Collaborative Localization Enhancement to the Global Positioning System using Inter-Receiver Range Measurements

Zachary J. Biskaduros

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Electrical Engineering

R. Michael Buehrer, Chair
Jeffery H. Reed
J. Michael Ruohoniemi

May 2, 2013
Blacksburg, Virginia

Keywords: localization, geolocation, collaborative position location, collaborative global positioning system

Collaborative Localization Enhancement to the Global Positioning System using Inter-Receiver Range Measurements

Zachary J. Biskaduros

ABSTRACT

The localization of wireless devices, e.g. mobile phones, laptops, and handheld GPS receivers, has gained much interest due to the benefits it provides, including quicker emergency personnel dispatch, location-aided routing, as well as commercial revenue opportunities through location based services. GPS is the dominant position location system in operation, with 31 operational satellites producing eight line of sight satellites available to users at all times making it very favorable for system implementation in all wireless networks. Unfortunately when a GPS receiver is in a challenging environment, such as an urban or indoor scenario, the signal quality often degrades causing poor accuracy in the position estimate or failure to localize altogether due to satellite availability.

Our goal is to introduce a new solution that has the ability to overcome this limitation by improving the accuracy and availability of a GPS receiver when in a challenging environment. To test this theory we created a simulated GPS receiver using a MATLAB simulation to mimic a standard GPS receiver with all 31 operational satellites. Here we are able to alter the environment of the user and examine the errors that occur due to noise and limited satellite availability. Then we introduce additional user(s) to the GPS solution with the knowledge (or estimate) of the distances between the users. The new solutions use inter-receiver distances along with pseudoranges to cooperatively determine all receiver location estimates simultaneously, resulting in improvement in both the accuracy of the position estimate and availability.

Acknowledgments

First, I want to acknowledge my advisor Dr. Michael Buehrer for his support and friendship. His guidance, help, and dedication helped me achieve my Master's, as well as a soccer championship for BCF in the New River United League.

Thanks to all the current and past professors at Virginia Tech, especially to Dr. Claudio da Silva for his exceptional teaching and encouragement to further my education and to Dr. Jeffrey Reed and Dr. J. Michael Ruohoniemi for their teachings and assistance in finalizing this thesis.

Finally, thanks to all students at Virginia Tech who gave a helping hand, especially my colleagues at Wireless@VT including Jacob Overfield, William Rogers, Javier Schloemann, Daniel Jakubisin, SaiDhiraj Amuru, and Jeffrey Poston for their support and research.

Dedication

This thesis is dedicated to my parents, Jim and Wanda Biskaduros, my brother, Eric Biskaduros, and my girlfriend, Rebecca Maine. I would not have been able to get to the point where I am today without their love, encouragement, and undying support. Love you all and thank you!

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The localization of wireless devices, e.g, mobile phones, laptops, and handheld GPS receivers, has gained much interest due to the benefits it provides, including quicker emergency personnel dispatch, location-aided routing, as well as commercial revenue opportunities through location-based services. Before we begin the discussion of how localization occurs we will define the term *node* as any wireless device that is used in localization, *anchor* as a node with a known location, and *unlocalized node* as a node without a known location and attempting to estimate its location. Therefore, in order to find the position estimate of an unlocalized node, we must first process a signal to collect measurements from an anchor (a node with a known location), or more likely a set of anchors. The parameters of the signal that can be used for geolocation include signal strength, time of arrival estimates, angle of arrival estimates, frequency estimates, or various hybrid combinations of the parameters [2]. Then using one of the optimal localization solutions discussed in Chapter 2.2 we are capable of achieving a position estimate.

Unfortunately, these collected measurements can be difficult to obtain in challenging environments such as urban or indoor environments, or can even introduce large noise components causing a poor position estimate. Thus several solutions have been developed to improve performance in such environments, one in particular being collaborative localization. Collaborative localization is when unlocalized nodes use the measurements between unlocalized

nodes in addition to the measurements between unlocalized nodes and anchors in order to estimate their positions simultaneously, in hopes of improving both the accuracy and availability of the position estimates.

This thesis focuses on improving the most prominent localization system in operation, the Global Positioning System (GPS), by introducing a new collaborative solution. The new solution uses inter-receiver distances between unlocalized nodes, along with pseudoranges (range measurements with a time offset) to cooperatively determine all receiver locations simultaneously, resulting in improvement in both the accuracy of the position estimate and availability.

1.1 Thesis Summary

In this section, each chapter is summarized with the final conclusion in Chapter 6 and back up data provided in the Appendices.

Chapter 2: The Non-Collaborative Geolocation Problem

The non-collaborative problem is discussed in Chapter 2. This chapter examines the various signal parameters used to collect measurements and examines the optimal solutions used for localization including the least squares estimator, the maximum likelihood estimator, and the maximum a posteriori probability estimator. To close, we explore the implementation of localization in commonly used systems, such as the Global Positioning System and mobile phones.

Chapter 3: The Collaborative Geolocation Problem

The collaborative problem is introduced in Chapter 3. This chapter examines the various types of collaborative localization and their classifications. This chapter also examines the optimal and sub-optimal solutions used for collaborative localization.

Chapter 4: Collaborative GPS

Chapter 4 introduces our new collaborative GPS solution to improve the accuracy and availability of a GPS position fix. The new collaborative solution uses the measurements from global positioning system satellites in cooperation with inter-receiver measurements. We begin by introducing a MATLAB simulation to mimic a standard GPS receiver with all 31 operational satellites for comparison purposes.
Next we examine and implement three of the most commonly used solutions for localization of a GPS receiver: the Newton-Raphson solution, the Extended Kalman Filter, and the Unscented Kalman Filter for comparison purposes. This lets us introduce our three new collaborative solutions: the Newton-Raphson-based Collaborative solution, the Collaborative Extended Kalman Filter, and the Collaborative Unscented Kalman Filter.

Chapter 5: Collaborative GPS Performance

The results of the MATLAB simulation are shown in Chapter 5 where we analyze and compare the effects of collaboration using the six solutions, three standard and three collaborative, previously discussed.

Contributions

The contributions of our work include the development of three Collaborative GPS solutions based on the three most popular GPS solutions and the development of a MATLAB simulation to analyze and compare the effects of various environmental scenarios for a GPS receiver and the improvements made through collaborating receivers.

1.2 Problem Formulation

Consider the following examples as two-dimensional where $\mathbf{\Theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, ..., \boldsymbol{\theta}_n]$ is a set of node locations in two-dimensions, where $\boldsymbol{\theta}_i = [x_i, y_i]^\top$, and $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_m]$ are the locations of

the anchors associated with those nodes (nodes with known locations), and $\mathbf{a}_i = [x_{s_i}, y_{s_i}]^\top$. Thus, we can relate our measurements $\mathbf{z}$ as a function of the positions between the anchors and unlocalized nodes plus noise, seen below:

$$\mathbf{z} = f(\boldsymbol{\Theta}, \mathbf{A}) + \boldsymbol{\eta} \tag{1.1}$$

where $f(\boldsymbol{\Theta}, \mathbf{A})$ is the non-linear function of the positions of the anchors and unlocalized nodes, typically an estimated range as seen in Chapter 2.1, and $\boldsymbol{\eta}$ is the measurement noise. Our goal is to estimate the positions of all the unlocalized nodes, $\boldsymbol{\Theta}$, using a set of N measurements.

The metrics focused on via simulations will be the accuracy and the availability. The accuracy can be defined as the mean error associated with the receiver's position, found by calculating the distance from the estimated position to the predetermined true location, $E|\mathbf{x} - \hat{\mathbf{x}}| = \frac{1}{N}\sum_{i=1}^{N} |\mathbf{x} - \hat{\mathbf{x}}|$. The availability can be defined as the number of occurrences in which a position estimate is localized. The availability fails when the number of satellites is less than four (or four minus the number of additional users in collaboration), if error is greater than 1 km, or the absolute value of the geometric dilution of precision (GDOP) is greater than 1000 due to a diverging solution.

Chapter 2

The Non-Collaborative Geolocation Problem

2.1 Signal Parameters and Measurements

In this section, the different types of measurements typically used in the localization problem are described. Additionally, the measurement models for 2-D coordinates are described, and the extension for 3-D coordinates is straightforward.

2.1.1 Time of Arrival

The time of arrival (TOA), also known as time of flight (TOF), signal parameter is the measurement of the distance between two nodes using precise timing and the signal's propagation time. TOA is one of the most prominently used signal parameters in position location, with the Global Positioning Systems (GPS) being the most prominent example. Using a signal's transmit time and receive time, we are able to calculate the range between nodes using the equation $d = vt$, where d is the distance, or range, v is the constant velocity of RF signal propagation and t is the travel time. Relating it to a free space model, we can assume the velocity is c (the speed of light, which is the propagation speed of an electromagnetic wave) and Δt is the time between when the signal was received and transmitted, thus $d = c\Delta t = c(t_r - t_t)$.

However, one must note that in order to perform range estimation in this manner, the transmitting and receiving nodes must have highly accurate clock synchronization, for a clock error of 1 μs can lead to a range error of 300 meters [2]. This synchronization has proven difficult to achieve, depending on the scenario and can lead to significant overall positioning errors. Additionally, the geometry and distance of the anchors plays an important role in the accuracy of this signal parameter, for if the anchors are within too small of a distance then the clock offsets will produce significant errors in the range estimation. Therefore it is important to have the anchors spread with an ideal geometry of a tetrahedron to produce the best position estimate.

TOA estimation is typically facilitated by transmitting signals using a spread spectrum wideband pseudorandom noise (PRN) sequence allowing the unlocalized node to identify which anchor is transmitting and to accurately measure TOA. Furthermore, transmitting signals using spread spectrum helps the receiver distinguish between line-of-sight (LOS) and delayed non-line-of-sight (NLOS) multipath components that can affect the ranging errors. After the TOA-based range estimation has been found relative to an anchor location, we can narrow the possible positions of the unlocalized node to a sphere, given 3-D or a circle given 2-D, with the radius being the estimated range and the sphere/circle being centered at the anchor location. Using the intersect of four spheres, or three circles for 2-D, helps us narrow down the position estimate of the unlocalized node. The TOA estimate can be modeled as a range estimate as seen in the equation below.

$$f_i(\mathbf{\Theta}, \mathbf{A}) = \sqrt{(x - x_{s_i})^2 + (y - y_{s_i})^2} \tag{2.1}$$

When TOA has an unknown clock bias then we can refer to it as pseudorange. Pseudorange is the estimated range from a satellite to a receiver, however it does not require a clock synchronization reducing cost of the hardware. This concept will be explained further in the discussion of GPS in Chapter 4. The TOA estimate with an unknown clock bias can be

modeled as a pseudorange value as seen in the equation below.

$$f_i(\mathbf{\Theta}, \mathbf{A}) = \sqrt{(x - x_{s_i})^2 + (y - y_{s_i})^2} + c(t_{s_i} - t) \tag{2.2}$$

2.1.2 Time Difference of Arrival

The time difference of arrival (TDOA) signal parameter is the measurement of the difference of distance between nodes using precise timing and the times at which the signal is received. The range difference measurement is found either by using two anchor locations transmitting simultaneously to one unlocalized node or two anchors measuring the receive time of unlocalized node's transmission. Using the signals' received times we are able to calculate the TDOA with $\Delta d = d_2 - d_1 = c(t_{received_2} - t_{received_1})$. By using the difference of TOA equations, we eliminate the need for clock synchronization at the unlocalized node, however it does require all transmitting or receiving anchors to have synchronized clocks. Since TDOA signals are transmitted the same way as TOA, nearly all the same errors as TOA affect it, including the geometry of the anchors as well as multipath errors. The estimated TDOA range difference estimate relative to the anchor location(s) creates a hyperboloid function, given 3-D or a hyperbola given 2-D, about the anchor locations sweeping in all directions for possible unlocalized node locations. Unfortunately, a minimum of two measurements are typically needed to find an intersection or near an intersection in order to avoid ambiguity, as seen below in Fig. 2.1. The TDOA estimate can be modeled as a range difference estimate as seen in the equation below.

$$f_i(\mathbf{\Theta}, \mathbf{A}) = \sqrt{(x - x_{s_2})^2 + (y - y_{s_2})^2} - \sqrt{(x - x_{s_1})^2 + (y - y_{s_1})^2} \tag{2.3}$$

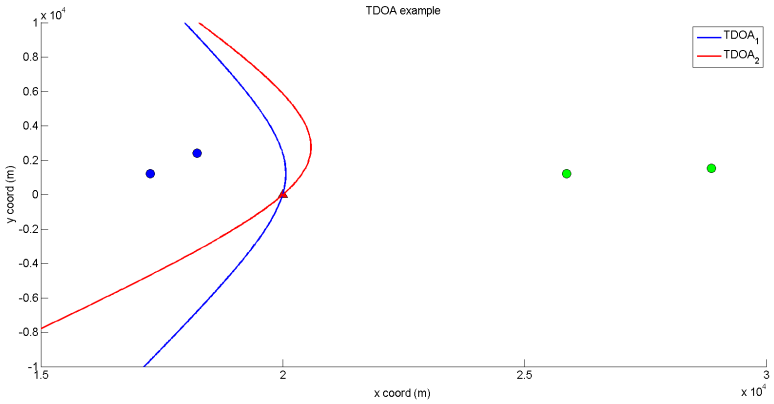


Figure 2.1: TDOA example

2.1.3 Frequency Difference of Arrival

The frequency difference of arrival (FDOA) signal parameter, also known as differential Doppler (DD), is the measurement of the difference of relative motion of the known node locations with respect to the unlocalized node. Using the known velocities and locations of the anchors, which cause an inherent Doppler shift in the received signals at the unlocalized node, the node's position can be estimated from the measured Doppler differences. This can be seen in the equation used to calculate the parameter, $\Delta f = \Delta f_1 - \Delta f_2 = \frac{f_0}{c}(\mathbf{v}_2^\top \mathbf{u}_2 - \mathbf{v}_1^\top \mathbf{u}_1)$, where $\mathbf{v}^\top$ is the transpose of relative velocity vector of the anchor node, $\mathbf{u}$ is the unit vector from each anchor node location to the estimated unlocalized node location, c is the speed of light, and f_0 is the carrier frequency. The FDOA Doppler shift estimate creates a complex quadratic function about the anchor node location(s) which sweeps in all directions of all possible target node locations. The curves that are created from FDOA are highly dependent on the geometry and velocities which can significantly alter the accuracy of the localization. This parameter is commonly used in parallel with TDOA to further reduce the positioning error, since the measurements are collected independent of each other [3]. Unfortunately because FDOA requires the transmitting nodes to be moving with known velocities, this limits the usefulness of this measurement. The FDOA estimate can be modeled as a frequency

difference estimate as seen in the equation below.

$$f_i(\mathbf{\Theta}, \mathbf{A}) = \frac{f_0}{c}(\mathbf{v}_2^\top \mathbf{u}_2 - \mathbf{v}_1^\top \mathbf{u}_1) \tag{2.4}$$

$$\mathbf{v}_i^\top \mathbf{u}_i = \frac{(v_x - v_{x_i})(x - x_i) + (v_y - v_{y_i})(y - y_i)}{\sqrt{(x - x_i)^2 + (y - y_i)^2}}$$

2.1.4 Received Signal Strength

Received Signal Strength (RSS), the simplest of the signal parameters, uses measured signal strength and signal propagation models to estimate the distance between nodes. The parameter uses knowledge of the signal powers at both the transmitter and the receiver along with a propagation model to determine the distance the signal has traveled from the transmitter to the receiver. This can be seen in the equation used to calculate the distance measurement, $d = d_o 10^{\frac{P_o - P_r}{10n}}$, where d_o is the reference distance, P_o is the power received at the reference distance, P_r is the observed received power, and n is the path loss exponent. Clock synchronization is not required for any of the nodes but the approach does rely on an accurate model of the signal's path loss versus distance. RSS is typically used for short distance range estimation only, since the estimation performance degrades over large distances due to shadowing and multipath fading caused by objects and obstructions (for power degrades with distance squared or worse and the signal's propagation is difficult to model in such a scenario). However, the advantage of using RSS is it is already readily available in most systems giving it low cost and complexity.

A variation of RSS, known as Differential Received Signal Strength (DRSS), allows one to disregard the transmit power by adding an extra transmitter (anchor) to the system. The parameter uses the ratio of the RSS measurements (difference in the log domain) from two different anchors transmitting at approximately the same time from their known locations or from one anchor at two different measurement locations in order to measure the distance to the unlocalized node [4].

If prior knowledge of the signal parameter is available then a type of fingerprinting (or

mapping) can be performed to localize. This uses a database of the previously measured parameters at their locations on a map that can be accessed to determine ones position based on recorded signal parameters. The advantage of fingerprinting is the several signal parameters can be measured and mapped and it has the ability to localize when strong shadowing and multipath components are present. Unfortunately, the main drawbacks are requiring prior access to the area of interest as well as enough resources for database storage. Furthermore, the fingerprinting needs constant updating due to changes that can occur in any environment making this unpractical in many implementations. The RSS estimate can be modeled as a range estimate as seen in the equation below, where α is a path loss component.

$$f_i(\mathbf{\Theta}, \mathbf{A}) = \alpha \ln(\sqrt{(x - x_{s_i})^2 + (y - y_{s_i})^2}) \tag{2.5}$$

2.1.5 Angle of Arrival

The angle of arrival (AOA) signal parameter is the measurement of the directions to the anchors rather than the range. If we have two antennas at the unlocalized node, the receiver can measure the phase difference of a signal at the two antennas in order to calculate the arrival direction using one of two techniques. The most common technique finds the direction by measuring the differences in arrival phases of the transmitted signal and using the known antenna array configuration. The other technique finds the direction by measuring the RSS ratio between multiple antennas (with different antenna patterns) on the unlocalized node [2]. In the presence of the measurement noise, the AOA measurements are modeled with noise as a zero-mean Gaussian random variable with a variance typically between 4 and 36 degrees which is dependent on the network deployment and distances [5].

While this parameter requires no clock synchronization and fewer anchors, the additional antenna requirement makes AOA much less practical by increasing cost and complexity of the system. Furthermore, in NLOS environments AOA can even be totally useless due to large noise components. The AOA estimate can be modeled as an angle estimate as seen in

the equation below.

$$f_i(\mathbf{\Theta}, \mathbf{A}) = \tan^{-1} \frac{y_s - y}{x_s - x} \tag{2.6}$$

2.1.6 Carrier Phase of Arrival

The phase of arrival (POA) signal parameter is the measurement of the distance between nodes using the carrier phase. This parameter requires all transmitting nodes to send pure sinusoidal signals of the same frequency with a phase lock between the anchor and unlocalized node in order to measure the phase [6]. The estimated range can be found using $d = \frac{c\phi_i}{2\pi f_0}$, where c is the speed of light, ϕ_i is the phase delay, and f_0 is the carrier frequency. The POA range estimation creates spheres centered at the anchor locations in order to narrow the possible position locations, like TOA.

A variation of this parameter is the phase difference of arrival (PDOA) approach which uses the phase difference between anchors to find an estimated range. Also, like TDOA, the found PDOA range estimation relative to the anchor location(s) creates a hyperboloid function about the anchor node location(s) sweeping in all directions for possible unlocalized node locations [7].

Unfortunately, like RSS, the POA and PDOA techniques only perform well over short distances because the unlocalized node needs a constant lock on the transmitting signals in order to continuously measure the phase. Due to this strict requirement and lack of practicality, these two parameters are rarely used. The POA estimate can be modeled as a range estimate as seen in the equation below.

$$f_i(\mathbf{\Theta}, \mathbf{A}) = \frac{c\phi_i}{2\pi f_0} = \sqrt{(x - x_{s_i})^2 + (y - y_{s_i})^2} \tag{2.7}$$

2.2 Optimal Localization Approaches

Reconsider our problem, previously defined in Chapter 1.2 where $\mathbf{\Theta}$ is the unlocalized node location and $\mathbf{A}$ are the locations of the anchors associated with those nodes. The measure-

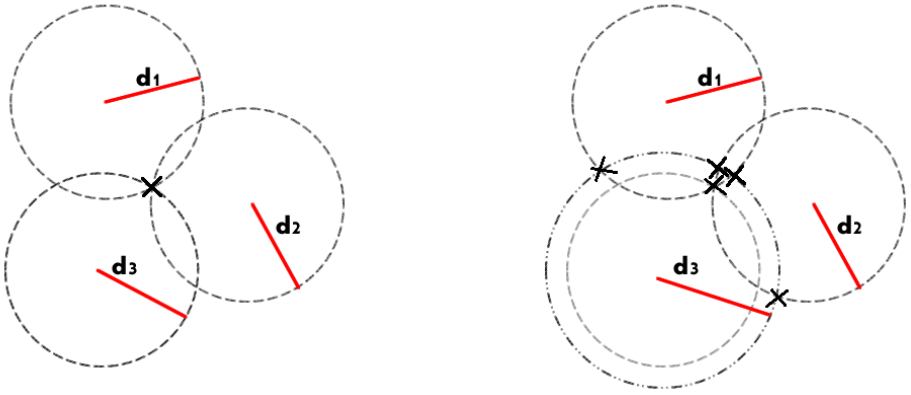
ments are a function of the node locations.

$$\mathbf{z} = f(\boldsymbol{\Theta}, \mathbf{A}) + \boldsymbol{\eta} \tag{2.8}$$

where $f(\boldsymbol{\Theta}, \mathbf{A})$ is the non-linear function of the positions of the anchors and unlocalized node (depending on the measurement taken) and $\boldsymbol{\eta}$ is the measurement noise. Our goal is to estimate the position of the unlocalized node, $\boldsymbol{\Theta}$, using one of two approaches. The most intuitive position location solution is the geometric solution, however when noise is present this solution is significantly hindered. Thus by using statistical methods, such as Bayesian's estimation or one of the classical estimations, we are able to localize in noisy environments.

2.2.1 Geometric Solution

The geometric method, the easiest to visualize, uses the intersections of the position lines that were found from the signal parameters in Chapter 2.1 to find a unique solution. The most commonly used geometric solutions are the trilateration and the triangulation techniques. Trilateration uses the range measurements from TOA, RSS, or POA to create circles centered at the known node locations, as shown in Fig. 2.2a. When noise is absent, a minimum of three range measurements are required in 2-D, or four measurements when in 3-D, in order to find a unique intersection and to avoid ambiguity. Triangulation is similar but uses direction estimates from AOA to determine the relative angles and to draw position lines to find the intersection, as visualized in Fig. 2.3. The advantage of triangulation is only two direction measurements are required in 2-D to obtain a unique solution versus three range measurements for trilateration.



(a) Trilateration technique without noise (b) Trilateration technique with noise

Figure 2.2: Trilateration techniques

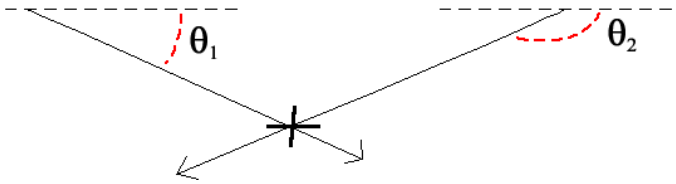


Figure 2.3: Triangulation technique

Using a technique called multilateration it is possible to solve for a unique position using the TDOA, FDOA, DRSS, and PDOA signal parameters, however these are much more difficult because of their more complicated position lines as previously discussed. Additionally, a combination of any of the signal parameters can be used together as a technique known as hybrid localization. Unfortunately, while it is easy to visualize, when noise is present this solution is very inefficient. By adding noise to just one of the range measurements using the trilateration technique from before, as seen in Fig. 2.2b, we go from one unique solution to several solutions causing ambiguity. Furthermore, if noise is present in all range measurements then there are even more intersections. Unfortunately by adding more signal parameters to a noisy environment only complicates the solution further, therefore we will examine statistical solutions in order to estimate the localization in noisy environments.

2.2.2 Statistical Solutions

Due to the presence of noise, the set of measurement equations is underdetermined with no unique solution. By using statistical methods, we are able to narrow down the unlocalized node's location to a confined area which is then used to estimate a unique solution (while the unique solution is occasionally the center of the confined area, one must remember that is not necessarily the case). When prior knowledge of the unlocalized node is not readily available we will use one of the classical estimators such as the Least Squares or Maximum Likelihood estimator and when prior knowledge is readily available we will use one of the Bayesian estimators for improved accuracy.

Least Squares (LS) Estimator

The simplest and most commonly used approach is the Least Squares (LS) Estimator. The LS estimator requires the least of amount information by minimizing the squared error between the measurements and the known relationship between the data and the node positions, as seen below:

$$\hat{\mathbf{\Theta}}_{LS} = \arg\min_{\Theta} \sum_{i=1}^{N} (z_i - f_i(\mathbf{\Theta}, \mathbf{A}))^2 \tag{2.9}$$

The relationship between the measurements, z_i, and the positions, $\mathbf{\Theta}$, are directly dependent on the signal parameters are used. Due to the fact that $f(\mathbf{\Theta}, \mathbf{A})$ is typically non-linear and non-convex, approximations, such as the Taylor Series, are often implemented to avoid increased complexity [8]. We will discuss this further in Chapter 4.2.

Maximum Likelihood Estimator (MLE)

When the measurement error has a known statistical distribution, we can use the maximum likelihood estimator (MLE) to estimate the position by maximizing the likelihood function, as seen below:

$$\hat{\mathbf{\Theta}}_{MLE} = \arg\max_{\Theta} P[\mathbf{z}|\mathbf{\Theta}] \tag{2.10}$$

where $P[\mathbf{z}|\mathbf{\Theta}]$ is the joint density function when using independent and identically distributed measurements. The MLE estimates the node location, $\mathbf{\Theta}$, by maximizing the likelihood of all the measurements in $\mathbf{z}$ over a defined finite interval. Furthermore, when using a non-differential signal parameter, such as TOA, we can assume all noise components to be independent and can model the noise to be a zero-mean Gaussian random variable with a variance σ for each of the measurements. This lets us classify the MLE as a weighted LS estimator, as seen below:

$$\begin{aligned}
P[\mathbf{z}|\mathbf{\Theta}] &= \prod_{i=1}^{N} \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp[-\frac{1}{2\sigma_i^2}(z_i - f_i(\mathbf{\Theta}, \mathbf{A})^2] \\
&= \frac{1}{(2\pi\sigma_i)^{N/2}} \exp[-\sum_{i=1}^{N} \frac{1}{2\sigma_i^2}(z_i - f_i(\mathbf{\Theta}, \mathbf{A}))^2] \\
\ln(P[\mathbf{z}|\mathbf{\Theta}]) &= -\frac{N}{2}\ln(2\pi + \sigma_i^2) - \frac{1}{2}\sum_{i=1}^{N} \frac{1}{\sigma_i^2}(z_i - f_i(\mathbf{\Theta}, \mathbf{A}))^2 \\
\arg\max_{\Theta} \ln(P[\mathbf{z}|\mathbf{\Theta}]) &= \arg\min_{\Theta} \sum_{i=1}^{N} \frac{1}{\sigma_i^2}(z_i - f_i(\mathbf{\Theta}, \mathbf{A}))^2 \\
\hat{\mathbf{\Theta}}_{MLE} = \hat{\mathbf{\Theta}}_{WLS} &= \arg\min_{\Theta} \sum_{i=1}^{N} \frac{1}{\sigma_i^2}(z_i - f_i(\mathbf{\Theta}, \mathbf{A}))^2
\end{aligned} \tag{2.11}$$

where the σ_i^2 are the variances for each measurement and the less reliable signal parameters or those with a larger noise component are given smaller weight.

Maximum a Posteriori (MAP) Estimator

If knowledge of the prior statistical data is available about the nodes' positions then we use the Bayesian approach. The maximum a posteriori (MAP) estimator maximizes the a posteriori distribution given the measurements, $\mathbf{z}$, seen below:

$$\hat{\mathbf{\Theta}}_{MAP} = \arg\max_{\Theta} p(\mathbf{z}|\mathbf{\Theta}, \mathbf{A}) p_{\Theta}(\mathbf{\Theta}) \tag{2.12}$$

This approach treats the unlocalized node's estimated locations as random variables with a known probability density function (PDF), $p_{\Theta}(\mathbf{\Theta})$. This approach typically results in

higher accuracy of the nodes' positions than the MLE, however the prior distribution of $\boldsymbol{\Theta}$ is difficult to support without physical constraints. Therefore more commonly, most localization estimators use the non-Bayesian approach.

2.3 Implementations

Using the solutions above, we now have the ability to estimate the position of the unlocalized node using the signal parameters we have measured. We will now examine the implementation of these signal parameters and solutions above in the two most common geolocation systems.

2.3.1 The Global Positioning System

The Global Positioning System (GPS) is the most well-known position location system today with common applications ranging from vehicle assistance to geocaching. GPS is a 31 satellite based system that uses TOA with unknown clock bias in order to calculate the location of a user positioned anywhere on the earth with at least four GPS satellites in view. The primary use of GPS is for navigation where it is used to calculate routes and to find the locations of user input navigation points. There are other applications as well such as time synchronization, the smart grid, and various other civil and military applications that include missile tracking and targeting. Other nations are either using or developing similar GPS systems, including Russia's GLONASS, China's Compass, European Union's Galileo, and India's Indian Regional Navigational Satellite System (IRNSS).

Since GPS uses TOA as its measured signal parameters, we know that in order to find a node's unique position in three-dimensions one must acquire a minimum of four satellites. This minimum requirement is normally not an issue due to the 31 operational satellites which produce eight line of sight satellites available to any user at all times. Problems occur if a user's visibility is blocked from large buildings, mountains, or is located indoors, since the number of visible satellites decreases making it extremely difficult to meet the requirement

of four satellites in order to find one's position. Even when four are available performance degrades substantially because fewer measurements constrain the solutions.

All GPS satellites broadcast their ranging data signal at both L1, civilian, and L2, military, frequencies, with GPS using CDMA spread spectrum with Pseudorandom Noise (PRN) Code sequences for each satellite. Using C/A coding for civilian and P(Y) encrypted coding for the military, the unique PRN codes, known as Gold Codes, allow receivers to identify each satellite to decode the satellite's ephemeris message. This ephemeris data contains all the necessary measurement information needed to determine the unlocalized node's position, including the noisy range measurements with an unknown clock bias known as the pseudorange data. Then by using a technique such as the Taylor series, we can convert the non-linear equations into linear equations as seen in Chapter 4.2, and by using Newton-Raphson's method we can perform LS estimation.

The largest source of error in GPS positioning can be attributed to the variance of the noise components of each pseudorange link, which include atmospheric effects, ephemeris errors, and satellite clock errors to name a few. Since each pseudorange link is a TOA signal parameter, we can assume they are independent zero-mean Gaussian random variables, with the variance being attributed to specific components called user equivalent range errors (UERE), described later in Chapter 4.1 in Table 4.1 [9]. The overall variance associated with the noise of each pseudorange link can be found by taking the root mean squared (RMS) error of the sum of the variance components from the table. This results in a range variance of around 8 m in clear sky conditions and an overall GPS position error around 12 m. Unfortunately, as mentioned before, since the user environment plays such a significant role in GPS, the noise component varies substantially based on environment thus increasing the error in positioning and decreasing the availability of satellites to the receiver if the noise is too strong to detect the signal.

Another large source of error is due to the geometry of the satellites when the availability of satellites is limited. Thus one way to measure the error associated with the geometry is by calculating the geometric dilution of precision (GDOP) to find which geometric shapes

produce the least amount of errors. Unfortunately since each signal parameter yields different favorable geometries, there is no single geometry for all systems that is ideal, however since GPS uses a form of TOA we can classify the most favorable geometry amongst the satellites as a tetrahedron in order to decrease errors produced by the GDOP. We will further examine GPS in Chapter 4 when we implement a standard GPS receiver to show how a receiver's position is localized.

As of recent, the modernized GPS satellites now transmit additional ranging data to civilians on the L2, known as L2C, and L5, known as Safety of Life. These additional signals acts as redundant signals in case of interference and help improve the accuracy and performance of localization.

2.3.2 Geolocation of Mobile Phones

Another popular system that has been become prevalent in recent years is geolocation through the use of mobile phones. Mobile phones are devices that can make and receive telephone calls by connecting to a cellular network across a geographic area. Most mobile phones provide the ability to send text messages, e-mail, access the internet, and much more. In order for mobile phones to have access to these services over a geographic area they must transmit and receive data from fixed locations called base stations, which provide coverage over defined areas. Cellular radio systems are approximately set up in a hexagonal shape with 3 sets of directional antennas on each towers separated by 120 degrees as seen in Fig. 2.4. The conventional hexagonal shapes allow for multiple frequencies to be reused in other cells to maximize bandwidth allocation. Furthermore, when cells have a high density of users they can be subdivided into smaller hexagonal cells to increase the number of channels for each geographic area.

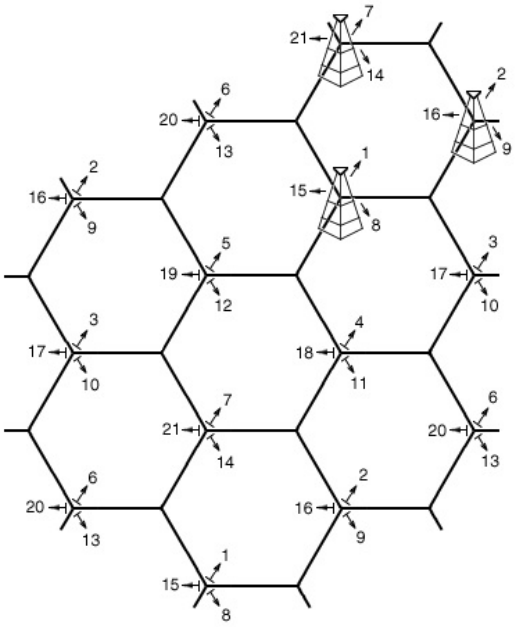


Figure 2.4: Mobile phone towers with cells. This figure shows mobile phone towers at the corners of each hexagonal cell separated by 120 degrees to allow multiple frequencies. Channel numbers are from expired U.S. Patent 4,144,411 Fig. 9

There are three main types of cellular networks: GSM, UMTS, and CDMA2000. The first is GSM which stands for Global System for Mobile Communications is the most widely used network across the globe and operates in the 900MHz or 1800MHz bands. The next is UMTS which stands for Universal Mobile Telecommunications Systems and is based on GSM but emphasizes multimedia access, the 3G version of GSM. The last network is CDMA2000 which is a family of 3G mobile technology standards. It is commonly referred as just CDMA which stands for Code Division Multiple Access, and uses CDMA channel access to send voice and data. These three networks are the standards that service providers offer globally in their networks in order to broadcast information to channels and to communicate. The commonality each uses regardless of the network type employed is the localization of mobile phones.

Mobile phones use a variety of techniques of which can be classified into two types: network-based and mobile-based. The network-based techniques have all measurements and positioning calculations done by the mobile phone's network. The easiest, but most robust, of

the network-based techniques is Cell ID (CID) which uses the identification of the cell size of the base station in use by the mobile phone to find a robust localization. This technique is available for all mobile networks and can provide a position accuracy of approximately 100m-3 km, depending on the size and density of the base station's cell. Furthermore, Enhanced Cell ID uses the assistance of Timing Advance, for time slot synchronization and to provide a rough range estimate, to further improve the position accuracy of CID, however this technique is only available on GSM networks. A much more accurate, but more expensive and complex, network-based technique is Uplink TDOA (UTDOA), which has the network estimate the signals received at various base stations to determine the TDOA. This technique is available for all mobile networks, however it is typically used by GSM, and can provide a position accuracy of approximately 50m [10]. The last network-based technique is Round Trip Time (RTT) which has the mobile device measure the time between reception and transmission to estimate the distance. This technique is available for only CDMA networks, due to their larger bandwidth allowing for higher timing accuracy, and can provide a position accuracy of approximately 100m [10].

The mobile-based techniques have all measurement and positioning calculations done by the mobile phone, through self-positioning. The first mobile-based technique is Advanced Forward Link Triangulation (AFLT), which uses timing information and base station locations to find the position with TDOA. AFLT is only available on CDMA, since all base station transmissions are synchronized, and can provide a position accuracy of approximately 100 m. A technique with higher accuracy is Assisted GPS (A-GPS), which uses GPS along with the measurements of reference receivers to improve accuracy and speed. This technique is available for all mobile networks, however it is typically used by CDMA, and can provide a position accuracy in the 10s of meters but relies on GPS satellite availability, suffering especially indoors. The last mobile-based technique is Enhanced Observed Time Difference (E-OTD) which is essentially multilateration with TDOA measurements. This technique is only available to GSM networks and provides a position accuracy approximately 50-200 m. While mobile-based techniques can potentially provide better results than network-based,

the disadvantage is they require extra software installed in order to perform localization, which is not always readily available to all mobile phones.

Table 2.1: Positioning techniques by US Carriers [1]

US Carrier	**Network**	**Technique**
T-Mobile	GSM	A-GPS/UTDOA
AT&T	GSM/TDMA	A-GPS/UTDOA
Sprint-Nextel	CDMA/IDEN	A-GPS/AFLT
Verizon	CDMA	A-GPS/AFLT

One of the main benefits of geolocating mobile phones is not the ability to track or find ones precise location but rather for location based services. These services include location-based advertising, finding someone or something nearby, proximity notifications, or more importantly for emergency tracking E911. Since mobile phones are not limited to their primary positioning technique, as seen in Table 2.1, they still have the ability to use either CI to find a coarse localization if necessary or use a built-in GPS receiver embedded in the mobile phone. This helps EMS, firefighters, or police dispatch quicker and head to a coarse location while information is still being revealed. Geolocation of mobile devices are still expanding and new techniques are being explore to further reduce the position error of the system.

Chapter 3

The Collaborative Geolocation Problem

3.1 Collaborative Localization

As previously mentioned, position location has gained much interest due to wide applications in controlling, tracking, and monitoring. Difficulties emerge when a node is in a challenging environment, such as an urban or indoor scenario, since the signal quality degrades causing poor accuracy in the position estimate or failure to localize altogether due to a lack of signal availability. Early advancements were developed to improve the signal quality, including transmitting higher signal power or using data from pre-deployed measurements [11]. However these early advancements demanded high system power requirements, lacked versatility in various environments, or more importantly were too computationally expensive. Therefore, in recent years new approaches have been developed that use collaborative (or cooperative) localization.

Collaborative localization is when wireless devices use both the measurements from anchors as well as the measurements between other unlocalized nodes to estimate their positions, in hopes of improving both the accuracy and availability of the wireless location estimates. The concept of collaborative localization can be explained further through a 2-D example

below, in which each unlocalized node requires at least three connections to anchors to find their estimated position. Although both nodes lack sufficient anchors for single-node (i.e. non-collaborative) localization, both nodes can achieve localization through collaborative localization. This is achieved by allowing the unlocalized nodes to use one another as unreferenced anchors so that both nodes can localize themselves simultaneously.

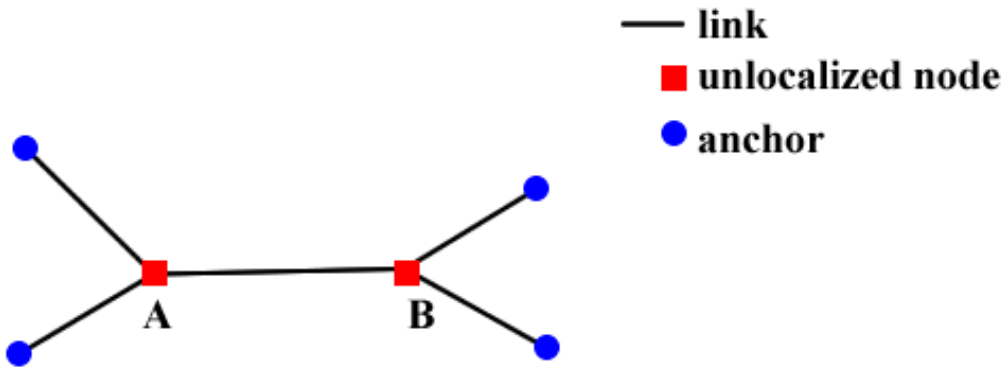


Figure 3.1: Localization Network Scenario 1

Through the use of collaborative localization we are able to increase the availability of localization and, in poor geometric conditions, we are able to improve the accuracy of the position estimates. There are drawbacks with this as not all methods are perfect when using collaborative localization. These drawbacks include scalability (with some approaches) as well as convergence to a local minima when a large amount of noise is present. The improvements in coverage heavily outweigh and offset these drawbacks for collaboration.

3.2 Classifications of Localization Solutions

Before examining the optimal collaborative localization solutions, we will explore the current localization solutions through four classifications [8].

- The first classification is defined by the type of measurement data used in localization such as distance-based, connectivity-based, AOA-based, RSS fingerprinting or hybrid methods. Distance-based solutions use measurements such as TOA are computational

complex, but result in accurate position estimates, where as connectivity-based solutions have less computational complexity but result in coarse position estimates.

- The second classification is defined by where the computations are performed, either by a central processor known as *centralized*, or jointly by the wireless devices known as *distributed*. Centralized solutions typically result in better position estimates, however since the global optimum is difficult to find without an efficient solver for large-scale non-linear optimization problems, distributed solutions are used more often due to their practicality and their robustness to node failure.

- The third classification is defined by how the computations are performed, either *sequentially* or *concurrently*. In sequential localization each localized node declares itself a *virtual anchor* in order to assist other unlocalized nodes, however this results in a sub-optimal approach as discussed in 3.4.1. Therefore the more commonly used method is concurrent localization, in which each unlocalized node uses other unlocalized nodes as unreferenced anchors in addition to the available anchors, as previously described in 3.1.

- The last classification is defined by how the problem is formulated, either *deterministic* or *probabilistic*. In deterministic solutions, a unique solution is found with no additional information about the quality or reliability, where as in probabilistic solutions a set of solutions is returned with a corresponding weight based on their reliability. As expected, the probabilistic solutions result in a more accurate position estimate but more complex.

Next, we will examine the collaborative efforts that have been explored including the optimal approaches as well as several sub-optimal collaborative approaches.

3.3 Optimal Collaborative Approaches

We can find the optimal localization solution by altering our previously defined localization solutions on the LS, MLE, MMSE, and MAP Estimators in Chapter 2.2.2 by setting $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, ..., \boldsymbol{\theta}_n]$ as a set of unlocalized node locations.

3.4 Sub-Optimal Collaborative Approaches

In addition to the optimal solutions, there were several early developments that attempted to use approximations of the optimal solution in order to lower complexity. In the next two sections we examine two specific sub-optimal approaches, the sequential LS and optimization-based approaches. For the following examples presented consider all scenarios as 2-D.

3.4.1 Sequential LS

Sequential collaboration is the simplest approach for collaborative localization and can be defined through the following steps [8]:

1. Estimate the range (or *pseudorange*) to all nodes within range
2. Use a LS estimator (or another position estimator) for single-node localization on all nodes with three or more range measurements to the anchors (or *virtual anchors*)
3. Declare all recently localized nodes as *virtual anchors*
4. Stop if all nodes are localized or if nodes can no longer be localized, else continue
5. Return to 2

To explore these steps further we explain sequential collaboration by using Fig. 3.2 as an example. We begin by estimating the ranges between all the nodes and discover that node A is the only unlocalized node capable of single-node localization with at least three anchors. After node A is localized, we declare it a *virtual anchor* and use to assist node B

in localization. After node B is localized, we declared it a *virtual anchor* to be used in the localization of node C. The advantage of sequential collaboration is the increased availability it provides for the nodes. Through single-node localization, only node A can localize its position estimate. However, through sequential collaboration all three nodes can determine their position estimates. One major drawback with this approach is localization error propagation. This occurs when a node uses a virtual anchor with an inaccurate position estimate as an anchor therefore passing on their error throughout the system to the ensuing node position estimates. Additionally, this approach lacks the ability to localize in cases where concurrent localization would succeed, as seen in the example in Fig. 3.1 in Chapter 3.1, therefore limiting the availability improvements.

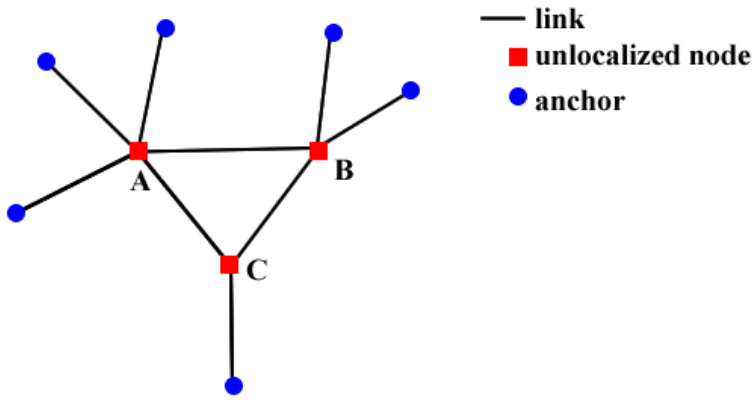


Figure 3.2: Localization Network Scenario 2

3.4.2 Optimization-Based Approaches

To begin the discussion of the optimization-based approach, again we recall our problem, previously defined in Chapter 1.2 where $\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \boldsymbol{\theta}_2, ..., \boldsymbol{\theta}_n]$ is a set of node locations and $\mathbf{A}$ are the locations of the anchors associated with those nodes. We will explain this approach through an example using the LS estimator as our objective function [8], as seen below:

$$\hat{\boldsymbol{\Theta}} = \arg\min_{\Theta} \sum_{i=1}^{n} \sum_{j=i+1}^{n+m} (z_{ij} - ||\boldsymbol{\theta}_i - \boldsymbol{\theta}_j||)^2 \tag{3.1}$$

By adding a relaxed equality constraint to the objective function we now establish a convex optimization problem, in which any local minimum must be a global minimum, as seen below:

$$\hat{\boldsymbol{\Theta}} = \arg\min_{\Theta} \sum_{i=1}^{n} \sum_{j=i+1}^{n+m} |z_{ij}^2 - y_{ij}| \tag{3.2}$$
$$s.t. \qquad y_{ij} = ||\boldsymbol{\theta}_i - \boldsymbol{\theta}_j||^2$$

In which we define the relaxed equality constraint as follows:

$$||\boldsymbol{\theta}_i - \boldsymbol{\theta}_j||^2 = (\boldsymbol{\theta}_i - \boldsymbol{\theta}_j)^\top (\boldsymbol{\theta}_i - \boldsymbol{\theta}_j) \tag{3.3}$$

Then defining $\mathbf{e}_i$ as an euclidean vector with a one in the ith position, $\mathbf{I}_n$ as an n x n identity matrix, and $\mathbf{A}$ as the anchor locations then:

$$b_{ij} = \begin{bmatrix} \mathbf{I}_n & 0 \\ 0 & \mathbf{A} \end{bmatrix} (\mathbf{e}_i - \mathbf{e}_j) \tag{3.4}$$

Then by declaring:

$$||\boldsymbol{\theta}_i - \boldsymbol{\theta}_j||^2 = \langle b_{ij} b_{ij}^\top, \begin{bmatrix} \boldsymbol{\Theta}^\top \boldsymbol{\Theta} & \boldsymbol{\Theta}^\top \\ \boldsymbol{\Theta} & \mathbf{I}_2 \end{bmatrix} \rangle_F \tag{3.5}$$

where $\langle \mathbf{A}, \mathbf{B} \rangle_F = \mathrm{tr}\,(\mathbf{AB})$, we can show that if $\begin{bmatrix} \mathbf{Y} & \mathbf{X}^\top \\ \mathbf{X} & \mathbf{I}_2 \end{bmatrix} \geq 0$ with rank 2, then $\mathbf{Y} = \mathbf{X}^\top \mathbf{X}$. Therefore, we can now rewrite the original non-convex optimization problem as follows:

$$\min_{X} \sum_{i=1}^{n} \sum_{j=i+1}^{n+m} |\langle b_{ij} b_{ij}^\top, \mathbf{X} \rangle_F - z_{ij}^2| \tag{3.6}$$
$$s.t. \qquad [X_{ij}]_{i,j \geq n+m} = \mathbf{I}_2$$
$$\mathbf{X} \geq 0$$
$$rank(\mathbf{X}) = 2$$

Then by relaxing the rank constraint and replacing the absolute value with slack variables, variables used to transform an inequality constraint into an equality constraint and a non-negative constraint, we now have a convex optimization problem as follows:

$$\begin{aligned} \min_{X} \sum_{i=1}^{n} \sum_{j=i+1}^{n+m} \langle b_{ij} b_{ij}^{\top}, \mathbf{X} \rangle_F - z_{ij}^2 + v_{ij} - u_{ij} \\ s.t. \qquad [X_{ij}]_{i,j \geq n_u + n_a} = \mathbf{I}_2 \\ \mathbf{X} \geq 0 \\ u_{ij}, v_{ij} \geq 0 \end{aligned} \tag{3.7}$$

While this replacement does result in a more restrictive solution, it becomes a very computationally expensive problem [8]. This holds true for other optimization-based approaches, therefore we typically use the approximated LS estimator for collaborative localization as seen in the Chapter 4.3 in the discussion of the new collaborative solution introduced.

Chapter 4

Collaborative GPS

As previously mentioned, GPS is the dominant position location system in operation and uses 31 operational satellites producing eight line of sight satellites available to users at all times. A standard GPS receiver requires a minimum of four satellites for localization. This can be a daunting task if one has limited visibility due to their environmental constraints. A variety of techniques have been introduced to improve coverage and reduce positional errors of GPS-based positioning in challenging environments. The most prominent is Differential-GPS (D-GPS) which uses GPS to establish the location of fixed base stations on earth allowing these base stations to act as a GPS satellite, effectively eliminating a substantial amount of positional error. Additionally, Collaborative Acquisition of Weak GPS signals uses shared signal data between GPS receivers to further aid acquisition and increase satellite availability [12]. To further improve GPS, we have proposed a new collaborative solution that has the ability to improve the accuracy and availability of a GPS receiver when in a challenging environment. The new solution uses inter-receiver distances between multiple GPS devices along with pseudoranges to cooperatively determine all receiver locations simultaneously, resulting in improvement in both the accuracy of the position estimate and availability [13].

4.1 Simulation Design

Before introducing the new collaborative solution we first developed a MATLAB simulation to mimic a standard GPS receiver with all 31 operational satellites. The MATLAB simulation supports comparative analysis and allows us to alter the environment of the user(s) while examining the errors that occur due to noise and limited satellite availability[1]. We began by creating a simulation of the GPS satellites by using a file originally created by Kai Borre called 'EASY17', [14], and editing it with ephemeris data from the National Geodetic Survey (NGS). With the baseline data, we then produced the constellation orbits as shown in Fig. 4.1. This also gave us the ability to simulate each satellite's position every two minutes in the sky to include one full orbital pass around the earth at 1440 different locations, or commonly referred to as 1440 time stamps.

After setting up the satellite orbits, we chose predetermined locations and clock biases for our GPS receivers, in the format of $[x\ y\ z\ ct_R]$. We then created our pseudorange data using the predetermined GPS receiver locations. For the sake of convenience we used Blacksburg, VA and its surrounding areas within 1 km as our predetermined locations, with all additional predetermined GPS receivers within 1 km of one another, and a random clock bias that varied between $\pm$10ns, based on average GPS clock bias data [15].

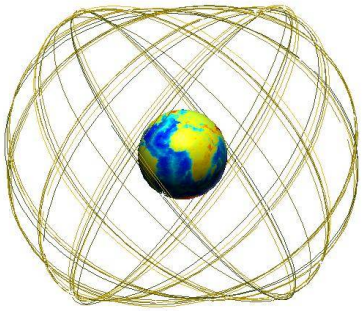


Figure 4.1: An illustration of all 31 active GPS satellites orbiting about the earth during a 24 hour period.

The last part of our MATLAB simulation is the simulation of pseudorange values. To find

[1]We are primarily interested in the impact of noise due to the environment causing the signal degradation and we are not concerned with error such as the doppler effects or significant ionospheric effects

the pseudorange associated with each satellite link we examined the effects of the reference views on the satellites as well as environmental effects. In order to account for the effects of the reference views of the satellites, we found the elevation and azimuth of each GPS satellite at each time stamp. We first determined which satellites are in full view to the predetermined location by checking the elevation of each GPS satellite to see if it is between 5 and 175 degrees. Then we split the full sky into three regions: a left view (between 5 and 45 degrees of elevation and between 180 and 360 degrees of azimuth), an above view (above 45 degrees of elevation), and an right view (between 5 and 45 degrees of elevation and between 0 and 180 degrees of azimuth) as shown in Fig 4.2. By separating the satellite views into the three regions, we find a satellite's carrier-to-noise (C/N_o) ratio depending on its reference region. We focus on a satellite's C/N_o versus the commonly used signal-to-noise (SNR) since GPS uses spread spectrum resulting in a SNR below the noise floor making it undetectable.

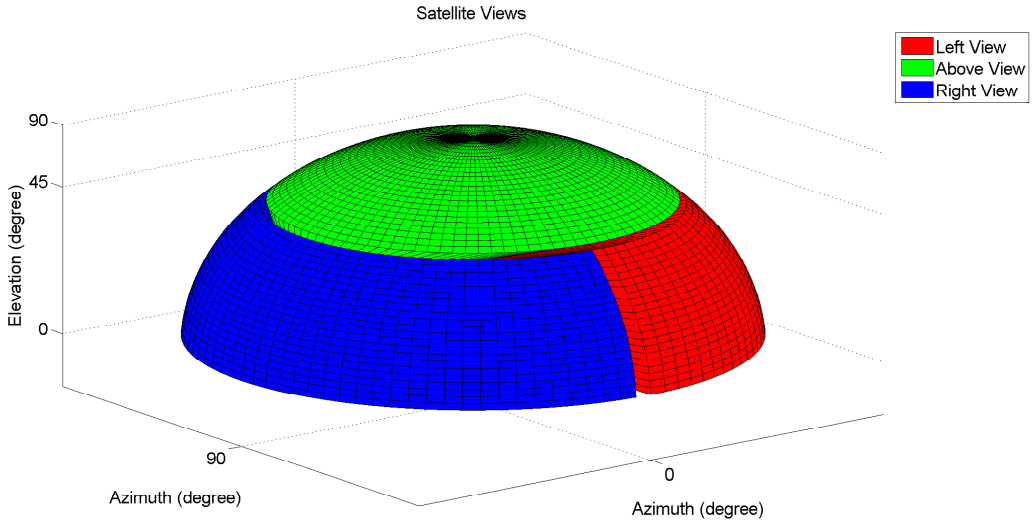


Figure 4.2: An illustration of the three regions of GPS satellites in view: Left, Right, and Above

In order to account for the effects of the environment we needed to add the noise component, η, to the pseudorange measurement. Since GPS uses TOA measurements we can assume that the measured error on each link, between satellite and receiver, is an independent Gaussian random variable with zero mean, and thus we only needed to examine the variance

of the noise component. This variance can be attributed to specific components called user equivalent range errors (UERE), shown in Table 4.1 [9]. By taking the root mean squared (RMS) error of the sum of the variance components in the table, we find the overall standard deviation associated with the noise of each pseudorange link.

Table 4.1: Source of User Equivalent Range Errors (UERE) in meters

Source Effects	**1σ Error (m)**
Signal arrival C/A, σ_S	± 1
Ionospheric effects, σ_I	± 5
Ephemeris errors, σ_E	± 4
Satellite clock errors, σ_C	± 1.5
Multipath distortion, σ_M	± 2.5
Tropospheric effects, σ_T	± 1.5

When in clear sky conditions the noise standard deviation is around 8 m resulting in an average positioning error, $E|\mathbf{x}-\hat{\mathbf{x}}| = \frac{1}{N}\sum_{i=1}^{N}|\mathbf{x}-\hat{\mathbf{x}}|$, around 12 m. Since environment plays such a significant role in GPS, the noise component is varied based on environment. Specifically the noise component is dependent on the C/N_o ratio and can drastically increase when a signal is not received in clear sky (a C/N_o equal to 46 dB-Hz). Various factors contribute to the decrease in C/N_o ratios at the receiver, including signal attenuation, elevation angle, antenna gain, and multipath fading. As an example, a brick wall attenuates a signal approximately 10 dB, while a steel reinforced concrete wall attenuates a signal by approximately 30 dB. This is significant since the threshold at which a standard GPS receiver can lock and track a satellite is approximately 26 dB-Hz [16]. Given this constraint, we simplify our simulation so any satellite has the ability lock a satellite at any given C/N_o but with less probability of being acquired. Thus for our simulation, any satellite link with a C/N_o less than 26 dB-Hz and greater than 10 dB-Hz is given a 50% chance of being acquired and any satellite link with less than 10 dB-Hz is given a 20% chance of being acquired.

Table 4.2: GPS simulation of various environments with estimated C/N_o ratios for each reference views

Environment	**Left View**	**Above**	**Right View**	**Inter-receiver**
Clear Sky	40 dB-Hz	46 dB-Hz	40 dB-Hz	46 dB-Hz
Urban	10 dB-Hz	30 dB-Hz	10 dB-Hz	20 dB-Hz
Window	26 dB-Hz	2 dB-Hz	2 dB-Hz	10 dB-Hz
Indoor	2 dB-Hz	10 dB-Hz	2 dB-Hz	10 dB-Hz

Using statistics from previous work, such as Soloviev and Dickman [16], we were able to model and estimate C/N_o for various environmental conditions for each of the reference views of the satellites as well as the inter-receiver distances, as seen in Table 4.2. The C/N_o for the various environmental conditions of the inter-receiver distances are estimated based off a low power transmitter where the predetermined GPS receivers are distributed randomly within 1 km of each other. These environments were selected based on common scenarios that occur during the localization of a GPS receiver. A clear sky environment occurs when there is nothing obtrusive blocking the view of the satellites to the receiver. An urban environment occurs in a city where large buildings affect the receivers side views creating multipath and the above view stays mostly unaffected. A window environment occurs when a receiver is near a window in a building with only a view from one satellite region. Lastly, an indoor environment occurs in a building where multipath is common, and only a satellite above can be acquired and all other satellite views struggle to be acquired. We can further estimate the effects of C/N_o on the variance of the noise by creating a model, shown in Fig. 4.3. This model allows us to determine the standard deviation of the noise component depending on the view and environment by using the C/N_o, given in Table 4.2. Additionally, we observed from the model how mean noise variance exponentially increases when it falls below the 26 dB-Hz satellite tracking threshold. The NLOS bias is modeled as a uniform random variable from 1 to 5 meters and is added randomly to any pseudorange measurement with a C/N_o below 30 dB-Hz. With these assumptions in place, the simulation has the ability to find the pseudorange for each satellite link to use in our GPS solution by using Eq. (4.1).

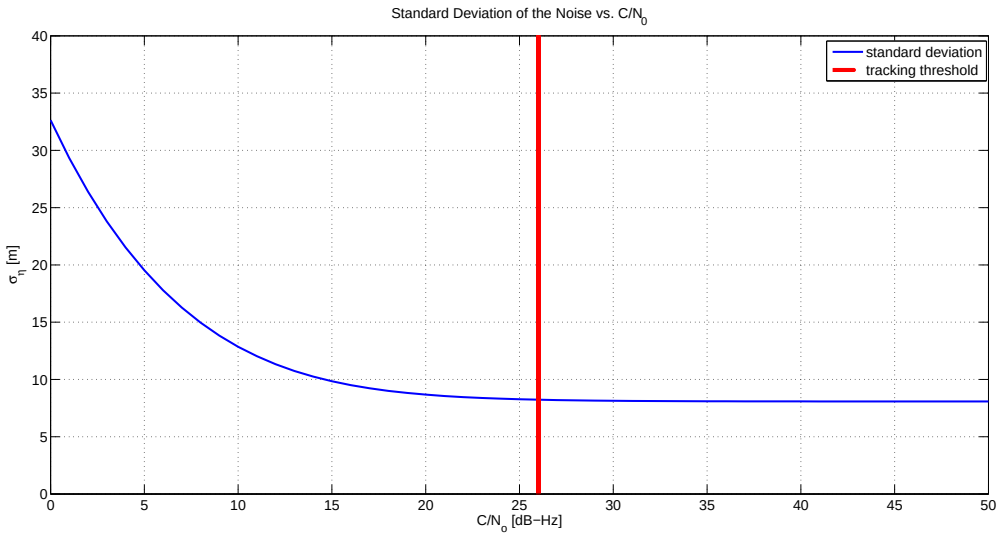


Figure 4.3: A model estimation of the effect of C/N_o on the standard deviation of the range error

The procedure of the standard GPS simulation is as follows:

1. Choose an environment given in the left most column of Table 4.2
2. Using our simulation of the GPS constellation, determine the positions of all 31 satellites
3. Determine which satellites are "Left View", "Above" and "Right View"
4. Determine C/N_o for each satellite in view using Table 4.2
5. Determine which satellites are "acquired" (based on C/N_o)
6. Determine the standard deviation of η for each link using Fig. 4.3
7. Randomly choose which pseudoranges will have a NLOS noise component added based on C/N_o.
8. Simulate pseudorange values for all visible satellites using Eq. (4.1)
9. Determine position estimate

This MATLAB simulation of a GPS receiver is for a single user allowing one to examine the errors associated with noise due to the environment and calibrate during the simulation.

4.2 Standard GPS

Before exploring the new collaborative GPS solution, we begin with the discussion of how a receiver's position is found in a standard GPS receiver [9]. The standard solution requires four or more pseudorange measurements, or TOA range measurements with a clock offset, in the form:

$$f_i(x, y, z, t) = P_i = \sqrt{(x - x_{sat_i})^2 + (y - y_{sat_i})^2 + (z - z_{sat_i})^2} + c(t_{sat_i} - t) \tag{4.1}$$

where P_i is the pseudorange, consisting of the true range from the ith satellite's location $[x_{sat_i} \ y_{sat_i} \ z_{sat_i}]$ to the receiver location, $(t_{sat_i} - t)$ is the clock bias between the known clock offset associated with the ith satellite and the unknown clock offset associated with the receiver, and c is the speed of light. In the next few sections three of the techniques commonly used for localization of a GPS receiver, Newton-Raphson, the Extended Kalman Filter, and the Unscented Kalman Filter are addressed.

4.2.1 Newton-Raphson Localization Solution

The Newton-Raphson solution, the simplest of the techniques, takes the non-linear pseudorange equations, linearizes those equations, and then solves for the position using an iterative process. It begins by using an initial position guess and partial derivatives to perform a steepest line of descent in order to find an approximate solution. The initial position estimate, which is based on a coarse estimate of the user's position, is updated during each iteration to find a better approximation of the solution.

We begin the Newton-Raphson solution by using the Taylor Series expansion of the pseudorange equation, Eq. (4.1), about the initial position guess $[x_0 \ y_0 \ z_0 \ t_0]^\top$ in order to linearize

the set of equations. A more efficient method will be discussed in the explanation of the Unscented Kalman Filter. The Taylor Series expansion of the pseudorange equation can be written as:

$$\begin{aligned} &f_i(x_0 + \Delta x, y_0 + \Delta y, z_0 + \Delta z, t_0 + \Delta t) = f_i(x_0, y_0, z_0, t_0) \\ &+ \tfrac{df_i}{dx_0}(x_0, y_0, z_0, t_0)\Delta x + \tfrac{df_i}{dy_0}(x_0, y_0, z_0, t_0)\Delta y \\ &+ \tfrac{df_i}{dz_0}(x_0, y_0, z_0, t_0)\Delta z + \tfrac{df_i}{dt_0}(x_0, y_0, z_0, t_0)\Delta t \\ &+ \ldots \textit{higher order terms} \end{aligned} \tag{4.2}$$

where $[\frac{df_i}{dx_0}\ \frac{df_i}{dy_0}\ \frac{df_i}{dz_0}\ \frac{df_i}{dt_0}]$ are the partial derivatives of the pseudorange equation with respect to the initial position guess and $[\Delta x\ \Delta y\ \Delta z\ \Delta t]$ are the change in position for each respective dimension. Ignoring the higher order terms of the Taylor Series expansion defined in Eq. (4.2), the partial derivatives are defined as:

$$\begin{aligned} \tfrac{df_i}{dx_0}(x_0, y_0, z_0, t_0) &= -\tfrac{x_{sat_i} - x_0}{r_{i_0}} \\ \tfrac{df_i}{dy_0}(x_0, y_0, z_0, t_0) &= -\tfrac{y_{sat_i} - y_0}{r_{i_0}} \\ \tfrac{df_i}{dz_0}(x_0, y_0, z_0, t_0) &= -\tfrac{z_{sat_i} - z_0}{r_{i_0}} \\ \tfrac{df_i}{dt_0}(x_0, y_0, z_0, t_0) &= -c \end{aligned} \tag{4.3}$$

where

$$r_{i_0} = \sqrt{(x_0 - x_{sat_i})^2 + (y_0 - y_{sat_i})^2 + (z_0 - z_{sat_i})^2} \tag{4.4}$$

Then we define the initial pseudorange as:

$$f_i(x_0, y_0, z_0, t_0) = P_{i_0} = r_{i_0} + ct_0 \tag{4.5}$$

where t_0 is the initial clock offset associated with the receiver and r_{i0} is the range from the initial position guess to the ith satellite as defined in Eq. (4.4).

Substituting the previous equations yields:

$$P_i - ct_{sat_i} \approx P_{i_0} - \tfrac{x_{sat_i} - x_0}{r_{i0}}\Delta x - \tfrac{y_{sat_i} - y_0}{r_{i0}}\Delta y - \tfrac{z_{sat_i} - z_0}{r_{i0}}\Delta z - c\Delta t \tag{4.6}$$

where $P_i - ct_{sat_i}$ is the pseudorange measurement with the known satellite clock offset associated with the ith satellite.

Then rearranging the equation above:

$$P_i - r_{i_0} - c(t_{sat_i} - t_0) = -\frac{x_{sat_i} - x_0}{r_i}\Delta x - \frac{y_{sat_i} - y_0}{r_i}\Delta y - \frac{z_{sat_i} - z_0}{r_i}\Delta z - c\Delta t \tag{4.7}$$

By putting these pseudorange equations into matrix form we obtain the following set of equations $\mathbf{A}\Delta\tilde{\mathbf{x}} = \mathbf{l}$ allowing us to solve for the change in position, $\Delta\tilde{\mathbf{x}} = [\Delta x\ \Delta y\ \Delta z\ c\Delta t]^\top$. We begin by setting up the $\mathbf{A}$ matrix using the partial derivatives from Eq. (4.3) as shown below:

$$\mathbf{A} = \begin{bmatrix} \frac{x_0 - x_{sat_1}}{r_{1_0}} & \frac{y_0 - y_{sat_1}}{r_{1_0}} & \frac{z_0 - z_{sat_1}}{r_{1_0}} & -1 \\ \frac{x_0 - x_{sat_2}}{r_{2_0}} & \frac{y_0 - y_{sat_2}}{r_{2_0}} & \frac{z_0 - z_{sat_2}}{r_{2_0}} & -1 \\ \frac{x_0 - x_{sat_3}}{r_{3_0}} & \frac{y_0 - y_{sat_3}}{r_{3_0}} & \frac{z_0 - z_{sat_3}}{r_{3_0}} & -1 \\ \frac{x_0 - x_{sat_4}}{r_{4_0}} & \frac{y_0 - y_{sat_4}}{r_{4_0}} & \frac{z_0 - z_{sat_4}}{r_{4_0}} & -1 \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{4.8}$$

where $[x_0\ y_0\ z_0]$ is the approximated solution of the user, $[x_{sat_i}\ y_{sat_i}\ z_{sat_i}]$ are the coordinates of the ith satellite, and r_{i_0} is the range from the satellites to the approximated solution as defined in Eq. (4.4).

Additionally by taking the left hand side of Eq. (4.7) for each satellite link we set up the $\mathbf{l}$ vector as follows:

$$\mathbf{l} = \begin{bmatrix} P_1 - c(t_{sat_1} - t_0) - r_{1_0} \\ P_2 - c(t_{sat_2} - t_0) - r_{2_0} \\ P_3 - c(t_{sat_3} - t_0) - r_{3_0} \\ P_4 - c(t_{sat_4} - t_0) - r_{4_0} \\ \vdots \end{bmatrix} \tag{4.9}$$

where P_i is the pseudorange measurement for each satellite, c is the speed of light, $(t_{sat_i} - t_0)$ is the clock bias between the known clock offset associated with the ith satellite and the estimated clock offset associated with the receiver clock bias, and r_{i_0} is the same from above. Using a Weighted Least Squares solution we are able to solve for the change in position using

the matrix and vector defined above:

$$\Delta\tilde{\mathbf{x}} = (\mathbf{A}^\top \mathbf{W}\mathbf{A})^{-1}(\mathbf{A}^\top \mathbf{W}\mathbf{l}) \tag{4.10}$$

where $\Delta\tilde{\mathbf{x}} = [\Delta x\ \Delta y\ \Delta z\ c\Delta t]^\top$ and $\mathbf{W}$ is a diagonal matrix with the inverse of the estimated pseudorange variances, σ_i^2, found using the known (or estimated) C/N_0 values:

$$\mathbf{W} = \begin{bmatrix} \frac{1}{\sigma_i^2} & 0 & \cdots \\ 0 & \frac{1}{\sigma_i^2} & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \tag{4.11}$$

We can then update our initial position estimate $\tilde{\mathbf{x}} + \Delta\tilde{\mathbf{x}}$ and repeat this process again with the updated position estimate. We continue through this iterative process until we converge to a position estimate for the GPS receiver, $[\hat{x}\ \hat{y}\ \hat{z}\ c\hat{t}_R]^\top$.

4.2.2 Extended Kalman Filter Localization Solution

The Kalman Filter was introduced to GPS for localization tracking by using the measurements collected over time to estimate the current mean, or position estimate, and covariance of the system in a two-step process. In the first step of the process, the filter predicts the current state and uncertainty using the previous estimates. In the second step the filter corrects the estimates using a weighted average of the measurements. The standard Kalman Filter (KF) solution assumes that the observables are a linear function of the state and that all errors are Gaussian. The Extended Kalman Filter (EKF) is used in substitution of the standard Kalman Filter to linearize the pseudorange measurements using the Taylor Series, much like in the Newton-Raphon solution [17].

Before using the EKF we start by defining our vectors and matrices. The first vector is the state vector equation to estimate our mean (i.e. position estimate) as seen below.

$$\boldsymbol{\delta} = \begin{bmatrix} x + v_x t & y + v_y t & z + v_z t & ct_R + ct_{d_R} t & v_x & v_y & v_z & ct_{d_R} \end{bmatrix}^\top \tag{4.12}$$

Where $[v_x\ v_y\ v_z]$ are the respective positional velocities and t_{d_R} is the clock drift associated with the receiver. Then in order to find a linear approximation of our state vector, we take the partial derivatives to find our state transition matrix as defined below:

$$\mathbf{\Phi} = \frac{d\boldsymbol{\delta}}{d\mathbf{x}} = \begin{bmatrix} 1 & 0 & 0 & 0 & t & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & t & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & t & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & t \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.13}$$

Next we define our process noise covariance matrix, $\mathbf{Q}$, as shown in Eq. (4.14) as an identity matrix to show there is little uncertainty in the state. It is important to note, if prior knowledge of the process noise covariance is available it can significantly increase the accuracy. For our purposes we will consider the process noise covariance as non-Bayesian with no prior information readily available. A large $\mathbf{Q}$ will result in greater change in the states, while a smaller $\mathbf{Q}$ will result in less change in states thus an adaptive process noise covariance can greatly improve state estimates.

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & \cdots \\ 0 & 1 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \tag{4.14}$$

Our measurement noise covariance matrix, $\mathbf{R}$, can be defined as the diagonal matrix of the estimated pseudorange variances, σ_i^2, found using the known (or estimated) C/N_0 values:

$$\mathbf{R} = \begin{bmatrix} \sigma_1^2 & 0 & \cdots \\ 0 & \sigma_2^2 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \tag{4.15}$$

To begin the EKF, we start by initializing the state vector and state error covariance matrix, also known as the uncertainty covariance matrix, as shown below:

$$\mathbf{x} = \begin{bmatrix} x_0 & y_0 & z_0 & ct_0 & v_{x0} & v_{y0} & v_{z0} & ct_{d0} \end{bmatrix}^\top \tag{4.16}$$

$$\mathbf{P} = \begin{bmatrix} 10 & 0 & \cdots \\ 0 & 10 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \tag{4.17}$$

where $\mathbf{x}$ is an n x 1 column vector (where n is the length of the state vector or eight) and $\mathbf{P}$ is an n x n matrix. The values in $\mathbf{x}$ are initially randomly chosen around the predetermined receiver location, as described in Ch. 4.1, with a known velocity and the values in $\mathbf{P}$ are chosen based on the uncertainty of the initialized state vector and recommendations in [18]. All sequential values in $\mathbf{x}$ and $\mathbf{P}$ are chosen by the state vector and state error covariance from the previous time measurement. Then we start step one of our two-step process by starting with the prediction of the state vector, $\mathbf{x}_p$, and state error covariance matrix, $\mathbf{P}_p$, defined below.

$$\mathbf{x}_p = \mathbf{\Phi}\mathbf{x} \tag{4.18}$$

$$\mathbf{P}_p = \mathbf{\Phi}\mathbf{P}\mathbf{\Phi}^\top + \mathbf{Q} \tag{4.19}$$

Where $\mathbf{\Phi}$ is our state transition matrix as defined in Eq. (4.13), $\mathbf{x}$ is our initialized state vector, $\mathbf{P}$ is our state error covariance, and $\mathbf{Q}$ is our process noise covariance as defined in Eq. (4.14).

Next we begin the correction step in our two-step process by calculating the filter gain, based on the uncertainty of the state vector and the measurements, that we will use for updating our predictions as shown below:

$$\mathbf{K} = \mathbf{P}_p\mathbf{H}^\top(\mathbf{H}\mathbf{P}_p\mathbf{H}^\top + \mathbf{R})^{-1} \tag{4.20}$$

Where $\mathbf{P}_p$ is our predicted state covariance, $\mathbf{R}$ is the measurement noise as defined in

Eq. (4.15), and **H** is our measurement matrix found by taking the partial derivatives of the pseudorange measurements, as shown in Eq. (4.3), and is defined below:

$$\mathbf{H} = \begin{bmatrix} \frac{x_0 - x_{sat_1}}{r_{1_0}} & \frac{y_0 - y_{sat_1}}{r_{1_0}} & \frac{z_0 - z_{sat_1}}{r_{1_0}} & -1 & 0 & 0 & 0 & 0 \\ \frac{x_0 - x_{sat_2}}{r_{2_0}} & \frac{y_0 - y_{sat_2}}{r_{2_0}} & \frac{z_0 - z_{sat_2}}{r_{22}0} & -1 & 0 & 0 & 0 & 0 \\ \frac{x_0 - x_{sat_3}}{r_{3_0}} & \frac{y_0 - y_{sat_3}}{r_{3_0}} & \frac{z_0 - z_{sat_3}}{r_{3_0}} & -1 & 0 & 0 & 0 & 0 \\ \frac{x_0 - x_{sat_4}}{r_{4_0}} & \frac{y_0 - y_{sat_4}}{r_{4_0}} & \frac{z_0 - z_{sat_4}}{r_{4_0}} & -1 & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \tag{4.21}$$

where $[x_0\ y_0\ z_0]$ is the approximated solution of the user, $[x_{sat_i}\ y_{sat_i}\ z_{sat_i}]$ are the coordinates of the ith satellites, and r_{i_0} is the range from the ith satellites to the approximated solution as defined in Eq. (4.4).

This resulting filter gain lets us correct and update our state and uncertainty predictions appropriately.

$$\mathbf{x} = \mathbf{x}_p + \mathbf{Kl} \tag{4.22}$$

$$\mathbf{P} = (\mathbf{I}_n - \mathbf{KH})\mathbf{P}_p \tag{4.23}$$

Resulting in **x** as our updated state vector of the predicted guess and **P** as our updated state error covariance matrix. Where $\mathbf{x}_p$ is the predicted guess, $\mathbf{I}_n$ is an n x n identity matrix, **K** is our filter gain, and **l** is the difference of the pseudoranges and the estimated ranges, as previously defined in the Newton-Raphson solution in Eq. (4.9) and redefined below:

$$\mathbf{l} = \begin{bmatrix} P_1 - c(t_{sat_1} - t_0) - r_{1_0} \\ P_2 - c(t_{sat_2} - t_0) - r_{2_0} \\ P_3 - c(t_{sat_3} - t_0) - r_{3_0} \\ P_4 - c(t_{sat_4} - t_0) - r_{4_0} \\ \vdots \end{bmatrix} \tag{4.24}$$

where P_i is the pseudorange for each satellite, c is the speed of light, $(t_{sat_i} - t_0)$ is the clock bias between the known clock offset associated with the ith satellite and the estimated clock

offset associated with the receiver clock bias, and r_{i_0} is the estimated range measurement. The procedure of the EKF is as follows:

1. Initialize state vector and state error covariance (or use the state vector and state error covariance from the previous time measurement):
   $\mathbf{x}$ and $\mathbf{P}$

2. Predict state vector and state error covariance:
   $\mathbf{x}_p = \mathbf{\Phi x}$
   $\mathbf{P}_p = \mathbf{\Phi P \Phi}^\top + \mathbf{Q}$

3. Compute Kalman Filter Gain:
   $\mathbf{K} = \mathbf{P}_p\mathbf{H}^\top(\mathbf{H}\mathbf{P}_p\mathbf{H}^\top + \mathbf{R})^{-1}$

4. Update and correct predicted state vector and state error covariance:
   $\mathbf{x} = \mathbf{x}_p + \mathbf{Kl}$
   $\mathbf{P} = (\mathbf{I}_n - \mathbf{KH})\mathbf{P}_p$

This process results in an estimated state vector for the GPS receiver $[\hat{x}\ \hat{y}\ \hat{z}\ \hat{ct_R}\ \hat{v_x}\ \hat{v_y}\ \hat{v_z}\ \hat{ct_{d_R}}]^\top$ and the state error covariance matrix that is used in the next measurement to further improve the estimated state vector over time.

4.2.3 Unscented Kalman Filter Localization Solution

An alternative to the Extended Kalman Filter is the Unscented Kalman Filter (UKF) to solve the non-linear problem. The UKF uses a distribution approximation method along with the true nonlinear functions, without the need of linearization. Then by generating a finite set of sigma points, or sample points, through an unscented transform, a method of determining the statistics of the states with a nonlinear function, we estimate the mean and covariance of the states [17]. This method requires the same cost and complexity of the EKF and typically outperforms the EKF when there is a strong non-linearity that results in a poor approximation using the partial derivatives, as used in Eq. 4.21.

Before describing the UKF, we start by defining our vectors and matrices, exactly as we did with the EKF. The first vector is the state vector equation to estimate our mean, or position estimate, as seen below.

$$\boldsymbol{\delta} = \begin{bmatrix} x + v_x t & y + v_y t & z + v_z t & ct_R + ct_{d_R} t & v_x & v_y & v_z & ct_{d_R} \end{bmatrix}^\top \tag{4.25}$$

Next we define our process noise covariance matrix, $\mathbf{Q}$, as we did with the EKF in Eq. (4.14), shown below.

$$\mathbf{Q} = \begin{bmatrix} 1 & 0 & \cdots \\ 0 & 1 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \tag{4.26}$$

Our measurement noise covariance matrix, $\mathbf{R}$, can be defined as the diagonal matrix of the estimated pseudorange variances, σ_i^2, found using the known (or estimated) C/N_0 values:

$$\mathbf{R} = \begin{bmatrix} \sigma_1^2 & 0 & \cdots \\ 0 & \sigma_2^2 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \tag{4.27}$$

To begin the UKF, we start by initializing the state vector and state error covariance matrix, also known as the uncertainty covariance matrix, as seen below:

$$\mathbf{x} = \begin{bmatrix} x_0 & y_0 & z_0 & ct_0 & v_{x0} & v_{y0} & v_{z0} & ct_{d0} \end{bmatrix}^\top \tag{4.28}$$

$$\mathbf{P} = \begin{bmatrix} 10 & 0 & \cdots \\ 0 & 10 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \tag{4.29}$$

where $\mathbf{x}$ is an n x 1 column vector (where n is the length of the state vector or eight) and $\mathbf{P}$ is an n x n matrix. The values in $\mathbf{x}$ are initially randomly chosen around the predetermined receiver location, as described in Ch. 4.1, with a known velocity and the values in $\mathbf{P}$ are

chosen based on the uncertainty of the initialized state vector and recommendations in [18]. All sequential values in $\mathbf{x}$ and $\mathbf{P}$ are chosen by the state vector and state error covariance from the previous time measurement.

Then in order to start the UKF we begin with the sigma points selection algorithm around $\mathbf{x}$ by generating the sigma points through the unscented transformation (UT). In order to find at least the second order of the state vector we must use $2n+1$ sigma points in the form below:

$$\mathbf{X}_{sigma} = \begin{bmatrix} \mathbf{x} & \mathbf{x} + \sqrt{(n+\lambda)\mathbf{P}} & \mathbf{x} - \sqrt{(n+\lambda)\mathbf{P}} \end{bmatrix} \tag{4.30}$$

This results in $\mathbf{X}_{sigma}$ a n x $2n+1$ matrix where $\mathbf{x}$ is the initial state vector, λ is a scaling factor as defined below in Eq. (4.31), $\mathbf{P}$ is the state error covariance given in Eq. (4.29), and Cholesky factorization is used for the square root of the matrices. Next we find the weights of the sigma points associated with both the mean and covariance as seen below.

$$\lambda = \alpha^2 n - n \tag{4.31}$$

$$\mathbf{w}_0^{(m)} = \frac{\lambda}{n+\lambda} \tag{4.32}$$

$$\mathbf{w}_0^{(c)} = \mathbf{w}_0^{(m)} + (1 - \alpha^2 + \beta) \tag{4.33}$$

$$\mathbf{w}_i^{(m)} = \mathbf{w}_i^{(c)} = \frac{1}{2(n+\lambda)} \qquad i = 1, ..., 2n \tag{4.34}$$

where α is the spread of the sigma points and is set to 1e-3 (a typical default value) and β is the distribution parameter and is set to 2 to declare a normal distribution.

To begin the UKF we start with the prediction of the state vector associated with each sigma point and state error covariance matrix as defined below:

$$\mathbf{X}_p = \boldsymbol{\delta}(\mathbf{X}_{sigma}) \tag{4.35}$$

$$\hat{\mathbf{x}} = \sum_{i=0}^{2n} \mathbf{w}_i^{(m)} \mathbf{X}_p \tag{4.36}$$

$$\mathbf{P}_p = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)} (\mathbf{X}_p - \hat{\mathbf{x}})(\mathbf{X}_p - \hat{\mathbf{x}})^\top + \mathbf{Q} \tag{4.37}$$

Resulting in $\mathbf{X}_p$ as a n x $2n+1$ matrix containing our sigma points passed through the state vector equation, $\boldsymbol{\delta}$ as defined in Eq. (4.25), $\hat{\mathbf{x}}$ as a n x 1 predicted mean state vector, and $\mathbf{P}_p$ as a n x n predicted state error covariance matrix. Where $\mathbf{w}_i^{(m)}$ is our sigma point weights associated with the mean, $\mathbf{w}_i^{(c)}$ is our sigma point weights associated with the covariance, and $\mathbf{Q}$ is the process noise covariance as defined in Eq. (4.26).

Next we find the range associated with each of the predicted sigma points, $\mathbf{X}_p$, and the predicted mean associated with the range measurements as seen below.

$$f_i(\mathbf{X}_p) = \sqrt{(x_{\mathbf{X}_p} - x_{sat_i})^2 + (y_{\mathbf{X}_p} - y_{sat_i})^2 + (z_{\mathbf{X}_p} - z_{sat_i})^2} + c(t_{sat_i} - t_{\mathbf{X}_p})$$

$$\mathbf{Z}_p = f_i(\mathbf{X}_p) \tag{4.38}$$

$$\hat{\mathbf{z}} = \sum_{i=0}^{2n} \mathbf{w}_i^{(m)} \mathbf{Z}_p \tag{4.39}$$

Where $f_i(\mathbf{X}_p)$ is the estimated range measurement from each of the predicted sigma points in $\mathbf{X}_p$ to the ith satellite using Eq. (4.1) and redefined above, $\mathbf{w}_i^{(m)}$ is our sigma point weights associated with the mean, and $\mathbf{w}_i^{(c)}$ is our sigma point weights associated with the covariance. Then to update our predicted mean and covariance we must calculate our filter gain, based on the uncertainty of the states and the measurements, by finding the innovation covariance, $\mathbf{P}_z$, and the cross-covariance, $\mathbf{P}_{xz}$, as defined below.

$$\mathbf{P}_z = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)} (\mathbf{Z}_p - \hat{\mathbf{z}})(\mathbf{Z}_p - \hat{\mathbf{z}})^\top + \mathbf{R} \tag{4.40}$$

$$\mathbf{P}_{xz} = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)} (\mathbf{X}_p - \hat{\mathbf{x}})(\mathbf{Z}_p - \hat{\mathbf{z}})^\top \tag{4.41}$$

Letting us define the filter gain as follows:

$$\mathbf{K} = \mathbf{P}_{xz}\mathbf{P}_z^{-1} \tag{4.42}$$

This resulting filter gain lets us correct and update our predicted mean and covariance appropriately:

$$\mathbf{x} = \hat{\mathbf{x}} + \mathbf{K}(\mathbf{z} - \hat{\mathbf{z}}) \tag{4.43}$$

$$\mathbf{P} = \mathbf{P}_p + \mathbf{K}\mathbf{P}_z\mathbf{K}^\top \tag{4.44}$$

The resulting $\mathbf{x}$ is our updated state vector of the predicted mean and $\mathbf{P}$ is our updated state error covariance matrix. Note that $\hat{\mathbf{x}}$ is the predicted mean, $\hat{\mathbf{z}}$ is the predicted mean associated with the range measurements, $\mathbf{P}_p$ is the predicted covariance, $\mathbf{K}$ is the filter gain, $\mathbf{z}$ is the vector of pseudorange measurements, and $\mathbf{P}_z$ is the innovation covariance.
The procedure of the UKF is as follows:

1. Initialize state vector and state error covariance (or use the state vector and state error covariance from the previous time measurement):
   $\mathbf{x}$ and $\mathbf{P}$

2. Sigma Point Selection Algorithm:
   $\mathbf{X}_{sigma} = \begin{bmatrix}\mathbf{x} & \mathbf{x} + \sqrt{(n+\lambda)\mathbf{P}} & \mathbf{x} - \sqrt{(n+\lambda)\mathbf{P}}\end{bmatrix}$

3. Predict state vector and state error covariance:
   $\mathbf{X}_p = \boldsymbol{\delta}(\mathbf{X}_{sigma})$
   $\hat{\mathbf{x}} = \sum_{i=0}^{2n} \mathbf{w}_i^{(m)}\mathbf{X}_p$
   $\mathbf{P}_p = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)}(\mathbf{X}_p - \hat{\mathbf{x}})(\mathbf{X}_p - \hat{\mathbf{x}})^\top + \mathbf{Q}$

4. Predict mean associated with the range measurements:
   $\mathbf{Z}_p = f_i(\mathbf{X}_p)$
   $\hat{\mathbf{z}} = \sum_{i=0}^{2n} \mathbf{w}_i^{(m)}\mathbf{Z}_p$

5. Calculate innovation covariance:
   $\mathbf{P}_z = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)} (\mathbf{Z}_p - \hat{\mathbf{z}})(\mathbf{Z}_p - \hat{\mathbf{z}})^\top + \mathbf{R}$
6. Calculate cross-covariance:
   $\mathbf{P}_{xz} = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)} (\mathbf{X}_p - \hat{\mathbf{x}})(\mathbf{Z}_p - \hat{\mathbf{z}})^\top$
7. Compute Kalman Filter Gain:
   $\mathbf{K} = \mathbf{P}_{xz}\mathbf{P}_z^{-1}$
8. Update and correct predicted state vector and state error covariance:
   $\mathbf{x} = \hat{\mathbf{x}} + \mathbf{K}(\mathbf{z} - \hat{\mathbf{z}})$
   $\mathbf{P} = \mathbf{P}_p + \mathbf{K}\mathbf{P}_z\mathbf{K}^\top$

This process results in the estimated state vector for the GPS receiver $[\hat{x}\ \hat{y}\ \hat{z}\ \hat{ct_R}\ \hat{v_x}\ \hat{v_y}\ \hat{v_z}\ \hat{ct_{d_R}}]^\top$ and the state error covariance matrix that is used in the next measurement to further improve the estimated state vector over time.

4.3 Collaborative Global Positioning System

Our new collaborative solution assumes an accurate estimate, but not perfect, of the distances between receivers. We consider the inter-receiver distance as an additional pseudorange (or *pseudodistance*) to the solution, allowing the use of both pseudoranges and the *pseudodistances* to solve for both user's positions simultaneously through concurrent collaboration. This collaboration between devices requires communication between the GPS devices with the additional feature of transmitting signals, if not already integrated in the device. The power requirements of the transmitted signal would not need to be very high given the typical receive power of GPS signals using spread spectrum and the shorter distances between devices. The signal would need to be transmitted at the same rate as the satellite data so it could be tracked and locked as if it were a satellite and would include similar ephemeris data as by a GPS satellite including, an estimated position and clock bias. The process of how quickly one device locks onto another device would depend on how accurate the initial

estimate position is based on the number of satellites and whether a cold (new and fresh data) or warm start (recently used with data) were used.

For simplicity purposes, we explain our solution through the examination of two collaborating receivers however our solution can be expanded beyond two receivers. We begin by adding the inter-receiver distance to our collaborative solution, as we introduce our *pseudodistance* equation below.

$$f_{ij}(x_i, x_j, y_i, y_j, z_i, z_j, t_i, t_j) = D_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} + c(t_j - t_i) \tag{4.45}$$

where D_{ij} is the *pseudodistance*, consisting of the true distance between users i and j, and $(t_j - t_i)$ is the difference between the clock biases of the two receivers.

The noise components that are introduced to the *pseudodistance* are η_{ij} which is a Gaussian random variable with zero mean and variance depending on the C/N_o as defined in Table 4.2 and b which is the NLOS noise component defined as a uniform random variable from 1 to 5 meters for those with C/N_o ratios below 30 dB-Hz.

4.3.1 A Newton-Raphson-based Collaborative Solution

In order to adjust the Newton-Raphson-based Collaborative solution to account for the inter-receiver distances and to collaborate concurrently, we make adjustments to the **A** and **l** components of the update, $\Delta\tilde{\mathbf{x}}$. Again, we linearize Eq. (4.45) using the Taylor Series expansion as defined in Eq. (4.2). Ignoring the higher order terms, the partial derivatives of the distance equation are defined as:

$$\begin{aligned}
\frac{df_{ij}()}{dx_{i_0}} &= -\frac{x_{j_0} - x_{i_0}}{d_{i_0 j_0}} & \frac{df_{ij}()}{dx_{j_0}} &= -\frac{x_{i_0} - x_{j_0}}{d_{i_0 j_0}} \\
\frac{df_{ij}()}{dy_{i_0}} &= -\frac{y_{j_0} - y_{i_0}}{d_{i_0 j_0}} & \frac{df_{ij}()}{dy_{j_0}} &= -\frac{y_{i_0} - y_{j_0}}{d_{i_0 j_0}} \\
\frac{df_{ij}()}{dz_{i_0}} &= -\frac{z_{j_0} - z_{i_0}}{d_{i_0 j_0}} & \frac{df_{ij}()}{dz_{j_0}} &= -\frac{z_{i_0} - z_{j_0}}{d_{i_0 j_0}} \\
\frac{df_{ij}()}{dt_{i_0}} &= -c & \frac{df_{ij}()}{dt_{j_0}} &= -c
\end{aligned} \tag{4.46}$$

where

$$d_{i_0 j_0} = \sqrt{(x_{i_0} - x_{j_0})^2 + (y_{i_0} - y_{j_0})^2 + (z_{i_0} - z_{j_0})^2} \tag{4.47}$$

Defining the initial *pseudodistance* as

$$f_{ij}(x_{i_0}, x_{j_0}, y_{i_0}, y_{j_0}, z_{i_0}, z_{j_0}, t_{i_0}, t_{j_0}) = D_{i_0 j_0} = d_{i_0 j_0} + c(t_{j_0} - t_{i_0}) \tag{4.48}$$

Where t_{i_0} and t_{j_0} are the initial clock offsets associated with the each receiver and $d_{i_0 j_0}$ is the inter-receiver distance between the estimated user positions as defined in Eq. (4.47). Substituting the previous equations yields:

$$\begin{aligned} D_{ij} \approx D_{i_0 j_0} &- \tfrac{x_{i_0} - x_{j_0}}{d_{i_0 j_0}} \Delta x_i - \tfrac{x_{j_0} - x_{i_0}}{d_{i_0 j_0}} \Delta x_j - \tfrac{y_{i_0} - y_{j_0}}{d_{i_0 j_0}} \Delta y_i \\ &- \tfrac{y_{j_0} - y_{i_0}}{d_{i_0 j_0}} \Delta y_j - \tfrac{z_{i_0} - z_{j_0}}{d_{i_0 j_0}} \Delta z_i - \tfrac{z_{j_0} - z_{i_0}}{d_{i_0 j_0}} \Delta z_j - c\Delta t_{i_0} - c\Delta t_{j_0} \end{aligned} \tag{4.49}$$

where D_{ij} is the *pseudodistance* measurement between users.
Then rearranging the equation above:

$$\begin{aligned} &D_{ij} - d_{i_0 j_0} - c(t_{j_0} - t_{i_0}) = \\ &- \tfrac{x_{i_0} - x_{j_0}}{d_{i_0 j_0}} \Delta x_i - \tfrac{x_{j_0} - x_{i_0}}{d_{i_0 j_0}} \Delta x_j - \tfrac{y_{i_0} - y_{j_0}}{d_{i_0 j_0}} \Delta y_i - \tfrac{y_{j_0} - y_{i_0}}{d_{i_0 j_0}} \Delta y_j \\ &- \tfrac{z_{i_0} - z_{j_0}}{d_{i_0 j_0}} \Delta z_i - \tfrac{z_{j_0} - z_{i_0}}{d_{i_0 j_0}} \Delta z_j - c\Delta t_{i_0} - c\Delta t_{j_0} \end{aligned} \tag{4.50}$$

By putting these *pseudodistance* equations into matrix form we obtain a new set of equations to solve for $\mathbf{A}\Delta\tilde{\mathbf{x}} = \mathbf{l}$. Therefore, we start by placing the partial derivatives from Eq. (4.46) in matrix form as shown below:

$$\begin{aligned} \mathbf{A}_{1-2} &= \begin{bmatrix} \frac{x_{1_0} - x_{2_0}}{d_{1_0 2_0}} & \frac{y_{1_0} - y_{2_0}}{d_{1_0 2_0}} & \frac{z_{1_0} - z_{2_0}}{d_{1_0 2_0}} & -1 \end{bmatrix} \\ \mathbf{A}_{2-1} &= \begin{bmatrix} \frac{x_{2_0} - x_{1_0}}{d_{1_0 2_0}} & \frac{y_{2_0} - y_{1_0}}{d_{1_0 2_0}} & \frac{z_{2_0} - z_{1_0}}{d_{1_0 2_0}} & -1 \end{bmatrix} \end{aligned} \tag{4.51}$$

where $[x_{1_0}\ y_{1_0}\ z_{1_0}]$ is the initial estimate of the solution of user 1, $[x_{2_0}\ y_{2_0}\ z_{2_0}]$ is the initial estimate of the solution of user 2, and $d_{1_0 2_0}$ is the inter-receiver distance between the estimated user positions as defined in Eq. (4.47).

To accommodate for the parallel collaboration, we create a new $\mathbf{A}$ matrix with the following components:

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{user_1} & 0 \\ 0 & \mathbf{A}_{user_2} \\ \mathbf{A}_{1-2} & \mathbf{A}_{2-1} \end{bmatrix} \tag{4.52}$$

where $\mathbf{A}_{user_i}$ is the appropriate $\mathbf{A}$ matrix for each individual user and can be found from Eq. (4.8) and $\mathbf{A}_{i-j}$ is the partial derivatives of the *pseudodistances* as defined above in Eq. (4.51). This matrix can be extended to additional users by placing the appropriate $\mathbf{A}$ matrix for each individual user on a diagonal then appending the appropriate collaborative matrices in their respective locations accompanied by zeros where collaboration does not occur, as shown in an example with three users below.

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{user_1} & 0 & 0 \\ 0 & \mathbf{A}_{user_2} & 0 \\ 0 & 0 & \mathbf{A}_{user_3} \\ \mathbf{A}_{1-2} & \mathbf{A}_{2-1} & 0 \\ \mathbf{A}_{1-3} & 0 & \mathbf{A}_{3-1} \\ 0 & \mathbf{A}_{2-3} & \mathbf{A}_{3-2} \end{bmatrix} \tag{4.53}$$

Then by taking the left hand side of Eq. (4.50) we can set up the *pseudodistance* $\mathbf{l}$ vector as follows:

$$\mathbf{l}_{1-2} = \left[D_{12} - c(t_{2_0} - t_{1_0}) - d_{1_0 2_0} \right] \tag{4.54}$$

where D_{12} is the *pseudodistance* measurement between users, t_{1_0} and t_{2_0} are the estimated clock offsets associated with each user, and $d_{1_0 2_0}$ is the inter-receiver distance between the estimated user positions. To accommodate for the parallel collaboration, we compose the new $\mathbf{l}$ vector as:

$$\mathbf{l} = \begin{bmatrix} \mathbf{l}_{user_1} \\ \mathbf{l}_{user_2} \\ \mathbf{l}_{1-2} \end{bmatrix} \tag{4.55}$$

where $\mathbf{l}_{user_i}$ is the appropriate $\mathbf{l}$ vector associated with each user and can be found from Eq. (4.9) and $\mathbf{l}_{i-j}$ is the *pseudodistances* $\mathbf{l}$ vector as defined above in Eq. (4.54). To extend this matrix beyond two users is straightforward, where all appropriate $\mathbf{l}$ vectors for each user are placed one after another in order and are then followed by the appropriate collaborative components, as shown in an example with three users.

$$\mathbf{l} = \begin{bmatrix} \mathbf{l}_{user_1} \\ \mathbf{l}_{user_2} \\ \mathbf{l}_{user_3} \\ \mathbf{l}_{1-2} \\ \mathbf{l}_{1-3} \\ \mathbf{l}_{2-3} \end{bmatrix} \tag{4.56}$$

Requiring both users to solve for both positions makes it necessary to share and communicate all needed information. Therefore using new $\mathbf{A}$ matrix and $\mathbf{l}$ vector we solve for the change in position for all users simultaneously using a Weighted Least Squares solution:

$$\Delta\tilde{\mathbf{x}} = (\mathbf{A}^\top \mathbf{W} \mathbf{A})^{-1} (\mathbf{A}^\top \mathbf{W} \mathbf{l}) \tag{4.57}$$

where we define $\mathbf{W}$ as a diagonal matrix with the inverse of estimated pseudorange and *pseudodistance* variances on the diagonal:

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_{user_1} & 0 & 0 \\ 0 & \mathbf{W}_{user_2} & 0 \\ 0 & 0 & \mathbf{W}_{1-2} \end{bmatrix} \tag{4.58}$$

where $\mathbf{W}_{user_i}$ corresponds to the inverse of the estimated pseudorange variances for the appropriate user as defined in Eq. (4.11) and $\mathbf{W}_{1-2}$ is the inverse of the estimated *pseudodistance* variance.

Finally, we define $\Delta\tilde{\mathbf{x}}$ as:

$$\Delta\tilde{\mathbf{x}} = \begin{bmatrix} \Delta x_1 & \Delta y_1 & \Delta z_1 & c\Delta t_{R1} & \Delta x_2 & \Delta y_2 & \Delta z_2 & c\Delta t_{R2} \end{bmatrix}^\top \tag{4.59}$$

As before, we update our initial position estimate $\tilde{\mathbf{x}} + \Delta\tilde{\mathbf{x}}$ for each user simultaneously and then repeat this process again with the updated position estimates. We continue through this iterative process until we converge to a position estimate for all GPS receivers.

4.3.2 Collaborative Extended Kalman Filter Solution

In order to adjust the Collaborative Extended Kalman Filter solution to account for the inter-receiver distances and to collaborate concurrently, we make adjustments to the $\mathbf{H}$ matrix and $\mathbf{l}$ vector.

Before we begin the description of the Collaborative EKF we start by defining our vectors and matrices. The first vector is the state vector equation to estimate our mean, or position estimate, as seen below.

$$\boldsymbol{\delta} = \begin{bmatrix} \boldsymbol{\delta}_{user_1} & \boldsymbol{\delta}_{user_2} \end{bmatrix}^\top \tag{4.60}$$

Where $\boldsymbol{\delta}_{user_i}$ represents the state vector corresponding to the appropriate user. Then to find a linear approximation of our state vector, we take the partial derivatives to find our state transition matrix as seen below:

$$\boldsymbol{\Phi} = \frac{d\boldsymbol{\delta}}{d\mathbf{x}} = \begin{bmatrix} \boldsymbol{\Phi}_{user_1} & 0 \\ 0 & \boldsymbol{\Phi}_{user_2} \end{bmatrix} \tag{4.61}$$

Where $\boldsymbol{\Phi}_{user_i}$ represents the partial derivatives corresponding to the appropriate user. Next we can define our process noise covariance matrix, $\mathbf{Q}$, and our measurement noise covariance matrix, $\mathbf{R}$, by extending the previous definitions to accommodate the additional states and measurements.

To begin the Collaborative EKF, we start by initializing the state vector and state error

covariance matrix, also known as the uncertainty covariance matrix, as seen below:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_{user_{1_0}} & \mathbf{x}_{user_{2_0}} \end{bmatrix}^\top \tag{4.62}$$

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{user_{1_0}} & 0 \\ 0 & \mathbf{P}_{user_{2_0}} \end{bmatrix} \tag{4.63}$$

where $\mathbf{x}$ is an n x 1 column vector (where n is the length of the state vector or sixteen) and $\mathbf{P}$ is an n x n matrix, $\mathbf{x}_{user_{i_0}}$ corresponds to the initialized state vector for the ith user, and $\mathbf{P}_{user_{i_0}}$ corresponds to the initialized state error covariance for the ith user. Then we start step one of our two-step process by starting our prediction of the state vector and state error covariance matrix as defined below.

$$\mathbf{x}_p = \mathbf{\Phi}\mathbf{x} \tag{4.64}$$

$$\mathbf{P}_p = \mathbf{\Phi}\mathbf{P}\mathbf{\Phi}^\top + \mathbf{Q} \tag{4.65}$$

Where $\mathbf{\Phi}$ is our state transition matrix as defined in Eq. (4.61), $\mathbf{x}$ is our initialized state vector, $\mathbf{P}$ is our state error covariance, and $\mathbf{Q}$ is our process noise covariance as defined in Eq. (4.14).

Next we begin the correction step of our two-step process by calculating the filter gain in which we will use for updating our predictions.

$$\mathbf{K} = \mathbf{P}_p\mathbf{H}^\top(\mathbf{H}\mathbf{P}_p\mathbf{H}^\top + \mathbf{R})^{-1} \tag{4.66}$$

Where $\mathbf{P}_p$ is our predicted state covariance, $\mathbf{R}$ is the measurement noise as defined in Eq. (4.15), and $\mathbf{H}$ is our measurement matrix found by taking the partial derivatives of the pseudoranges and *pseudodistance* measurements, as shown in Eq. (4.3) and Eq. (4.46) respectively, with the coupling matrix, $\mathbf{H}_{i-j}$, containing the *pseudodistance* defined as follows:

$$\begin{aligned} \mathbf{H}_{1-2} &= \begin{bmatrix} \frac{x_{1_0}-x_{2_0}}{d_{1_0 2_0}} & \frac{y_{1_0}-y_{2_0}}{d_{1_0 2_0}} & \frac{z_{1_0}-z_{2_0}}{d_{1_0 2_0}} & -1 & 0 & 0 & 0 & 0 \end{bmatrix} \\ \mathbf{H}_{2-1} &= \begin{bmatrix} \frac{x_{2_0}-x_{1_0}}{d_{1_0 2_0}} & \frac{y_{2_0}-y_{1_0}}{d_{1_0 2_0}} & \frac{z_{2_0}-z_{1_0}}{d_{1_0 2_0}} & -1 & 0 & 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{4.67}$$

where $[x_{1_0}\ y_{1_0}\ z_{1_0}]$ is the initial estimate of the solution of user 1, $[x_{2_0}\ y_{2_0}\ z_{2_0}]$ is the initial estimate of the solution of the user 2, and $d_{1_0 2_0}$ is the inter-receiver distance between the estimated user positions as defined in Eq. (4.47).

To accommodate for the parallel collaboration, we edit the $\mathbf{H}$ matrix with the following components:

$$\mathbf{H} = \begin{bmatrix} \mathbf{H}_{user_1} & 0 \\ 0 & \mathbf{H}_{user_2} \\ \mathbf{H}_{1-2} & \mathbf{H}_{2-1} \end{bmatrix} \tag{4.68}$$

where $\mathbf{H}_{user_i}$ is the appropriate $\mathbf{H}$ matrix for each individual user and can be found from Eq. (4.21) and $\mathbf{H}_{i-j}$ is the partial derivatives of the *pseudodistances* as defined above in Eq. (4.67). To extend the solution beyond two users is the same process that occurred in the Newton-Raphson-based Collaborative solution as seen in the three user example in Eq. (4.53).

This resulting filter gain lets us correct and update our position and uncertainty predictions appropriately.

$$\mathbf{x} = \mathbf{x}_p + \mathbf{K}\mathbf{l} \tag{4.69}$$

$$\mathbf{P} = (\mathbf{I}_n - \mathbf{K}\mathbf{H})\mathbf{P}_p \tag{4.70}$$

Resulting in $\mathbf{x}$ as our updated state vector of the predicted guess and $\mathbf{P}$ as our updated state error covariance matrix. Where $\mathbf{x}_p$ is the predicted guess, $\mathbf{I}_n$ is an n x n identity matrix, $\mathbf{K}$ is our filter gain, and $\mathbf{l}$ is the the difference of the pseudoranges and *pseudodistance* measurements and the estimated ranges and inter-receiver distances, as previously defined in the Newton-Raphson-based Collaborative solution in Eq. (4.54) and redefined below:

$$\mathbf{l}_{1-2} = \left[D_{12} - c(t_{2_0} - t_{1_0}) - d_{1_0 2_0} \right] \tag{4.71}$$

where D_{12} is the *pseudodistance* measurement between users, t_{1_0} and t_{2_0} are the estimated clock offsets associated with each user, and $d_{1_0 2_0}$ is the inter-receiver distance between the estimated user positions. Then to accommodate for the parallel collaboration, we compose

the new $\mathbf{l}$ vector as:

$$\mathbf{l} = \begin{bmatrix} \mathbf{l}_{user_1} \\ \mathbf{l}_{user_2} \\ \mathbf{l}_{1-2} \end{bmatrix} \tag{4.72}$$

where $\mathbf{l}_{user_i}$ is the appropriate $\mathbf{l}$ vector for each individual user and can be found from Eq. (4.9) and $\mathbf{l}_{i-j}$ is the *pseudodistances* $\mathbf{l}$ vector as defined above in Eq. (4.71). Again in order to extend this beyond two users is the same process that occurred in the Newton-Raphson-based Collaborative solution as seen in the three user example in Eq. (4.56).

The same procedure holds for the Collaborative EKF as did for the EKF as listed below:

1. Initialize state vector and state error covariance (or use the state vector and state error covariance from the previous time measurement):
   $\mathbf{x}$ and $\mathbf{P}$

2. Predict state vector and state error covariance:
   $\mathbf{x}_p = \mathbf{\Phi x}$
   $\mathbf{P}_p = \mathbf{\Phi P \Phi}^\top + \mathbf{Q}$

3. Compute Kalman Filter Gain:
   $\mathbf{K} = \mathbf{P}_p\mathbf{H}^\top(\mathbf{H}\mathbf{P}_p\mathbf{H}^\top + \mathbf{R})^{-1}$

4. Update and correct predicted state vector and state error covariance:
   $\mathbf{x} = \mathbf{x}_p + \mathbf{Kl}$
   $\mathbf{P} = (\mathbf{I}_n - \mathbf{KH})\mathbf{P}_p$

This process results in an estimated state vector for the two GPS receivers simultaneously, $[\mathbf{x}_{user_1}\ \mathbf{x}_{user_2}]^\top$ and the state error covariance matrix that is used in the next measurement to further improve the estimated state vectors over time.

4.3.3 Collaborative Unscented Kalman Filter Solution

In order to adjust the Collaborative Unscented Kalman Filter solution to account for the inter-receiver distances and to collaborate concurrently, the predicted mean associated with measurements is affected.

Before we begin the Collaborative UKF we start by defining our vectors and matrices, exactly as we did with the Collaborative EKF. The first vector is the state vector equation to estimate our mean, or position estimate, as seen below.

$$\boldsymbol{\delta} = \begin{bmatrix} \boldsymbol{\delta}_{user_1} & \boldsymbol{\delta}_{user_2} \end{bmatrix}^\top \tag{4.73}$$

Where $\boldsymbol{\delta}_{user_i}$ represents the state vector corresponding to the appropriate user.

Next we can define our process noise covariance matrix, $\mathbf{Q}$, and our measurement noise covariance matrix, $\mathbf{R}$, by extending the previous definitions to accommodate the additional states and measurements.

To begin the Collaborative UKF, we start by initializing the state vector and state error covariance matrix, also known as the uncertainty covariance matrix, as seen below:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_{user_{1_0}} & \mathbf{x}_{user_{2_0}} \end{bmatrix}^\top \tag{4.74}$$

$$\mathbf{P} = \begin{bmatrix} \mathbf{P}_{user_{1_0}} & 0 \\ 0 & \mathbf{P}_{user_{2_0}} \end{bmatrix} \tag{4.75}$$

where $\mathbf{x}$ is an n x 1 column vector (where n is the length of the state vector or sixteen) and $\mathbf{P}$ is an n x n matrix, $\mathbf{x}_{user_{i_0}}$ corresponds to the initialized state vector for the ith user, and $\mathbf{P}_{user_{i_0}}$ corresponds to the initialized state error covariance for the ith user.

Then in order to start the Collaborative UKF we begin with the sigma points selection algorithm around $\mathbf{x}$ by generating the sigma points through the unscented transformation (UT). In order to find at least the second order of the state vector we must use $2n+1$ sigma

points in the form below.

$$\mathbf{X}_{sigma} = \begin{bmatrix} \mathbf{x} & \mathbf{x} + \sqrt{(n+\lambda)\mathbf{P}} & \mathbf{x} - \sqrt{(n+\lambda)\mathbf{P}} \end{bmatrix} \tag{4.76}$$

This results in $\mathbf{X}_{sigma}$ a n x $2n+1$ matrix where $\mathbf{x}$ is the initial state vector, λ is a scaling factor as defined in Eq. (4.31), $\mathbf{P}$ is the state error covariance, and Cholesky factorization is used for the square root of the matrices. Next, we find the weights of the sigma points associated with both the mean and covariance as seen below.

$$\mathbf{w}_0^{(m)} = \frac{\lambda}{n+\lambda} \tag{4.77}$$

$$\mathbf{w}_0^{(c)} = \mathbf{w}_0^{(m)} + (1 - \alpha^2 + \beta) \tag{4.78}$$

$$\mathbf{w}_i^{(m)} = \mathbf{w}_i^{(c)} = \frac{1}{2(n+\lambda)} \qquad i = 1, ..., 2n \tag{4.79}$$

where λ is again the scaling factor as defined in Eq. (4.31), α is the spread of the sigma points and is set to 1e-3 (a typical default value), and β is the distribution parameter and is set to 2 to declare a normal distribution.

Next start with the prediction of the state vector and state error covariance matrix as defined below.

$$\mathbf{X}_p = \boldsymbol{\delta}(\mathbf{X}_{sigma}) \tag{4.80}$$

$$\hat{\mathbf{x}} = \sum_{i=0}^{2n} \mathbf{w}_i^{(m)} \mathbf{X}_p \tag{4.81}$$

$$\mathbf{P}_p = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)} (\mathbf{X}_p - \hat{\mathbf{x}})(\mathbf{X}_p - \hat{\mathbf{x}})^\top + \mathbf{Q} \tag{4.82}$$

Resulting in $\mathbf{X}_p$ a n x $2n+1$ containing our sigma points passed through the state vector equation in Eq. (4.73), $\hat{\mathbf{x}}$ as a n x 1 predicted mean state vector, and $\mathbf{P}_p$ as a n x n predicted state error covariance matrix. Where $\mathbf{w}_i^{(m)}$ is our sigma point weights associated with the mean, $\mathbf{w}_i^{(c)}$ is our sigma point weights associated with the covariance, and $\mathbf{Q}$ is our process noise covariance.

Next we find the range and inter-receiver associated with each of the predicted sigma points, $\mathbf{X}_p$, and the predicted mean associated with the range and inter-receiver measurements. Before we show this process we include the collaboration of the inter-receiver measurements as shown below:

$$\begin{aligned} f_{1-2}(\mathbf{X}_p) &= \sqrt{(x_{1_{\mathbf{X}_p}} - x_{2_{\mathbf{X}_p}})^2 + (y_{1_{\mathbf{X}_p}} - y_{2_{\mathbf{X}_p}})^2 + (z_{1_{\mathbf{X}_p}} - z_{2_{\mathbf{X}_p}})^2} + c(t_{2_{\mathbf{X}_p}} - t_{1_{\mathbf{X}_p}}) \\ f_{2-1}(\mathbf{X}_p) &= \sqrt{(x_{2_{\mathbf{X}_p}} - x_{1_{\mathbf{X}_p}})^2 + (y_{2_{\mathbf{X}_p}} - y_{1_{\mathbf{X}_p}})^2 + (z_{2_{\mathbf{X}_p}} - z_{1_{\mathbf{X}_p}})^2} + c(t_{1_{\mathbf{X}_p}} - t_{2_{\mathbf{X}_p}}) \end{aligned} \tag{4.83}$$

where $f_{1-2}(\mathbf{X}_p)$ and $f_{2-1}(\mathbf{X}_p)$ are the estimated inter-receiver range measurement between users at the predicted sigma points, $\mathbf{X}_p$, using Eq. (4.45) and redefined above. Then to accommodate for the parallel collaboration, we compose the new $f(\mathbf{X}_p)$ function including the following:

$$f(\mathbf{X}_p) = \begin{bmatrix} f_{user_1}(\mathbf{X}_p) \\ f_{user_2}(\mathbf{X}_p) \\ f_{1-2}(\mathbf{X}_p) \\ f_{2-1}(\mathbf{X}_p) \end{bmatrix} \tag{4.84}$$

Where $f_{user_i}(\mathbf{X}_p)$ is the range measurements from each of the predicted sigma points associated with the appropriate user in $\mathbf{X}_p$ to the ith satellite using Eq. (4.1), and $f_{i-j}(\mathbf{X}_p)$ is the estimated inter-receiver range measurement between users at the predicted sigma points as defined above in Eq. (4.83). The repeated inter-receiver range estimate in Eq. (4.84), $f_{1-2}(\mathbf{X}_p)$ and $f_{2-1}(\mathbf{X}_p)$, is necessary to associate the *pseudodistance* measurement for each user when using the Collaborative UKF and unfortunately is a result to the initial degradation that will be seen in Chapter 5.1. Then to find the range and inter-receiver associated with each of the predicted sigma points, $\mathbf{X}_p$, and the predicted mean associated with the range and inter-receiver measurements, we use the equations below.

$$\mathbf{Z}_p = f(\mathbf{X}_p) \tag{4.85}$$

$$\hat{\mathbf{z}} = \sum_{i=0}^{2n} \mathbf{w}_i^{(m)} \mathbf{Z}_p \tag{4.86}$$

Where $f(\mathbf{X}_p)$ is the estimated measurements at each of the predicted sigma points as defined above in Eq. (4.84), $\mathbf{w}_i^{(m)}$ is our sigma point weights associated with the mean, and $\mathbf{w}_i^{(c)}$ is our sigma point weights associated with the covariance.

Then in order to calculate our filter gain and update our predicted mean and covariance, we first must find the innovation covariance, $\mathbf{P}_z$ and the cross-covariance, $\mathbf{P}_{xz}$, as defined below.

$$\mathbf{P}_z = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)} (\mathbf{Z}_p - \hat{\mathbf{z}})(\mathbf{Z}_p - \hat{\mathbf{z}})^\top + \mathbf{R} \tag{4.87}$$

$$\mathbf{P}_{xz} = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)} (\mathbf{X}_p - \hat{\mathbf{x}})(\mathbf{Z}_p - \hat{\mathbf{z}})^\top \tag{4.88}$$

Letting us define the filter gain as follows:

$$\mathbf{K} = \mathbf{P}_{xz}\mathbf{P}_z^{-1} \tag{4.89}$$

This resulting filter gain lets us correct and update our predicted mean and covariance appropriately.

$$\mathbf{x} = \hat{\mathbf{x}} + \mathbf{K}(\mathbf{z} - \hat{\mathbf{z}}) \tag{4.90}$$

$$\mathbf{P} = \mathbf{P}_p + \mathbf{K}\mathbf{P}_z\mathbf{K}^\top \tag{4.91}$$

This results in $\mathbf{x}$ as our updated state vector of the predicted mean and $\mathbf{P}$ as our updated state error covariance matrix. Note that $\hat{\mathbf{x}}$ is the predicted mean, $\hat{\mathbf{z}}$ is the the predicted mean associated with the range measurements, $\mathbf{P}_p$ is the predicted covariance, $\mathbf{K}$ is our filter gain, $\mathbf{z}$ is the vector of the pseudorange and *pseudodistance* measurements, and $\mathbf{P}_z$ is the innovation covariance.

The same procedure holds for the Collaborative UKF as did for the UKF as listed below:

1. Initialize state vector and state error covariance (or use the state vector and state error covariance from the previous time measurement):
   $\mathbf{x}$ and $\mathbf{P}$

2. Sigma Point Selection Algorithm:

   $$\mathbf{X}_{sigma} = \begin{bmatrix} \mathbf{x} & \mathbf{x} + \sqrt{(n+\lambda)\mathbf{P}} & \mathbf{x} - \sqrt{(n+\lambda)\mathbf{P}} \end{bmatrix}$$

3. Predict state vector and state error covariance:

   $$\mathbf{X}_p = \boldsymbol{\delta}(\mathbf{X}_{sigma})$$
   $$\hat{\mathbf{x}} = \sum_{i=0}^{2n} \mathbf{w}_i^{(m)} \mathbf{X}_p$$
   $$\mathbf{P}_p = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)} (\mathbf{X}_p - \hat{\mathbf{x}})(\mathbf{X}_p - \hat{\mathbf{x}})^\top + \mathbf{Q}$$

4. Predict mean associated with the range and inter-receiver range measurements:

   $$\mathbf{Z}_p = f(\mathbf{X}_p)$$
   $$\hat{\mathbf{z}} = \sum_{i=0}^{2n} \mathbf{w}_i^{(m)} \mathbf{Z}_p$$

5. Calculate innovation covariance:

   $$\mathbf{P}_z = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)} (\mathbf{Z}_p - \hat{\mathbf{z}})(\mathbf{Z}_p - \hat{\mathbf{z}})^\top + \mathbf{R}$$

6. Calculate cross-covariance:

   $$\mathbf{P}_{xz} = \sum_{i=0}^{2n} \mathbf{w}_i^{(c)} (\mathbf{X}_p - \hat{\mathbf{x}})(\mathbf{Z}_p - \hat{\mathbf{z}})^\top$$

7. Compute Kalman Filter Gain:

   $$\mathbf{K} = \mathbf{P}_{xz} \mathbf{P}_z^{-1}$$

8. Update and correct predicted state vector and state error covariance:

   $$\mathbf{x} = \hat{\mathbf{x}} + \mathbf{K}(\mathbf{z} - \hat{\mathbf{z}})$$
   $$\mathbf{P} = \mathbf{P}_p + \mathbf{K}\mathbf{P}_z\mathbf{K}^\top$$

This process results in an estimated state vector for the GPS receivers simultaneously, $[\mathbf{x}_{user_1}\ \mathbf{x}_{user_2}]^\top$ and the state error covariance matrix that is used in the next measurement to further improve the estimated state vectors over time.

Chapter 5

Collaborative GPS Performance

We begin by examining the solutions of the standard GPS receiver to find a basis for comparison. After running our Monte Carlo simulation, we were able to find statistics of the mean error associated with the receiver's positioning as well as the availability of the position estimate. The position error is calculated by finding the distance from the estimated position to the predetermined true location, $E|\mathbf{x} - \hat{\mathbf{x}}|$, and the availability is found by the number of occurrences in which the number of satellites is less than four, the error is greater than 1 km, or the absolute value of the GDOP is greater than 1000, divided by the total number of simulation runs. The results, as seen in Table 5.1, are consistent with available data [9] [16].

Table 5.1: Standard GPS simulated Performance in various environments

Clear Sky	**Solution**	**Error**	**Availability**
	Newton-Raphson	11.763 m	100.00%
	Extended KF	8.198 m	100.00%
	Unscented KF	8.276 m	100.00%
Urban	**Solution**	**Error**	**Availability**
	Newton-Raphson	26.079 m	99.15%
	Extended KF	13.653 m	99.21%
	Unscented KF	13.630 m	99.21%
Window	**Solution**	**Error**	**Availability**
	Newton-Raphson	68.645 m	72.08%
	Extended KF	26.352 m	72.78%
	Unscented KF	26.349 m	72.85%
Indoor	**Solution**	**Error**	**Availability**
	Newton-Raphson	101.046 m	36.00%
	Extended KF	35.277 m	36.86%
	Unscented KF	35.239 m	36.95%

As seen in the table above, both of the Kalman Filter solutions outperform the Newton-Raphson solution in regards to accuracy of the position estimate and availability. Further, the accuracy of the position estimate for both the Kalman Filter solutions are comparable to one another as expected. However the UKF results in an equivalent or better availability in all scenarios over the EKF due its resistance to the bias of NLOS component.

An example is shown below in Fig. 5.1, where the three solutions position errors are tracked over 100 different time stamps, or the next satellite position in the sky, for a Clear Sky user with the UKF solution providing the best results.

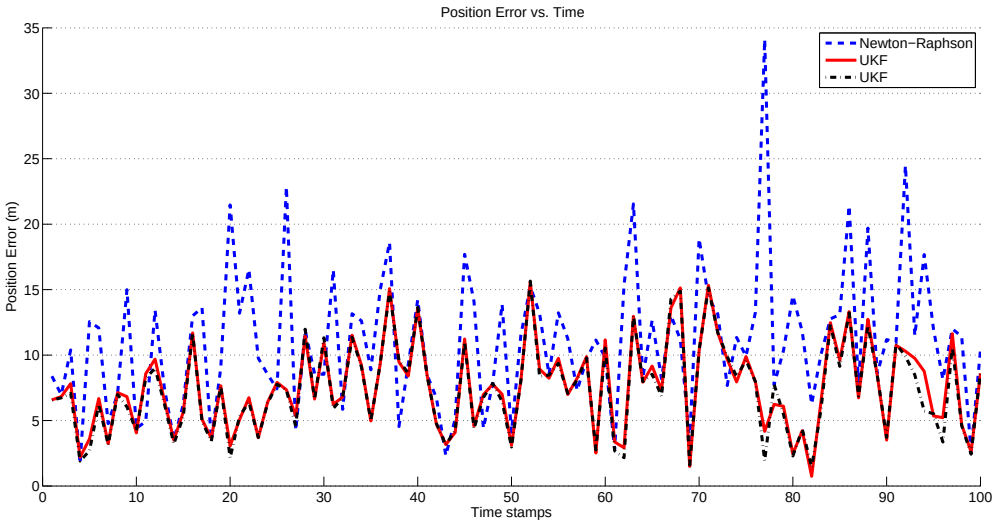


Figure 5.1: Example Position Error of a Clear Sky User vs Time

5.1 Collaborative Performance

After running our new solutions Monte Carlo for two collaborating users, we again find the mean error associated with the receivers' positioning as well as the availability of a position estimate with the results seen in Table 5.2 and Table 5.3.

Table 5.2: Two User Collaborative GPS simulated Performance Enhancements Part I

Clear Sky & Clear Sky	**Solution**	**User**	**Error**	**Availability**
	Newton-Raphson's	Clear Sky	10.125 m	100.00%
		Clear Sky	10.154 m	100.00%
	Extended KF	Clear Sky	7.462 m	100.00%
		Clear Sky	7.530 m	100.00%
	Unscented KF	Clear Sky	7.509 m	100.00%
		Clear Sky	7.517 m	100.00%
Clear Sky & Urban	**Solution**	**User**	**Error**	**Availability**
	Newton-Raphson	Clear Sky	10.614 m	100.00%
		Urban	18.995 m	99.94%
	Extended KF	Clear Sky	7.646 m	100.00%
		Urban	11.990 m	99.94%
	Unscented KF	Clear Sky	7.613 m	100.00%
		Urban	12.335 m	99.95%
Clear Sky & Window	**Solution**	**User**	**Error**	**Availability**
	Newton-Raphson	Clear Sky	11.460 m	100.00%
		Window	47.949 m	90.50%
	Extended KF	Clear Sky	7.914 m	100.00%
		Window	21.232 m	90.66%
	Unscented KF	Clear Sky	7.861 m	100.00%
		Window	32.330 m	90.42%
Clear Sky & Indoor	**Solution**	**User**	**Error**	**Availability**
	Newton-Raphson	Clear Sky	11.634 m	100.00%
		Indoor	84.835 m	63.69%
	Extended KF	Clear Sky	8.063 m	100.00%
		Indoor	30.708 m	64.04%
	Unscented KF	Clear Sky	7.977 m	100.00%
		Indoor	51.114 m	63.94%

Table 5.3: Two User Collaborative GPS simulated Performance Enhancements Part II

Urban & Urban	**Solution**	**User**	**Error**	**Availability**
	Newton-Raphson	Urban	19.713 m	99.93%
		Urban	20.987 m	99.92%
	Extended KF	Urban	12.284 m	99.93%
		Urban	12.478 m	99.93%
	Unscented KF	Urban	13.394 m	99.95%
		Urban	13.438 m	99.95%
Window & Window	**Solution**	**User**	**Error**	**Availability**
	Newton-Raphson	Window	67.904 m	79.54%
		Window	63.815 m	79.46%
	Extended KF	Window	24.585 m	81.85%
		Window	24.625 m	81.80%
	Unscented KF	Window	30.527 m	81.86%
		Window	30.608 m	81.96%
Indoor & Indoor	**Solution**	**User**	**Error**	**Availability**
	Newton-Raphson	Indoor	111.842 m	40.88%
		Indoor	108.459 m	40.87%
	Extended KF	Indoor	34.618 m	46.25%
		Indoor	35.348 m	46.10%
	Unscented KF	Indoor	43.227 m	46.12%
		Indoor	45.489 m	45.89%

From the tables above, we can see that the two user scenarios resulted in significant improvements in the availability for every paired scenario, especially in the Clear Sky & Indoor scenario where the availability nearly doubled from roughly 37% to 64%. Furthermore, the accuracy of the users position errors improved for nearly all scenarios with the exception of a few (specifically Indoor-Indoor), that had minor degradation to the position error. The result of this minor degradation in the position error is a result of cases where localization was not possible previously but can now occur do to collaboration thus improving the availability but degrading the position error. This concept is explored further in Chapter 5.3.

5.1.1 Position Error

After introducing additional collaborating users to the new solution we examined the results of the mean position error for the Newton-Raphson-based Collaborative Solution, the Col-

laborative EKF Solution, and the Collaborative UKF Solution as seen in Fig. 5.2, Fig. 5.3, and Fig. 5.4 respectively. All of the figures showed continuous improvements in the position error for nearly all of the simulated scenarios with the exceptions of the initial degradation, specifically in Indoor-Indoor, that will be addressed in Chapter 5.3.

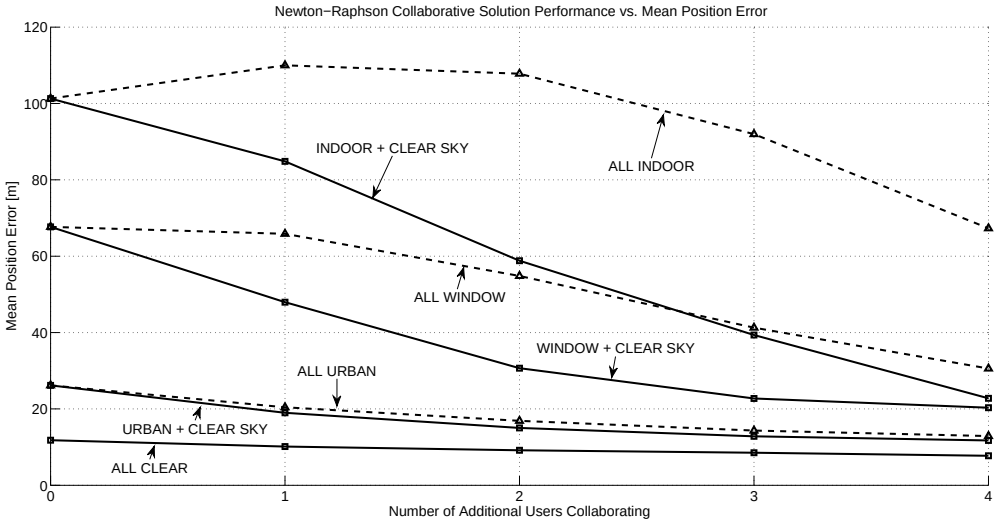


Figure 5.2: The mean position error of users using the Newton-Raphson-based Collaborative Solution

As seen in the figure above for the Newton-Raphson-based Collaborative Solution, the Indoor-Indoor scenario struggled initially before showing improvement in the position error with three additional users collaborating due to the increased availability. On the other hand, in the Indoor-Clear Sky scenario, the indoor user showed significant improvements in the position error as additional users were introduced.

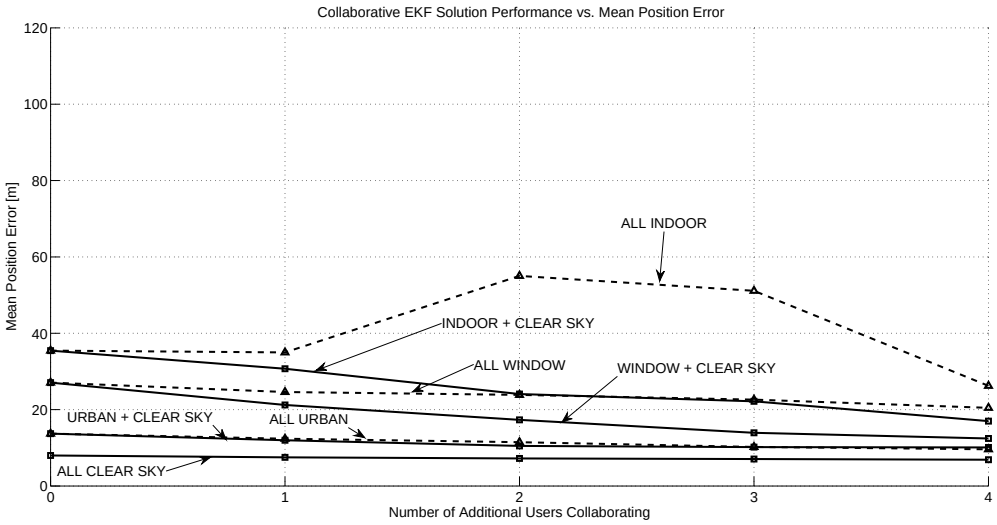


Figure 5.3: The mean position error of users using the Collaborative EKF Solution

In the figure above for the Collaborative EKF Solution, the Indoor-Indoor scenario also struggled before showing improvement in the position error when four additional users were collaborating, again due to the cases where localization was previously not possible. All other scenarios showed a steady trend in improving the position estimate as additional users were introduced.

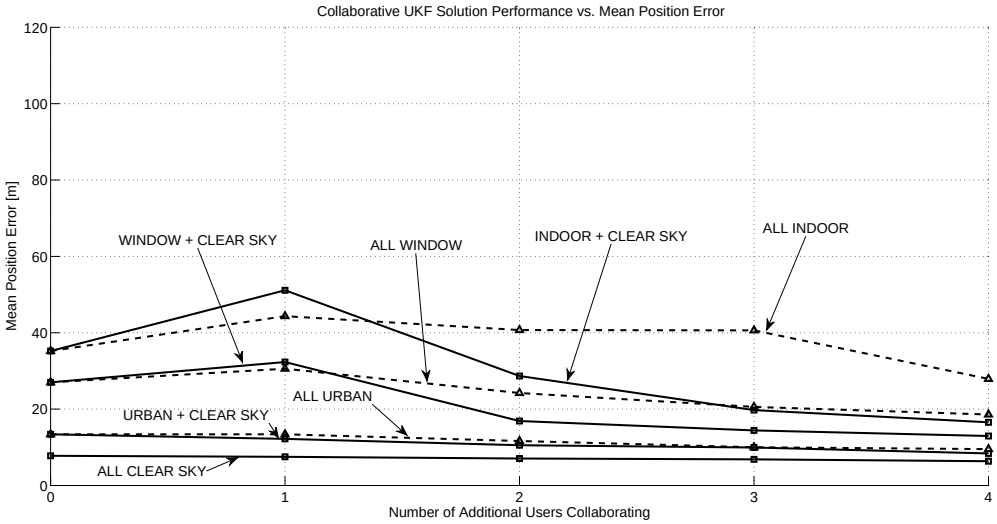


Figure 5.4: The mean position error of users using the Collaborative UKF Solution

In the figure above for the Collaborative UKF Solution, despite the Indoor-Indoor scenario

showing less degradation in the position error than the Collaborative EKF it showed less of an improvement in the position error when four additional users were collaborating. All other scenarios showed a steady trend in improving the position estimate as additional users were introduced with the exception of the Indoor-Clear Sky and Window-Clear Sky that initially degraded before showing improvement in the position error.

Next we examined this degradation further by comparing the mean position error of an indoor user when collaborating with a number of clear sky users as shown in Fig. 5.5 below.

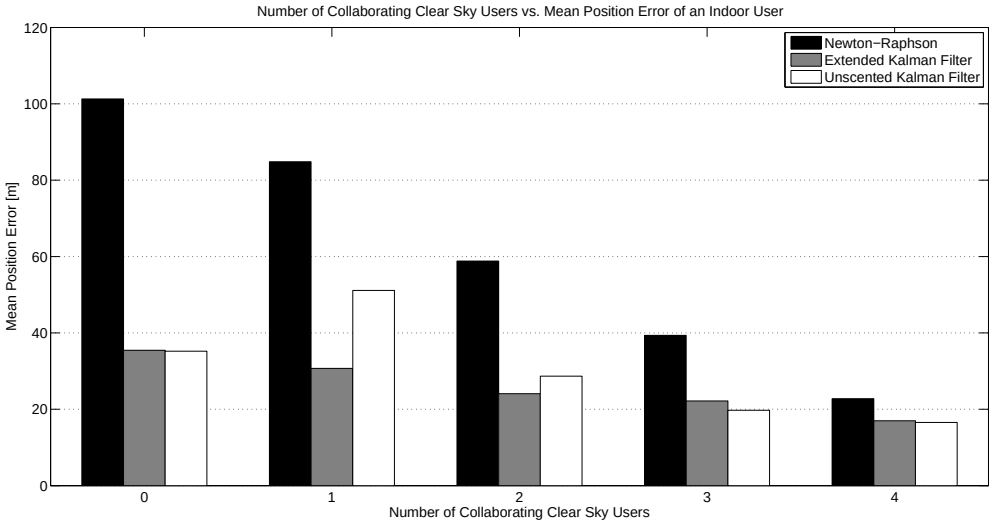


Figure 5.5: The mean position error of an indoor user vs. the number of collaborating clear sky users

After a thorough comparison, we concluded that the Collaborative EKF solution provided the best results in regards to the mean position error.

5.1.2 Availability

Next we examine the mean availability for the Newton-Raphson-based Collaborative Solution, the Collaborative EKF Solution, and the Collaborative UKF Solution as seen below in Fig. 5.6, Fig. 5.7, and Fig. 5.8 respectively. All of the figures below show continuous improvements in the availability for all simulated scenarios and with nearly 100% availability for all scenarios when five users are collaborating and when using either one of the Collaborative

Kalman Filter solutions, although not for the all indoor scenarios.

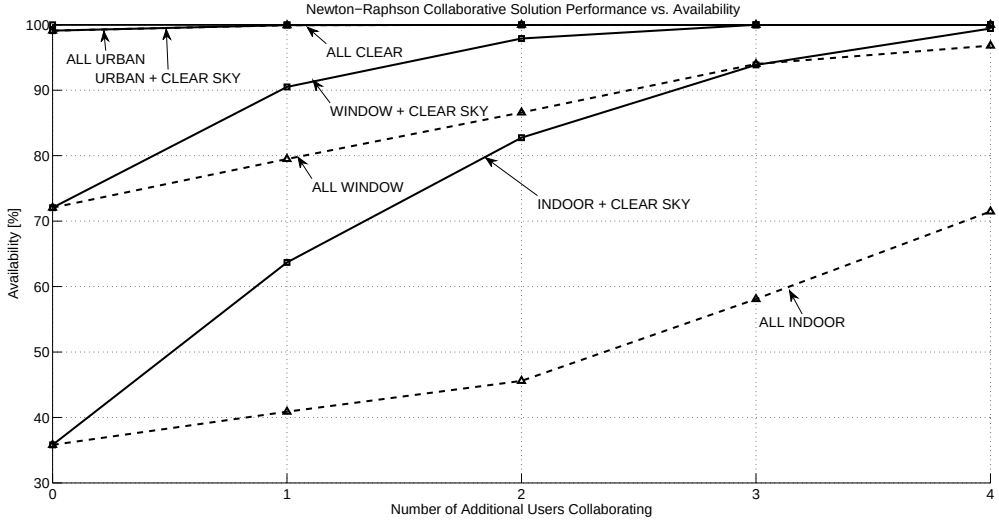


Figure 5.6: The availability of position estimate using the Newton-Raphson-based Collaborative Solution

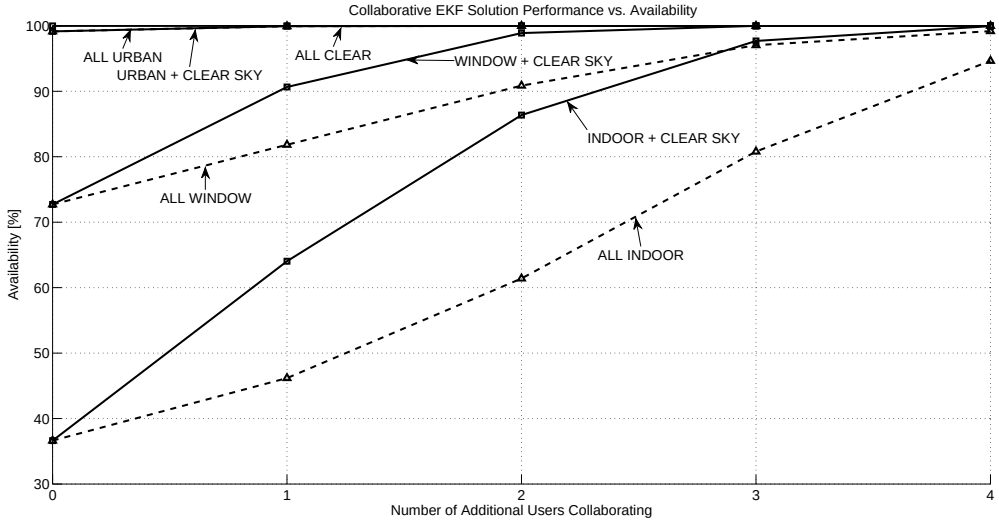


Figure 5.7: The availability of position estimate using the Collaborative EKF Solution

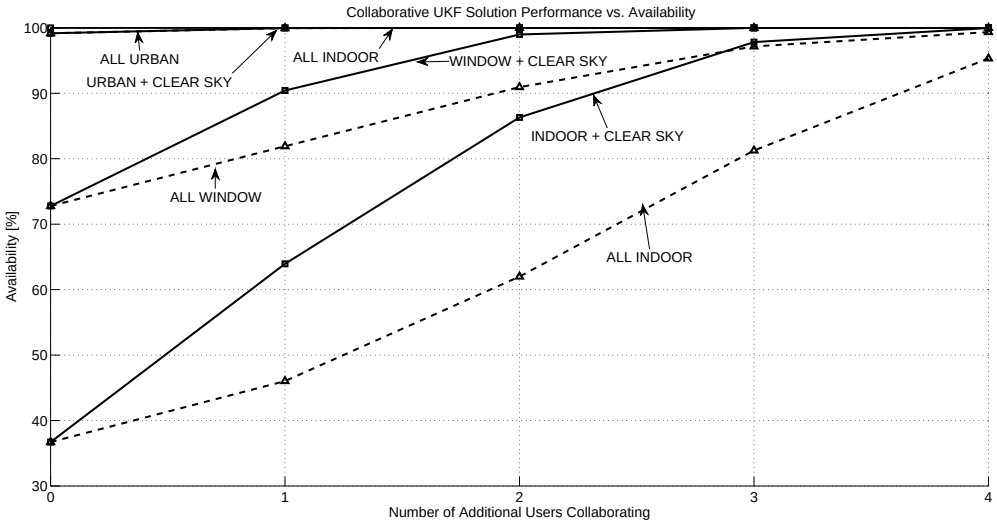


Figure 5.8: The availability of position estimate using the Collaborative UKF Solution

We examined the results further by comparing the availability of an indoor user when collaborating with a number of clear sky users as shown in Fig. 5.9 below. Here we examined the most drastic improvements that each solution can achieve to one another as we did before with the mean position error.

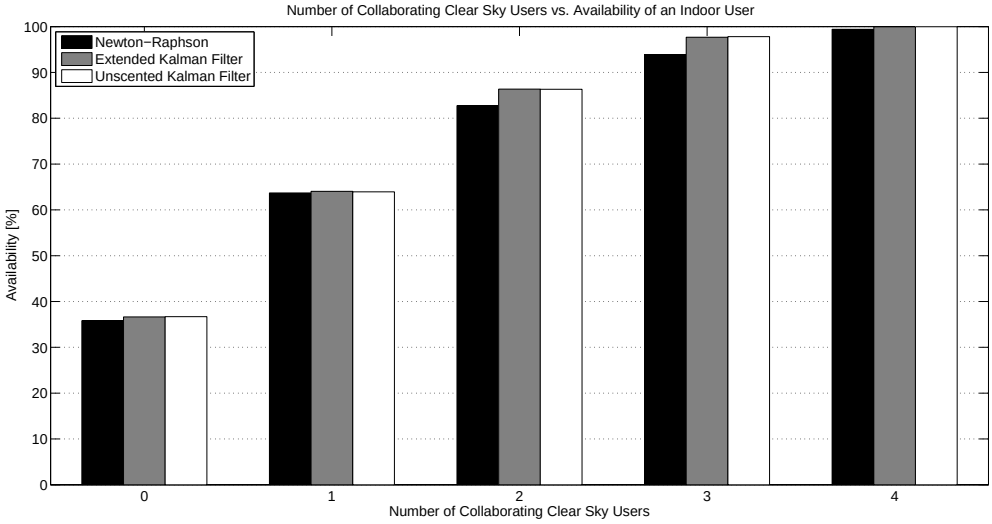


Figure 5.9: The availability of an indoor user vs. the number of collaborating clear sky users

After a thorough comparison, we concluded that both the Collaborative UKF solution and Collaborative EKF solution provided the best results in regards to the availability of users

for all of the simulated scenarios by outperforming the Newton-Raphson-based Collaborative solution. This can be contributed to the better convergence properties the Kalman Filter holds over the Newton-Raphson solution.

5.2 Positioning Accuracy using the 95% CEP

To further explore the position accuracy of the new solutions, we examined the statistical data of the position errors for each scenario and the corresponding cumulative density functions (CDFs). Using CDFs gives us the ability to measure the Circular Error Probability (CEP), or the radius of a circle centered about the true position whose boundary is expected to include a percentage of the estimated positions within it, for every scenario. We focus on the 95% CEP, where nearly all the position estimates fall, in order to show the improvements the new collaborative solutions makes over the standard GPS solution. An example of a 95% CEP with a Clear Sky user with the Newton-Raphson solution is shown below in Fig. 5.10.

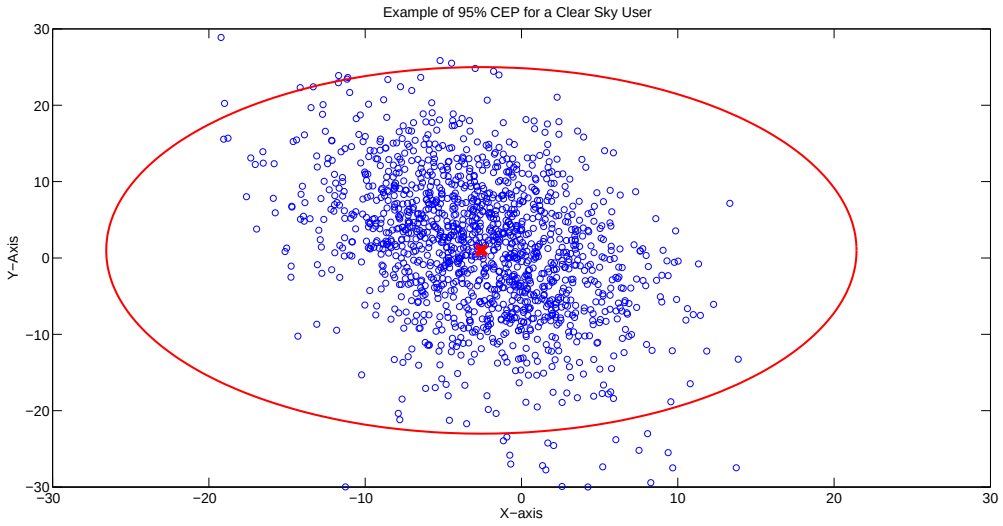


Figure 5.10: Example of a 95% CEP for a Clear Sky user using the Newton-Raphon Solution

For simplicity purposes we only cover the Newton-Raphson Collaborative solution, since all three collaborative solutions follow similar distributions as seen in Appendix A.

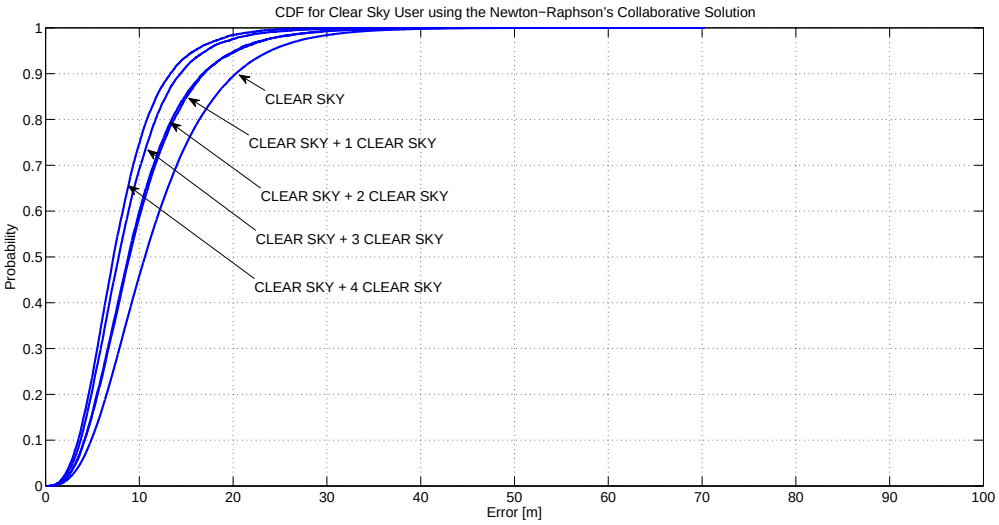


Figure 5.11: CDF of Position Error for Clear Sky user using the Newton-Raphon Collaborative Solution

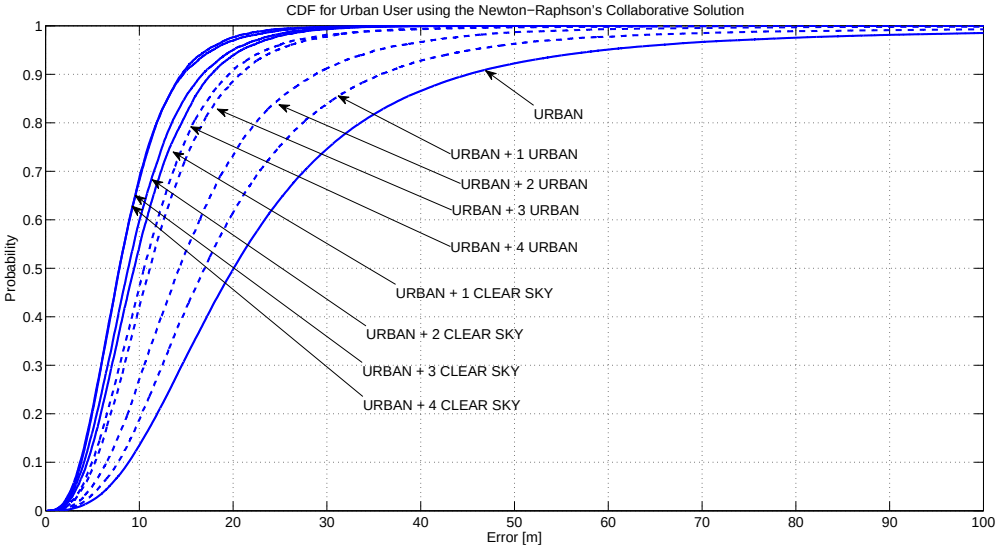


Figure 5.12: CDF of Position Error for Urban User using the Newton-Raphon Collaborative Solution

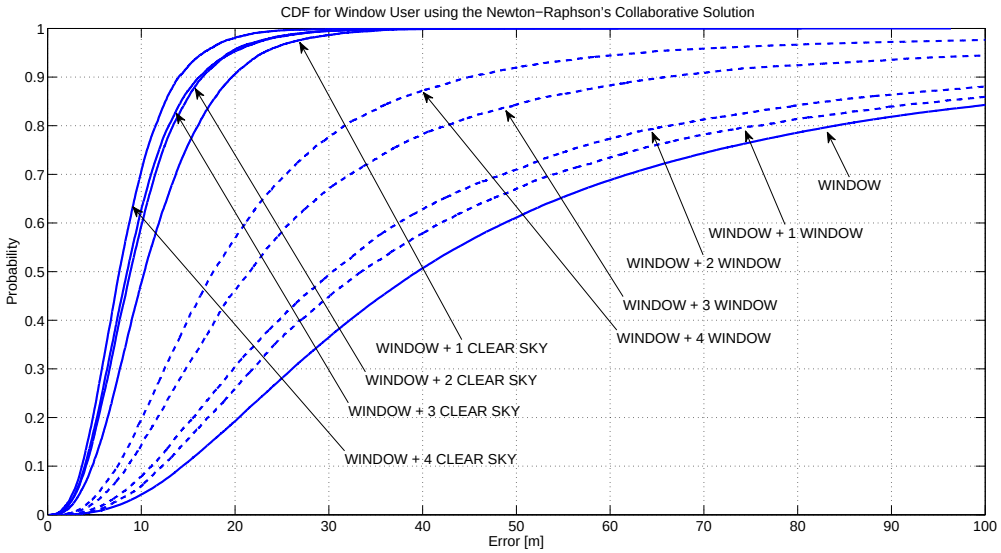


Figure 5.13: CDF of Position Error for Window User using the Newton-Raphon Collaborative Solution

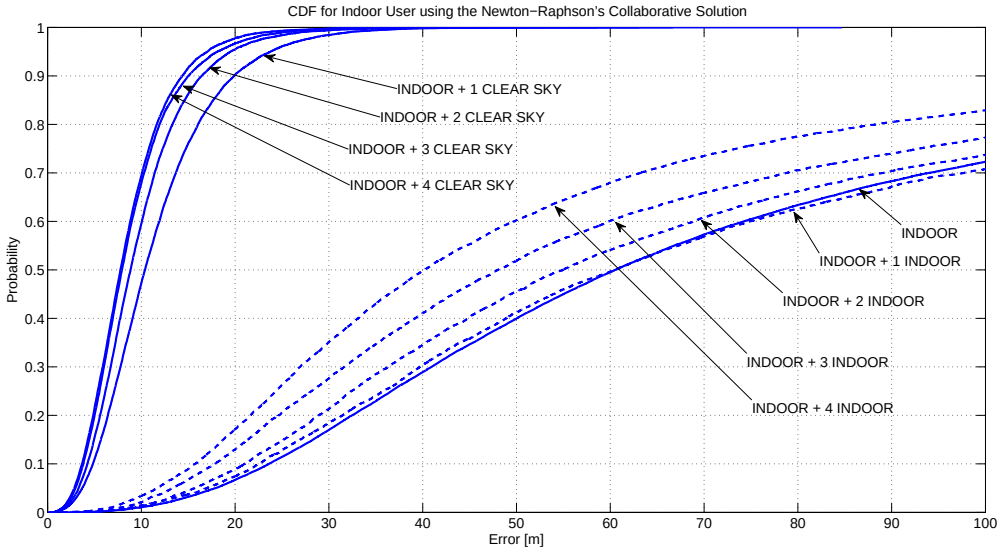


Figure 5.14: CDF of Position Error for Indoor User using the Newton-Raphon Collaborative Solution

Beginning with clear sky users in Fig. 5.11, we observed a moderate improvement in their 95% CEP from 24 m to 15 m when collaborating with four other clear sky users. Urban users in Fig. 5.12 made an impressive enhancement in their 95% CEP improving from 60 m to 23 m when collaborating with four other urban users and improved even further when collaborating with four clear sky users from 60 m to 17 m. Window users in Fig. 5.13

made an impressive enhancement in their 95% CEP improving from 214 m to 64 m when collaborating with four other window users, however when collaborating with four clear sky users their 95% CEP improved notably from 214 m to 17 m. Indoor users in Fig. 5.14 made a modest enhancement in their 95% CEP improving from 338 m to 240 m when collaborating with four other indoor users, however when collaborating with four clear sky users their 95% CEP improved drastically from 338 m to 18 m. The improvements made in the 95% CEP for all of the new collaborative solutions can be seen in the tables in Appendix A. This reveals significant improvements in position estimate that the new collaborative solutions are able to achieve over the standard GPS receiver. More importantly, several scenarios that were unable to meet the 95% E911 requirement of 150 m can now meet or exceed this requirement through collaborative GPS [28].

5.3 Sensitivity to Satellites

In exploring the initial degradation in position error, we examined the sensitivity of the solutions to the number of satellites used. By comparing the number of satellites used in localization versus the accuracy of the position estimate we concluded they are correlated, as seen in Fig. 5.15. When less satellites are available to a user then the solution typically reduces the accuracy of the position estimate, as one would expect.

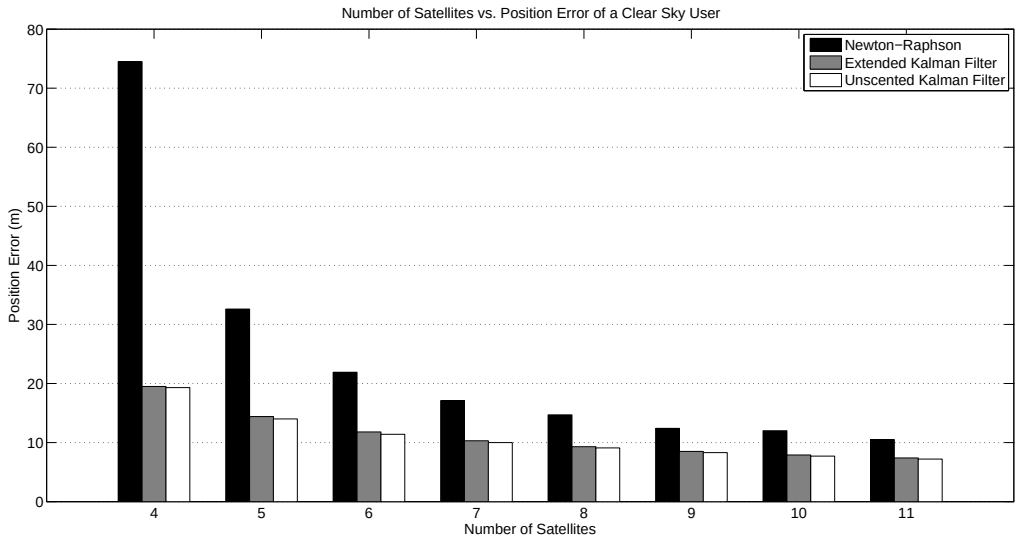


Figure 5.15: The mean position error for a Clear Sky user versus the number of satellites

As seen in the figure above, this correlation holds true for all of the solutions resulting in further degradation to the position estimate when less than four satellites are used in collaborative localization and helps explain the initial degradation that occurs in the position error, as seen in the results in Chapter 5.1. As an example, we examined the results of a user in a challenging environment collaborating with a clear sky user when using the Newton-Raphson-based Collaborative solution. When a window user using four or more satellites collaborates with a clear sky user, localization is achieved for the window user 73% of the time with a mean position error of 33 m. However when a window user using only three satellites collaborates with a clear sky user, localization is achieved for the window user 18% of the time with mean position error of 110 m. This results in the overall mean position error of 48 m with an availability of 91% for the window user in collaboration with a clear sky user. When an indoor user using four or more satellites collaborates with a clear sky user localization is achieved for the indoor user 37% of the time with a mean position error of 55 m. However when an indoor user using only three satellites collaborates with a clear sky users, localization is achieved for the indoor user 27% of the time with a position error of 126 m. This results in the overall mean position error of 85 m with an availability of 64% for the indoor user in collaboration with a clear sky user.

As expected, the large error that occurs from three satellites in collaboration causes degradation in the position estimate. Further degradation occurs in the position estimate when one receiver uses two satellites in collaboration with several Clear Sky users (with access to several satellites) and even further when one receiver uses only one satellite in collaboration with several Clear Sky users (with access to several satellites). Thus if accuracy of the position estimate was a priority over availability one could implement a constraint to require the standard four satellite minimum in order to only improve the position estimate.

When users share the same satellites in collaboration their solution degrades further due to the scale of the system. In comparison to the distance from the satellites to the users, the distance between users is minuscule causing ambiguity in the solution and possible convergence to a local minima.

To get a deeper understanding of the where the gains in performance occur in collaboration, we find the probability associated with the number of available satellites in each environmental scenario as shown in Fig. 5.16. With this information we are able to show that the largest source of improvement occurs in the availability where previous solutions were not possible. For example, an indoor user with three satellites has a probability of occurrence of 27%. When using a standard GPS, localization is not possible, however when the indoor user collaborates with a clear sky user, localization is now possible for the indoor user in all occurrences where only three satellites are available by having the additional user act as a replacement for a satellite.

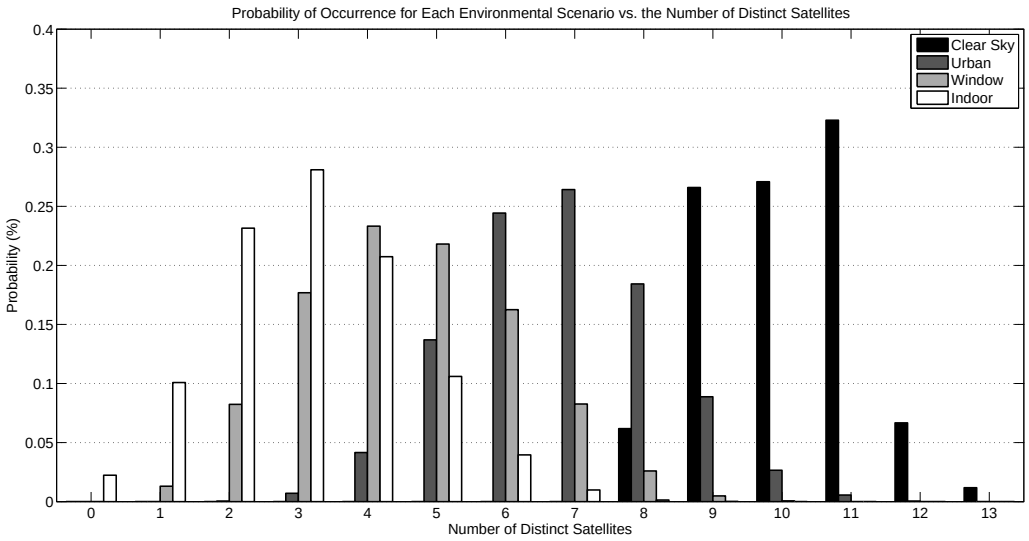


Figure 5.16: Probability of occurrence for each environmental scenario versus the number of distinct satellites

Not all satellite scenarios result in localization. When using any of the new collaborative solutions, a user with three satellites may collaborate with another user with four or more satellites. However, if both users only have three satellites available then neither can achieve localization. A minimum of three users are required for localization when using three satellites each and a minimum of five users are required for the localization when using two satellites each. Furthermore, if all users have one satellite each, localization will not occur for all collaborative solutions due to a lack of reference from sufficient measurements and a poor dilution of precision.

When examining the sensitivity of a receiver to the initial position guess it is found that regardless of the starting location a solution is found that meets the accuracy of a receiver in a given environment. However it is important to note that the further the initial position guess is from the true receiver location the more iterations or time measurements are required to find a solution that meets the accuracy. A closer initial position guess does improve performance by reducing the number of computations needed to resolve a solution. Lastly, when examining the sensitivity to the inter-receiver range error it is found that the signal strength of the inter-receiver does not affect the solutions noticeably. This is a result of the kalman gain and the weighted matrix giving a smaller weight on the weaker measurements

including the inter-receiver range measurements.

5.4 Time Synchronization

Another important feature that GPS uses is the ability to time synchronize all nodes within nanoseconds, depending on the node's environment, for successful data communication. By using high accuracy clocks for time synchronization this supports additional network capabilities, such as advanced modulation schemes yielding higher data rates. The timing synchronization capability makes GPS receivers popular for implementation, even if localization is unnecessary, since many wireless networks have relatively inexpensive hardware clocks. As mentioned above, the environment not only affects the position measurement but also the accuracy of the time synchronization. So we examined the accuracy of time offset when using the new collaborative solution to further improve the time synchronization.
We began with the solution of the standard GPS receiver with respect to the timing to find a basis for comparison. After running our Monte Carlo simulation, we were able to find statistics of the mean timing error associated with the receiver's time compared to the actual time. The results, as shown in Table 5.4, are consistent with available data [19].

Table 5.4: Standard GPS simulated Performance in various environments

Clear Sky	**Solution**	**Time Offset**
	Newton-Raphson's	2.130e-08 s
	Extended KF	6.495e-09 s
	Unscented KF	5.481e-09 s
Urban	**Solution**	**Time Offset**
	Newton-Raphson's	5.341e-08 s
	Extended KF	8.217e-09 s
	Unscented KF	5.828e-09 s
Window	**Solution**	**Time Offset**
	Newton-Raphson's	1.128e-07 s
	Extended KF	1.042e-08 s
	Unscented KF	8.112e-09 s
Indoor	**Solution**	**Time Offset**
	Newton-Raphson's	2.074e-07 s
	Extended KF	1.211e-08 s
	Unscented KF	8.365e-09 s

From the table above, both the Kalman Filter solutions produced the best results in accuracy of the time synchronization over the Newton-Raphson's when using a standard GPS receiver. The accuracy of the Kalman Filter solutions can time synchronize with the accuracy on the order of nanoseconds while the Newton-Raphson's solution can only time synchronize in the accuracy range of tens to hundreds of nanoseconds.

After introducing additional collaborating users to the new solutions we observed the results of the mean timing errors for the Newton-Raphson-based Collaborative Solution as shown in Fig. 5.17. This shows the improvements in the timing errors for nearly all scenarios with the exception of a few where there was minor degradation, much like the position error. The timing errors follow the same trend as the position errors for the Newton-Raphson-based Collaborative solution. The significance of these results is seen in the improvements made by an indoor user collaborating with clear sky users, improving from hundreds of nanoseconds to tens of nanoseconds.

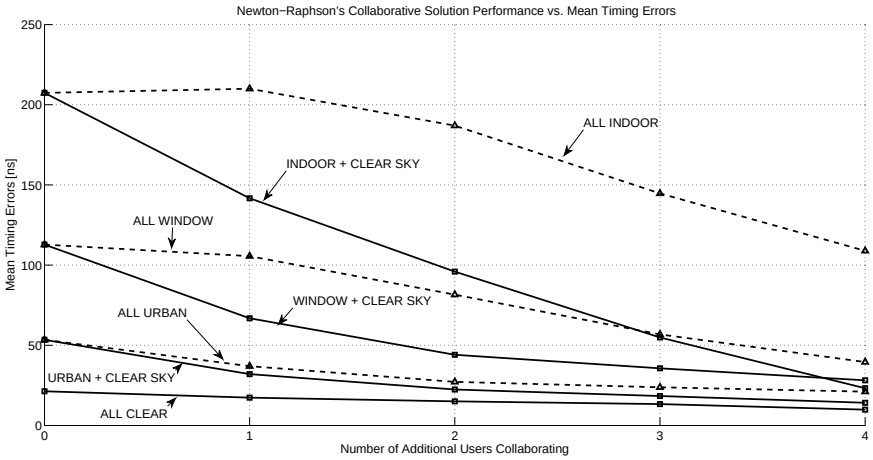


Figure 5.17: The mean timing error of users using the Newton-Raphson-based Collaborative solution

Both Kalman Filter solutions do not provide any improvements to the timing errors when collaboration is introduced as shown in Fig. 5.18, where we compare the mean timing errors of an indoor user when collaborating with a number of clear sky users. This figure shows no convincing trend in the Kalman Filter solutions and is contributed to the fact that the Kalman Filter solutions are unable to achieve a higher accuracy than the solution when using a standard GPS receiver.

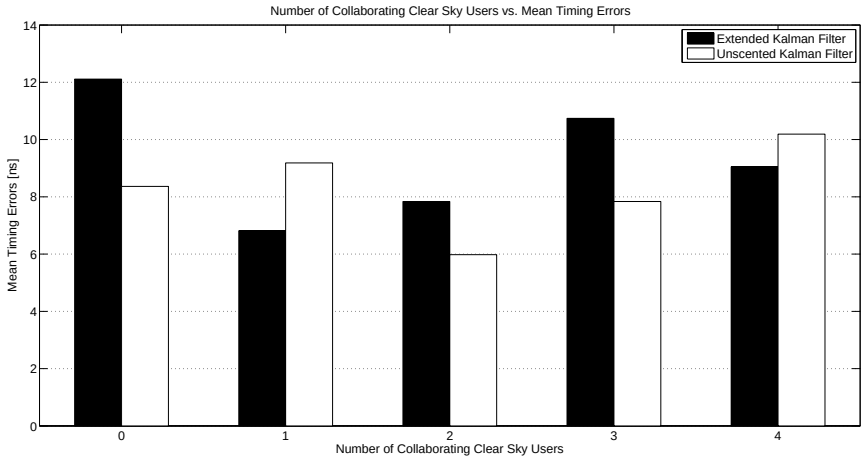


Figure 5.18: The mean timing error of an indoor user vs. the number of collaborating clear sky users

Chapter 6

Conclusion

After a thorough examination, our simulated results showed significant improvements in availability for users in all scenarios and in position error when enough users are available for collaboration. Overall, the Collaborative EKF solution performed the best in regards to both the position error and availability. Our new solutions showed significant improvements in the position estimate, allowing indoor users to meet the E911 requirements, as seen in the examination of the 95% CEP. After examining the number of satellites and the position error, a correlation was found that helped explain the initial minor degradation that occurred in select scenarios. The main source of improvement in the position estimate can be contributed to collaboration between users with at least four satellites available for each user. Most importantly though, the improvements in availability in all of those scenarios outweigh the initial minor degradation in position error that occurred allowing for the minimum of a coarse position estimate. The cost would require minimal adjustments to current GPS devices in order to share information. By sending a low power transmitted signal in a similar format as the pseudorange with ephemeris data attached to each signal one could keep the system the very similar. Our new solution shows great promise as a new means of geolocation through the use of the pre-existing GPS along with collaboration, improving both position error and availability.

Appendix A

Positioning Accuracy using the 95% CEP Figures & Tables

A.1 Collaborative Extended Kalman Filter Solution

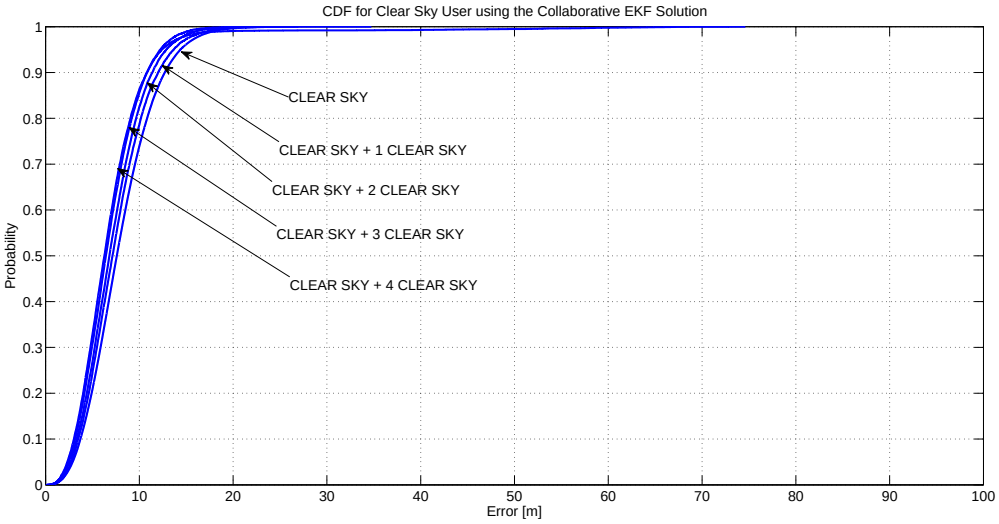


Figure A.1: CDF of Position Error for Clear Sky user using the Collaborative EKF Solution

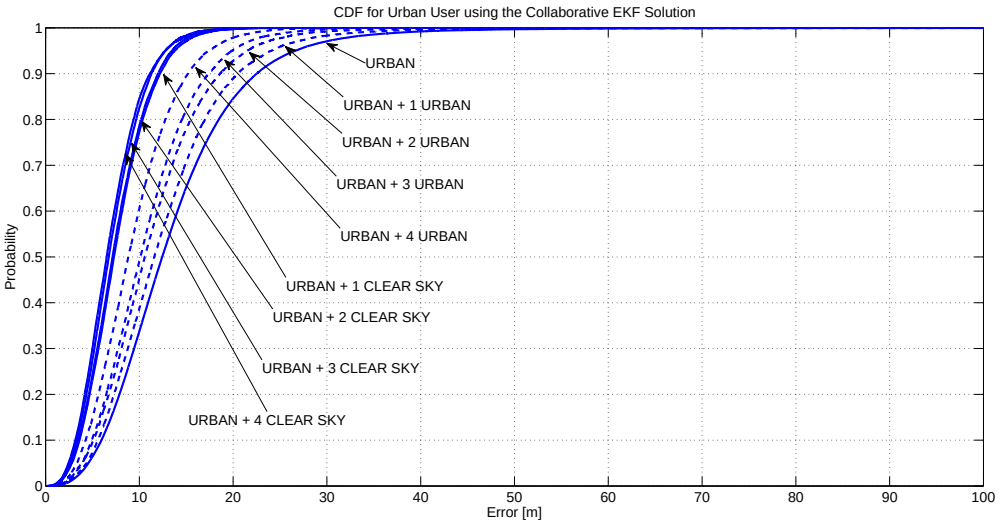


Figure A.2: CDF of Position Error for Urban User using the Collaborative EKF Solution

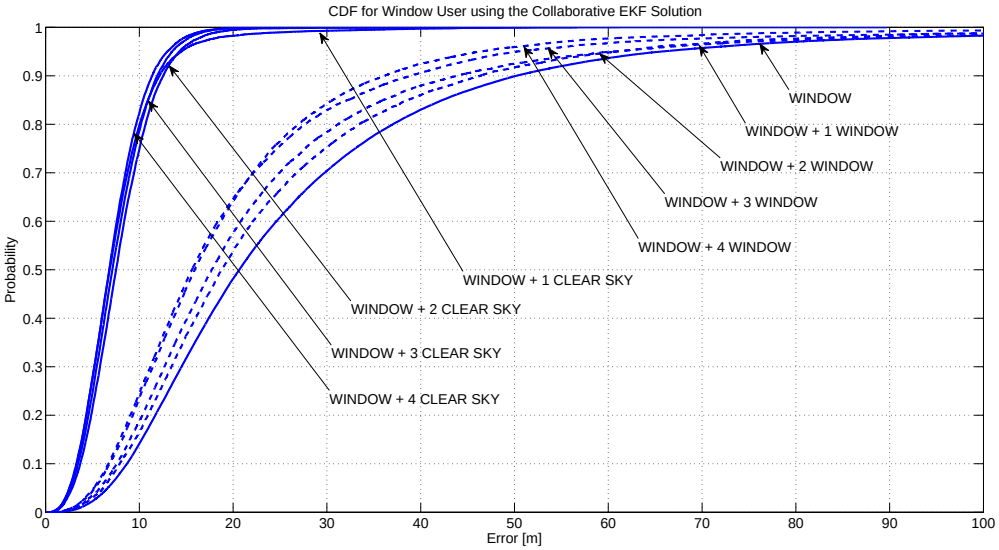


Figure A.3: CDF of Position Error for Window User using the Collaborative EKF Solution

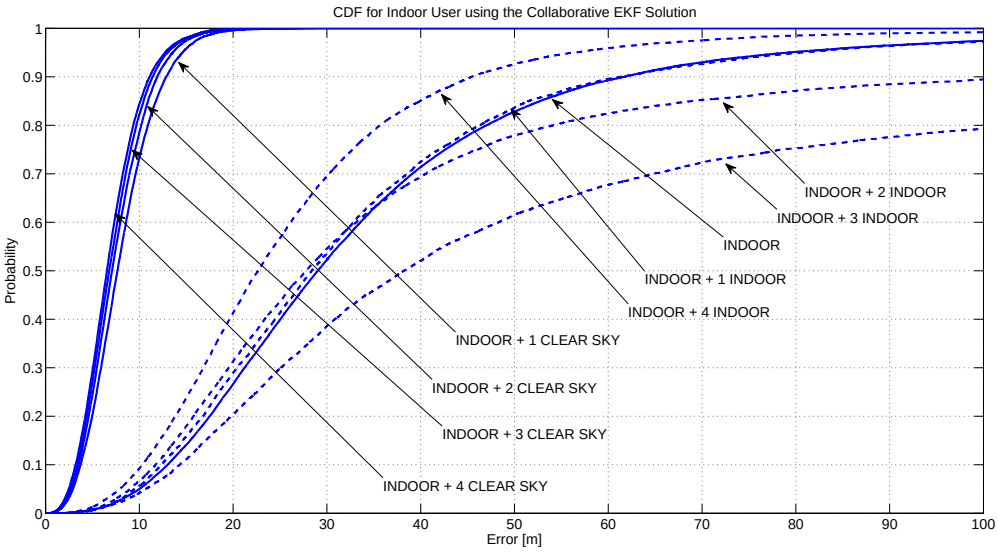


Figure A.4: CDF of Position Error for Indoor User using the Collaborative EKF Solution

A.2 Collaborative Unscented Kalman Filter Solution

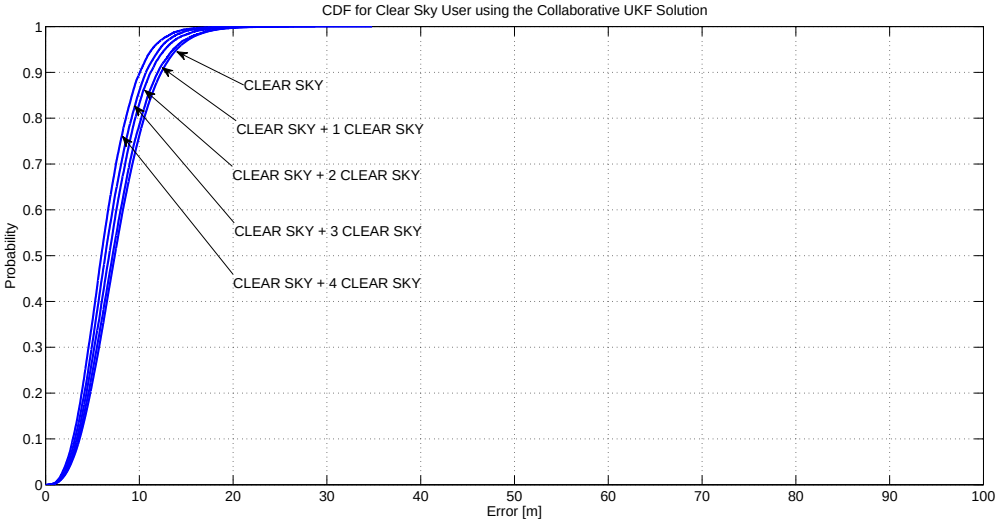


Figure A.5: CDF of Position Error for Clear Sky user using the Collaborative UKF Solution

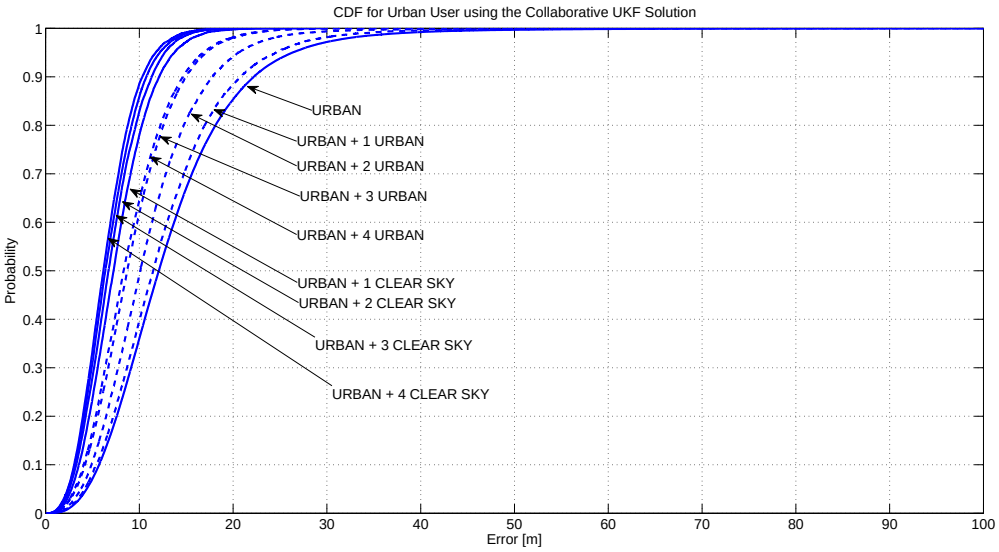


Figure A.6: CDF of Position Error for Urban User using the Collaborative UKF Solution

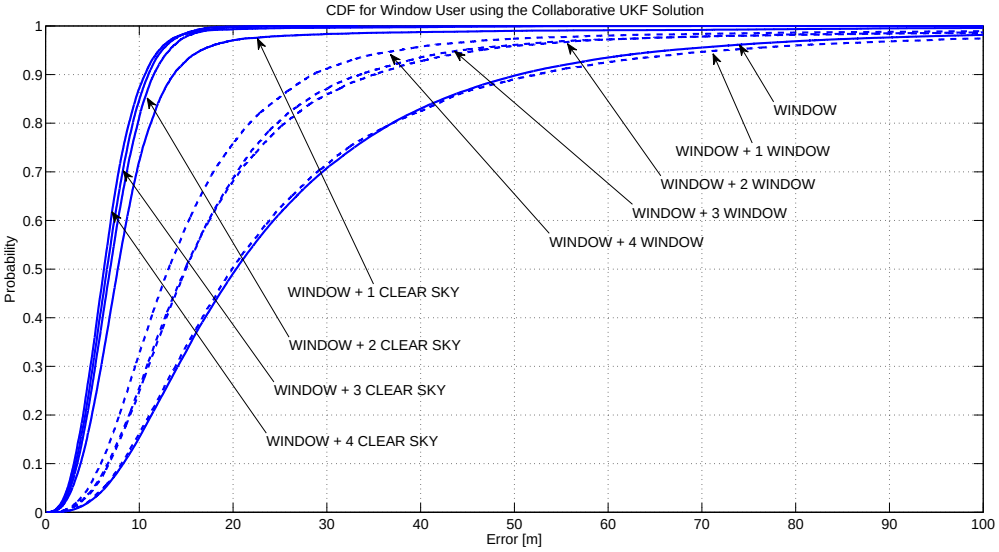


Figure A.7: CDF of Position Error for Window User using the Collaborative UKF Solution

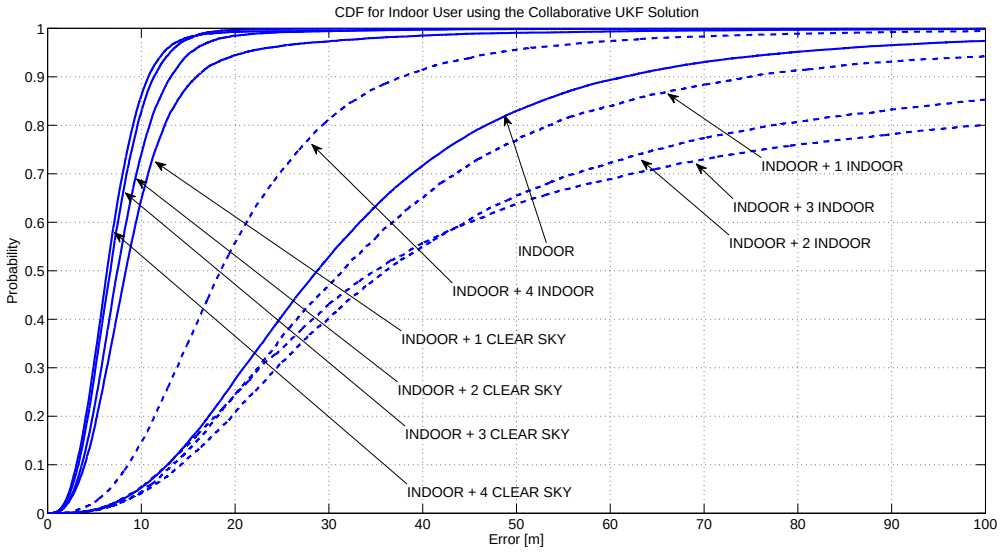


Figure A.8: CDF of Position Error for Indoor User using the Collaborative UKF Solution

A.3 95% CEP Improvements

Table A.1: 95% CEP improvements for Clear User with the new Collaborative GPS solutions

Solution	**Environment**	**Collaboration**	95% **CEP**
Newton-Raphson	Clear Sky	——	24 m
	Clear Sky	1 Clear Sky	21 m
	Clear Sky	2 Clear Sky	20 m
	Clear Sky	3 Clear Sky	17 m
	Clear Sky	4 Clear Sky	15 m
Extended KF	Clear Sky	——	14 m
	Clear Sky	1 Clear Sky	13 m
	Clear Sky	2 Clear Sky	13 m
	Clear Sky	3 Clear Sky	12 m
	Clear Sky	4 Clear Sky	12 m
Unscented KF	Clear Sky	——	14 m
	Clear Sky	1 Clear Sky	13 m
	Clear Sky	2 Clear Sky	13 m
	Clear Sky	3 Clear Sky	12 m
	Clear Sky	4 Clear Sky	11 m

Table A.2: 95% CEP improvements for Urban User with the new Collaborative GPS solutions

Solution	**Environment**	**Collaboration**	95% **CEP**
Newton-Raphson	Urban	——	60 m
	Urban	1 Urban	45 m
	Urban	2 Urban	36 m
	Urban	3 Urban	33 m
	Urban	4 Urban	23 m
	Urban	1 Clear Sky	21 m
	Urban	2 Clear Sky	20 m
	Urban	3 Clear Sky	18 m
	Urban	4 Clear Sky	17 m
Extended KF	Urban	——	27 m
	Urban	1 Urban	24 m
	Urban	2 Urban	21 m
	Urban	3 Urban	19 m
	Urban	4 Urban	17 m
	Urban	1 Clear Sky	14 m
	Urban	2 Clear Sky	14 m
	Urban	3 Clear Sky	13 m
	Urban	4 Clear Sky	12 m
Unscented KF	Urban	——	26 m
	Urban	1 Urban	25 m
	Urban	2 Urban	21 m
	Urban	3 Urban	19 m
	Urban	4 Urban	17 m
	Urban	1 Clear Sky	14 m
	Urban	2 Clear Sky	13 m
	Urban	3 Clear Sky	12 m
	Urban	4 Clear Sky	11 m

Table A.3: 95% CEP improvements for Window User with the new Collaborative GPS solutions

Solution	**Environment**	**Collaboration**	95% **CEP**
Newton-Raphson	Window	——	214 m
	Window	1 Window	218 m
	Window	2 Window	202 m
	Window	3 Window	107 m
	Window	4 Window	63 m
	Window	1 Clear Sky	23 m
	Window	2 Clear Sky	20 m
	Window	3 Clear Sky	19 m
	Window	4 Clear Sky	17 m
Extended KF	Window	——	66 m
	Window	1 Window	61 m
	Window	2 Window	60 m
	Window	3 Window	46 m
	Window	4 Window	50 m
	Window	1 Clear Sky	14 m
	Window	2 Clear Sky	13 m
	Window	3 Clear Sky	13 m
	Window	4 Clear Sky	14 m
Unscented KF	Window	——	66 m
	Window	1 Window	72 m
	Window	2 Window	44 m
	Window	3 Window	46 m
	Window	4 Window	38 m
	Window	1 Clear Sky	17 m
	Window	2 Clear Sky	13 m
	Window	3 Clear Sky	13 m
	Window	4 Clear Sky	12 m

Table A.4: 95% CEP improvements for Indoor User with the new Collaborative GPS solutions

Solution	**Environment**	**Collaboration**	95% **CEP**
Newton-Raphson	Indoor	——	338 m
	Indoor	1 Indoor	395 m
	Indoor	2 Indoor	398 m
	Indoor	3 Indoor	344 m
	Indoor	4 Indoor	240 m
	Indoor	1 Clear Sky	24 m
	Indoor	2 Clear Sky	20 m
	Indoor	3 Clear Sky	19 m
	Indoor	4 Clear Sky	18 m
Extended KF	Indoor	——	79 m
	Indoor	1 Indoor	80 m
	Indoor	2 Indoor	377 m
	Indoor	3 Indoor	242 m
	Indoor	4 Indoor	56 m
	Indoor	1 Clear Sky	15 m
	Indoor	2 Clear Sky	13 m
	Indoor	3 Clear Sky	13 m
	Indoor	4 Clear Sky	12 m
Unscented KF	Indoor	——	79 m
	Indoor	1 Indoor	106 m
	Indoor	2 Indoor	226 m
	Indoor	3 Indoor	312 m
	Indoor	4 Indoor	48 m
	Indoor	1 Clear Sky	21 m
	Indoor	2 Clear Sky	15 m
	Indoor	3 Clear Sky	13 m
	Indoor	4 Clear Sky	12 m

Bibliography

[1] S. Wang, J. Min, and B. K. Yi. Location Based Services for Mobiles: Technologies and Standards. In *IEEE International Conference on Communication (ICC)*, 2008.

[2] N. Patwari, J. N. Ash, S. Kyperountas, A.O. Hero, III, R. L. Moses, and N. S. Correal. Locating the nodes: cooperative localization in wireless sensor networks, 2005.

[3] M. L. Fowler and X. Hu. Signal Models for TDOA/FDOA Estimation. *IEEE Transactions on Aerospace and Electronic Systems*, 44(4), October 2008.

[4] J. H. Lee and R. M. Buehrer. Location estimation using differential RSS with spatially correlated shadowing. In *IEEE Global Telecommunications Conference*, page 46134618, December 2009.

[5] S. L. Collier and D. K. Wilson. Performance bounds for passive sensor arrays operating in a turbulent medium: Spherical-wave analysis. *The Journal of the Acoustical Society of America*, 116(2):987–1001, 2004.

[6] H. Liu, H. Darabi, P. Banerjee, and J. Liu. Survey of Wireless Indoor Positioning Techniques and Systems. *IEEE Transactions on Systems, Man, and Cybernetics-Part C: Applications and Reviews*, 37(6):1067–1080, November 2007.

[7] L. Zhao, G. Yao, and J.W. Mark. Mobile positioning based on relaying capability of mobile stations in hybrid wireless networks. *IEE Proceedings Communications*, 153(5):762–770, October 2006.

[8] R. Zekavat and R. M. Buehrer. *Handbook of Position Location: Theory, Practice and Advances (IEEE Series on Digital & Mobile Communication)*. Wiley-IEEE Press, 2011.

[9] P. M. Kintner Jr. *Global Positioning System Theory and Design*. Cornell University, 1999.

[10] R. M. Buehrer. Position Location: Fundamentals, Practice and Recent Research. Presentation, 2011.

[11] T. Jia. *Collaborative Position Location for Wireless Networks in Harsh Environments*. PhD thesis, Virginia Polytechnic Institute and State University, 2010.

[12] P. Luo and M. G. Petovello. Collaborative Acquisition of Weak GPS Signals. In *Position Location and Navigation Symposium (PLANS), 2010 IEEE/ION*, pages 787–795. 2010.

[13] Z. Biskaduros and R. M. Buehrer. Collaborative Localization Enhancement to the Global Positioning System. In *9th Workshop on Positioning, Navigation, and Communication*, March 2012.

[14] K. Borre. GPS EASY Suite II: A MATLAB Companion. In *Inside GNSS*, volume 4, pages pp. 48–52. March/April 2009.

[15] Hewlett-Packard Company. Gps and precision timing applications, May 1996. Application Note 1272.

[16] A. Soloviev and T. J. Dickman. Deeply Integrated GPS for Indoor Navigation. In *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, September 2010.

[17] C. Tseng D. Jwo, M. Chen and T. Cho. Adaptive and Nonlinear Kalman Filtering for GPS Navigation Processing. In *Kalman Filter: Recent Advances and Applications*, chapter 14, pages 321–346. InTech, 2009.

[18] M. Lei and C. Han. Sequential nonlinear tracking using ukf and raw range-rate measurements. *Aerospace and Electronic Systems, IEEE Transactions on*, 43(1):239–250, 2007.

[19] P. H. Dan and B. M. Penrod. The role of gps in precise time an frequency dissemination. *GPS World*, July/August 1990.

[20] T. Pattison and S.I. Chou. Sensitivity Analysis of Dual-Satellite Geolocation. *IEEE Transactions on Aerospace and Electronic Systems*, 36(1), January 2000.

[21] B. Thompson. Characterizing and Improving the Non-Collaborative and Collaborative Localization Problems. Master's thesis, Virginia Polytechnic Institute and State University, 2011.

[22] Swedish Defence Research Agency (FOI). *Collaborative GPS/INS Navigation in Urban Environment*, January 2004.

[23] R. M. Vaghefi, M. R. Gholami, and E. G. Ström. Bearing-only target localization with uncertainties in observer position. In *Proc. IEEE PIMRC*, pages 238–242, September 2010.

[24] F. Sun Y. Hao, Z. Xiong and X. Wang. Comparison of Unscented Kalman Filters. In *International Conference on Mechatronics and Automation*, August 2007.

[25] G. Welch and G. Bishop. An Introduction to the Kalman Filter, 2001.

[26] A. Tiwari A. Chou W. Mann A. Sahai J. Stone P. Enge, R. Fan and B. Van Roy. Improving GPS Coverage and Continuity: Indoors and Downtown.

[27] L. Galleani and P. Tavella. Time and the Kalman Filter. In *IEEE CONTROL SYSTEMS MAGAZINE*. April 2010.

[28] Federal Communications Commission. Fcc wireless 911 requirements, January 2011.