factor required to include the emitter in the estimate was 1.243 with a max of 3.845 and a minimum of .0043. The mean distance from the ellipse center to the emitter's true location was .4581 nm, an improvement of 10.8712 nm. The SF95 for this scenario is 2.324.

At location (50,10), the mean area after applying the Best Fit Comparison algorithm was .7911 nm^2, a mean reduction of 92.95 % or 11.8134 nm^2. The mean scale factor required to include the emitter in the estimate was 1.1437 with a max of

4.158 and a minimum of .000126. The mean distance from the ellipse center to the emitter's true location was .5230 nm, an improvement of 1.1195 nm. The SF95 for this scenario is 2.525.

At location (50,75), the mean area after applying the Best Fit Comparison algorithm was 1.106 nm^2, a mean reduction of 98.95 % or 107.007 nm^2. The mean scale factor required to include the emitter in the estimate was 1.259 with a max of 3.823 and a minimum of .0075. The mean distance from the ellipse center to the emitter's true location was .415 nm, an improvement of 5.528 nm. The SF95 for this scenario is 2.323. The results for this experiment are summarized in table 4.3.

Table 4.3: BFC Method Emitter Proximity Results

| Position | Emitter Distance (nm) | Mean Area ($nm^2$) | Area Δ ($nm^2$) | Error Distance (nm) | Dist Δ (nm) | Mean SF | SF95 |
|---|---|---|---|---|---|---|---|
| 20, 35 | 51.995 | .3847 | 45.348 | .4581 | 10.8712 | 1.243 | 2.324 |
| 50, 10 | 28.039 | .7911 | 11.8134 | .523 | 1.1195 | 1.1437 | 2.525 |
| 50, 75 | 80.333 | 1.106 | 107.007 | .415 | 5.528 | 1.259 | 2.323 |

*4.5.4 Conclusions*

In every tested scenario, estimating location after using the Best Fit Comparison algorithm yielded a smaller estimate area, lower error distance, and smaller scale factor values when compared to untreated data and RoCS data. As before, in order to understand this result we must examine the effectiveness of our outlier selection method. Comparison of each data point not only to its neighbors but to the trend of the data greatly improves outlier selection accuracy. This leads to a system where the data being used in location estimation is much cleaner than even the RoCS scenario. For this reason we see a greater reduction in the estimate area, lower required scaling factors and reduced error distances.

### *4.6 Analysis of Results*

In order to understand the value of each method, it's worth comparing their results directly to each other. The conclusions drawn in section 4.2.1 give us sound justification for leaving DftM mitigation out of this comparison. Table 4.4 shows the results of CPLE based elliptical estimation as the emitter gets further from the flight path. The mean distance from every point along the flightpath was calculated for an emitter as it was moved through space in increments of 5nm. At each of 15 different locations we tracked mean area of ellipse, mean distance from center of ellipse to the true location of the emitter, mean scaling factor to contain the emitter, and scaling factor required to contain the emitter in 95% of the trials. This process was conducted on untreated data, RoCS treated data, and Best Fit Comparison treated data. Figure 4.7 shows how Area of the estimate is affected by average distance between the emitter and the flight path for each method. Figure 4.8 shows how mean Scale Factor required to include the emitter in the ellipse changes as a function of distance between the emitter and the flight path for each method. Figure 4.9 shows how distance from the center of the ellipse to the emitter changes as a function of distance between the emitter and the flight path for each method. Figure 4.10 shows the relationship between distance from the emitter to the flight path and the scale factor required to include the emitter in our estimate in 95% of cases for each method.

Each method has some merits, and some shortfalls. Comparison of figure 3.3, figure 4.5, and figure 4.6 demonstrate well the different results that our processes yield. Ultimately, the merits of the Best Fit Comparison method far outweigh the merits of the other methods. While it is computationally expensive it does what each of the other methods do better and more effectively. In the scenarios where Rate of Change or Distance from the Mean do perform well, Best Fit Comparison gets the same results at worst. More importantly, it yields results with much higher accuracy in situations with complex flightpaths, very noisy data, and less than ideal emitter locations where the first two methods break down. Ultimately, the Best Fit Comparison tends to reduce the area of our estimate and move it closer to the emitter's true location more effectively than the other methods. This characterization makes it so that the Best Fit Comparison method could be applied to a real-world system.

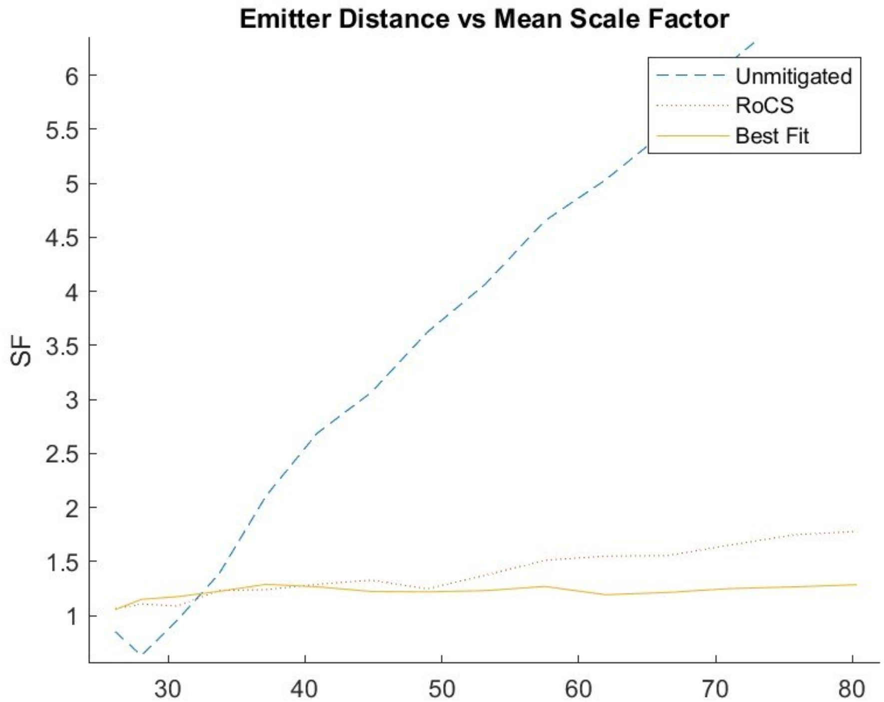


Figure 4.7: Effect of Emitter Distance on Mean Scale Factor

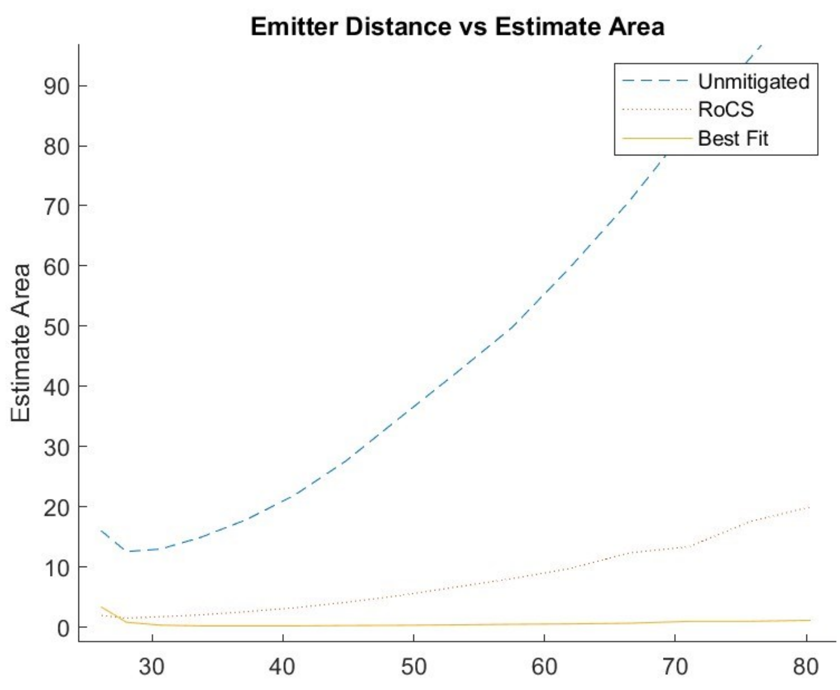


Figure 4.8: Effect of Emitter Distance on Estimate Area

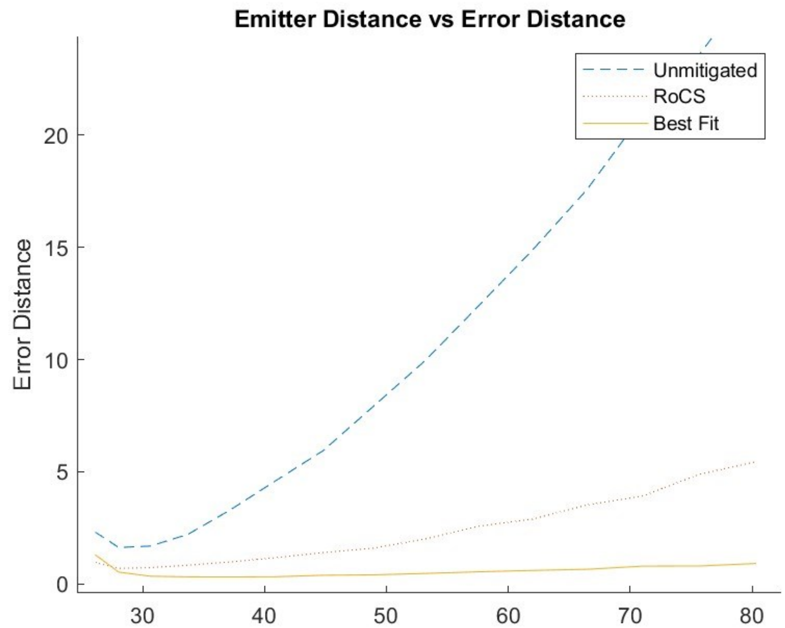


Figure 4.9: Effect of Emitter Distance on Error Distance

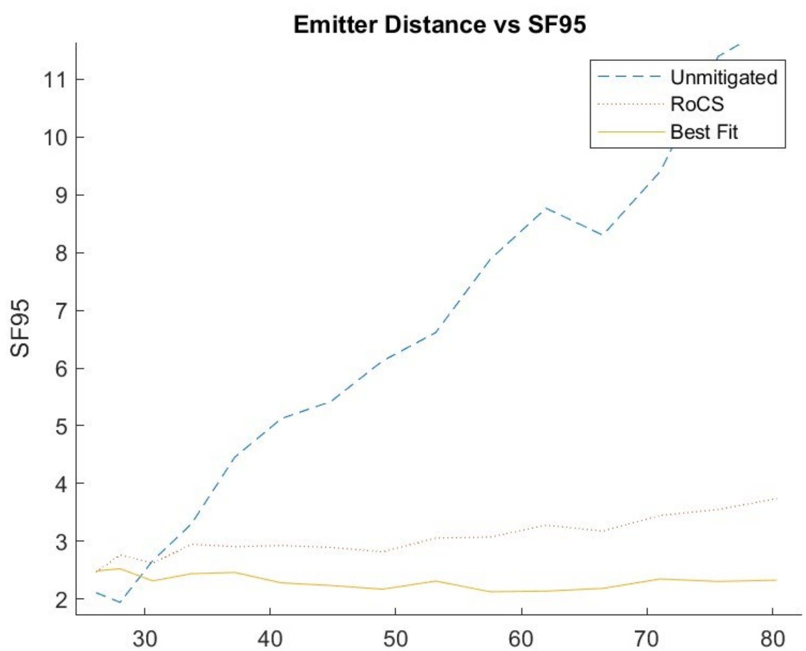


Figure 4.10: Effect of Distance on SF95

Table 4.4: Direct Comparison of Method Results

| Ex | Ey | Average distance | SF95 | Mean Area | Mean D2E | Mean SF | RoCS SF95 | RoCS mean Area | RoCS mean D2E | RoCS mean SF | BFC SF95 | BFC mean Area | BFC mean D2E | BFC mean SF |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 50 | 5 | 2.111 | 16.008 | 2.316 | 0.85 | 2.467 | 1.946 | 0.965 | 1.062 | 2.487 | 3.38 | 1.304 | 1.053 | 26.117 |
| 50 | 10 | 1.945 | 12.566 | 1.63 | 0.629 | 2.765 | 1.509 | 0.703 | 1.106 | 2.526 | 0.823 | 0.534 | 1.149 | 28.039 |
| 50 | 15 | 2.666 | 12.965 | 1.695 | 0.953 | 2.619 | 1.741 | 0.735 | 1.087 | 2.315 | 0.335 | 0.351 | 1.173 | 30.636 |
| 50 | 20 | 3.302 | 14.919 | 2.216 | 1.386 | 2.95 | 2.043 | 0.842 | 1.231 | 2.439 | 0.219 | 0.323 | 1.223 | 33.718 |
| 50 | 25 | 4.455 | 17.816 | 3.296 | 2.109 | 2.908 | 2.574 | 0.986 | 1.239 | 2.461 | 0.213 | 0.318 | 1.288 | 37.163 |
| 50 | 30 | 5.125 | 21.917 | 4.594 | 2.686 | 2.928 | 3.222 | 1.175 | 1.288 | 2.281 | 0.213 | 0.327 | 1.265 | 40.885 |
| 50 | 35 | 5.42 | 27.623 | 5.949 | 3.062 | 2.895 | 4.127 | 1.406 | 1.327 | 2.236 | 0.274 | 0.394 | 1.222 | 44.822 |
| 50 | 40 | 6.121 | 34.778 | 7.916 | 3.621 | 2.819 | 5.255 | 1.601 | 1.246 | 2.171 | 0.306 | 0.412 | 1.219 | 48.929 |
| 50 | 45 | 6.614 | 42.104 | 9.952 | 4.066 | 3.056 | 6.649 | 2.01 | 1.372 | 2.312 | 0.38 | 0.475 | 1.229 | 53.172 |
| 50 | 50 | 7.883 | 49.805 | 12.371 | 4.652 | 3.073 | 8.082 | 2.572 | 1.51 | 2.127 | 0.482 | 0.548 | 1.268 | 57.523 |
| 50 | 55 | 8.766 | 59.796 | 14.864 | 5.031 | 3.28 | 9.742 | 2.896 | 1.549 | 2.138 | 0.545 | 0.609 | 1.192 | 61.962 |
| 50 | 60 | 8.3 | 70.768 | 17.585 | 5.493 | 3.177 | 12.346 | 3.528 | 1.553 | 2.185 | 0.655 | 0.661 | 1.213 | 66.474 |
| 50 | 65 | 9.401 | 83.233 | 20.966 | 6.118 | 3.448 | 13.37 | 3.927 | 1.651 | 2.349 | 0.968 | 0.8 | 1.248 | 71.046 |
| 50 | 70 | 11.395 | 94.43 | 23.589 | 6.595 | 3.551 | 17.521 | 4.89 | 1.745 | 2.305 | 0.971 | 0.807 | 1.264 | 75.668 |

# CHAPTER FIVE

# Conclusion

## *5.1 Conclusions*

In this thesis we have demonstrated that outlier mitigation methods, when applied to the process of Cartesian Pseudo-Linear Estimation, can lead to vastly improved location estimations. We've applied techniques motivated bywell-researched methods for resampling on data to make location estimates based only on bearing data. We've shown through thorough simulation based experiments that there is real value added in the use of effective methodologies for identifying those outliers. Most importantly we've identified one method which, when applied to noisy biased data, yields reliable bearing data from which we can make a reasonable location estimate. The application of this method in tandem with resampling has the potential to advance the study of CPLE as it applies to geolocation with data gathered by moving receivers.

## *5.2 Future Work*

The field of bearings only localization will always have new and interesting areas of study available. Given the opportunity, it would be helpful to explore the application of these outlier identification methods to real world data and scenarios. While the simulations described in this thesis are thorough and form representative models, there is a certain level of validation associated with real physical trials. Additionally, improving the metric tracking related to scale factor poses an interesting challenge. We calculate plausibility ellipses but extended the fidelity of that concept to form true confidence intervals poses an interesting challenge. Continuation of this research in those areas could lead to further novel results.

BIBLIOGRAPHY

[1] K. Dogancay, “Bias compensation for the bearings-only pseudolinear target track estimator," Signal Processing, IEEE Transactions on, vol. 54, no. 1, pp. 59{68,Jan 2006.

[2] D. Koks, “numerical calculations for passive geolocation scenarios,” DTIC Document, Tech. Rep., 2007

[3] S. Nardone, A. Lindgren, and K. Gong, “Fundamental properties and performance of conventional bearings-only target motion analysis,” IEEE Transactions on Automatic Control, vol. 29, no. 9, pp. 775–787, Sep 1984.

[4] R. Stansfield, “Statistical theory of d.f. fixing,” Electrical Engineers - Part IIIA: Radiocommunication, Journal of the Institution of, vol. 94, no. 15, pp. 762– 770, March 1947

[5] Stone, Joshua, “Method for the Application of the Emprirical Bootstrap to Confidence Ellipse Estimation in Bearings-only Localization,” M.S. thesis, Dept. of Electrical and Computer Eng., Baylor University, Waco, Texas, 2016

[6] K. Singh and M. Xie, “Bootstrap: A statistical method - rutgers university,” *Min-ge Xie Research Papers*, 2008. [Online]. Available: https://www.stat.rutgers.edu/home/mxie/RCPapers/bootstrap.pdf. [Accessed: 01-Oct-2022].

[7] B. Efron. "Bootstrap Methods: Another Look at the Jackknife." Ann. Statist. 7 (1) 1 - 26, January, 1979. https://doi.org/10.1214/aos/1176344552

[8] J. Frost, “Introduction to bootstrapping in statistics with an example,” *Statistics By Jim*, 12-Jun-2020. [Online]. Available: https://statisticsbyjim.com/hypothesis-testing/bootstrapping/. [Accessed: 02-Oct-2022].

[9] S. Ozeki, Y. Sumiya and M. Shirakawa, "AOA measurement error under the antenna couplings," Proceedings of Digital Processing Applications (TENCON '96), 1996, pp. 804-809 vol.2, doi: 10.1109/TENCON.1996.608449.

[10] S. A. Kassam and H. V. Poor, "Robust techniques for signal processing: A survey," in Proceedings of the IEEE, vol. 73, no. 3, pp. 433-481, March 1985, doi: 10.1109/PROC.1985.13167