

**Autonomous Error Bounding of Position  
Estimates from GPS and Galileo**

by

Thomas J. Temple

Bachelor of Arts in Computer Science and Engineering-Physics  
Dartmouth College 2003

Submitted to the Department of Aeronautics and Astronautics  
in partial fulfillment of the requirements for the degree of

Master of Science in Aeronautics and Astronautics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2006

© Thomas J. Temple 2006. All rights reserved.

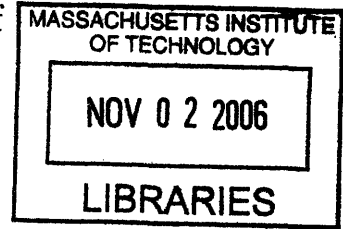
**ARCHIVES**

The author hereby grants to MIT permission to reproduce and  
distribute publicly paper and electronic copies of this thesis document  
in whole or in part.

Author .....  
Department of Aeronautics and Astronautics  
August 24, 2006

Certified by .....  
Jonathan P. How  
Associate Professor  
MIT Aeronautics and Astronautics

Accepted by .....  
Jaime Peraire  
Professor of Aeronautics and Astronautics  
Chair, Committee on Graduate Students





# Autonomous Error Bounding of Position Estimates from GPS and Galileo

by

Thomas J. Temple

## Abstract

In safety-of-life applications of satellite-based navigation, such as the guided approach and landing of an aircraft, the most important question is whether the navigation error is tolerable. Although differentially corrected GPS is accurate enough for the task most of the time, anomalous measurement errors can create situations where the navigation error is intolerably large. Detection of such situations is referred to as integrity monitoring. Due to the non-stationary nature of the error sources, it is impossible to predetermine an adequate error-bound with the required confidence. Since the errors at the airplane can be different from the errors at reference stations, integrity can't be assured by ground monitoring. It is therefore necessary for the receiver on the airplane to autonomously assess the integrity of the position estimate in real-time.

In the presence of multiple errors it is possible for a set of measurements to remain self-consistent despite containing errors. This is the primary reason why GPS has been unable to provide adequate integrity for aircraft approach. When the Galileo system become operational, there will be many more independent measurements. The more measurements that are available, the more unlikely it becomes that the errors happen to be self-consistent by chance. This thesis will quantify this relationship. In particular, we determine the maximum level of navigation error at a given probability as a function of the redundancy and consistency of the measurements.

Rather than approach this problem with statistical tests in mind, we approach this as a machine learning problem in which we empirically determine an optimal mapping from the measurements to an error bound. In so doing we will examine a broader class of tests than has been considered before. With a sufficiently large and demanding training data, this approach provides error-bounding functions that meet even the strictest integrity requirements of precision approaches.

We determine the optimal error-bounding function and show that in a GPS + Galileo constellation, it can meet the requirements of Category I, II and III precision approach—a feat that has proven difficult for GPS alone. This function is shown to underestimate the level of error at a rate of less than  $10^{-7}$  per snapshot regardless of the pseudorange error distribution. This corresponds to a rate of missed detection of

less than  $10^{-9}$  for all approach categorizations. At the same time, in a 54 satellite constellation, the level of availability for Category I precision approaches availability exceeds 99.999%. For Category II and III precision approaches, it can provide availability exceeding 99.9% with either a 60 satellite constellation, or with a modest improvement over existing LAAS corrections.

# Contents

<b>1</b>	<b>Introduction</b>	<b>13</b>
1.1	Motivation . . . . .	14
1.1.1	Integrity of a Position Estimate . . . . .	14
1.1.2	Current Approach . . . . .	16
1.1.3	Previous Difficulty . . . . .	16
1.1.4	Research Context . . . . .	18
1.2	Problem Statement . . . . .	19
1.2.1	Unknown Error . . . . .	20
1.2.2	Discrete Decisions . . . . .	20
1.2.3	Continuous Error Bounds . . . . .	21
1.2.4	Under-Estimation of Error . . . . .	22
1.3	Thesis Outline . . . . .	25
<b>2</b>	<b>Problem Formalism for Precision Approaches</b>	<b>27</b>
2.1	Objective Function . . . . .	27
2.1.1	Discrete Classification . . . . .	28
2.1.2	Minimum Expected Cost . . . . .	29
2.2	Algorithm Design . . . . .	30
2.2.1	Future Data . . . . .	30

2.2.2	Algorithmic Framework . . . . .	32
2.2.3	Scatter . . . . .	33
2.2.4	Geometric Considerations . . . . .	34
2.2.5	Metrics . . . . .	39
2.2.6	Weights, $\beta$ . . . . .	41
2.2.7	Scale, $\alpha(k)$ . . . . .	41
<b>3</b>	<b>Methodology</b>	<b>43</b>
3.1	Search . . . . .	43
3.1.1	Summary . . . . .	43
3.1.2	Subset Size . . . . .	44
3.1.3	Scale Topology . . . . .	45
3.1.4	Weight Topology . . . . .	47
3.1.5	Norms . . . . .	48
3.1.6	Model Order . . . . .	48
3.2	Simulation Environment . . . . .	49
3.2.1	Error Model . . . . .	50
3.2.2	Size Limitations . . . . .	52
<b>4</b>	<b>Results</b>	<b>55</b>
4.1	Search Results . . . . .	55
4.1.1	Optimal Subset Size . . . . .	55
4.1.2	Number of Metrics . . . . .	60
4.1.3	Norms . . . . .	61
4.1.4	Weights . . . . .	61
4.1.5	Scale . . . . .	66
4.2	Algorithm Summary . . . . .	69

4.3	Validation Results . . . . .	71
<b>5</b>	<b>Conclusions</b>	<b>75</b>
5.1	Future Work . . . . .	77



# List of Figures

1-1	Error-Bound outcomes . . . . .	22
1-2	Empirical Pseudorange Error Distribution . . . . .	24
2-1	CDFs of Vertical Error and DOP Adjusted Vertical Error . . . . .	36
3-1	Scatter-plot of Single Metric and Error . . . . .	46
3-2	Experimental Range Error . . . . .	51
4-1	PDF of Number of Satellites Visible . . . . .	56
4-2	Performance by Subset Size . . . . .	58
4-3	Convergence of Single-Metric Search . . . . .	62
4-4	Convergence of Two-Metric Search . . . . .	63
4-5	Convergence of Three-Metric Search . . . . .	64
4-6	Level of Service Availability by Norm . . . . .	65
4-7	Performance by Satellites Visible . . . . .	67
4-8	Optimal Scaling Factors . . . . .	68
4-9	Progression of Optimal Scaling Factors . . . . .	70
4-10	CDF of Under-Estimation Rate . . . . .	73



# List of Tables

- 1.1 Precision Approach Categories . . . . . 15
- 1.2 Outcome Classifications . . . . . 21
  
- 3.1 Search Components . . . . . 44
  
- 4.1 Precision Approach Integrity Requirements . . . . . 71
- 4.2 Category I False Alarm Rates . . . . . 74
- 4.3 Category II/III False Alarm Rates . . . . . 74



# Chapter 1

## Introduction

Imagine yourself as an airplane pilot approaching a runway surrounded in fog. You must rely on your navigational instruments for your position relative to the runway. You don't need to know the exact position of the airplane; as long as the airplane is within some predetermined safe region, say, within  $x$  meters of a nominal  $3^\circ$  glide-slope, you can be confident that you will land safely. You would like to be very confident that you are within that region.

You have a number of measurements from different sensors available to you and when you consider them all, you can estimate that you are within the safe region. You have enough sensors so that there is some redundancy. Under what conditions would you abort the landing?

1. All 5 sensors agree *almost exactly*.
2. All 22 sensors agree *almost exactly*.
3. Some of the 5 sensors disagree *substantially*.
4. Some of the 22 sensors disagree *substantially*.

Clearly, closer agreement is more appealing and more sensors are preferable to fewer. One can make that determination without regard to the error distribution from which the sensor readings were drawn. The goal of this thesis is to quantify this intuition. The key idea is that the larger the number of measurements, the more representative they become of the distribution from which they were selected and the less reliant we need to be on assumptions regarding the *a priori* distribution.

## 1.1 Motivation

### 1.1.1 Integrity of a Position Estimate

In order to use a Global Navigation Satellite System (GNSS) for navigation, it must meet the users' requirements for accuracy, availability and integrity. The availability of the system is the fraction of the time that the system is deemed safe to use for a particular application while the integrity is the ability of the system to protect against errors. More specifically, the integrity is defined as the probability that a position error of a given magnitude will be detected. There is a trade-off between availability and integrity, *e.g.*, one can trivially provide perfect integrity at zero availability.

In safety-of-life applications integrity is of the utmost importance. Suppose that a GNSS system meets the accuracy requirements of an application nearly all time, but in some rare cases, say, one in a million estimates, the level of inaccuracy puts lives in danger. If the system is unable to identify most of those cases, it cannot be used. This is the situation in which the differentially corrected GPS finds itself today in the application of precision approach.

A precision approach is one in which both vertical and horizontal guidance are used to acquire the runway. In this thesis we will only focus on the vertical component of the guidance since it is more demanding. Typically, if the runway is not visible from

Category	Decision Height (ft)	Vertical Error Limit (m)	Integrity (per approach)
I	200	10	$10^{-7}$
II	100	5	$10^{-9}$
III	0-50	5	$10^{-9}$

Table 1.1: Integrity requirements for precision approach. These requirements are still under discussion within the aviation community.

a height of 700-1000 feet above the runway (depending on the obstruction around the airport), then the pilot must receive horizontal and vertical guidance to descend lower. Such approaches are called precision approaches. How low the airplane can descend on instruments depends on the pilots training and the plane's avionics. There are three categories of precision approaches numbered I, II and III with Category III subdivided in to IIIa, b and c for very low decision heights. These have increasingly exacting requirements as the airplane gets closer to the runway as shown in Table 1.1. If the runway is not visible from the decision height, the landing is aborted.

Currently, precision approaches in the United States require the Instrument Landing System (ILS), which provides of a set of radio beams aligned with the glide slope, or angle of approach, of the landing airplane. The ILS is more expensive than a GPS solution since it requires installation and maintenance of precise equipment at every runway end. Local differential correction would only require one signal for the entire airport and would be cheaper than a single ILS beacon. The ILS is scheduled to begin phase-out in 2015[3].

The integrity required for precision approach is given in Table 1.1. Despite high-hopes and a truly astonishing level of accuracy, differential GPS is finding it a challenge to meet the strict integrity requirement[11].

### 1.1.2 Current Approach

In GPS, the task of providing integrity has been handled by a receiver function called receiver autonomous integrity monitoring (RAIM). There are two different ways of approaching RAIM depending on whether temporal information is also considered, which divides RAIM techniques into “snapshot” and “filtered” techniques. snapshot techniques attempt to determine the degree of agreement among the measurements and generally use various least-squares statistical methods. Filtered algorithms generally use some form of Kalman filter, which requires knowledge of temporal error covariances. Filtered algorithms are also typically based on least-squares estimation. In this thesis we will concentrate on snapshot techniques because it relies less heavily on knowing *a priori* the nature of the error. However, we will reconsider whether least-squares is the best approach for assessing integrity.

A diverse group of snapshot RAIM algorithms have been proposed based on various methods of detecting and quantifying the disagreement among the measurements. These methods vary in the way measurements are compared and they vary in the test statistic used to assess those comparisons. This thesis seeks to resolve these differences empirically. In particular, we propose a space of error-bounding algorithms that includes practically all existing snapshot RAIM methods as well as yet unconsidered algorithms of similar form. We then employ search algorithms to determine the error-bounding algorithm that we expect to perform the best in terms of integrity and availability.

### 1.1.3 Previous Difficulty

The difficulty in detecting multiple errors of similar magnitude results from the possibility that the errors may be self-consistent. For instance if every measurement were the correct pseudorange to the same (although in incorrect) position, they would

agree perfectly although there would still be error. To be more precise, let  $\rho$  denote the a vector of the pseudorange measurements. The position equations are linearized which makes the position estimate,  $\hat{x}$ , the solution to the over-constrained system of linear equations[18]

$$\rho = G\hat{x}.$$

Where  $G$  is called the geometry matrix and it's rows are the cosines of direction of the satellite from the position estimate in the local reference frame.<sup>1</sup> This is commonly solved using the least-squares fit.

$$\hat{x} = (G'G)^{-1}G'\rho$$

Let  $\rho = \tilde{\rho} + \varepsilon$  where  $\tilde{\rho}$  is the true measurement and  $\varepsilon$  is the error. Let  $\tilde{x}$  denote the true position.

$$\begin{aligned}\hat{x} &= (G'G)^{-1}G'(\tilde{\rho} + \varepsilon) \\ \hat{x} - \tilde{x} &= (G'G)^{-1}G'\varepsilon\end{aligned}\tag{1.1}$$

Now consider the residuals.

$$\begin{aligned}\rho - G\hat{x} &= (I - G(G'G)^{-1}G')\rho \\ &= (I - G(G'G)^{-1}G')(\tilde{\rho} + \varepsilon) \\ &= (I - G(G'G)^{-1}G')(G\tilde{x} + \varepsilon) \\ &= (G(I - I)\tilde{x} + (I - G(G'G)^{-1}G')\varepsilon) \\ &= (I - G(G'G)^{-1}G')\varepsilon\end{aligned}\tag{1.2}$$

---

<sup>1</sup>The cosine in the time direction is one.

Suppose that  $\varepsilon = Gb$  for some vector  $b$ .

$$\begin{aligned}\hat{x} - \tilde{x} &= (G'G)^{-1}G'Gb = b \\ \rho - G\hat{x} &= (I - G(G'G)^{-1}G')Gb = (G - G)b = 0\end{aligned}$$

The residuals are zero while the position error is precisely  $b$ . This case is intractable. Our only chance is for such an arrangement of errors to be extremely unlikely. For this to happen,  $\varepsilon$  must lie in the span of the columns of  $G$ , which is  $k \times 4$ , where  $k$  is the number of satellites visible. As  $k$  increases, it becomes more and more unlikely that such an event will occur. Or more accurately, it becomes increasingly unlikely that the errors will be *close* to the span of  $G$ . With a single constellation where it is common to have 7 or fewer satellites visible, this probability is simply unmanageable for safety-of-life applications. However with the much larger constellation provided by GPS + Galileo, the probability will decrease to a manageable level.

#### 1.1.4 Research Context

Originally, snapshot methods consisted of an algorithm to determine if one of the satellite ranges was faulty[19]. “Faulty” in this context meant *grossly* erroneous. But as pointed out in [14], for the application of precision approach, detecting single satellite “faults” is not enough. In the information rich environment provided by the combined GPS + Galileo constellations, gross faults will be stand out obviously and the differential reference stations will reliably detect them. We will instead concern ourselves with a metric notion of the quality of the ranges *as a whole* as is done in [5].

An intuitive approach is to define a test statistic based on a measure of the disagreement among the measurements and define a threshold for that statistic below which the system is declared safe to use. Most of the existing techniques take this ap-

proach, including [8–10, 17, 20], some of which employ search to determine optimality. Fundamentally, the same approach is taken here. In particular, the algorithm space explored in this thesis is considered to be a generalization of [17]. However, we would like to make a conceptual distinction.

We adopt the change of focus introduced in [5], moving from the discrete notion of faults to a metric notion of quality, specifically maximum error at a given probability. Instead of focusing on a test statistic and application-specific thresholds, we will instead provide a set of quality metrics and a function that maps them to a safe error-bound. If we can show the mapping to be reliable, it can be used more generally since it does not require any application specific parameters, *e.g.*, thresholds and error models. In practice, the algorithm could be used “out of the box” without consideration of signal environment; it will be equally reliable on corrected or uncorrected, filtered or unfiltered ranges. In particular, one would be free to exclude measurements using traditional RAIM *before* handing those measurements to the error-bounding algorithm.

## 1.2 Problem Statement

Rather than detect faults *per se*, our goal is to allow an airplane’s GPS receiver to place an upper-bound on the error in its position estimate obtained from a set of pseudorange measurements. Primarily we require the probability that the true error is greater than the bound be very small, thereby ensuring integrity. Given that, we would like the upper-bound to be tight, *i.e.*, under-estimate as little as possible, to maximize availability of service.

### 1.2.1 Unknown Error

A difficulty comes from the fact that we don't know enough about the pseudorange errors *a priori*. We don't know how they correlate temporally which will require that the error-bound be determined in real-time, *i.e.*, based only on the “snapshot” of current sensor readings. Nor do we adequately know their distribution. This will cost a great deal of analytical traction. We will treat this issue by analyzing error-bounding algorithms using simulated data. This data will include realistic as well as extremely challenging pseudorange error distributions. Finally, we don't know the spacial covariances between pseudorange errors from different satellites. Fundamentally, this is intractable—if the only information available is from the sensors and the sensors are allowed to collude to hide as discussed in Section 1.1.3, nothing can be done. For this thesis we will assume the pseudorange errors in measurements from different satellites to be independent.

### 1.2.2 Discrete Decisions

In the precision approach problem, the aviation community has designated an “alert limit” on position error that it considers to be the limit of safety. If the Integrity Monitor can only choose to raise an alarm or not, that dictates four discrete scenarios (see Table 1.2). If true error was greater than the alert limit and we raised the alarm we will call that a “correct detection” and if we failed to raise the alarm we will call it a “missed detection”. If the true error was less than the alert limit and we raised the alarm we will call that a “false alarm,” while the cases in which we do not raise an alarm will be referred to as “service available”. For each scenario there are relative costs and allowable rates of occurrence. This specification defines an optimization problem, in particular we want to find the algorithm that is most likely to satisfy it.

Figure 1-1 illustrates how this specification can be applied to a particular algo-

	Error Tolerable	Error Intolerable
Alarm Raised	False Alarm	Correct Detection
Alarm Not Raised	Service Available	Missed Detection

Table 1.2: The possible outcomes of raising, or not raising, an alarm.

rithm. The ellipse represents a scatter-plot of the results of running a particular algorithm on a data set. If we replaced “error bound” with “test statistic” on the abscissa, this would be a traditional threshold setting problem. In such a problem we would be looking for the vertical line that segments the data with minimum cost. In our approach however, the threshold is not a free parameter, instead it is held fixed at the value given in the specifications. Pictorially, rather than moving the threshold, we can move and reshape the data by changing the error-bounding algorithm. For instance, simply by scaling the algorithm one can segment the data in any way that a threshold could. But by changing the algorithm in other ways we can perhaps change the shape of the distribution to a more favorable one.

### 1.2.3 Continuous Error Bounds

The discrete notions of false alarms and missed detections are somewhat incompatible with the continuous notion of an error-bound. For instance in Figure 1-1 there are data-points in which the error-bound under-estimates although those points are in the “Service Available” or “Correct Detection” region. But clearly with a different threshold, the classification of any one of these points could be changed to a missed detection. This enforces an intrinsic notion of scale which is antithetical to our goal of avoiding assumptions regarding the measurement error distribution. To avoid this, we will map the discrete notions of missed detections and false alarms to the continuous notions of error-bound under- and over-estimates.

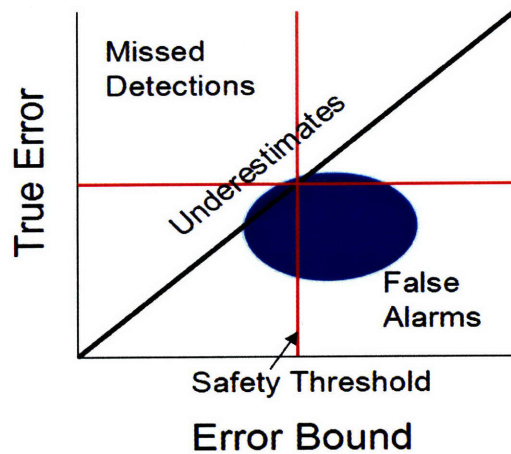


Figure 1-1: Scatter-plot of error/error-bound pairs for a particular algorithm on a data set.

#### 1.2.4 Under-Estimation of Error

Instead of promising a certain level of missed detections, we will promise a certain level of under-estimates. All missed detections are also under-estimates. Based on our *empirical* knowledge of the measurement error distribution, we can put an upper bound on the fraction of over-estimates that could be missed detections. This approach gives us a great deal of traction in dealing with a largely unknown pseudorange error distribution.

For example, suppose we have enough real world data to confidently say that a particular error model, for instance a Gaussian mixture, is a good description of the pseudorange error in 99% of snapshots. Suppose that we can show analytically that the fault rate for this error distribution is practically nil. This tells us that a missed detection is only *possible* 1% of the time. This is sometimes referred to as the “unconditional fault rate”. Suppose also that we can show analytically or in

simulation that our algorithm underestimates only 0.01% of the time for an arbitrary distribution. Finally if we can show that whether our algorithm under-estimates or not is independent of the size of the true error, then we can claim a 0.0001% rate of missed detections.

Figure 1-2 shows the cumulative probability distribution function (CDF) of pseudorange error for a real receiver using local differential corrections. The errors are of a completely tolerable magnitude in every case. However, there aren't enough cases to prove adequate integrity. Furthermore, the experiment was conducted at a single location. However, at this location at least, a Gaussian mixture could describe at least 99.9% of the data very well.

Similarly, we can use empirical data to determine the mapping from over-estimates to false alarms. However, the false alarm rate doesn't need to be known to as extreme a precision as the missed detection rate. This means that we are free to determine false alarm rate based on actual empirical data or more conveniently an idealized model of it.

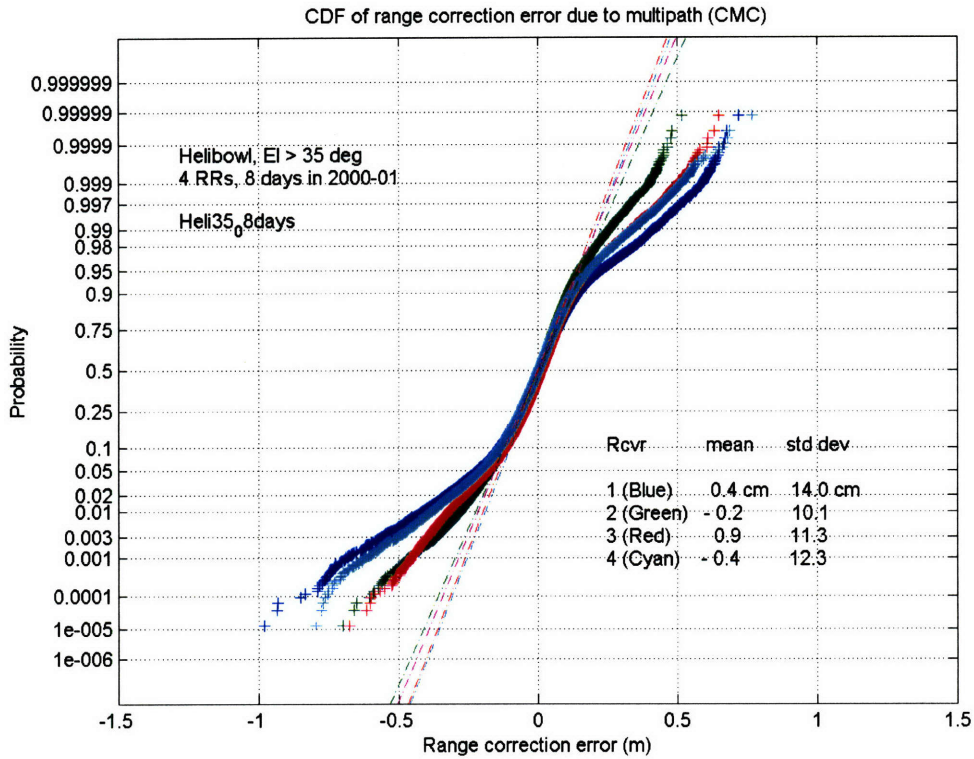


Figure 1-2: Empirical distribution of pseudorange error after applying a local differential corrections from four different reference stations. A Gaussian mixture, for instance  $0.9\mathcal{N}(0, 0.10m) + 0.1\mathcal{N}(0, 0.30m)$ , is a very good representation of this distribution. (Courtesy: FAA technical Center, Atlantic City, NJ)

## 1.3 Thesis Outline

The goal of this thesis is to determine the best algorithm for placing an error-bound on the position estimate of an airplane executing a precision approach. This work is arranged as follows. Section 2.1 quantitatively defines the notion of performance by which we will judge algorithms. Section 2.2 defines the space of algorithms we will consider. Chapter 3 poses the search problem and discusses the methodology and algorithms for conducting it. Section 4.1 discusses the results of the search with the optimal algorithm given by Equation 4.4. The best algorithm found is then validated in Section 4.3. We conclude in Chapter 5 that these results demonstrate the adequacy of this algorithm for all categories of precision approaches. Finally, Section 5.1 discusses the implications of this work.



## Chapter 2

# Problem Formalism for Precision Approaches

In this chapter, we pose the problem of determining the best algorithm to generate error-bounds from a snapshot of pseudoranges as a non-linear optimization problem. In Section 2.1 we quantitatively define our objectives and in Section 2.2 we define the space of algorithms that fall under consideration. In Chapter 3 we describe how this space is searched for the algorithm that optimizes our choice of objective function.

### 2.1 Objective Function

We are trying to find an algorithm that can determine whether or not it is safe to continue with an approach. If we treat this as a discrete decision and also treat safety as binary, the four cases in Table 1.2 are exhaustive and mutually exclusive—we only need to specify three of the four rates to fully describe the performance of an algorithm. Furthermore, the algorithm has no control over whether the true error is greater or less than the alert limit. This allows us to characterize performance by

specifying only two of these.

We seek to maximize the time when the landing service is available while keeping the missed detection rate below a mandated level, which we will denote  $T$ . Since the error rate is fixed, minimizing false alarms is equivalent to maximizing availability in our context. Let us approach the problem as a non-linear optimization problem. Let  $A$  denote the space of all error bounding algorithms.

$$\begin{aligned}
 \text{objective :} & \quad \arg \min_{a \in A} P(\text{False Alarm}) \\
 \text{such that :} & \quad P(\text{Missed Detection}) < T \qquad (2.1)
 \end{aligned}$$

### 2.1.1 Discrete Classification

Suppose that we have a large and representative data set of  $n$  measurement–error pairs,  $(m_i, e_i)$ . Then a good approximation to Problem 2.1 would be to replace the probabilities with their maximum likelihood estimates. Let FA, MD, SA and CD denote zero-one functions defined over errors and error bounds that indicate whether a particular event is a false alarm, missed detection, service available or correct detection, respectively. Let

$$\begin{aligned}
 P_{FA} & \equiv \frac{1}{n} \sum_{i=1}^n \text{FA}(a(m_i), e_i) \\
 P_{MD} & \equiv \frac{1}{n} \sum_{i=1}^n \text{MD}(a(m_i), e_i) \\
 P_{SA} & \equiv \frac{1}{n} \sum_{i=1}^n \text{SA}(a(m_i), e_i) \\
 P_{CD} & \equiv \frac{1}{n} \sum_{i=1}^n \text{CD}(a(m_i), e_i)
 \end{aligned}$$

making the problem

$$\begin{aligned}
 \text{objective :} & \qquad \qquad \qquad \arg \min_{a \in A} P_{FA} \\
 \text{such that :} & \qquad \qquad \qquad P_{MD} < T. \qquad \qquad \qquad (2.2)
 \end{aligned}$$

But it might not be prudent to suppose that the missed detection rate is *zero* simply because we haven't had a missed detection *yet*. We could instead be Bayesian and use artificial counts to represent a uniform prior probability distribution. This would make the estimates used in problem 2.2

$$\begin{aligned}
 P_{FA} &\equiv \frac{1}{n+1} \left( 1 + \sum_{i=1}^n \text{FA}(a(m_i), e_i) \right) \\
 P_{MD} &\equiv \frac{1}{n+1} \left( 1 + \sum_{i=1}^n \text{MD}(a(m_i), e_i) \right) \\
 P_{SA} &\equiv \frac{1}{n+1} \left( 1 + \sum_{i=1}^n \text{SA}(a(m_i), e_i) \right) \\
 P_{CD} &\equiv \frac{1}{n+1} \left( 1 + \sum_{i=1}^n \text{CD}(a(m_i), e_i) \right). \qquad \qquad \qquad (2.3)
 \end{aligned}$$

We can convert Problem 2.2 into an unconstrained optimization problem using a barrier or penalty function. Instead we take a slightly less artificial tack.

### 2.1.2 Minimum Expected Cost

If we assign a relative cost,  $c$ , to each of the four possible outcomes, we can pose the problem as minimum expected cost.

$$\arg \min_{a \in A} (c_{MD} \cdot P_{MD} + c_{FA} \cdot P_{FA} + c_{SA} \cdot P_{SA} + c_{CD} \cdot P_{CD}) \qquad \qquad \qquad (2.4)$$

The rate at which errors of unsafe magnitude occurs is fixed, let it be,  $P_e$ . We can rewrite Objective 2.4 as

$$\arg \min_{a \in A} (c_{MD} \cdot P_{MD} + c_{FA} \cdot P_{FA} + c_{SA}(1 - P_e - P_{FA}) + c_{CD}(P_e - P_{MD}))$$

or

$$\arg \min_{a \in A} (P_{MD}(c_{MD} - c_{CD}) + P_{FA}(c_{FA} - c_{SA}) + \text{constants})$$

Since the optimization is unaffected by a linear function, we can subtract out the constant terms and divide by  $c_{FA} - c_{SA}$ .

$$\text{Defining } C \equiv \frac{c_{MD} - c_{CD}}{c_{FA} - c_{SA}}$$

we have

$$\arg \min_{a \in A} (C \cdot P_{MD} + P_{FA}) \tag{2.5}$$

It should be noted that Problems 2.1 and 2.5 are not the same. Solving 2.5 will not ensure that the constraint on missed detections is met. Instead it will balance the rate of missed detections appropriately against the rate of false alarms. We must decide in the end whether or not the result is satisfactory.

## 2.2 Algorithm Design

### 2.2.1 Future Data

The primary problem is that it is impossible to obtain a data set on which to assess the probabilities in Equation 2.5. What we want is a data set that is representative of *future* conditions. Since the missed detection rate is to be kept extremely low, it would require an unrealistic level of prescience to acquire a data set that would let

us predict the probabilities with sufficient accuracy. Clearly we're going to have to make some sort of assumptions about the kind of data that we're going to see.

Suppose that we have enough empirical knowledge to say that at least some fraction of the time,  $\tau$ , the measurement error will conform to some nominal distribution,  $N$ . Let's assume that the measurement errors within a pseudorange snapshot are independent and identically-distributed (IID). At the same time, this implies that there is another, unknown distribution,  $X$ , from which the errors are drawn  $1 - \tau$  of the time. Either analytically or numerically, we can determine the performance of an algorithm on errors from  $N$  and for particular choices of  $X$ . For instance we might suppose that  $X$  is  $\pm 1$  meter uniform error and that  $N$  is  $0.9\mathcal{N}(0, 0.14m) + 0.1\mathcal{N}(0, 0.42m)$ . But notice that empirically we can say relatively little about the validity of this choice of  $X$ . For instance, suppose that we define a new distribution  $Y = \gamma X$  for some scalar  $\gamma$ , making  $Y = \pm \gamma m$  uniform. It is impossible to say whether we should consider  $X$  or  $Y$  when analyzing an error-bounding algorithm. In fact, the errors are non-stationary. They might conform to  $X$  some of the time and  $Y$  the rest.

This leads us to an important feature of the algorithms that we will consider. We will attempt to make the algorithm robust to the scale of the error. Let  $\rho = \tilde{\rho} + \epsilon$  where  $\tilde{\rho}$  is the vector of true pseudoranges and  $\epsilon$  is the vector of pseudorange errors. The least squares solution to the position equations is given by

$$\begin{aligned} x &= (G'G)^{-1}G'\rho \\ &= (G'G)^{-1}G'\tilde{\rho} + (G'G)^{-1}G'\epsilon \end{aligned}$$

since  $(G'G)^{-1}G'\rho$  is the true position, then the position error is simply  $(G'G)^{-1}G'\epsilon$  which is a linear function of  $\epsilon$ . That is to say if we replaced  $\epsilon_i$  with  $\gamma\epsilon_i$  the position error would become  $\gamma\tilde{\epsilon}_i$ . Therefore we would like our error-bound to scale to  $\gamma\hat{\epsilon}_i$ .

Recall from Equation 1.2 that the residuals which are given by  $(I - G(G'G)^{-1}G')\epsilon$ ,

a linear function of the range error. Furthermore the position error is a linear function of the residuals. Similarly, any least squares position computed from subsets of the ranges will move from the true position linearly. This implies that the distance between such position estimates will also scale.

## 2.2.2 Algorithmic Framework

To ensure that the space of algorithms is manageable, let us break the error-bounding algorithm into separate components. For a snapshot method, the only information available to the algorithm is

- the pseudorange measurements,  $\rho$ , and
- the geometry matrix,  $G$

Implicitly we have one more piece of information, the number of satellites visible,  $k$ . We will examine algorithms of the form

$$\text{Error Bound} = \alpha(k) \cdot \beta \cdot \text{METRICS}(\text{SCATTER}(\rho, G)) \quad (2.6)$$

Where

**Scatter** is intended to be suggestive of the disagreements among the measurements.

The Scatter function maps the measurements into a set on the real line

**Metrics** characterizes the “spread” of the scatter by extracting a vector of the  $n$  most informative features of that set,

$\beta$  is a normalized vector of  $n$  weights assigning relative importance to the different metrics and,

$\alpha$  is a scaling factor that converts these scaled features into an error bound.

### 2.2.3 Scatter

Misra and Bednarz [17] originally introduced the notion of scatter. They proposed that one collect a number of subsets of  $j$  satellites each and compute a position estimate for each such subset. Then by comparing the solutions we can estimate the degree of inconsistency in the system. Here we use essentially the same methodology. Primarily, we would like to determine the best choice  $j$ .

For a subset of size four or greater, we compute the least squares position estimate for that subset and compare it to the position estimate using all the satellites. For a subset of size less than four, we first compute the position estimate for the *excluded* satellites and then compute the residuals from that position to the pseudoranges from the small subset. By analogy, we will consider the ordinary residuals as estimates for a subset size of zero.

Two natural choices for the number of satellites to use per subset are  $k - 1$  and 1, *i.e.*, the leave-one-out cross-validation estimates and the residuals to those estimates. This would mean that  $k$  subsets would clearly be sufficient seeing as that is be all of the subsets possible. For other choices, there will be a much larger number of subsets available.

First note that, while there are  $j$ -choose- $k$  different subsets that we could choose, we are still solving a set of  $k$  linear equations. Regardless of the number of subsets used, there are only  $k$  independent measurements. This implies that all the positions computed will be a linear combination of some  $k$  position subset, with the relation dictated by the  $G$  matrix. This suggests that it might not be necessary to choose a large number of subsets.

It would be unwise to choose the satellite subsets randomly. Such randomness inevitably leads to over- and under-emphasis of certain satellites and can lead to disagreement going unnoticed. Although we will still choose satellites randomly, we

will mitigate this affect by including each satellite in the same number of subsets.

Misra and Bednarz divided the sky into bins to determine which satellites would be included in subsets. This is problematic because a statically positioned bin will occasionally have only a small number of measurements in it, overemphasizing them. It is also very difficult to dynamically choose bins to avoid this. The reason for the bins was to avoid poor geometry subsets.<sup>1</sup> Here we avoid bad geometry by requiring the VDOP of a subset be bellow a predefined threshold. The impact of this threshold is mitigated by making certain adjustments to the scatter as will be discussed in Section 2.2.4.

To determine how many satellites to use per subset we compute the optimal single metric algorithms for each subset size of a fixed number of visible satellites. This is done for every subset size and for the ordinary residuals. For each we need to determine the norm to use and the scaling factor by which that norm is converted into an error bound. This is essentially a brute force search of a simplified space. It was not computationally realistic to consider multiple-metric algorithms while determining the best notion of scatter. Furthermore, while it would be possible to mix norms of wholly different notions of scatter, to consider this possibility the scale of the search would have to be considerably expanded.

## 2.2.4 Geometric Considerations

From the geometry matrix one is able to determine how the measurements interrelate. We will use the geometry information to adjust the scatter to reflect our confidence due to the interrelation of the measurements.

First and simplest, the covariance of the position estimate is proportional to

---

<sup>1</sup>The quality of the geometry is given by  $(G'G)^{-1}$ . The square-root of the sum of the diagonal of this matrix is referred to as the “dilution of precision” or DOP, while the square-root of the third element of the diagonal is referred to as the “vertical dilution of precision”, or VDOP.

$(G'G)^{-1}$ [18]. It seems obvious then that we should scale the scatter by the VDOP of all the satellites in view. Figure 2-1 shows how such a correction tightens the CDF of the error. Even though it is a relatively small change, it is an important improvement—accounting for uncertainty in the position estimate due to the geometry.

If we compute the position estimate (using all the satellites) and then compute a estimate using a subset, we can determine the position that would be computed from the opposite subset that is it's exclusion simply by examining the geometry. Let  $S_a$  and  $S_b$  be two subsets of satellites that are mutually exclusive and collectively exhaustive, *i.e.*, every satellite is included in exactly one of the subsets. Let  $G_a, G_b$  be the rows of  $G$  corresponding to  $S_a, S_b$  respectively with zeros in the rows corresponding to satellites from the other subset. The position estimates are given by

$$\begin{aligned}x_a &= (G'_a G_a)^{-1} G'_a \rho \\x_b &= (G'_b G_b)^{-1} G'_b \rho\end{aligned}$$

Rearranging and summing we have

$$(G'_a G_a)x_a + (G'_b G_b)x_b = G'_a \rho + G'_b \rho$$

Since  $G = G_a + G_b$

$$(G'_a G_a)x_a + (G'_b G_b)x_b = G' \rho$$

The overall estimate,  $x$  is given by  $x = (G'G)^{-1} G' \rho$  so,

$$(G'_a G_a)x_a + (G'_b G_b)x_b = (G'G)x$$

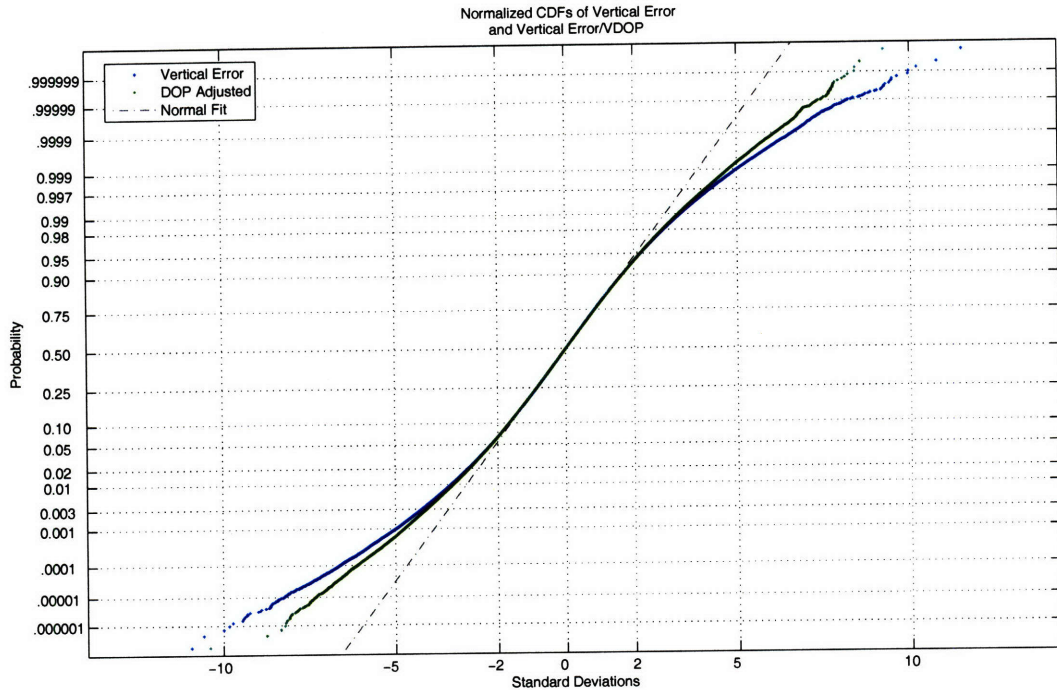


Figure 2-1: Normalized CDFs of vertical error and vertical error divided by VDOP. The pseudorange errors were selected from a Gaussian mixture. Normalization ensures that both distributions have zero mean and unity covariance. The VDOP adjusted error has tighter tails, indicating that vertical error divided by VDOP is a “better behaved” variable<sup>3</sup> than vertical error. Although the curves only deviate in the extreme ends of the tails, we are trying to predict *extreme* behavior and the tightness of the tails is our primary concern.

which expands to

$$\begin{aligned}
(G'_a G_a)x_a + (G'_b G_b)x_b &= ((G_a + G_b)'(G_a + G_b))x \\
x_b &= (G'_b G_b)^{-1}(((G_a + G_b)'(G_a + G_b))x - (G'_a G_a)x_a) \\
x_b &= x + (G'_b G_b)^{-1}(G'_a G_a)(x - x_a) + (G'_a G_b + G'_b G_a)x.
\end{aligned}$$

Note that  $G_a$  and  $G_b$  have complimenting zeros making the mixed terms zero. This lets us conclude the following relationship.

$$x_b - x = (G'_b G_b)^{-1}(G'_a G_a)(x - x_a) \quad (2.7)$$

The difference between the position estimate and an estimate based on a subset of the satellites is linearly related to the difference between the position estimate and the estimate based on the subset acquired by excluding those satellites. The relation is given by Equation 2.7. We can use this result to adjust the scatter such that there becomes no difference between a subset of size  $j$  and one of  $k - j$ . This result does not rely on the invertability of  $(G'_a G_a)$ , except for the implicit definition of  $x_a$ . If  $(G'_a G_a)$  is not full rank, we can replace  $(G'_a G_a)x_a$  with  $G'_a \rho$  giving us

$$x_b - x = (G'_b G_b)^{-1}(G'_a G_a)x - G'_a \rho. \quad (2.8)$$

In particular,  $G'_a$  could consist of a single row, making  $G'_a \rho$  a single directed pseudorange, while  $x_b$  would then be the leave-one-out solution. This corresponds to Lee's famous result [13] showing the equivalence of residual and position comparison techniques. More importantly for our purposes, this proves that subset positions differences and pseudorange residuals are functionally equivalent in that they both scale linearly with position error.

The amount of disagreement between two position estimates is a vector, while we have defined scatter as a set of scalars. We will simply project this vector on the direction that we care about—the vertical. If we were concerned with the magnitude of position error in general, we could take its norm instead. Analogously, the residuals can be considered to be *directed* in that they point towards a satellite. Somewhat surprisingly however, because of the first term in Equation 2.8, they don’t necessarily move the estimate in that direction. Nonetheless we can determine their direction of influence and likewise project this direction against the vertical.

Suppose that a satellite subset has a bad geometry. That suggests that it will estimate a position far from the overall estimate, even if the pseudoranges are relatively consistent. Bad subset geometry will exaggerate the disagreement between the measurements. In the overall estimate, bad geometry will increase the error bound while a bad subset geometry will make us less confident that the disagreement exists.

In the case of residuals, an analogue to the above is not immediately obvious. We would like to know how “susceptible” the position estimate is to influence from the satellite whose residual we are considering.<sup>4</sup> For illustration, imagine that all of the satellites lie in a plane except one: if we compute a position by excluding the range from the out-of-plane satellite, the residual for the out-of-plane satellite could be extremely large. We correct for this with the a notion of “directed DOP” of the estimate.

$$\text{Directed DOP} \equiv G_a(G'_b G_b)^{1/2} \tag{2.9}$$

---

<sup>4</sup>This “susceptibility” is very close to the notion of “slope” presented in [8]. It is different in that slope determines how much of the pseudorange error appears in the estimate while here we are concerned with how much the pseudorange error will appear in the residual.

Making the  $i^{\text{th}}$  adjusted residual

$$(G_{ai}x - \rho_i)G_{ai}(G'_bG_b)^{1/2}e_3$$

where  $e_3 \equiv \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix}$  is the direction of interest.

For symmetry, the DOP adjustment for position scatter was changed to also be “directional” in the same sense. That is to say we proceeded with

$$(x_b - x)'(G'_bG_b)^{1/2}e_3$$

rather than

$$(x_b - x)'e_3(e'_3(G'_bG_b)^{-1}e_3)^{-1/2}.$$

It should be noted that this is not the only means by which this adjustment could be pursued. We found this method to work better than using the slope in [8] or ordinary DOP. We suspect that this is due to better utilization of information in the off-diagonal terms of  $G'G$  and  $(G'G)^{-1}$ .

## 2.2.5 Metrics

To ensure a manageable space to search, we must predefine a space of metrics. Since scatter scales appropriately, any function of the scatter that scales linearly would also scale. For example, with a fixed number of measurements, the standard deviation would scale appropriately. Let  $S$  denote the scatter set. Consider the following

functions.

$$l_0 = \exp\left(\frac{\sum_{s \in S} \log(s)}{k-4}\right) \quad (2.10)$$

$$l_p = \left(\frac{\sum_{s \in S} |s|^p}{k-4}\right)^{1/p} \quad (2.11)$$

$$l_\infty = \max_{s \in S} |s| \quad (2.12)$$

Function 2.10 and 2.12 are the limits of Function 2.11 as  $p$  approaches 0 and  $\infty$  respectively. We will refer to these functions as “norms” although for  $p$  less than one, they do not obey the triangle inequality. Also the normalization term,  $k - 4$ , makes it appear that we are going to compare the norms of sets with different cardinalities, which would break the vector space on which the norms are defined. This is not strictly true since the algorithm is conditioned at the top level by the number of satellites visible. This term is simply a constant that can be absorbed by the scale function  $\alpha(k)$ . It is only included here in the hope of later simplifying the form of  $\alpha(k)$ .

A probability distribution becomes increasingly well described by an increasing number of it’s moments[15]. The fact that these norms are isomorphic to moments, suggests that the set of 2.10 through 2.12 are sufficiently descriptive representation of the scatter. In particular, we expect some finite (preferably small) subset of them to form an adequate description for estimating the magnitude of the error. As such, the problem of determining the optimal metrics consists of finding the optimal set of  $p$ -parameters.

### 2.2.6 Weights, $\beta$

If we map the measurements and position errors  $(\rho, G, \tilde{e})$  to the space,  $S$ , defined by

$$S \equiv (\text{METRICS}(\text{SCATTER}(\rho, G)), \tilde{e}),$$

then the remainder of the algorithm, *i.e.*,  $\alpha$  and  $\beta$ , is simply a hyperplane separating over- and under-estimates. Because of the requirement that our prediction be robust to the scale of the error, all these hyperplanes must pass through the origin. We will allow there to be different hyperplanes depending on the number of satellites visible but to avoid having a large number additional parameters we will require that they all be the same in the subspace that is  $S$  with the errors removed. The weights,  $\beta$  chooses the most informative “side” of the the measurements in this subspace.

To be conservative about over-fitting this plane to a small feature specific to the data set, we will require that all of the weights be non-negative. This has the important side effect of *drastically* reducing the number of local minima in the space of weights.

### 2.2.7 Scale, $\alpha(k)$

Conceptually, the scale function distinguishes the error-bounding approach from an ordinary test statistic. Although  $\beta$  times the metrics is a test statistic, we have taken pains to ensure that it scale linearly with error. At this point we simply provide a constant factor to relate our statistic to worst case position error at a given probability.

We have yet to address the central question of how to account for the number of measurements, *i.e.*, the number of satellites visible,  $k$ . The algorithm clearly should consider the number of measurements available. It would be possible to use this information to condition the algorithm at the innermost level. We could use a different

number of metrics with different norms and weights for every different number of satellites we can see. We could even use different scatter functions. The benefit (and also the price) for doing so would be that there would be a larger number of free parameters in the overall algorithm (for instance the number of  $p$ -parameters would go from  $n$  to  $kn$ ). More parameters allow the algorithm to more tightly bound the error in the training data. At the same time, the more parameters we have, the less well this improved performance will generalize to future data. Since performance on future data is of utmost importance, we should only introduce new parameters where it is substantially beneficial to do so.

So where is the natural place to condition on the number of satellites visible? In this work we will place it in the absolute scaling. Primarily because it introduces the fewest free parameters if it is inserted at that point. We do not expect the particular norms are the most informative about scatter to be sensitive (or at least not overly sensitive) to the number of points making up the scatter distribution. Intuitively, one would expect additional measurements to translate directly into additional confidence. To mitigate the risk of over-fit, we will enforce this intuition, requiring that the absolute scaling be monotone decreasing as more measurements are made available.

It may be the case that the most informative norms stay the same but their relative importance varies. This would recommend conditioning on the number of measurements at the relative weighting point. If this were the case we would expect a large degree of continuity between weights of neighboring numbers of measurements. We assume that the optimal relative weights change slowly (if at all) with varying numbers of measurements available.

Another reason to keep the influence of  $k$  at the top level is that  $\alpha(k)$  encapsulates the additional confidence we get from having more measurements available. This is a valuable thing in and of itself. Armed with this function, system designers would know exactly how much additional confidence would be provided by an additional signal.

# Chapter 3

## Methodology

Section 3.1 describes the methodology that was employed to search the space of algorithms defined in Section 2.2 for the one which minimizes Objective 2.5. Section 3.2 describes the data on which algorithms are to be evaluated. The results of this search are presented in Section 4.1.

### 3.1 Search

#### 3.1.1 Summary

Recall that we are looking the algorithm that solves

$$\arg \min_{\alpha \in A} (C \cdot P_{MD} + P_{FA}) \quad (3.1)$$

Wherein  $A$  is the set of algorithms of the form

$$\text{Error Bound} = \alpha(k) \cdot \beta \cdot \text{METRICS}(\text{SCATTER}(\rho, G)) \quad (3.2)$$

where  $P_{MD}$  and  $P_{FA}$  are the Bayesian estimates described in Equation 2.3. The value used for  $C$  is  $10^7$  to be consistent with our strict demands on missed detection rate.

The algorithm space,  $A$ , can be defined by a relatively small set of parameters which are enumerated in Table 3.1. We will search this parameter space to determine the best algorithm. The rest of this section is devoted to discussing the types of search used to determine these parameters which are summarized in Table 3.1.

Algorithm Component	Search Method
The size of subsets that determine scatter	by exhaustive search on a restricted space of metrics
The number of metrics to consider	determined iteratively,
the norm parameters, $p$ , to be used for those metrics	determined by randomized gradient search,
the relative weights, $\beta$ , of those norms	also by randomized gradient search
the absolute scaling, $\alpha$ , of those weights (conditioned on the number of satellites visible, $k$ ).	determined by line search.

Table 3.1: Components of the search

### 3.1.2 Subset Size

We will determine the size of subsets to use for scatter by determining the optimal algorithm for every subset size and then comparing their performance. Computationally, it is impractical to determine optimal algorithms for every subset size using the entire algorithm space and an adequately large data set. Therefore this portion of the search is restricted to single-metric algorithms.

### 3.1.3 Scale Topology

To determine how best to search for optimal weights and scale, we must carefully consider the topology of the objective function. If it is convex, or even quasi-convex we should be able to easily find its minimum. Conversely if it is highly non-convex we will have to be more judicious about how we conduct the search.

Given a set of metrics, the space of weights is a convex space. First consider all of the data in the space  $S$  in which each data point  $(\rho, G, \tilde{e}_i)$  is plotted

$$(\text{METRICS}(\text{SCATTER}((\rho, G), \tilde{e}_i)).$$

For example, for a particular one-dimensional metric function, the results of this mapping is shown in figure 3-1. Assume for the moment that we have so much data that we can treat it as a continuous density and assume further that this density is convex in the space  $S$ . Since the error-bound is a convex function of metrics and we've assumed the the metrics are a convex density over our data, then the density of error/error-prediction pairs is convex for any choices of  $\alpha$  and  $\beta$ .

Let  $O(a)$  denote the objective function 2.5. Given a particular METRICS function and fixed  $\beta$ , we can define a new objective function  $O'(\alpha)$ .

$$O'(\alpha) = O(a)$$

$$\text{such that: } a = \alpha \cdot \beta \cdot \text{METRICS}$$

For an individual data point, the objective is quasi-convex in  $\alpha$ , *i.e.*, it has a point about which the function is non-decreasing above and below. This is because on the one hand as  $\alpha$  decreases, the bound tightens until it is no longer a false alarm, but at some point it becomes an under-estimate. The result of integrating a quasi-convex function over the convex density is quasi-convex[16]. Considering the entire data set,

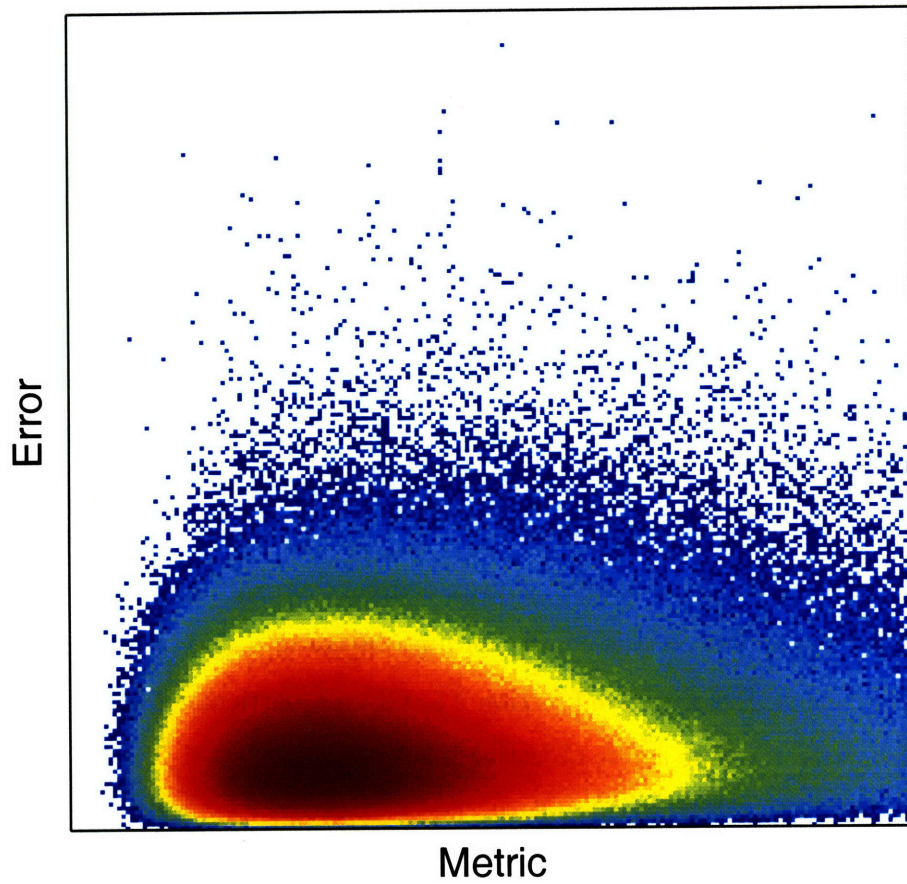


Figure 3-1: Scatter-plot of a data set of 5 million snapshots mapped into the metric/error space,  $S$ , for a particular one-dimensional metric function. Roughly speaking, the density is convex except at the fringe.

$O'(\alpha)$  is also quasi-convex.

But this is based on the assumption that the the measurement/error pairs have a continuous density. However, our data are discrete. This means that  $O'(\alpha)$  isn't necessarily quasi-convex. However, in practice, the scale over which  $O'$  varies is very large with respect to the distance between points. For the contrary to be the case, one would have to squeeze more than  $10^7$  false alarms between two consecutive missed detections. In practice, this only happens at or near the global optimum. For finding the optimum weights given relative weights, the problem is quasi-convex, allowing an efficient solution using line search.

### 3.1.4 Weight Topology

Even with a continuous density of data, we are not assured that the objective function is convex over the space of weights. That is to say that

$$O''(\beta) = \sup_{\alpha} O(a)$$

such that  $a = \alpha \cdot \beta \cdot \text{METRICS}$

is not quasi-convex. This is because the quasi-convexity of convolution only holds in one dimension[16]. Despite this lack of guarantee, if the gradient of  $O''$ ,  $\nabla_{\beta}O''$ , changes slowly we can still find an optimum.

But discrete data also means that  $\nabla_{\beta}O''$  is undefined. We can get around this by estimating a gradient by examining nearby points. To use this gradient, the continuity assumption needs to be examined more closely. One might expect that at fine resolution, level sets of  $O''(\beta)$  will be very jagged since they will follow individual data points along the boundary. At a rougher resolution however,  $O''$  varies smoothly with  $\beta$ . We proceed by smoothing the gradient and employing standard

randomization techniques—those being to follow the empirical gradient from many randomly chosen starting points and to allow the search randomly move contrary to the gradient.

### 3.1.5 Norms

Given a fixed number of norms to use, we can attempt to find the particular ones that perform the best. This has as a sub-problem, the problem of finding the best weights and scale for each set of norms. The norms are continuous in their  $p$ -parameter as defined in Equation 2.11. This lets us define an empirical gradient by varying these parameters.

In this search, all bets are off in terms of favorable topology. It is unrealistic to hope that the outcome is sensitive to these parameters in a smooth way. To accomplish this non-convex optimization we also use the empirical gradient and randomized search.

### 3.1.6 Model Order

We must also decide on the *number* of metrics to use. Since the class of  $(n+1)$ -metric algorithms includes the space of  $n$  metric algorithms (*e.g.*, by setting one weight to zero) these define a set of nested hypothesis classes. We will begin by finding the best single metric algorithm. Then we will search for the best algorithm after adding a second metric. We will continue adding metrics until either performance levels off or the problem becomes computationally intractable.

Our ability to perform the gradient smoothing discussed in Section 3.1.4 is dependant on having a lot of data nearby. But data naturally spread out in higher dimensional spaces. Also, computing many empirical gradients becomes increasingly costly. This will put pressure on the number of metrics we will be able to consider at

a time. This is compounded by the fact that we are most interested in fringe cases that represent rare but large errors.

A more important reason to limit the dimensionality is the risk of over-fit. Since we are optimizing over a maximum likelihood estimate, we should not expect actual performance to be entirely commensurate with the performance on the training data. Even if we are to use a less optimistic estimator, the risk of over-fit continues. If the algorithm is allowed too much freedom, it can pick out features that are peculiar to the training data and not indicative of future data.

## 3.2 Simulation Environment

To generate the data on which this search will be performed, we simulate a suite of constellations and randomly chosen user positions. The constellations are Walker  $n/m/1$  with  $m$  denoting the number of satellites, between 42 and 64, divided between  $m$  evenly spaced orbital planes. The third Walker parameter is the phase between satellites in adjacent planes and is equal to  $360^\circ/n$ . The elevation and inclination are equal to those of the GPS constellation. The simulated user is positioned randomly (uniform by area, not by lat/lon) between the arctic and antarctic circles. The time is chosen uniformly within one period of the constellations orbit.

To save on computational overhead, each constellation is used for 10 geometry snapshots (*i.e.*, time). Each of these geometry snapshots is used for 10 user positions. For each of these positions, different pseudorange errors are generated. A satellite is considered “visible” if it is more than 5 degrees above the horizon.

Pseudorange errors are drawn independently from a predefined error model and the user estimates his position using the ordinary least squares solution to the position equations.<sup>1</sup> The error between this estimate and the users actual position is considered

---

<sup>1</sup>Since improving the accuracy is a non-issue, we will continue to use the least-squares fit for the

the true error,  $\tilde{\epsilon}$ .

### 3.2.1 Error Model

The data set consists of two parts, a false-alarm data set and a missed detection (or rather under-estimation) data set. Since the rate of false alarms does not need to be known to such a high level of certainty, it is relatively small and has its pseudorange errors drawn from a realistic nominal distribution. The missed detection data set on the other hand is much larger and has pseudorange errors drawn from a very pessimistic distribution.

The error model used to determine false alarm rate is based on the empirical data in Figure 3-2. This is represented by a Gaussian mixture drawing from mean zero, standard deviation 14cm 90% of the time and from mean zero standard deviation of 42cm 10% of the time (which we will write as  $0.9\mathcal{N}(0, 0.14m)+0.1\mathcal{N}(0, 0.42m)$ ), which is only modestly conservative. This mixture relationship is a common representation of the realities of multipath. It is important to note that the fact that the distribution is mean zero is inconsequential because all the bias ends up in the receiver clock.

The error model used for determining under-estimation rate is uniform. Since the algorithm is designed to be invariant to the scale of the distribution, the width of the distribution is irrelevant and set arbitrarily to  $\pm 1$  meter.

The uniform distribution was found by [17] to be the most challenging. Most over-bounding distributions (*e.g.*, [10, 21]) although they are pessimistically wide, their tails still fade like a Gaussian and this provides traction. It is worth noting that this distribution is not *necessarily* the most pessimistic one possible. For a particular constellation, time and receiver position, there will be worse distributions possible. Such a distribution would introduce range errors in such that they correspond to a 

---

position estimate even while we use other methods to assess that fit. This is to avoid risking the creation of a new source of bias.

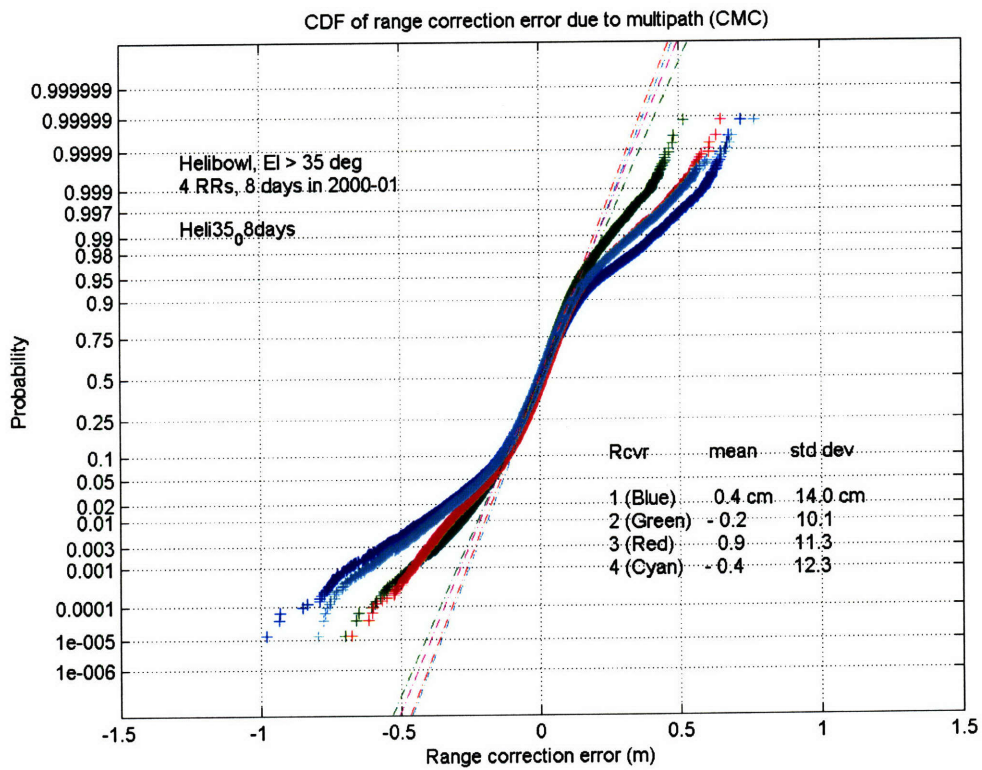


Figure 3-2: Empirical distribution of pseudorange error after applying a local differential corrections from four different reference stations.

multimodal position error distribution. Put another way, the errors could be selected in such a way as to maximize alignment with  $G$ . However,  $G$  is constantly changing. With a randomly chosen receiver position at a randomly chosen time, it is impossible to construct an error model that will create a multimodal position error distribution without dispatching the assumption that pseudorange errors are independent.

### 3.2.2 Size Limitations

The aviation community, [1, 2], has specified a missed detection rate of only one per ten million approaches for Category I landings. If we do not assume an unconditional fault rate, this suggests that we need at least ten million data points. But to be confident in our estimate, we should have substantially more. Suppose we have  $n$  data points, none of which are missed detections, how confident are can we be that the probability of missed detection is less than one in ten million? The probability that the “true” probability of missed detection is less than that can be determined from the CDF of the binomial distribution.

To have a level of confidence commensurate with the estimate, that is to say that the probability that we are wrong to only be one in ten million would require about 16 times that much data. That is to say that the that the true parameter is less than  $10^{-7}$  with a  $10^{-7}$  chance of being wrong would require  $1.6 \times 10^8$  data points. However this is perhaps an unnecessarily high level of confidence. Since the parameter is binomial, even if we are wrong about the true parameter, the probability that it is *substantially* more than the nominal value becomes *vanishingly* small.

This is an unrealistically large number for some portions of the search. Searching for the best metrics involves computing many norms of all the pseudoranges for every data point. Determining the best notion of scatter also involves determining the best metrics. Those norms are not particularly cheap operations. With the computational

power available, it was only reasonable to perform this search on a data set of one million snapshots. We must assume that the metrics that effectively predict large error at a lower occurrence rate are the same as those which effectively predict larger, more rare errors.

Once the metrics have been chosen, we can expand the data set since the remaining operations are linear ones. Nonetheless, we would like to avoid millions of vector multiplications. Instead of storing all the norms, we can store their convex hull and try to maintain a density inside using discretization. In fact, since we have chosen to be so conservative and can only allow a tiny number of missed detections, only the convex hull and points very close to it will end up mattering. This allows the search for the optimal weights to be performed efficiently on a data set of tens of millions of data points.

The search for optimal scale with given weights has a favorable topology which will allow for it to be determined with hundreds of millions of data points.



# Chapter 4

## Results

### 4.1 Search Results

#### 4.1.1 Optimal Subset Size

In this portion of the search we attempted to determine the optimal subset size to use to assess the agreement in the measurements. The data set consisted of  $10^6$  range-error pairs for missed detection and  $10^5$  for false alarms. For every point, exactly 20 satellites were considered “visible”. This number was chosen because it is a common number to have visible with constellations the size of GPS + Galileo, as shown in Figure 4-1. If there were fewer than 20 visible, the point was discarded while if more were visible, a randomly chosen subset of 20 would be marked as “visible”. It is important that this number be held fixed—consider by way of example that of size 8 from 16 satellites is different than a subset of size 8 from 24 satellites. Once we find the optimal subset size for a fixed number of visible satellites, we must determine whether the optimal subset size is an absolute number or a fraction of the satellites visible. The number 20 is therefore more appealing than 19 and 21 since it has more divisors.

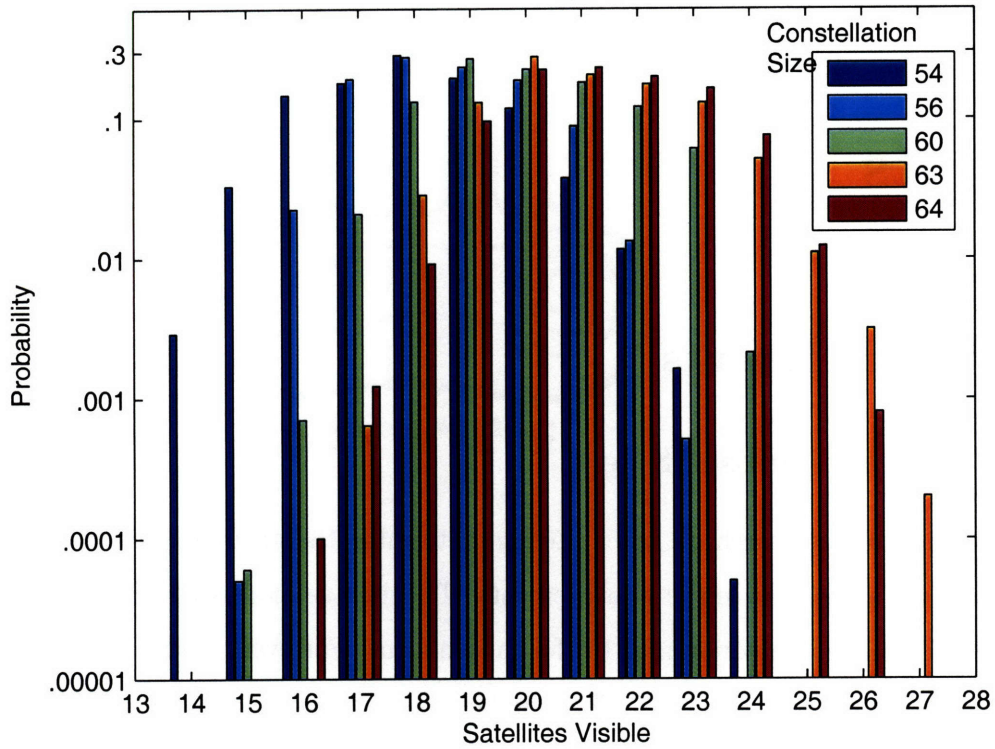


Figure 4-1: PDF of the number of satellites visible for a constellations with a similar number of satellites as GPS + Galileo. These numbers are for a mid-latitude receiver *i.e.*, between the arctic and antarctic circles. The GPS + Galileo constellation can be expected to show slightly higher variance in the number of satellites visible since they will not have as well synchronized of orbits as the constellations used here.

To proceed, we need a metric in order to evaluate different subset sizes. For a broad range of subset sizes we conducted an initial search for the best norm to use for the metric. We found that for all of them a norm slightly less than the  $l_1$  norm to be the best. That they agreed is reassuring in that it suggests that the optimal norm is independent of the subset size.

The  $l_1$  norm is very appealing computationally and analytically. It is faster to compute than any non-integer norm. In a wide class of non-Gaussian distributions, the estimate that minimizes the  $l_1$  norm is better estimate than the one that minimizes the  $l_2$  norm that it traditionally used.<sup>1</sup> For these reasons, we fixed the metric to be the  $l_1$  norm in order to compare the performance of different subset sizes. Since there are no weights to be determined, the remainder of the search is for the best scaling factor for each which is easily done on a data set of this size.

Figure 4-2 shows the relative levels of availability and under-estimation rate for each size subset. The y-axis shows the number of under-estimates allowed in the training data set. For a data set of this size, the optimal algorithm will have precisely zero under-estimates. The front edge of the surface is therefore more informative. However, it is also much noisier since it is informed by very few data points. The rest of the surface is shown in order to smooth this noise. Similarly, with a nominal signal error of  $0.9\mathcal{N}(0, 0.14m) + 0.1\mathcal{N}(0, 0.42m)$  and only  $10^5$  data points, the rates of false alarm show little or no differentiation at a 10 meter decision threshold. To exaggerate the difference, the availability shown on the z-axis is the availability with a 1.2 meter decision threshold. This exaggeration does not qualitatively affect the result. As long as differentiation is possible, it mirrors the relative differences in Figure 4-2.

Recall that the zero subset size refers the ordinary residuals to the overall position

---

<sup>1</sup>It was noted in [22] that the  $l_1$  norm of a very similar notion of scatter developed here was the most conservative bound available, although he was assuming a single "fault". At the time it was deemed too conservative an estimator. It seems that we now have enough information to be able to spend some on such conservatism.

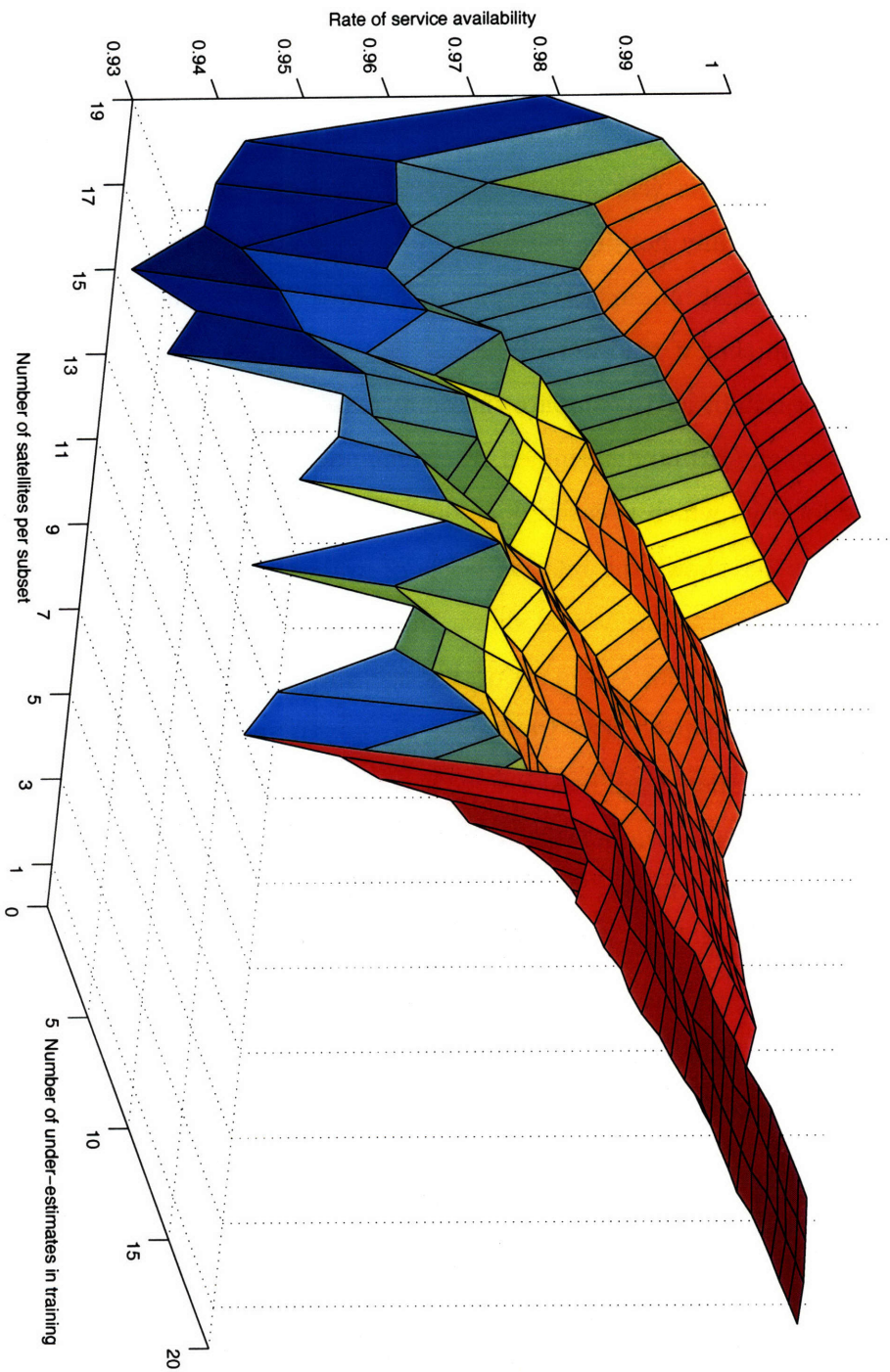


Figure 4-2: For each subset size, the optimal availability rates is shown for a given number of false alarms allowed in training. Training consisted of only  $10^6$  points. Availability is shown at a 1.2 meter alert limit to exaggerate features

estimate *i.e.*, the one including all 20 satellites. Also recall that the subset sizes 1–3 refer to instead the residuals to the position computed if those 1–3 satellites are excluded. This relationship is described in Section 2.2.3.

From Figure 4-2 it is clear that subsets using about half the satellites are a poorer estimator than subsets with 1-3 or 17-19 satellites. This is explained by the central limit theorem. By the central limit theorem, the more ranges that are included in a subset, the more that the estimates will converge, whether or not the pseudorange agree. With a large number of satellites in both the excluded and included subsets, it is much more likely for the pseudorange errors within each subset to cancel another out. In such a case, the subset position estimates agree even if the pseudorange errors do not.

There are also clear troughs at 4 and 16 satellite subsets. The 4 satellite case is explained simply by geometry. With only four satellites, the geometry of the position equations can be pretty bad. Although we screened out bad geometry, we did so using a threshold. With only 4 satellites per subset, a much higher percentage of estimates are close to, but still below, that threshold. The 16 satellite case is bad, in fact worse, for the same reason. While the geometry for the sixteen satellites might be very good, the the geometry of those excluded might be very bad. We did not screen the geometry of the excluded satellites. Recalling Equation 2.7, it is possible for  $G'_a G_a$  to be nearly singular, and so the variation could be largely hidden.

Even if we were to somehow correct these geometry issues, the fact would remain that subset sizes at the ends perform better. In the ends, the trend continues all the way to subset sizes 1 and 19. The central limit theorem goes a long way in explaining the trend. Also, these are the only subset sizes for which there is no randomness in selection. For instance of the 380 possible subsets of size 18 that could be chosen, we are randomly choosing 20 such that each satellite is included in exactly 20 subsets. Surprisingly often, 40% of the time, the same subset might be chosen twice and it's

features will be over-emphasized. This random over-emphasis, can only hurt our estimate on the average. But note that, with a 17 satellite subset, this only occurs 3% of the time and a 17 satellite subset seems to perform markedly less well than one with 18, suggesting that this effect is small.

Comparing a the 1 and 19 satellite cases against each other, we see a small but consistent advantage held by the one satellite case. The difference between them is that while the 19 satellite case measures how much the position moves when one satellite is removed, the 1 satellite case is the residual from the satellite that is removed to the new position. It appears that the latter captures the “relative geometry” better. The 1 satellite residual is in a better position to detect cases where excluding one satellite changes the geometry substantially.

Finally notice that the ordinary residuals do surprisingly well. One might expect the ordinary residuals to be identical to the 19 satellite subsets. In fact the ordinary residuals outperform the leave-one-out case and are only slightly outperformed by the one-satellite subset case. We conclude difference must be attributable to the geometric adjustments discussed in Section 2.2.4. Noting that the ordinary residuals are drastically easier to compute than the size 1 subsets, we capitalize on this result by proceeding with the ordinary residuals.

### 4.1.2 Number of Metrics

We begin by searching for the best algorithm whose metric component consists of a single norm. The norm used for that metric converged very convincingly as shown in Figure 4-3 (see also Figure 4-6). The optimal value of the  $p$ -parameter of the norm was about 0.7.

After adding a second metric, the search still converged well. The norms to which it converged are shown in Figure 4-4. One of the norms always converged to the same

norm as in the single-metric case. When this happened, it ended up getting nearly all of the relative weight. The end result was that the optimal two-metric algorithm ended up being almost identical to the single-metric algorithm. Since the relative weights are prohibited from being negative, it could not appreciably outperform the single-metric case. If this prohibition were lifted, the two metric algorithm could almost certainly find some way to improve over the one-metric algorithm. Exploratory investigation suggested that the improvement was scant and failed to generalize to unseen data.

For completeness the results for searching for a three-metric algorithm are shown in Figure 4-5. The only difference between the three- and two-metric cases was that the three-metric search converged more slowly. In the best algorithms found, the only non-negligible weights were assigned to norms with the same value as that of the single-metric case.

### 4.1.3 Norms

As the searches described in Section 4.1.2 show, the search converged rather convincingly to a  $p$ -parameter around 0.7. Figure 4-6 shows the relative level of performance of different norms.<sup>2</sup> This norm only marginally outperformed the  $l_1$  norm however, as shown in Figure 4-6. We proceed using the  $l_1$  norm since it is analytically more appealing and vastly easier to compute.

### 4.1.4 Weights

It was found that a single metric was sufficient. Relative weights are no longer needed.

---

<sup>2</sup>Rather than putting the value of the objective function on the ordinate, which has rather meaningless units, we instead put the rate of false alarms. This is allowable because all comparable cases have the same number of under-estimates—which in the case of Figure 4-6 is zero.

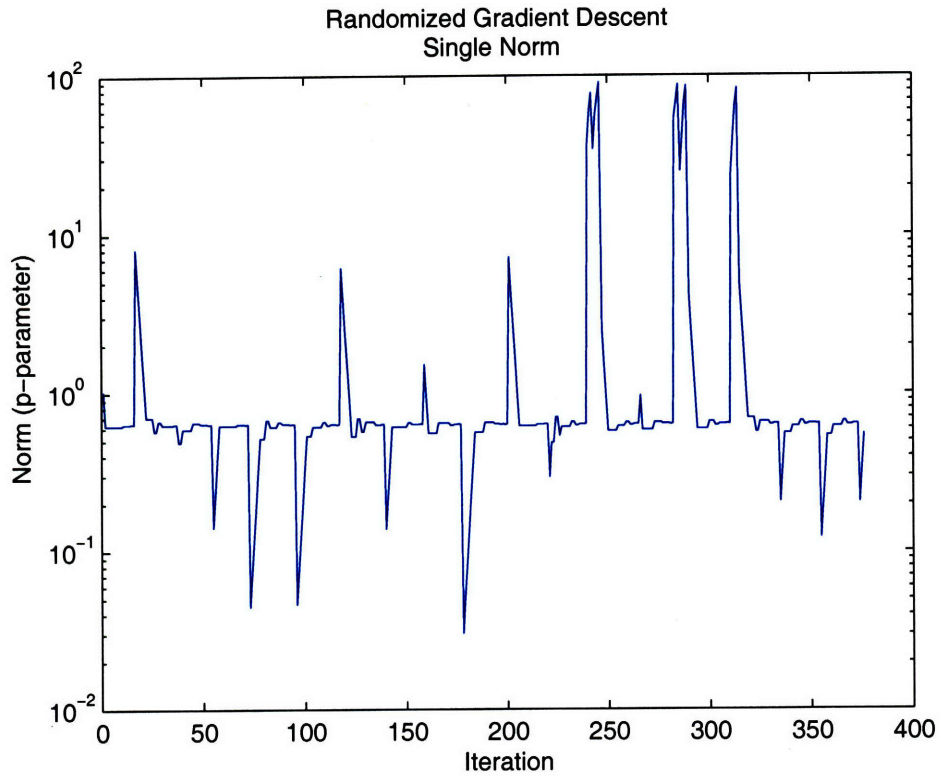


Figure 4-3: The chronological progression of the norm ( $p$ -parameter) for the optimal single-metric algorithm. At each iteration, the search determines the optimal scale for the current norm as well as for nearby norms to estimate the gradient. The spikes are due to random motion allowed by the search algorithm. It always settled back down to the optimum near 0.7

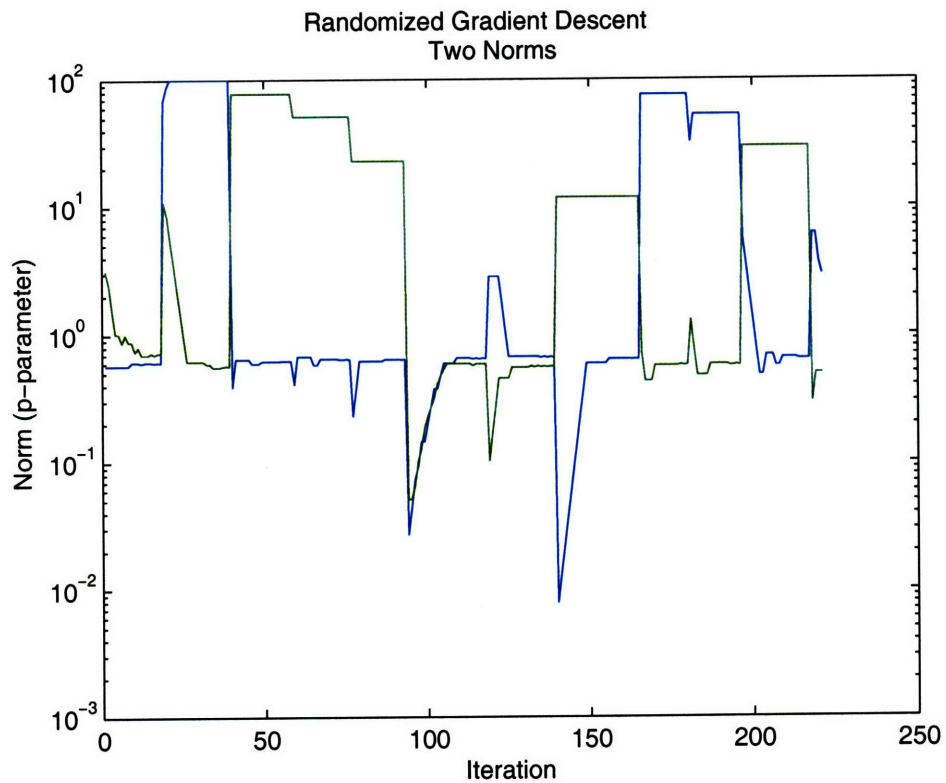


Figure 4-4: The chronological progression of the norms for the optimal two-metric algorithm. At each iteration the search determines optimal weights and scale for the current norms as well as for nearby norms to estimate the gradient. The spikes are due to random motion allowed by the search algorithm. Flat areas far from the optimum found in the single-metric case belong to a norm whose relative weight has gone to zero.

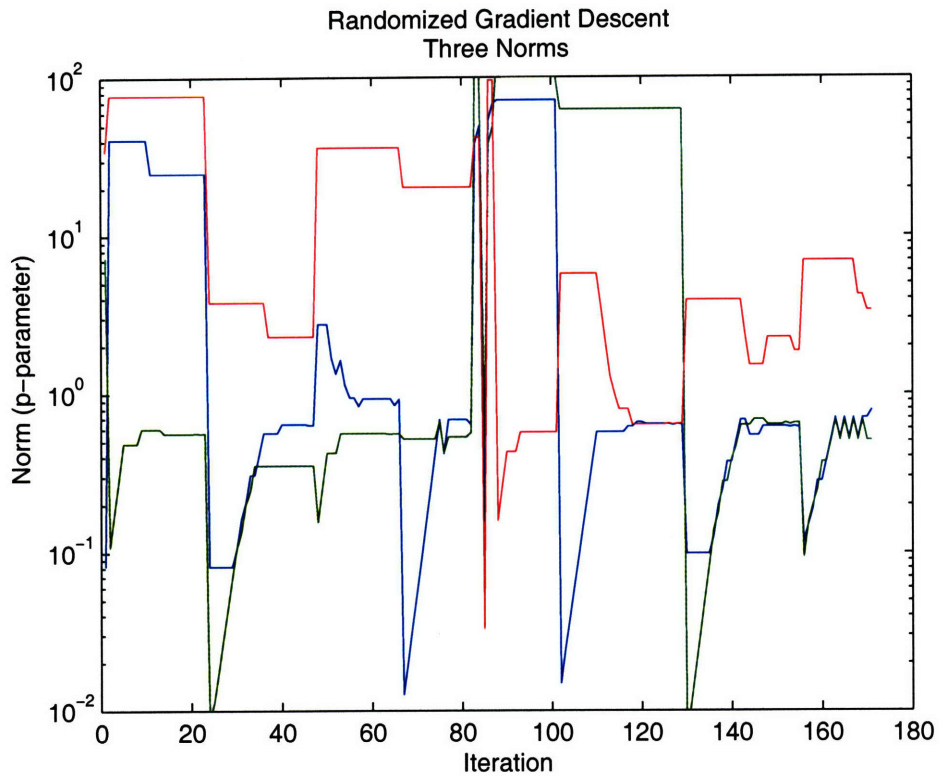


Figure 4-5: The chronological progression of the norms in the optimal three-metric error-bounding algorithm. The spikes are due to random motion allowed by the search algorithm. Flat areas far from the optimal single-metric optimum belong to a norm whose relative weight has gone to zero.

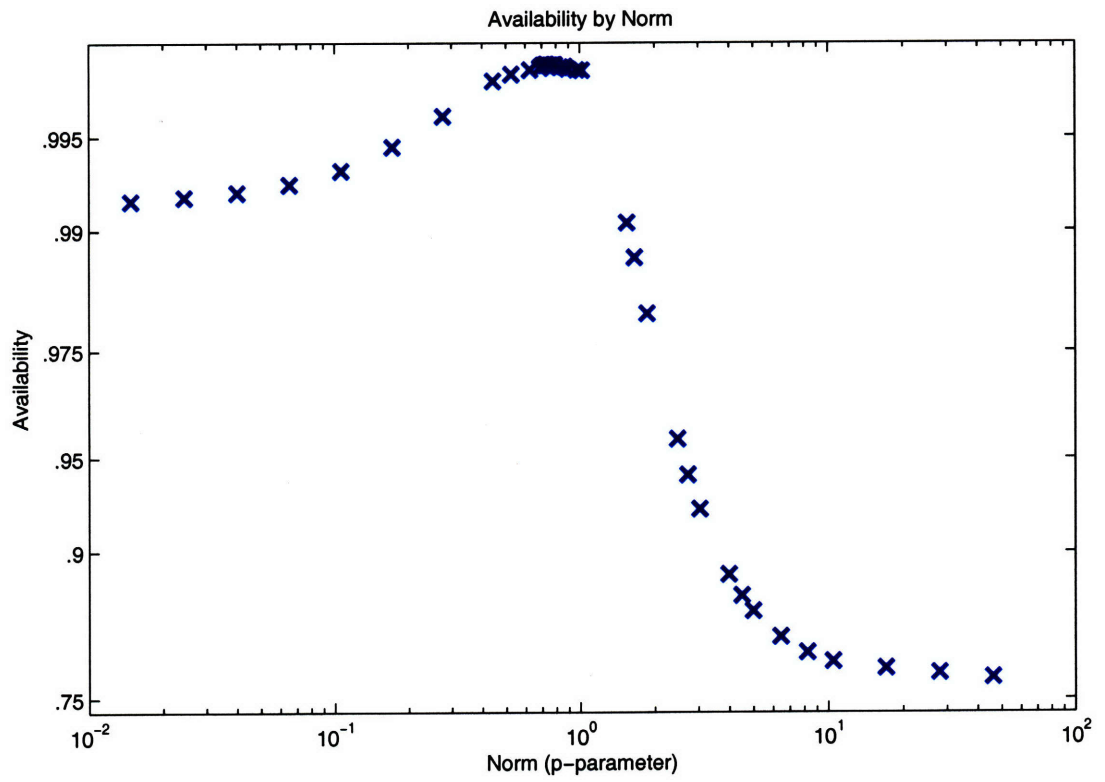


Figure 4-6: Level of availability by norm ( $p$ -parameter) for zero under-estimates and an alert limit of 1.2 meters.

### 4.1.5 Scale

The scaling function encapsulates the effect of the number of satellites visible. If we used a fixed scale as in [17] then we would have to be so conservative as to treat cases when there are 25 satellites visible the same as cases when there might be only 13. Figure 4-7 shows the relationship between rate of under-estimation and false alarms as a function of the number of satellites visible.

The data set on which the Figure 4-7 is based consisted of over half a billion snapshots from the uniform pseudorange error model. One can imagine a smoothed version which has some parametric form and whose parameters can be determined using some notion of best fit. Such an equation could be used to determine the algorithm we might use for *any* requisite level of missed detections.

We will concern ourselves, however, with one in particular, the  $10^{-7}$  level. Assuming a  $10^{-2}$  unconditional “failure” rate (which we consider *extremely* conservative), this would put the missed detection rate at  $10^{-9}$  which would meet meet the integrity requirements for Category III precision approach. At this level, there is enough data to confidently estimate parameters, while at more extreme rates we are much more at the mercy of singular points. Notably, a  $10^{-7}$  miss rate meets the requirement for Category I approach, even if the pseudorange errors were drawn from an arbitrarily wide uniform distribution *all of the time*.

We determined the optimal scale using one third of the data, 193 million snapshots, and reserved the other two thirds for validation. The scale function could be a simple table look-up of the optimal values that we’ve found. However, we have no reason to suspect that this will generalize well. Rather, we might try to find a simple functional form that is a good representation. The optimal values found in training are plotted in Figure 4-8. We could attempt to fit any function we would like to this curve.

We need not limit ourselves to any notion of least squares fit. Times when the

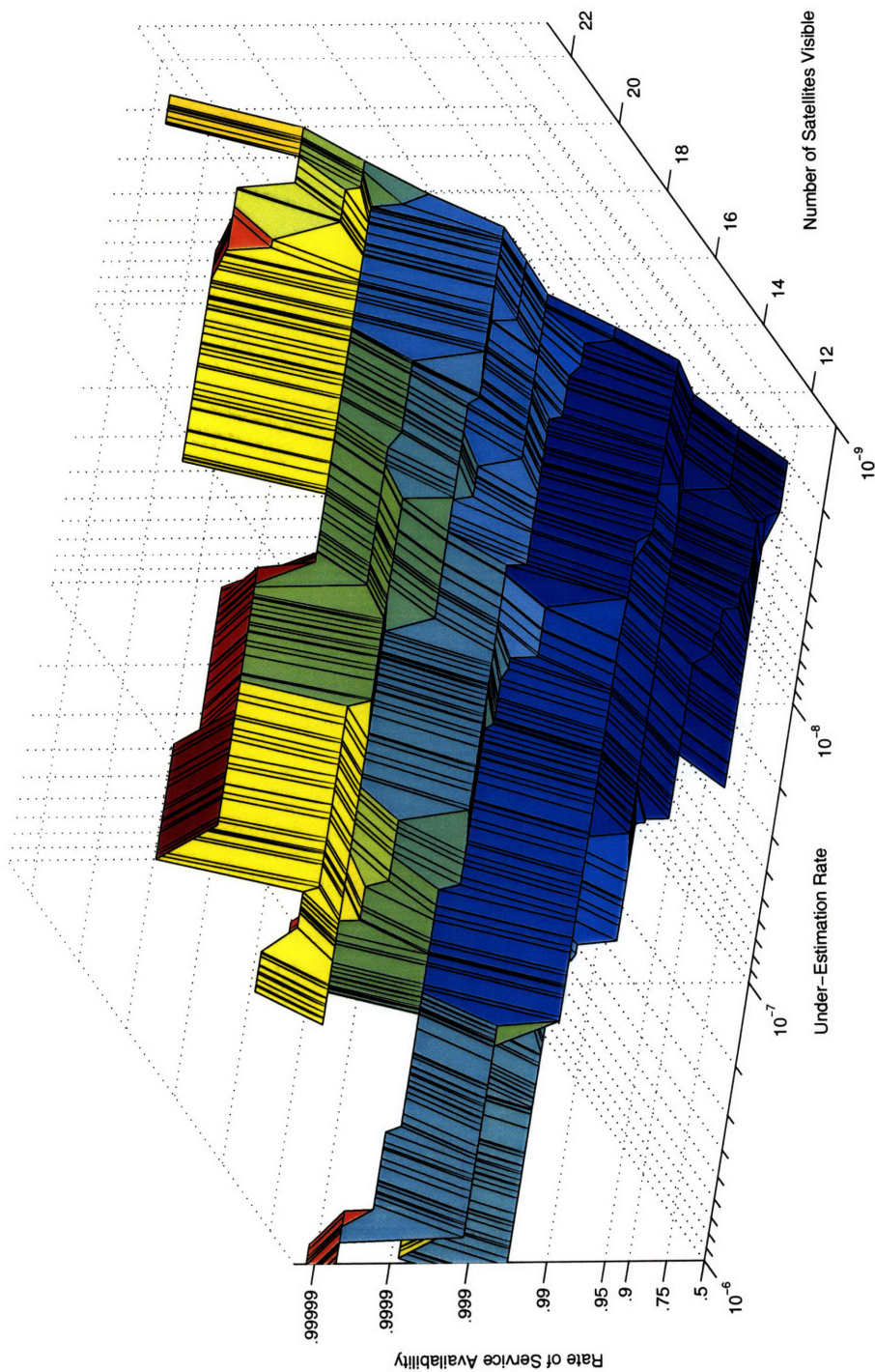


Figure 4-7: The trade-off between availability and under-estimation by the number of satellites visible. The vertical scale is logarithical so the values of 100% availability are not shown. For perspective, the surface is roughly planar sloping down toward the front corner. The availabilities shown are for Category II/III, *i.e.*, an alert limit of 5 meters and pseudorange errors drawn from  $0.9\mathcal{N}(0, 0.14m) + 0.1\mathcal{N}(0, 0.42m)$ . The under-estimation rate shown is the rate in both training and validation data with pseudorange errors drawn from a uniform distribution. For more than 22 satellites, availability was 100%.

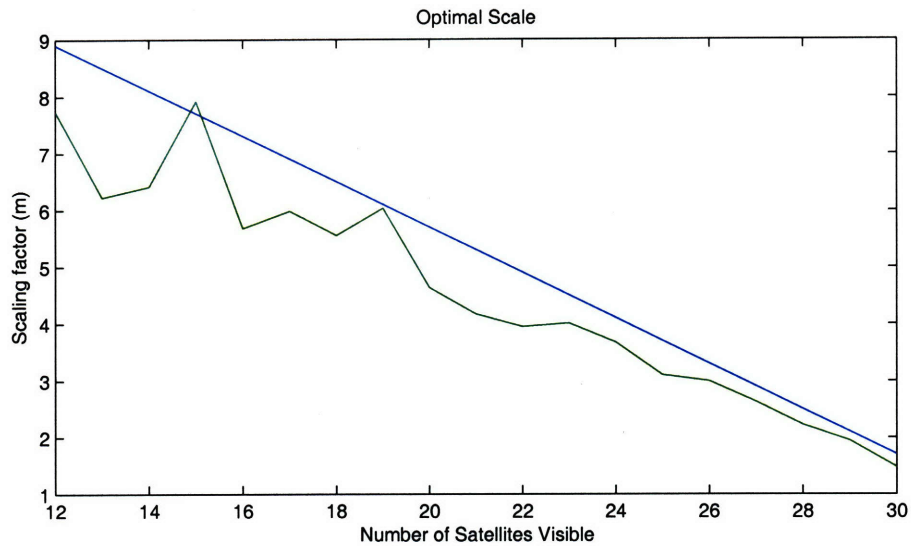


Figure 4-8: The optimal values for the scale parameter,  $\alpha$ , determined in training. Also shown is the linear approximation used for validation. The line is offset to allow for more confidence in generalization.

scale is below the optimal value are riskier than times when it is above. We needn't be *overly* cautious in this regard however since the choice of scale itself attempts to balance this trade-off. Our primary concern is that the generalization error will include a substantial number of under-estimates. It is possible, likely in fact, that if we fit the scale too closely we could see several times the number of under-estimates than we saw in training. For this reason we need to be pessimistic when fitting a function to the optimal scale values that we've found.

The function that we chose is simply

$$\alpha(k) = 13.7 - 0.4k \quad (4.1)$$

which is the least-squares linear fit plus an offset and is shown in Figure 4-8. The use of a least-squares fit in this situation is arbitrary. A linear fit is a good choice since it

increases the model order the least. To determine the offset consider Figure 4-9. The offset was determined by fitting a plane to Figure 4-9 and projecting one a factor of 16 further in the “number of trials” direction. The value 16 is somewhat arbitrary and comes from the fact that the tails of the binomial CDF are exponential and  $e^{-16} \approx 10^{-7}$ , *i.e.*, the confidence we would like to have.

## 4.2 Algorithm Summary

The optimal algorithm determined by the search is summarized as follows. Let,  $k$  denote the number of ranges available.

$$\text{Vertical Error Bound} = (13.7 - 0.4k)\text{VDOP} \cdot \frac{\|R\|_1}{k - 4} \quad (4.2)$$

where

$$\text{VDOP} \equiv (e_3'(G'G)^{-1}e_3)^{1/2}$$

$$\text{letting } e_3 \equiv \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix},$$

and where  $R$  denotes the adjusted residuals, described in Section 2.2.4. Letting,  $G_i$  denote the  $i^{\text{th}}$  row of  $G$ ,

$$R_i = (G_i\hat{x} - \rho_i)(G_i(G'G)^{1/2}e_3). \quad (4.3)$$

The algorithm is applicable if there are 12 or more satellites in view, while this equation holds with up to 30 satellites are in view. With fewer than twelve (which has a probability of less than  $10^{-7}$  in a 54 satellite constellation) service is unavailable.

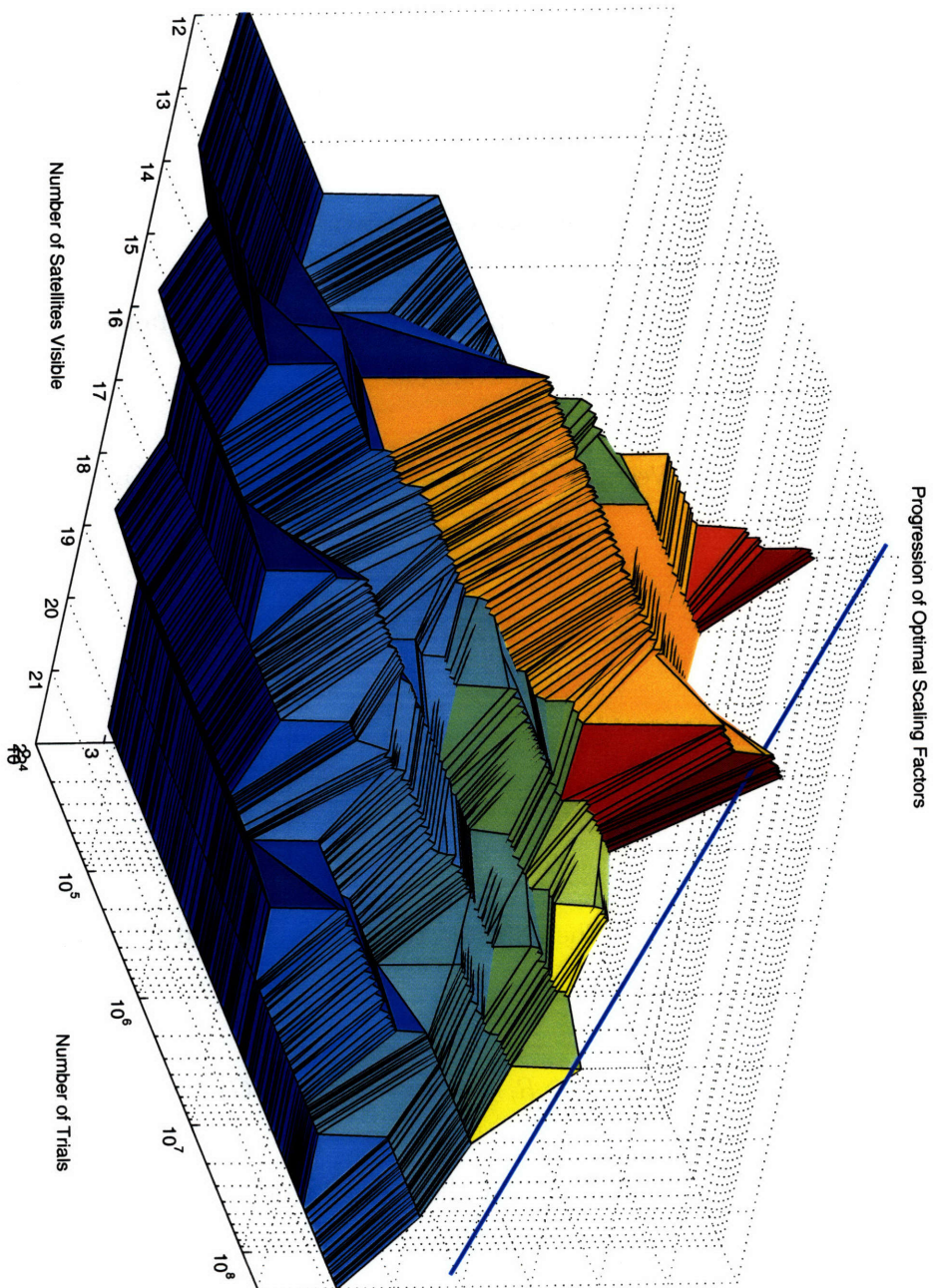


Figure 4-9: The optimal scaling factors increase as we attempt to avoid under-estimation on increasingly difficult cases. For perspective, the surface is roughly planar sloping up to the top corner. The line drawn is the scaling function. A planar fit to this surface was projected to 1.6 billion trials to determine the offset of the line.

Category	Alert Limit (m)	Integrity per Approach	False Alarm
I	10	$2 \times 10^{-7}$	$10^{-3}$ — $10^{-5}$
II	5.3	$10^{-9}$	$10^{-3}$ — $10^{-5}$
III	5.3	$10^{-9}$	$10^{-3}$ — $10^{-5}$

Table 4.1: Precision Approach Integrity Requirements

With more than 30, we fix  $\alpha$  at a value of 1.7 for which we are assured perfect availability under nominal conditions.

We can write Equation 4.2 with minimal notation abuse if we define the absolute value of a vector as the absolute values of it’s components.

$$\text{Vertical Error Bound} = \frac{13.7 - 0.4k}{k - 4} (e_3'(G'G)^{-1}e_3)^{1/2} |G\hat{x} - \rho'| |G(G'G)^{1/2}e_3| \quad (4.4)$$

### 4.3 Validation Results

The search described above finds the algorithm that performs best on a particular data set. To estimate how well that algorithm will perform on future data, we need to test it on a new data set. In particular if we wish to show that the algorithm satisfies the FAA’s integrity requirements shown in Table 4.1.

One would think that the binomial distribution would apply to the training data *i.e.*, if we have 160 million data points we should be able to estimate the probability of failure at the 1 per 10 million level with an accuracy of roughly  $e^{-16}$ . This is not the case. In the training data, the cases aren’t independent given the algorithm. For instance, if we know that the algorithm is simply a tightest fit to a given set of data points (support vectors), those points are not independent of the rest of the data — in particular we know that all the other data points are *not* support vectors and are

therefore more likely to be far from the decision surface. In fact, in the case just described, the probability that our estimate is too optimistic is more like 1/16. Since this is unsatisfactory, we must use independent data.

It was for this reason that we reserved two-thirds of our data, 387 million snapshots for validation. The optimal algorithm determined by the search was then run on this unseen data. On the validation data set there were only four under-estimates. This suggests that the probability of under-estimation is roughly  $10^{-8}$  although with so few samples, this could be an under-estimate. The relevant portion of the binomial CDF for the probability of under-estimate is shown in Figure 4-10. The y-axis tells the probability that the true probability of under-estimation is greater than the estimated probability of overestimation shown on the x-axis. Notably the curve crosses the  $x = y$  line at  $6.8 \times 10^{-8}$ , which is the point where our confidence in our under-estimation rate is equal to the under-estimation rate itself.

For validating the false alarm rate we generated new data for a range of constellation sizes and using a range of pseudorange error distributions similar to the one in Figure 3-2. All were Gaussian mixtures of the form  $0.9\mathcal{N}(0, xm) + 0.1\mathcal{N}(0, 3xm)$ . Tables 4.2 and 4.3 show the false alarm rate for for different sized nominal error distributions in different constellations. From Table 4.2 it is clear that we can meet the requirements of Category I landing with a minimum constellation and less accurate ranges than currently available. Table 4.3 shows that we should be able to meet the requirements of Categories II and III as well. These requirements can be met by some combination of maintaining high-quality ranging (*e.g.*, the green curve in Figure 3-2) or a 60+ satellite constellation.

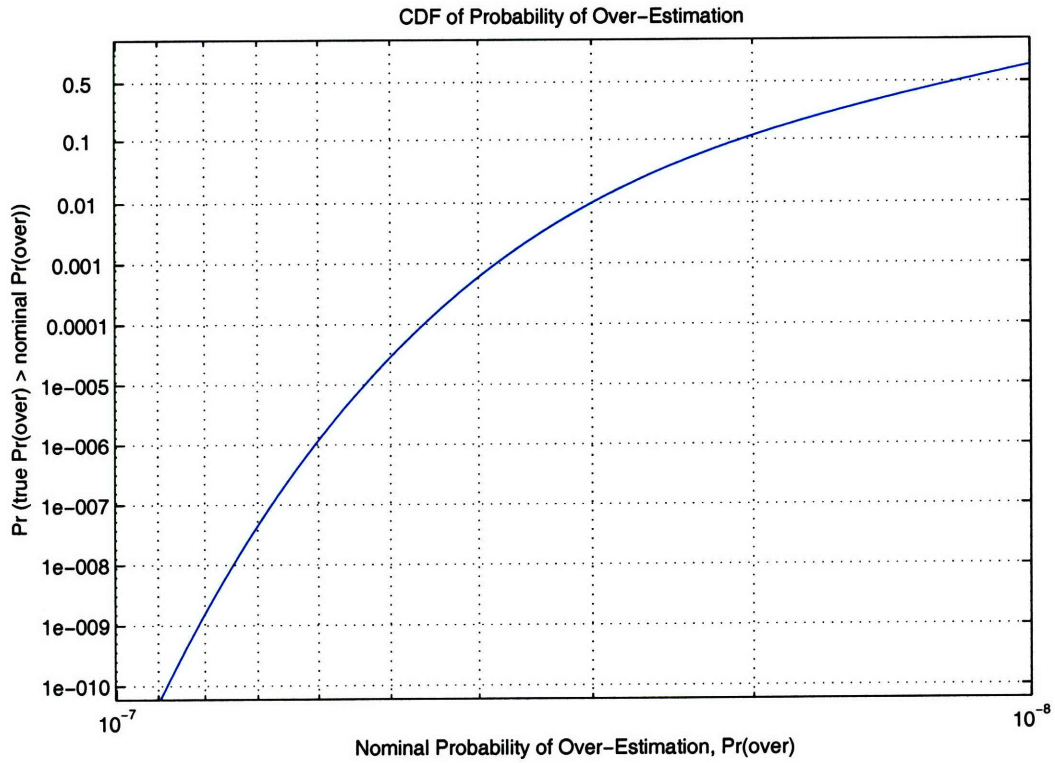


Figure 4-10: A close-up on the CDF of under-estimation rate. The abscissa shows nominal values for the probability of under-estimation while the ordinate gives the probability that the true value is less than the nominal. This curve comes from the binomial distribution for validation consisting of 4 under-estimates out of 387 million trials.

Distribution	Walker Constellations				
	54/6/1	56/7/1	60/6/1	63/7/1	64/8/1
$0.9\mathcal{N}(0, 0.14m) + 0.1\mathcal{N}(0, 0.42m)$	<1E-6	<1E-6	<1E-6	<1E-6	<1E-6
$0.9\mathcal{N}(0, 0.16m) + 0.1\mathcal{N}(0, 0.48m)$	8E-6	1E-6	<1E-6	<1E-6	<1E-6
$0.9\mathcal{N}(0, 0.18m) + 0.1\mathcal{N}(0, 0.56m)$	8E-5	1E-5	1E-6	<1E-6	<1E-6
$0.9\mathcal{N}(0, 0.20m) + 0.1\mathcal{N}(0, 0.60m)$	2.7E-4	7E-5	1E-5	3E-6	3E-6

Table 4.2: Category I False Alarm Rates

Distribution	Constellation				
	54/6/1	56/7/1	60/6/1	63/7/1	64/8/1
$0.9\mathcal{N}(0, 0.08m) + 0.1\mathcal{N}(0, 0.24m)$	2E-6	<1E-6	<1E-6	<1E-6	<1E-6
$0.9\mathcal{N}(0, 0.10m) + 0.1\mathcal{N}(0, 0.30m)$	1.6E-4	3E-5	5E-6	1E-6	<1E-6
$0.9\mathcal{N}(0, 0.12m) + 0.1\mathcal{N}(0, 0.36m)$	1.2E-3	4.7E-4	1.4E-4	5E-5	3E-5
$0.9\mathcal{N}(0, 0.14m) + 0.1\mathcal{N}(0, 0.42m)$	6.2E-3	3.1E-3	1.1E-3	4.41E-4	3.0E-4
$0.9\mathcal{N}(0, 0.16m) + 0.1\mathcal{N}(0, 0.48m)$	2.0E-2	1.1E-2	5.5E-3	2.6E-3	1.9E-3

Table 4.3: Category II/III False Alarm Rates

# Chapter 5

## Conclusions

We have determined an error-bounding algorithm that meets the integrity requirements of Category I, II and III precision approach using the combined GPS + Galileo constellation. This algorithm is reproduced as Algorithm 5.1. By means of search, as described in Chapter 3, we determine that this algorithm provides near optimal availability at this level of integrity. We show that this level of availability also meets the requirements of the aviation community.

$$\text{Vertical Error Bound} = \frac{13.7 - 0.4k}{k - 4} (e_3'(G'G)^{-1}e_3)^{1/2} |G\hat{x} - \rho'| |G(G'G)^{1/2}e_3| \quad (5.1)$$

We can be extremely confident of Algorithm 5.1 having an under-estimation rate of no more than  $6.8 \times 10^{-8}$  as long as the pseudorange errors are drawn independently. If we assume a one to one-hundred ratio of under-estimation to missed detection we can promise a missed detection rate of  $6.8 \times 10^{-10}$  which is sufficient for all approaches. Furthermore, we consider such an assumption *extremely* conservative. For compari-

son, [12] assumed unconditional rate of  $10^{-4}$  for faults, while only a small fraction of our under-estimates would correspond to faults. This is because for error-bounding, the errors must not only be abnormally large—“faulty” in the traditional sense—they must also not be so large that the bound is larger than the alert limit despite being an under-estimate. Recall that only a tiny wedge of the under-estimates in Figure 1-1 correspond to missed detections.

Tables 4.2 and 4.3 list the false alarm rates for realistic signal environments. It should be noted that a combined GPS + Galileo constellation will show slightly higher variance in the number of satellites visible, which will result in slightly lower availability. For the minimum constellation and nominal errors conforming to Figure 3-2 we can easily meet all the requirements for Category I approach. For Categories II and III, availability is adequate with high quality ranges or a non-minimum constellation. Table 4.3 demonstrates that this availability will rapidly improve with even a small improvement in range accuracy.

This algorithm gives a quantitative basis for fault exclusion. If by excluding a satellite the error-bound improves, it would be advantageous to exclude that satellite as “faulty”. Also, there is now a quantitative valuation for the additional confidence that can be bought by adding new sources of ranges.

Finally a more fundamental conclusion is that the optimal norm is *not* the  $l_2$  norm which has hitherto been the standard for snapshot methods. This suggests that traditional least squares techniques have been holding us back. In particular, the position estimate with the minimum  $l_1$  norm is perhaps a better estimate than minimum  $l_2$  norm.

## 5.1 Future Work

The assumption that needs further consideration is the assumption of error independence. We have been working under the assumption that differential corrections eliminate nearly all spatially correlated errors. As local differential GPS is more broadly deployed and its details settled upon, this assumption must be assessed empirically.

The availability improves very rapidly as more ranges become available. If in the future satellites closer to the horizon can be used, this will allow us to tighten the error bound and improve availability. However, it is known that the errors tend to be larger from such low elevation-angle satellites. In such a case we must reconsider whether pseudorange errors can be safely considered identically distributed. Similarly, we do not consider the possibility of mixed range sources. If there are additional sources like pseudolites or a barometric sensor, those could be included as well. These also have errors from different distributions. In the position equations, the rows of the geometry matrix,  $G$ , corresponding to the different sensors are typically scaled so that the errors are of similar magnitude. However we treat every row of the geometry matrix as if it were the same. Introducing this scale could affect our results.

We do not consider the possibility of bias since the error common to all the pseudoranges ends up in the clock and does not concern us. However, if this algorithm were to be incorporated in a receiver with substantial time information *e.g.*, alongside the clock smoothing technique of [4], this bias needs to be considered.

Although this work only considers snap-shot algorithms, as opposed to filtering algorithms, a metric error-bound can also be useful dynamically without making broad assumptions about the auto-covariance of the measurement error. For example, suppose that a short time ago the error-bounding algorithm assured you that you were within 1 meter of the estimate, but now it can only promise that you are within 15 meters of the estimate. Suppose also that from inertial sensors, you know that the

error could not have grown so quickly. Instead of using the 15 meter error-bound, you could propagate the 1 meter error-bound forward according to the maximum error-rate of your inertial sensors[7]. This has similar properties to the Kalman filter approach of [6], but avoids many the distribution assumptions inherent in Kalman filtering although it still requires knowledge of the temporal correlation of the range errors.

This work raises important new questions. Primarily it raises questions about the validity of least-squares techniques in unknown, non-Gaussian distributions. We show that the median is a more sure estimate than the mean but we also suggest that a concave norm might be a still better measure of fit. Future work will explore this possibility and will likely have to address the analytical ramifications of such a discovery.

# Bibliography

- [1] Federal radionavigation systems. US Departments of Defense and Transportation, 2001.
- [2] Minimum operational performance standards for GPS local area augmentation system airborne equipment. RTCA Inc, Washington, DC, 2001.
- [3] Federal radionavigation plan. US Departments of Defense and Transportation, 2005.
- [4] BEDNARZ, S., AND MISRA, P. Receiver clock-based integrity monitoring for GPS precision approaches. *IEEE Trans on Aerospace and Electronic Systems* 42, 2 (2006).
- [5] BLANCH, J., WALTER, T., AND ENGE, P. Protection level calculation in the presence of heavy tail errors using measurement residuals. In *Proc of the European Navigation Conference GNSS* (July 2005).
- [6] BRENNER, M. Integrated GPS/inertial fault detection availability. *Navigation* 43, 2 (1996).
- [7] BROOKS, R. A. Symbolic error analysis and robot planning. *Int. J. Robotics Research* 1, 4 (1982).

- [8] BROWN, R. G., AND CHIN, G. Y. GPS RAIM: Calculation of threshold and protection radius using chi-square methods. In *Global Positioning System (Red Book Series)*, vol. 4. The Institute of Navigation, Alexandria, VA, 1998.
- [9] BROWN, R. G., AND MCBURNEY, P. W. Self-contained GPS integrity check using maximum solution separation. *Navigation* 35, 1 (1998).
- [10] DECLEENE, B. Defining pseudorange integrity-overbounding. In *Proc of the Institute of Navigation-GPS* (2000).
- [11] DIVIS, D. A. Augmented GPS: All in the family. *GPS World* (April 2003).
- [12] HWANG, P., AND BROWN, R. G. RAIM FDE revisited: A new breakthrough in availability performance with NIORAIM (novel integrity-optimized RAIM). In *Proc of the Institute of Navigation Technical Meeting* (January 2005).
- [13] LEE, Y. Analysis of range and position comparison methods. In *Proc of the Institute of Navigation* (June 1986).
- [14] LEE, Y. Performance of receiver autonomous integrity monitoring (RAIM) in the presence of simultaneous multiple satellite faults. In *Proc. of the 60th Annual Meeting of the Institute of Navigation* (June 2004).
- [15] LEVINE, J. H., AND ROOS, T. B. *Introduction to Data Analysis*. On-line Manuscript, 2002. <http://www.dartmouth.edu/~jlevine/>.
- [16] LUC, D. T., AND VOLLE, M. Level sets infimal convolution and level addition. *Optimization Theory and Applications* 94, 3 (1997).
- [17] MISRA, P., AND BEDNARZ, S. Robust integrity monitoring. *GPS World* (April 2004).

- [18] MISRA, P., AND ENGE, P. *Global Positioning System*, second ed. Ganga-Jamuna Press, 2006.
- [19] PARKINSON, B. W., AND AXELRAD, P. Autonomous GPS integrity monitoring using the pseudorange residual. *Navigation* 35, 2 (1988).
- [20] PULLEN, S., AND PARKINSON, B. A new approach to GPS integrity monitoring using prior probabilities and optimal threshold search. In *Proc. of IEEE PLANS* (1994).
- [21] RIFE, J., AND PULLEN, S. Core overbounding and its implications for LAAS integrity. In *Proc. of the Institute of Navigation-GNSS* (2004).
- [22] WALTER, T., AND ENGE, P. A proposed integrity equation for WAAS MOPS. In *Proc. of the Institute of Navigation GPS-97* (September 1997).