Major Qualifying Project

# Adversary UAV Localization With Software Defined Radio

Ian Gelman, Abdul Hassan,
John Loftus, Brian Mahan

2019

WPI

# ADVERSARY UAV LOCALIZATION WITH SOFTWARE DEFINED RADIO

A Major Qualifying Project submitted to the Faculty of Worcester Polytechnic Institute in partial fulfilment of the requirements for the Degree of

Bachelor of Science in Electrical and Computer Engineering

by

Ian Gelman
Abdul Hassan
John Loftus
Brian Mahan

May 2019

Ian Gelman, Abdul Hassan, John Loftus, Brian Mahan:
*Adversary UAV Localization With Software Defined Radio* (2019)


The work in this MQP Report was made in the:


Center for Wireless Information Network Studies
Department of Electrical and Computer Engineering
Faculty of Electrical and Computer Engineering
Worcester Polytechnic Institute

Supervisor: Dr. Kaveh Pahlavan

# ABSTRACT

Unmanned Aerial Vehicles (UAVs) continue to pose an immediate threat to personal privacy and national security. In an effort to detect and mitigate the threat of unwanted drones, our team designed a RSS-Based 3D localization system utilizing software-defined radio. Localization occurred in an urban outdoor and line of sight environment. This report focused on localization of hobbyist drones by detecting and quantifying the received signal strength (RSS) of the video stream emitted by the drone to the remote controller. The adaptive filter algorithm, recursive least squares (RLS) was used to numerically estimate the drone's 3D position. The precision and accuracy of our system was quantified by distance measurement error (DME) and in comparison with Cramer-Rao lower bound (CRLB).

# ACKNOWLEDGEMENTS

This three-term project has four team members: Ian Gelman, Abdul Hassan, John Loftus, and Brian Mahan. We would like to express our sincere gratitude to our advisor, Professor Kaveh Pahlavan for his mentorship, and would like to congratulate Dr. Pahlavan on the release of his first solo-written textbook, Indoor Geolocation Science and Technology - At the Emergence of Smart World and IoT. We would also like to thank Dr. Pahlavan for his offering of ECE 5307 - Wireless Access and Localization. We would finally like to thank Professor Alexander Wyglinski for his generosity in allowing us to use his Ettus Research radio equipment. We would not have been able to complete the project without their assistance.

# CONTENTS

# LIST OF FIGURES

# LIST OF TABLES

# ACRONYMS

| ADALM-PLUTO | Analog Devices Active Learning Module - PLUTO |
|---|---|
| API | Application Programming Interface |
| ASIC | Applicant Specific Integrated Circuit |
| BLIT | Bit Block Transfer |
| CRLB | Cramer-Rao Lower Bound |
| CWINS | Center for Wireless Information Network Studies |
| DDC | Digital Downconverter |
| DME | Distance Measurement Error |
| DJI | Da-Jiang Innovations |
| DPD | Digital Predistortion Filter |
| DSP | Digital Signal Processor |
| DUC | Digital Upconverter |
| FAA | Federal Aviation Administration |
| FCC | Federal Communications Commission |
| FFT | Fast Fourier Transform |
| FPGA | Field Programmable Gate Array |
| GUI | Graphical User Interface |
| IC | Integrated Circuit |
| LAN | Local Area Network |
| LNPL | Log-Normal Path Loss |
| LO | Local Oscillator |
| MLE | Maximum Likelihood Estimation |
| PL | Programmable Logic |
| RF | Radio Frequency |
| RFFE | Radio Frequency Front End |
| RFNoC | Radio Frequency Network on a Chip |
| RLS | Recursive Least Squares |
| RSS | Received Signal Strength |
| SD | Standard Deviation |
| SDR | Software Defined Radio |
| SNR | Signal to Noise Ratio |
| SoC | System on a Chip |
| UAS | Unmanned Aircraft System |
| UAV | Unmanned Aerial Vehicle |
| UDP | Undirected Path |
| UDP | User Datagram Packet |
| UHD | Universal Software Radio Peripheral Hardware Driver |
| US | United States |
| USB-OTG | USB On The Go |
| USRP | Universal Software Radio Peripheral |
| VHDL | Very High Speed Integrated Circuit Hardware Description Language |
| WC | Weighted Centroid |
| WPI | Worcester Polytechnic Institute |

# 1 INTRODUCTION

Unmanned Aerial Vehicles (UAVs) continue to pose an immediate threat to personal and national security. On Aug 10, 2018 there was a failed drone attack on Venezuelan president Nicolas Maduro. Two DJI Matrice hobbyist UAVs laden with explosives detonated but failed to harm President Maduro[7]. The DJI Matrice 600 is commonly used for professional photography. The DJI Matrice 600 is capable of carrying 13.2 pounds of additional payload [6]. The threat and danger of drones creates a need for real-time detection and localization of UAVs within restricted air-space. In the past 5 years, hobbyist drones have had significant growth in the consumer market. From 2015-2017, the FAA distributed 788,570 Unmanned Aircraft System (UAS) licences, but predicted over 1.1 million UAS-type aircraft to be flying in that same year [6]. This implies that hobbyist drones flying in restricted air-spaces across the United States (US), and possibly the globe. Individuals have utilized UAVs to carry payloads including explosives and firearms. Our team focused on localization of DJI Phantom 3 Professional hobbyist drone. The Phantom 3 has a 4K camera, a flight time of 23 minutes and max speed of 16m/s. In this report, we designed a localization system utilizing Software Defined Radio (SDR). The adversarial drone is detected using the transmitted video stream from the drone to the controller. We assume we know the type of drone, and the drone's video stream protocol and transmit frequency.

## 1.1 PROJECT DESCRIPTION

This project focused on using software defined radios to localize off a UAV's emitted video stream. This proprietary waveform is measured and its re-


**Figure 1.1:** Drone Attack on Venezuelan President Nicolas Maduro. The explosions occurred mid-air, leaving scorch marks and exterior wall damage on neighboring buildings.

ceived signal strength (RSS) is used to calculate the position of the drone. The drone used in the project is the DJI Phantom 3 Pro, which camera feed transmits between 2.4 to 2.48GHz frequency range depending on the channel used for camera streaming. For radios, we utilized five Ettus Universal Software Radio Peripherals (USRP)s, which were designed to measure frequencies in the 2.4 GHz as well as the 5.9 GHz range. All five ETTUS SDRs are connected by Ethernet to a Local Area Network (LAN). All devices on the LAN are interfaced with a 16-port 1Gbps unmanaged ethernet switch. All terminals on the LAN are statically addressed with Internet Protocol Version 4 (IPv4) protocol. The path loss model used in the localization algorithm is derived by finding the least square fit of data collected of received strength signals at different distances. The pathloss model is evaluated by Cramer Rao Lower Bound (CRLB). This report outlines our real-time system design while comparing our real-time results CRLB and distance measurement error (DME). The system's performance was quantified by the reduction in DME by averaging RSS samples prior to calculating RLS to derive location.

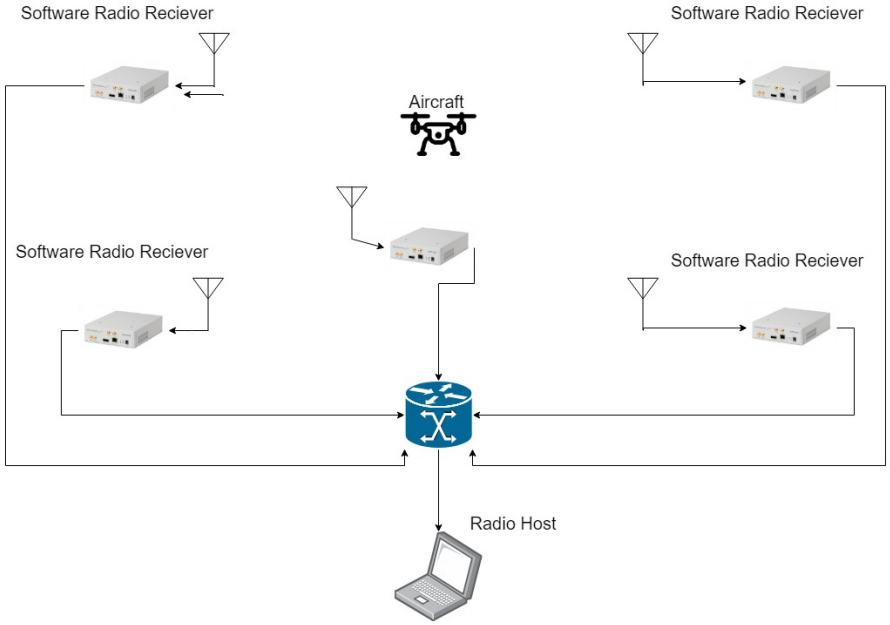


**Figure 1.2:** High Level Deployment Diagram. This figure shows the physical deployment of the localization system. Five software radio receivers are connected on a flat switched network, with the processing being offloaded to one or more host PCs.

## 1.2 REPORT OUTLINE

The layout of this report is created in a manner in which the reader is able to learn about Software-Defined Radio, Received Signal Strength (RSS) acquisition and modeling, real-time localization using RSS, and verification using CRLB. In Chapter 2, Theoretical Background, the theory behind all of the path loss models, localization algorithms, and verification testing using CRLB in 2D and 3D are discussed. Chapter 3, System Architecture, describes the hardware and software implementations. Chapter 4, Results and Discussion, discusses the RSS ranging results, the proposed path-loss model, empirical performance analysis using CRLB in 2D and 3D, and empirical Distance Measurement Error analyses. Finally, Chapter 5, Conclusion and Future Recommendations, concludes our work and gives directions on what can be done for future additions to this project.

# 2 THEORETICAL BACKGROUND

This chapter will provide the theoretical background necessary for understanding details of this project. Path loss principles for wireless propagation will be discussed, as well as the multipath phenomena which occurs in indoor and outdoor areas. Theoretical background for distance measurement error and the estimation algorithm Recursive Least Square (RLS) will be discussed in detail. System modeling using distance measurement error will be discussed via frequency histograms, and the most probable sources of error within the system are examined. The theoretical background behind RLS and its derivation are explored. Decision making for the best localization algorithm to use for this system and our reasoning behind using this specific algorithm is analyzed in this section. System verification using Cramer Rao Lower Bound (CRLB) for ranging in 2D and 3D will also be examined. The principles of derivation of CRLB for RSS positioning are mentioned in great detail in this section of the report. Finally, the operating principles of a Software Defined Radio (SDR) system will also be explained. The hardware and software, as well as the operation mechanics behind an SDR, will be discussed in detail.

## 2.1 PATH-LOSS MODEL

A RSS Path Loss Model is a linear regression model that illustrates the relation of the RSS with the logarithmic distance between the transmitter and receiver based off of Equation 2.1

$$P_r = P_0 - 10\alpha log_{10}\left(\frac{d}{d_1}\right) + \chi \tag{2.1}$$

The Power Received ($P_r$) is equivalent to Power Loss in the First Meter ($P_0$) deducted by the ten times the Distance Power Gradient ($\alpha$) multiplied by log distance from transmitted divided by one meter. $P_r$ is expressed in decibels and distance is in meters. Alpha ($\alpha$) is the derived slope of a linear regression model. Alpha ($\alpha$) represents the decay of the Radio Frequency (RF) signal strength in dB over distance. $\chi$ represents the standard deviation of shadow fading. Shadow fading is the variation in attenuation from a transmitted wireless signal. Power $x[mW]$ can be converted to dBm by Equation 2.2.

$$x[mW] = 10log_{10}\left(\frac{x}{1mW}\right)[dBm] \tag{2.2}$$

In Equation 2.1. $\chi$ is a 1-dimensional Gaussian random variable representing Shadow Fading. Shadow Fading is the main cause of fluctuation of received signal strength at certain locations. As an illustration of this phenomenon, multipath propagation for IEEE 802.11 is shown in Figure 2.1.

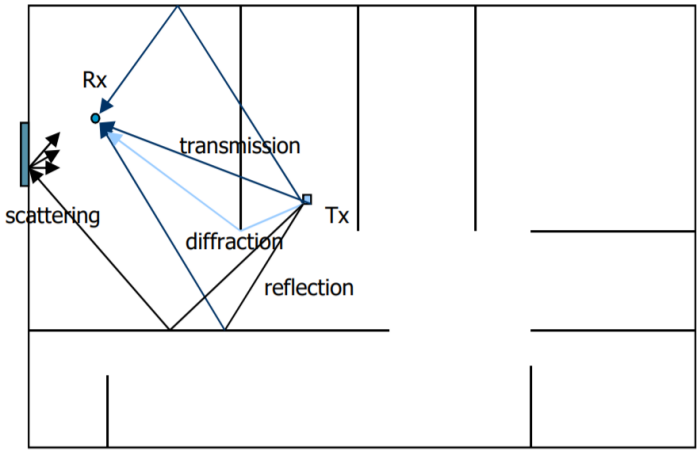


**Figure 2.1:** Indoor Multipath Fading of 802.11. The same RF propagation arrives at point Rx with different times, phases and powers. These variations can be due to the signal being affected by walls and other objects. Scattering, diffraction and reflection are illustrated.[Pahlavan, 40]

The same RF propagation arrives at *Rx* at different times, phases and powers. These variations can be due to the signal being affected by walls and other objects. In Figure 2.1, The same RF propagation arrives by scattering, diffraction and reflection.

## 2.2 INTRODUCTION TO LOCALIZATION ALGORITHMS AND THE RLS LOCALIZATION ALGORITHM

After empirically deriving parameters for the log-normal path loss channel model (LNPL), an algorithm must be employed in order to convert the calculated distances from each receiver into latitude, longitude, and height. There are many methods to choose from, therefore further literature must be consulted in order to determine the method that we would like to implement. The following is a synopsis of joint graduate/undergraduate research undertaken by Worcester Polytechnic Institute students entitled, "RSS-Based 3-D Drone Localization and Performance Evaluation." The research identifies popular RSS-based algorithms for 3-D drone localization such as Weighted Centroid, Maximum Likelihood Estimation (MLE), and Recursive Least Squares (RLS). In the paper, each methods computational complexity was evaluated, as well as its performance. The Cramer-Rao Lower Bound (CRLB) for an unbiased estimator was employed to compute the error bound of each algorithm. Three different Localization Algorithms were explored: Weighted Centroid, Maximum Likelihood Estimation (MLE), and Recursive Least Squares (RLS). The performance and efficiency of these popular localization algorithms were evaluated, through the algorithm's accuracy based off of Cramer Rao Lower Bound, as well as the time complexity of each algorithm. Through comparison with the CRLB, and total computation times for each algorithm, a table summarizing Tian's findings is shown in Table Table 2.1. For our project, we decided to utilize the Recursive Least Squares, as its accuracy was seen to be sufficient while also not being overly time-complex. This algorithm employs Gauss-Newton method to numeri-

| Algorithm | Accuracy | Time Complexity |
|---|---|---|
| Maximum Likelihood Estimation | Best | High |
| Recursive Least Squares | Better | Medium |
| Weighted Centroid | Good | Low |

**Table 2.1:** Feature Matrix for RSS-Based Localization Algorithms. Complexity and accuracy of each algorithm is shown.

cally minimize the expression for $E$, which is represented as the distance measurement error.

$$E_1(x,y) = \sum_{i=1}^{n} f_{1i}^2 = \sum_{i=1}^{n} \left( \sqrt{(x_i - x)^2 + (y_i - y)^2} - d_i \right)^2 \tag{2.3}$$

From the log-normal Path Loss Model, distances are able to be derived based off of the received signal strength. When given multiple reference points to receive the signal from the drone, multiple distances at different locations can be derived and the location of the drone can be calculated using localization algorithms. In the 2D Recursive Least Squares Algorithm, the function reflecting ranging error from an adversarial drone and a software radio receiver is defined as:

$$f_i(x,y) = (x_i - x)^2 + (y_i - x)^2 - d_i^2 \tag{2.4}$$

Where $(x,y)$ is the location of the adversarial drone to be localized, $(x_i, y_i)$ is the location of the $i$-th SDR and $d_i$ is the calculated distance from the SDR and the drone. When combining the functions into the quadratic vector function $F$, we obtain:

$$F = \begin{bmatrix} f_1(x,y) & f_2(x,y) & f_3(x,y) & \dots & f_N(x,y) \end{bmatrix}^T \tag{2.5}$$

We are able to convert the expression given in Equation 2.5 into a Jacobian Matrix $J$:

$$J = \begin{bmatrix} \frac{\partial f_1(x,y)}{\partial x} & \frac{\partial f_1(x,y)}{\partial y} \\ \frac{\partial f_2(x,y)}{\partial x} & \frac{\partial f_2(x,y)}{\partial y} \\ \vdots & \vdots \\ \frac{\partial f_i(x,y)}{\partial x} & \frac{\partial f_i(x,y)}{\partial y} \end{bmatrix} \tag{2.6}$$

With this Jacobian Matrix, we are able to then estimate the location. If we start with a location:

$$l(n) = \begin{bmatrix} x(n) & y(n) \end{bmatrix} \tag{2.7}$$

We can then update this location through the equation:

$$l(n+1) = l(n) + E_n \tag{2.8}$$

Where:

$$E_n = -(J^T J)^{-1} J^T F \tag{2.9}$$

## 2.3 CRLB OF RSS-BASED POSITIONING FOR PERFORMANCE ANALYSIS

In localization systems, the performance of the ranging and localization can be compared with the Cramer Rao Lower Bound. This bound is compared

with the standard deviation of a localization system, which comes from the spread of error against the estimated distance or location. A lower variance indicates a lower chance of high error from the location estimate. From [], CRLB gives the smallest variance of a probability distribution function $f(O|\alpha)$ such that:

$$Var[\hat{a}(O) - \alpha] \geq CRLB \tag{2.10}$$

The CRLB is given by the calculating the inverse of the Fisher Information Matrix,

$$F = E\left[\frac{\partial lnf(O|\alpha)}{\partial \alpha}\right]^2 = -E\left[\frac{\partial^2 lnf(O|\alpha)}{\partial \alpha^2}\right] \tag{2.11}$$

Making the overall CRLB equation:

$$CRLB = Var[\hat{a}(O) - \alpha] \geq F^{-1} \tag{2.12}$$

When looking at observations that are corrupted by zero mean Gaussian noise, the observations $O$ can be seen as:

$$O = \alpha + \eta \tag{2.13}$$

Where $\eta$ is the Gaussian noise with variance $\sigma^2$. The conditional probability density function for $O$ is given by the equation:

$$f(O|\alpha) = \frac{1}{\sqrt{2\pi\sigma}} exp\left(-\frac{(O-\alpha)^2}{2\sigma^2}\right) \tag{2.14}$$

When put through the Fisher matrix, the function simplifies to $\frac{1}{\sigma}$. Therefore:

$$CRLB \geq F^{-1} = \sigma^2 \tag{2.15}$$

### 2.3.1 Theoretical Distance Measurement Error

The difference between the estimated and the actual value of the error is the distance measurement error (DME) and it is given by:

$$DME = \epsilon = \hat{r} - r \tag{2.16}$$

The known radius from the transmitting device $(Tx)$ is $r$. The measured distance is $\hat{r}$. $r$ is derived from the RSS of the receiver by a path-loss model. $r$ is equal to distance $d$ from Equation 2.1. $r$ is the green ring and the $R$ is the black line in Figure 2.2. In modeling the distance measurement error, we differentiate the small errors caused by multipath from the large errors produced by the occurrence of undetected direct path (UDP) conditions. Figure 2.3 is an example of a DME histogram. We refer to the small distance errors caused by multipath as the multipath distance measurement error. The histogram's peak is the center of DME. The histogram's skewness is caused by the UDP condition as UDP distance measurement error. The multipath error caused by neighboring paths always exists and the UDP error exists only when the UDP condition occurs. DME is commonly analyzed by resistive measures, which include mean and quartile statistics. Means and Standard deviations of DMEs are not resistive measures due to the UDP conditions.

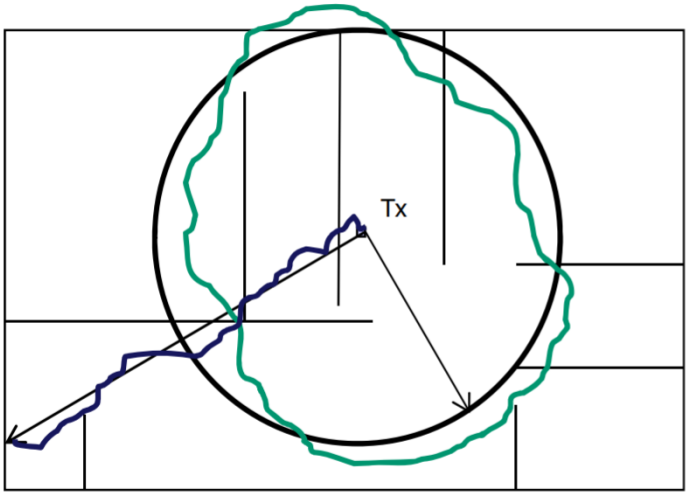


**Figure 2.2:** Distance Measurement Error for $Tx$. Multipath propagation error for a $Rx$ node processing in a circle around point $Tx$. The known radius from the transmitting device $(Tx)$ is $r$. The measured distance is $\hat{r}$. $\hat{r}$ is derived from the RSS of the receiver by a path-loss model. $\hat{r}$ is equal to distance $d$ from Equation 2.1. $\hat{r}$ is denoted by the green ring and $R$ is denoted by the black line.[Pahlavan, 46]

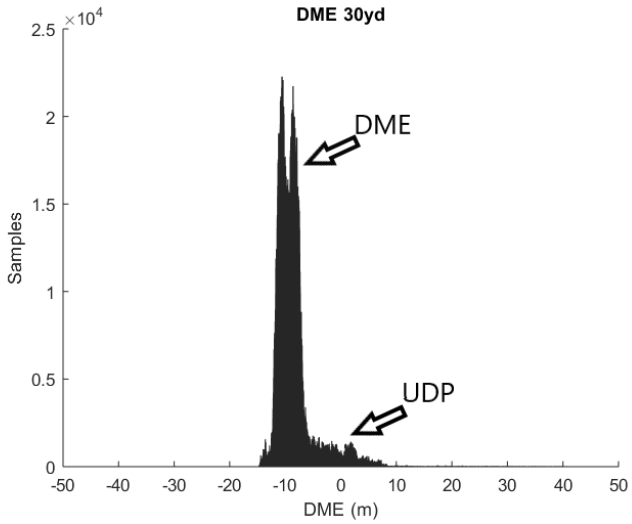


**Figure 2.3:** Frequency Histogram of Distance Measurement Errors for 30 Yard Measurements. Errors are relatively small and are mostly around -10m error due to multipath.

### 2.3.2 CRLB for Ranging

In our RSS localization systems, as mentioned, the signal strength of the drone is the observed power that is used to estimate $d$. In this case, our path loss model is our observation function from Equation 2.1. From this function, we are able to then convert it into our probability distribution function from []:

$$F(d) = \frac{1}{2\pi\sigma} exp\left(-\frac{(P_r - P_0 + 10\alpha log_{10}(d))^2}{2\sigma^2}\right) \tag{2.17}$$

Evaluating the Fisher Information Matrix with the PDF from Equation 2.17, we obtain this expression:

$$\frac{10^2\alpha^2}{(ln(10)^2\alpha^2 d^2)} \tag{2.18}$$

Taking the inverse of this expression yields the CRLB for 1-dimensional ranging.

$$CRLB \geq \frac{ln(10)^2}{100}\frac{\sigma^2}{\alpha^2}d^2 \tag{2.19}$$

To obtain the standard deviation of error for comparison $p$, we need to take the square root of this equation since the CRLB is the variance. This will give us the needed CRLB for ranging comparison:

$$CRLB \geq \frac{ln(10)^2}{100}\frac{\sigma^2}{\alpha^2}d^2 \tag{2.20}$$

### 2.3.3 CRLB for Positioning in 2D

In positioning across a 2D plane, the path loss model is similar to in ranging where:

$$P_r = P_0 - 10\alpha log_{10}(r_i) + \chi, i = 1, 2, \ldots, N \tag{2.21}$$

and the distance $d$ from the ranging is replaced with the distance in relation to multiple axes $r_i$.

$$r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2} \tag{2.22}$$

Where $(x, y)$ is the location of the device and $(x_i, y_i)$ is the location of the reference points for N given reference points. From the shadow fading, positioning algorithms experience fluctuations in received power, defined by [] where:

$$dP_i(x, y) = -\frac{10\alpha_i}{ln(10)}\left(\frac{x - x_i}{r_i^2}dx + \frac{y - y_i}{r_i^2}dy\right) \tag{2.23}$$

which leads to variations in ranging estimates $dr$. Then looking at these parameters in vector form, their relationship can be seen as:

$$dP = Hdr \tag{2.24}$$

and

$$dr = (H^T H)^{-1} H^T dP \tag{2.25}$$

where

$$dP = \begin{bmatrix} dP_1 \\ dP_2 \\ \vdots \\ dP_N \end{bmatrix} \tag{2.26}$$

$$dr = \begin{bmatrix} d_x \; d_y \end{bmatrix} \tag{2.27}$$

$$H = -\frac{10}{ln(10)} \begin{bmatrix} \sigma_1 \; \sigma_2 \; \dots \; \sigma_N \end{bmatrix} \begin{bmatrix} \frac{x-x_1}{r_1^2} & \frac{y-y_1}{r_1^2} \\ \frac{x-x_2}{r_2^2} & \frac{y-y_2}{r_2^2} \\ \vdots & \vdots \\ \frac{x-x_N}{r_N^2} & \frac{y-y_N}{r_N^2} \end{bmatrix} \tag{2.28}$$

From here, we are able to find the covariance of the location estimate $dr$ from Equation 2.29:

$$cov(dr) = \sigma^2 (H^T H)^{-1} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy}^2 \\ \sigma_{xy}^2 & \sigma_y^2 \end{bmatrix} \tag{2.29}$$

And then take the standard deviation of the location error $r$, off of the resulting matrix from Equation 2.29

$$\sigma_r = \sqrt{\sigma_x^2 + \sigma_y^2} \tag{2.30}$$

This standard deviation, similar to that of ranging, gives us the maximum accuracy our localization system can have in 2D at a certain $(x, y)$ point.

### 2.3.4 3D Expansion of RLS for RSS-Based Positioning

One challenge that we faced in our project was taking the 2D algorithms and converting them to 3D which was needed for our localization system analysis. To convert these into 3D, another dimension, accounting for height, had to be put into consideration for these algorithms. Similar to the 2D Recursive Least Square Algorithm, the 3D RLS algorithm follows the same steps, except an extra dimension is added. In the 3D RLS algorithm, the function reflecting ranging error from an adversarial UAV and a software radio receiver is defined in equation Equation 2.31

$$f_i(x, y, z) = (x_i - x)^2 + (y_i - y)^2 + (z_i - z)^2 - d_i^2 \tag{2.31}$$

Where $(x, y, z)$ is the location of the device to be localized, $(x_i, y_i, z_i)$ is the location of the $i$-th radio and $d_i$ is the calculated distance from the radio and the device. Combining the functions into the quadratic vector function F, produces a similar outcome as in 2D RLS shown in Equation 2.32

$$F = \begin{bmatrix} f_1(x, y, z) & f_2(x, y, z) & \dots & f_N(x, y, z) \end{bmatrix}^T \tag{2.32}$$

From here, we convert to a Jacobian Matrix J, as shown in Equation 2.33.

$$J = \begin{bmatrix} \frac{\partial f_1(x,y,z)}{\partial_x} & \frac{\partial f_1(x,y,z)}{\partial_y} & \frac{\partial f_1(x,y,z)}{\partial_z} \\ \frac{\partial f_2(x,y,z)}{\partial_x} & \frac{\partial f_2(x,y,z)}{\partial_y} & \frac{\partial f_2(x,y,z)}{\partial_z} \\ \vdots & \vdots & \vdots \\ \frac{\partial f_i(x,y,z)}{\partial_x} & \frac{\partial f_i(x,y,z)}{\partial_y} & \frac{\partial f_i(x,y,z)}{\partial_z} \end{bmatrix} \tag{2.33}$$

Similar to 2D RLS, we are able to then estimate the location where if we start with a location, as shown in Equation 2.34

$$l(n) = \begin{bmatrix} x(n) & y(n) & z(n) \end{bmatrix} \tag{2.34}$$

We can then update the location estimate with Equation 2.34

$$l(n+1) = l(n) + E_n \tag{2.35}$$

### 2.3.5 3D Expansion of CRLB for RSS-Based Positioning

In the 3D expansion of CRLB, the definitions are similar to that of 2D, but like with the expansion of RLS there now is the extra height dimension z. Now considering a X, Y, Z coordinate system, the path loss model is shown in Equation 2.36

$$P_r = P_0 - 10\alpha log(r_i) + \chi; \quad i = 1,\ 2,\ \ldots,\ N \tag{2.36}$$

Where

$$r_i = \sqrt{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2} \tag{2.37}$$

Variations in recieved power are now defined by Equation 2.38

$$dP_i(x, y, z) = -\frac{10\alpha_i}{ln(10)} \left( \frac{x - x_i}{r_i^2} dx + \frac{y - y_i}{r_i^2} dy + \frac{z - z_i}{r_i^2} dz \right) \tag{2.38}$$

The expression for $dP$ is still defined by Equation 2.26, but $dr$ and $H$ are now expanded in Equation 2.39 and Equation 2.40, respectively.

$$dr = \begin{bmatrix} d_x & d_y & d_z \end{bmatrix} \tag{2.39}$$

$$H = -\frac{10}{ln(10)} \begin{bmatrix} \sigma_1 & \sigma_2 & \ldots & \sigma_N \end{bmatrix} \begin{bmatrix} \frac{x-x_1}{r_1^2} & \frac{y-y_1}{r_1^2} & \frac{z-z_1}{r_1^2} \\ \frac{x-x_2}{r_2^2} & \frac{y-y_2}{r_2^2} & \frac{z-z_2}{r_2^2} \\ \vdots & \vdots & \vdots \\ \frac{x-x_N}{r_N^2} & \frac{y-y_N}{r_N^2} & \frac{z-z_N}{r_N^2} \end{bmatrix} \tag{2.40}$$

This also changes the covariance function as shown in Equation 2.41

$$cov(dr) = \sigma^2 (H^T H)^{-1} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy}^2 & \sigma_{xz}^2 \\ \sigma_{xy}^2 & \sigma_y^2 & \sigma_{yz}^2 \\ \sigma_{xz}^2 & \sigma_{yz}^2 & \sigma_z^2 \end{bmatrix} \tag{2.41}$$

From here, similar to the 2-dimensional case, we are able to take the standard deviation of the location error $r$, off of the resulting matrix defined in Equation 2.42

$$\sigma_r = \sqrt{\sigma_x^2 + \sigma_y^2 + \sigma_z^2} \tag{2.42}$$

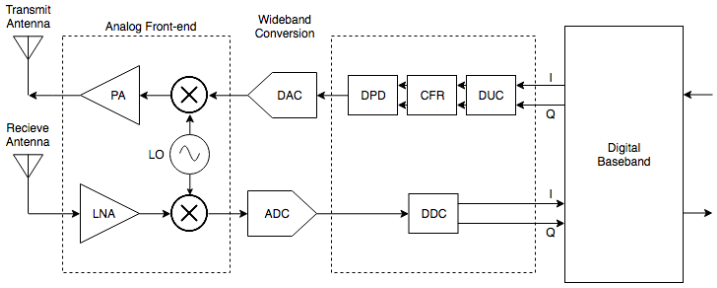


**Figure 2.4:** Functional Block Diagram of SDR Internals. The Motherboard is split into two main sections, the analog front-end and the digital baseband.

## 2.4 SOFTWARE DEFINED RADIO

As evinced in earlier sections of this report, software-defined radios are used instead of the conventional tools utilized by the Center for Wireless Information Network Studies (CWINS) in the past. As guiding notes for future work, the operating principles of software-defined radio will be discussed in this section.

### 2.4.1 SDR Hardware

Software defined radios have been implemented in several ways, and in many different form factors. The general definition of a software-defined radio is a technology where software modules interface with a generic hardware system consisting of Digital Signal Processors (DSPs), embedded general microprocessors, and analog radio frequency (RF) modulation/demodulation circuits. The purpose of the system is to implement functions that transmit and receive RF signals. The benefit of an extensively configurable system such as this is that link-layer and waveform-specific changes can be made to the radio platform without having to design additional hardware. The more attractive benefit for the user is that all operations can happen in real time. From a localization engineer's standpoint, this allows the capture of vast amounts of data in a short amount of time to feed positioning engines. Figure 2.4 shows the typical block diagram for the hardware inside of an SDR. The user application is either built on, or interfaces with the Digital Baseband block. This is where signals are generated and measured. This block can either be implemented on the same hardware as the SDR, or it can be offloaded to an external PC to be processed by software packages such as GNURadio, GQRX, or UHD. On the transmit side, the byte representation of the signal is converted into In-Phase, and Quadrature components in the phasor domain. This is commonly represented as IQ-pairing. The real to IQ conversion uses the Euler formula shown in Equation 2.43 to generate values for I and Q.

$$Ae^{jx} - Icos(x) + jQsin(x) \tag{2.43}$$

In this equation, a real signal A is represented as the in-phase component I, and the quadrature component Q. From there, the IQ pair is sent to a digital up-converter (DUC), which upsamples the signal to the sampling rate of the digital-analog converter (DAC) further down the chain. The upsampled signal is processed by crest factor reduction (CFR) and digital pre-distortion (DPD) to eliminate harmonics in the output signal. The DAC converts the

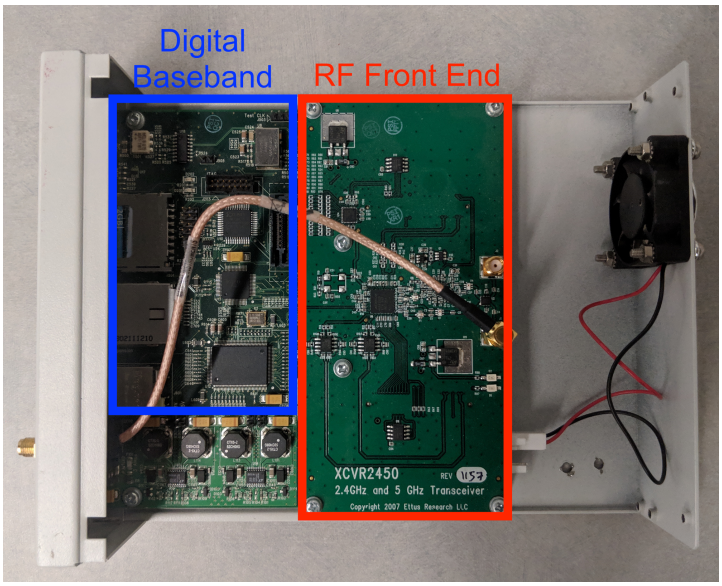


**Figure 2.5:** RFFE and Digital Isolation on USRP N210 SDR. Photograph of USRP Motherboard shows the digital baseband and the RF front end hardware. Only one receive channel per radio is used in this project.

output of the DPD into an analog signal that is then modulated to passband by mixing with the local oscillator (LO), and is finally amplified by the power amplifier (PA) before reaching the transmit antenna. To receive a signal, the reverse process is performed, but the signal is instead amplified by a low-noise amplifier (LNA), and the resultant signal is then processed by a digital downconverter (DDC) to synchronize the sampling rate of the RF signal to the rate of the baseband signal. All of the analog blocks are isolated into what is known as the Analog or RF front-end (RFFE). Some devices like the USRP choose to put the RFFE on a separate daughterboard to the motherboard (Figure 2.5), and other manufacturers choose to use a single PCB, but topologically group the analog blocks together.

### 2.4.2 SDR Software

As mentioned in the introduction to the parent section, SDRs are useless without accompanying software to process the samples being recorded by the hardware. Users have multiple ways of interfacing with SDRs, each with varying levels of proximity to and granular control of the hardware, which is shown in Figure 2.6. SDR software can be stratified into three hierarchical layers as described in Figure 2.6. These layers are; the Programmable Logic (PL) layer, the Embedded layer, and the Host layer. PL-layer software is implemented on the Digital Baseband block inside of the SDR itself. The Digital Baseband can consist of an Application-Specific Integrated Circuit (ASIC), a Field Programmable Gate Array (FPGA) combined with an embedded microprocessor, or solely a monolithic FPGA. Most early SDRs (USRP1, USRP2, USRP N210) contain either a monolithic FPGA or an ASIC to handle the digital baseband processing. More modern units (Nutaq, ADALM-PLUTO, USRP X310, USRP N310) contain either a hybrid SoC like the Xilinx MPSoC which contains the embedded and programmable logic on the same die, or they will include a discrete FPGA and embedded processor. Any application that can be done in higher layers of software can also be done in

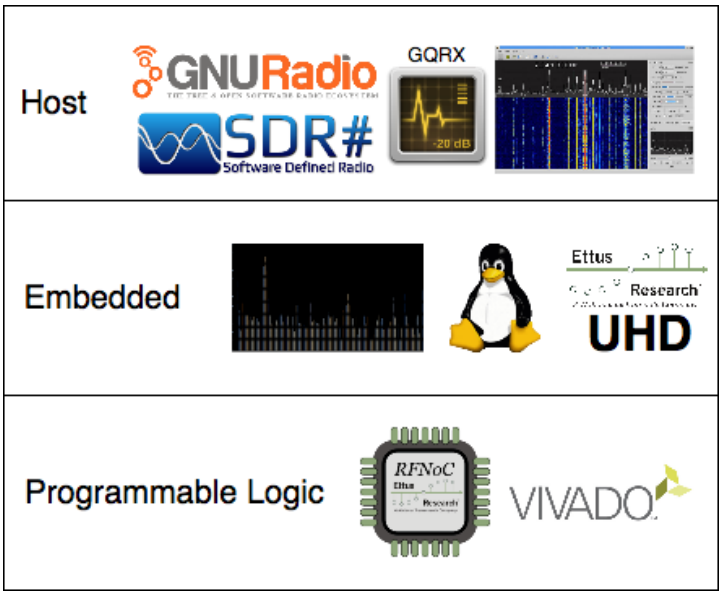


**Figure 2.6:** Landscape of SDR Software Packages. RFNoC allows the user to write software at the programmable logic layer (FPGA). UHD allows the user to write software on an embedded CPU, and GNURadio allows the user to write applications for the host. UHD can also be run on the host layer.

the PL-layer. Due to the speed of FPGA circuits, this layer is most useful for implementing communication waveform blocks like modulators, demodulators, interleavers, and timing/frequency offset feedback loops. However, this speed comes at the cost of user-friendliness, and the increased time required for design and prototyping. One of the technologies that aims to simplify this process is known as the Ettus RF Network on a Chip (RFNoC). It resides at the FPGA layer and allows the user to create circuits, represented as 'User IP', that can interface with GUIs further up the software hierarchy, which are represented by the Crossbar (Figure 2.7). All PL-Layer applications are written in a Hardware Description Language (HDL) such as Verilog, or VHDL. Another way to interface with SDRs is at the Embedded level. Embedded level software resides on the digital baseband, and is executed on the embedded CPU or MPSoC. These applications are typically written in C or C++, and interface directly with the hardware registers using software supplied by the manufacturer. The most common way to interface with SDRs is at the Host level. Host level software resides on the SDR host, which is either a PC or an external embedded system. These applications are typically written in C, C++, Python, or Java, and require the use of a hardware driver to retrieve samples from the SDR. Additionally, a serial or network connection must be established in order to retrieve the samples from the radio by the host, which is illustrated in Figure 2.8. Each SDR manufacturer supplies their own hardware driver, complete with Application Programming Interface (APIs) and associated documentation. Host-layer applications are then built using these APIs like UHD, which then result in the popular software packages that engineers use today like GQRX, GNURadio, and SDR#. Since localization does not require FPGA-level speeds, and because user-friendliness is a priority, our team elected to go with the host-layer approach as described in this section. We chose GNURadio as our software application because of the abundant support for our hardware, as well as the extensive configurability. GNURadio is a C++ and

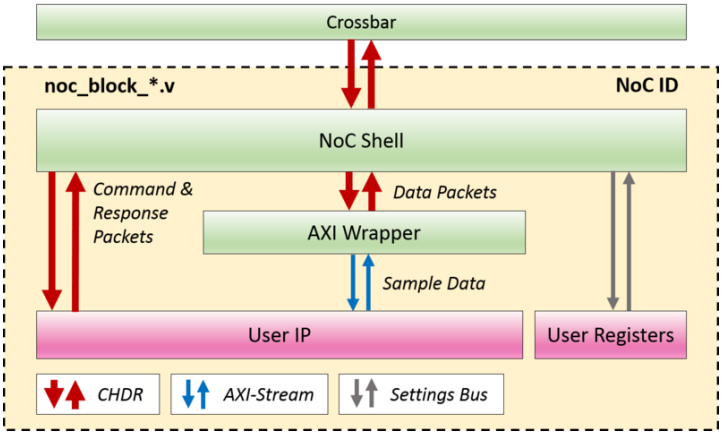


**Figure 2.7:** Structure of a RFNoC Block. The crossbar is the digital baseband used in earlier examples, with the user-created module wrapped in a packet interchange format called AXI Stream.
© Ettus Research https://kb.ettus.com/images/thumb/6/6b/rfnoc_gsg_an_4.png/800px-rfnoc_gsg_an_4.png

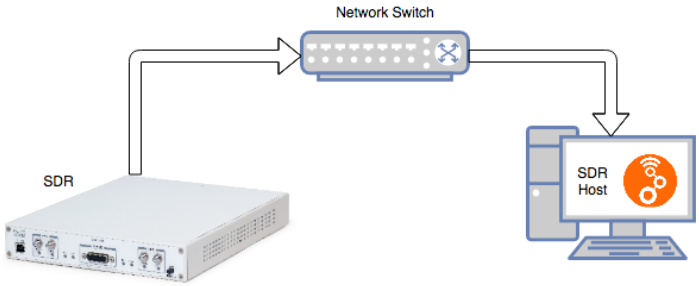


**Figure 2.8:** Host Layer SDR Operation. The SDR is connected to an IPv4 switched network, and packets containing IQ samples are ferried between the radio and the PC.

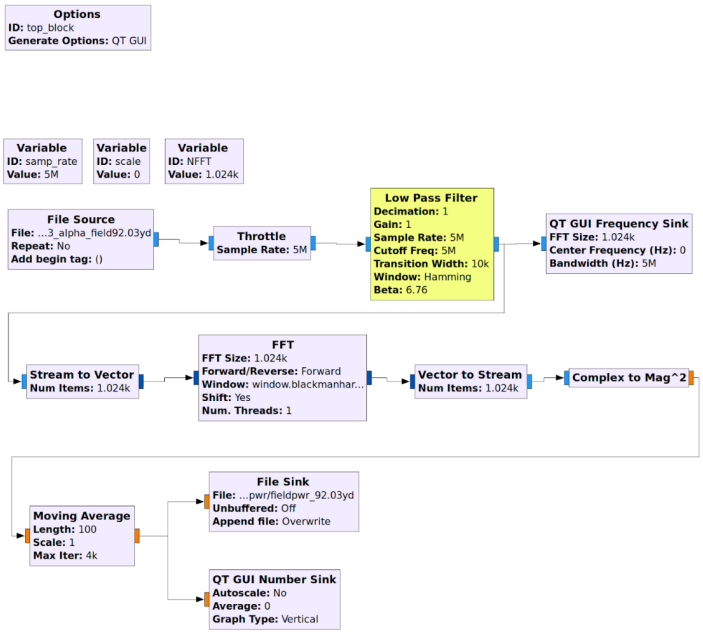


**Figure 2.9:** GNU Radio Flowgraph. This block diagram shows the steps taken to calculate the RSS. First, the Fast Fourier Transform (FFT) is taken, then the magnitude is computed and averaged along a sliding window. The measurements are then packaged into a UDP packet to be used by the Python app.

Python framework that allows engineers to prototype their SDR designs using a sample flow, block-based metaphor. Samples flow from source blocks to sink blocks, connected by application-specific processing blocks. The collection of these blocks is known as a Flowgraph. In the example shown in Figure 2.9, Samples are being processed from a file, and then written to another file. Additional blocks can be written by the user to fulfill more specific tasks, or downloaded from the Web. GNURadio applications can also utilize GUIs written in QT, a graphical framework. These applications can also communicate with other apps over the network, which is what our team does in order to perform ranging and RLS calculations. Figure Figure 2.10 shows a more complex flowgraph where the USRP devices are being read from in real time in order to drive software further down the chain that calculates the position of the UAV.

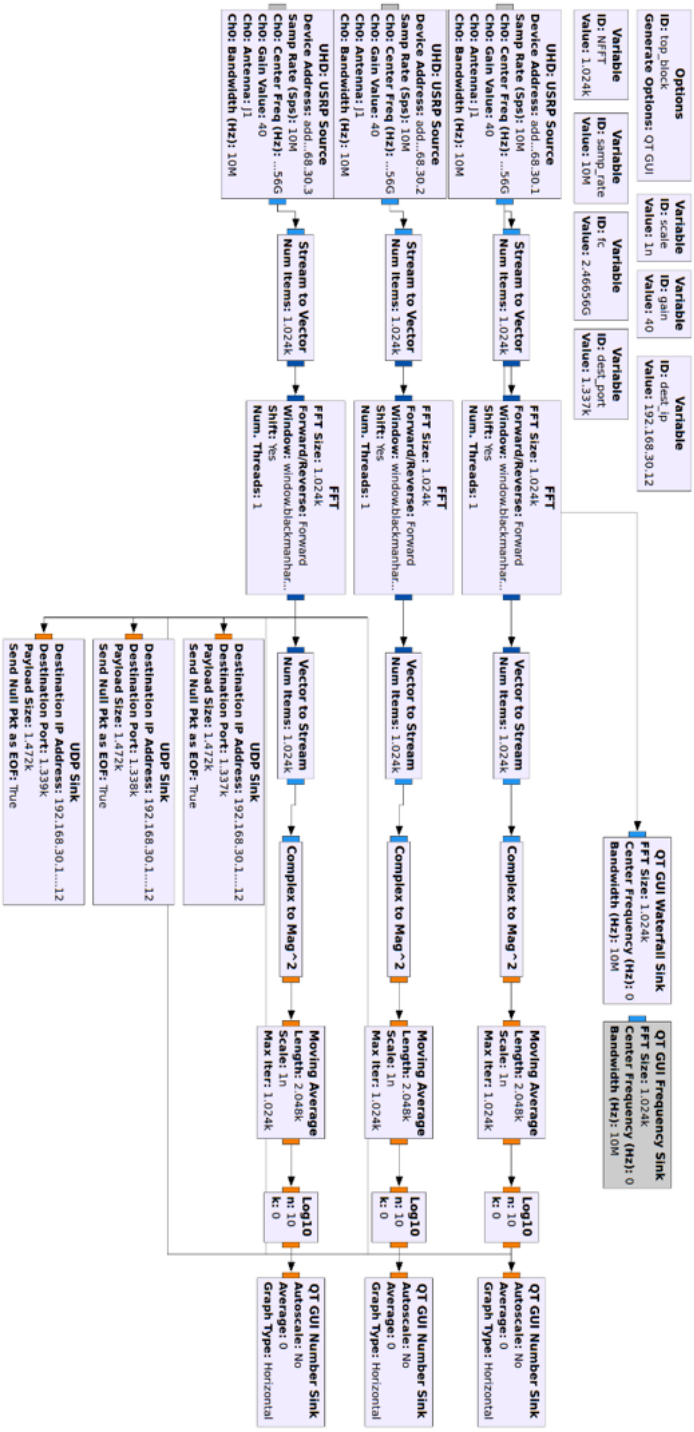


Figure 2.10: Multireciever USRP Flowgraph. This flowgraph shows the act of receiving IQ frames from the USRP (inserted as a data source), processing them to generate the RSS readings, and then writing the results to a UDP socket. This socket is then read from by other applications to calculate the position of the UAV.

# 3 SYSTEM ARCHITECTURE

This project leverages software defined radio for increased sampling rates and signal processing development. In this chapter, the overall outline of our system and system modules will be discussed. First, a functional block diagram of the system hardware and their relative connections are examined. The localization system LAN and the decision to why creating the LAN for the system are discussed. Comparisons between the functional block diagram discussed in Chapter 1 and the realized version discussed in this section are made. Next, each individual hardware module used in the design of the localization system are examined. The specifications are included with each of the hardware module descriptions. Finally, the software flow of the system is analyzed in this section. RF signal conversion, signal processing, data acquisition and processing, algorithm computation, and real-time visualization are all discussed within a software perspective in this section. Different software applications, such as GNURadio, Python, and C are described due to their use in the creation of this localization system.

## 3.1 FUNCTIONAL BLOCK DIAGRAM

The RSS-based UAV localization system is constructed with five SDRs connected by ethernet to a local area network (LAN). The system involves five Ettus Research USRP2 software defined radios which are connected to a central 16-port network switch. The LAN also contains a host computer that will receive transmission and perform signal processing. The configuration of the LAN enables accessibility to multiple hosts for parallelization of processing. The system is shown in Figure 1.2. Featured are 4 Ettus Radios, each radio has its own static IP. Each Ettus Radio is connected to a 16-port network switch. All of the hardware components of the system are connected via ethernet to a central switch, to which a host computer and any other user may also connect. All of the components in this network require 120V power.

## 3.2 HARDWARE MODULES

Provided below are specifications of the hardware modules shown in the functional block diagram and Figure 1.2. The specification of each of these hardware components are described below in 3.2.


Figure 3.1: Physical System Architecture.

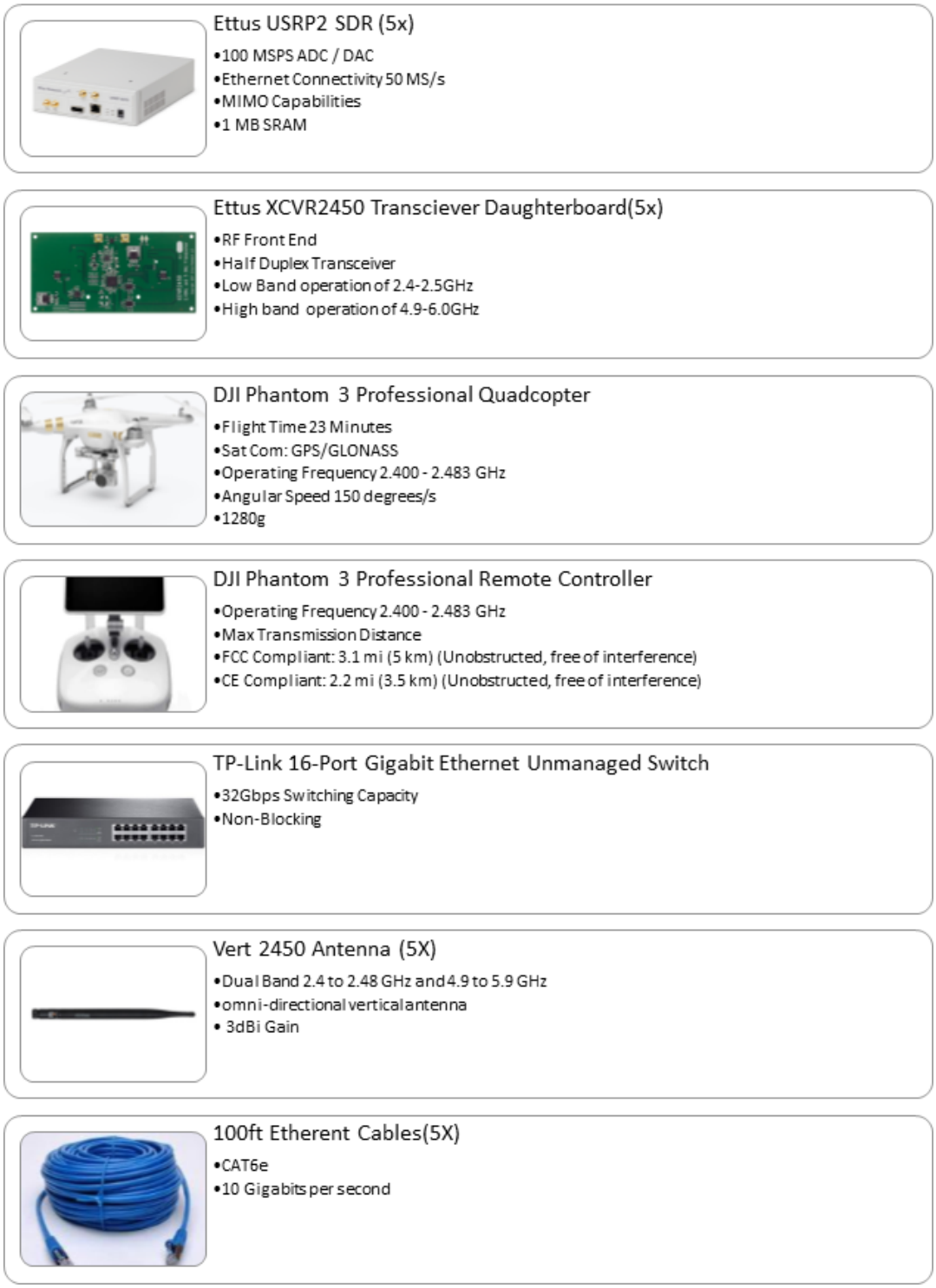


**Figure 3.2:** List of System Hardware Components. This info graphic distills the table of hardware modules, and provides relevant specifications for each module that contributed to the final product

## 3.3 SOFTWARE FLOW

The system is configured as a Local Area Network (LAN). All devices on the network are statically addressed with IPV4 protocol. The USRPs transmit CHDR ("cheddar") packets. CHDR is a proprietary ETTUS packet format for data transmission between the USRP and GNU radio application. The CHDR packets transmit from each USRP to a GNU Radio Application hosted on a statically address terminal. The GNU radio application is executed on a 16.04 Ubuntu distribution. Within the GNU Radio Application, a GNU Radio Companion Flowgraph performs the signal processing and sends UDP packets containing RSS measurements in dBm to a python server hosted on a second statically addressed terminal. The Python server then maps each RSS measurement to the path loss model, and runs the RLS localization algorithm. The server then sends a UDP packet to a third statically addressed terminal. The third terminal is responsible for plotting and visualizing the localized coordinates.

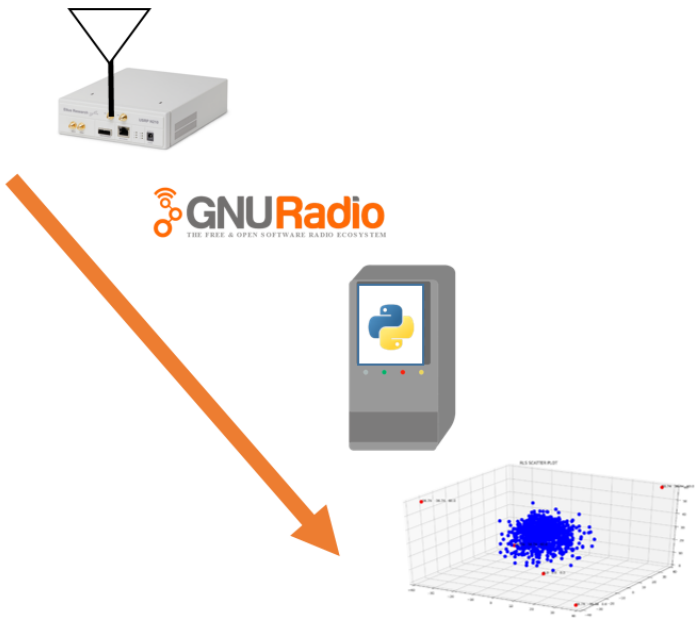


**Figure 3.3:** Raw data from each radio was passed through to the GNURadio application via UDP. Once the RSS was calculated, another UDP packet was formed to send the measurements to a python application to perform localization and display the results.

# 4 RESULTS AND DISCUSSION

In this section, the preliminary data acquisition results are described. First, the drone communication ISM bands are described and their relation to FCC findings are described. Next, using channel modeling, the first meter path loss and distance power gradient are derived. Methods as to how these results were obtained are discussed. The techniques to data acquisition used in this project are described and reasoning behind the use of these methods are examined. The received signal strength (RSS) measurements from drone emissions are shown between distances ranging from 1 to 91 meters. Using these results, a general path loss model for the channel in which the system will be set is introduced. Also, the RSS samples compared to distance are measured and plotted in real time, shown at the end of this section. The CRLB contour plots which were created using our channel model and used for verification of the system in 2D and 3D. Experimental distance measurement error which was a result of our 1D ranging are shown for multiple distance ranges. Next, the outdoor setup of the system in 2D is shown for testing, and the simulated results are described. The real-time 2D localization results are then discussed. Finally, the the simulated and theoretical 3D results are described, as well as the 3D outdoor setup. All of these results are compared to their respective CRLB plots to verify system functionality and accuracy. The effects of RSS averaging on system accuracy is explored.

## 4.1 PRELIMINARY FINDINGS

Prior to developing the system described in Chapter 3, preliminary experiments had to be conducted in order to identify critical system parameters. This entailed mapping the internal network of the DJI system to isolate waveforms of interest. Once settled on a waveform, we had to study how the drone would communicate using it, which involved measuring the RF spectrum for all possible transmit frequencies. The path loss model also had to be created by recording controlled RSS measurements in known distance increments across the WPI football field. Simulations were also carried out to verify performance of the system. Finally, FCC reports for the DJI Phantom 3 were referenced to identify more channel parameters.

### 4.1.1 Drone Communication Frequency Allocation and FCC Findings

Our first objective was to identify the communication protocols of the DJI Phantom 3 Professional. The smart device utilizes USB On The Go (USB-OTG) serial to communicate between the itself and the controller. The controller was determined to communicate between the controller and the aircraft at 2.4 GHz. The communication between the drone and the controller is set in the DJI Go mobile application to automatically frequency hop.

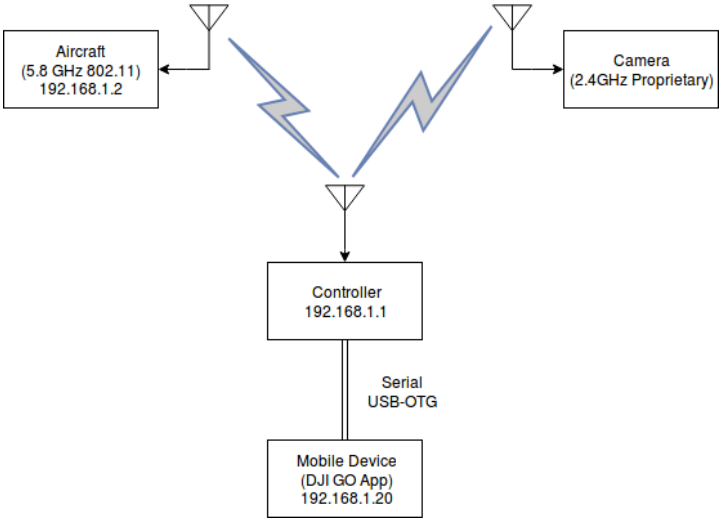


**Figure 4.1:** DJI Phantom 3 Professional Communication Protocols. Each block shows each node's IP address, and the mode of communication used. The aircraft/camera speak to the controller wirelessly, and the connection between the mobile device and the controller is handled via USB.

A screenshot of the DJI GO App is shown in 4.2, and allows the operator to specify the ISM channel for communications between the drone and the operator. DJI uses a proprietary transceiver called LightBridge. Lightbridge provides the operator with 2.4GHz HD video stream from the drone. In Figure 4.2 ISM channel 19 is chosen as the current operating frequency between the drone and the remote controller. Channel 19 was statically set for testing.

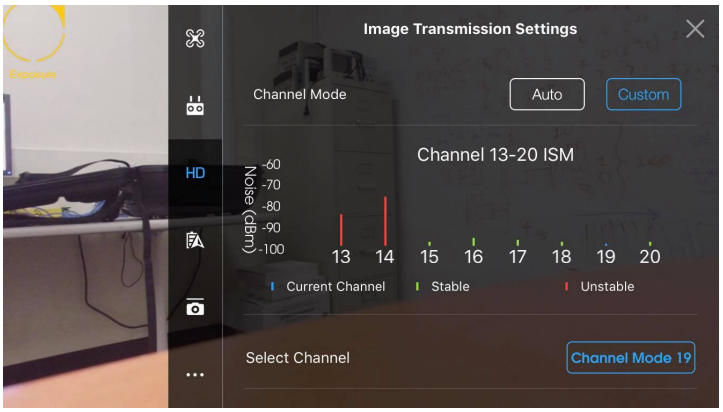


**Figure 4.2:** ISM Channel Modeling DJI Go App. This screenshot is from the "DJI Go" controller application, and it's showing signal-to-noise ratio (SNR) for each ISM channel

Our next objective was to identify the channel characteristics of the ISM Channels from the drone. Using the ETTUS N210, and GNURadio's built in FFT block, the center frequency and bandwidth of each channel was recorded. Table 4.3 shows that the average bandwidth of the channels are close to 10 MHz, with the center frequencies ranging from 2.40664 to 2.47652 GHz.

| Channel Name | Center Frequency | Lower Band Limit | Upper Band Limit | Total Bandwidth (MHz) |
|---|---|---|---|---|
| 13 | 2.40664 | 2.40179 | 2.41148 | 9.69 |
| 14 | 2.41647 | 2.41172 | 2.42122 | 9.5 |
| 15 | 2.42649 | 2.42172 | 2.43125 | 9.53 |
| 16 | 2.43651 | 2.43184 | 2.44117 | 9.33 |
| 17 | 2.44656 | 2.44185 | 2.45127 | 9.42 |
| 18 | 2.45653 | 2.45181 | 2.46125 | 9.44 |
| 19 | 2.46656 | 2.46184 | 2.47128 | 9.44 |
| 20 | 2.47652 | 2.47176 | 2.48127 | 9.51 |

**Figure 4.3:** Channels, Center Frequency, and Bandwidth for DJI Lightbridge Technology. Each channel takes up around 9.5 MHz bandwidth. Channel 19 is chosen for all of our experimentation.

The Federal Communication Commission (FCC) enforces rigorous standards on RF emissions for consumer devices. All U.S consumer drones are rigorously tested for compliance by the FCC. The FCC test reports give much insight on waveform specifications such as frequency hopping, transmit power levels, frequency allocations, and bandwidths.

| Frequency (MHz) | Antenna Gain | | Conducted Power | | Evaluation Distance (cm) | Power Density ($mW/cm^2$) | MPE Limit ($mW/cm^2$) |
|---|---|---|---|---|---|---|---|
| | (dBi) | (numeric) | (dBm) | (mW) | | | |
| 2406.5 | 2 | 1.58 | 27.87 | 612.35 | 20.00 | 0.19258 | 1.0 |

**Figure 4.4:** FCC Power Analysis of the Phantom 3

FCC Power Analysis of the Phantom 3. This metric was used as the basis for the first meter path loss Lo, and by extension, the channel model used in the localization system

Table 4.4 is from the power analysis by the FFC 2013 review of the DJI Phantom 3 UAV. The FCC lists the consumer product's maximum transmit power, antenna gain and power density. The FCC also addresses frequency hopping parameters and modulation schemes. The DJI Phantom 3 professional had a maximum conducted power of 27.87dB or 612.35mW.

### 4.1.2 Data Acquisitions and First Meter Path-Loss

The first meter path loss of the camera feed signal was measured inside the WPI CWINS anechoic chamber. The drone was placed exactly a meter away from the USRP, and the RSS measurements were recorded. As expected, the signal strength was very strong because because of the minimal distance as well as no network interference with unwanted signals. The power at 1m was measured to be approximately -35 dB. In our ranging data collection, our first test examined the accuracy of our system in 1-D. To do this we set up a similar test environment as the path-loss model tests. We set up our radio at one spot and calculated distances derived from the signal of the drone at different locations of the field. The locations ranged between 5-85yd and 93.01yd as shown in Figure 4.6.

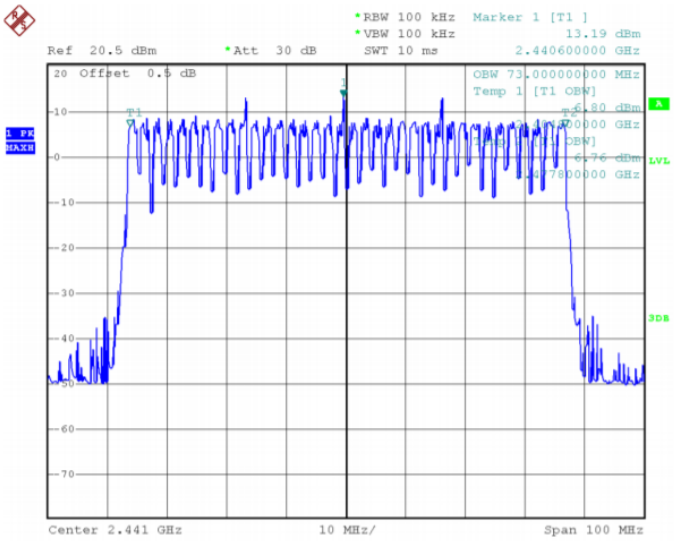


**Figure 4.5:** DJI Video Wavefourm This figure shows the FFT plot of the drone's camera stream waveform. It is a multicarrier OFDM waveform with a bandwidth of 10MHz.

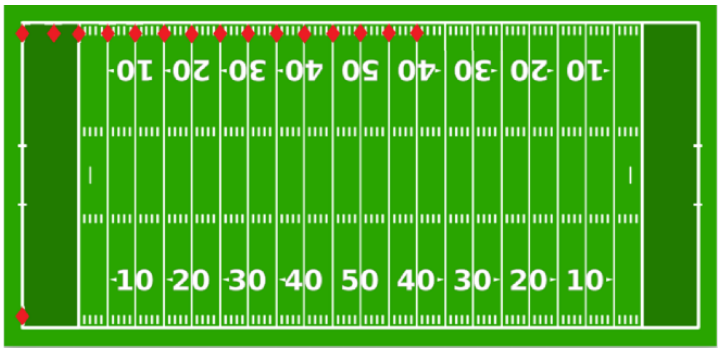


**Figure 4.6:** Football Field Measurement Locations. Denoted by red Triangles, 16 total measurements were taken with the drone situated farther from the receiver in 5 yard increments. The longest distance measurement was taken with the drone being across the field from the receiver, approximately 93.01 yards.

The RSS values were recorded at the each of the denoted locations in Figure 4.6. The Ettus USRP2 samples at 10 million samples per second and each recording lasted 10 seconds. At each location 100 million RSS values were recorded.

$$RSS\ Samples = 100,000,000 = 10MS/s \cdot 10s \tag{4.1}$$

With the 100 million IQ values, RSS samples power calculations were made with a 1024 bin FFT. Resulting in approximately 97650 power calculations per second.

### 4.1.3 Path Loss Model and Real-Time RSS

The RSS of the drone was captured at 1 meter and 5 to 91.76 meters in 5 meter increments. The drone was placed at a constant height of 0.712 m

**Figure 4.7:** Outdoor RSS Acquisition (RSS Ranging). Shows one half of the experimental setup and propped the setup above the ground to avoid signal degradation coming from the ground.

for the measurements to account for signal degradation due to the ground. Attenuation of the signal ranged from -35dB at 1 m to -74dB at 91.7 m.The RSS values were plotted against distance. Distance ranged from 1 meter to 91.7 meters. Figure 4.8 shows the plot of RSS (dBm) vs 10*log10(d)[dB].

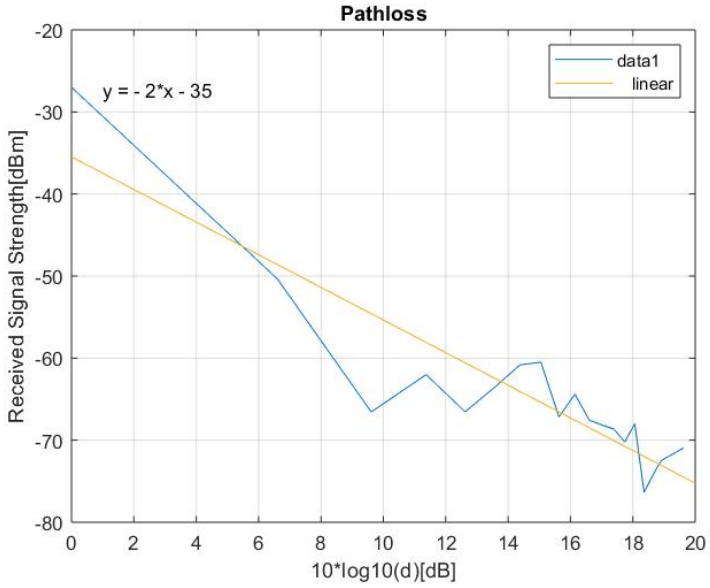


**Figure 4.8:** Path Loss Plot. Shows linearity with 10*log(d) distances between 0 and 10 m, but nonlinearity between distances 10 and 20 m. A line of best fit is constructed to account for this.

| $\alpha$ | $P_0$ (dBm) | $\chi$ (dBm) |
|---|---|---|
| -1.988 | -35.476 | 4.5124 |

**Table 4.1:** Distance Power Gradient and Po. The first meter path loss is -35.48, with an distance power gradient around 2, and a standard deviation of shadow fading around 4.5. These values served as a basis for our path-loss model

The team developed a real-time RSS vs samples Graphical User Interface (GUI). The GUI used Bit Block Transfer (BLIT) to display the Real-Time RSS samples. BLIT multi-threads a python plot method with enough accuracy to plot in real-time.

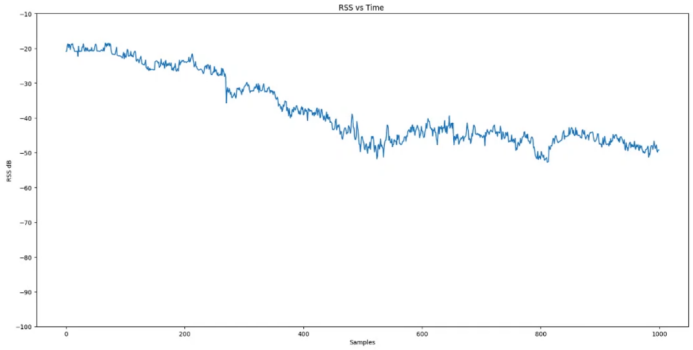


**Figure 4.9:** Real-Time RSS Acquistion. As the drone flew further away from the receiver, the average RSS value decreased as expected, shown in the figure.

Figure 4.9, displays the RSS of the drone to a radio receiver with varying distances. As the drone gets farther away from the receiver, the RSS values decrease, as seen in 4.9. This recording was taken in real-time. The RSS samples are filtered through a Fast Fourier Transform (FFT) of size 1024 resulting in approximately 97650 power calculations per second. When looking at signal strength vs time, it can clearly be seen how the shadow fading constantly effects the signal. To test for shadow fading, the drone along with a holder were placed at increments of 5 meters away from the software defined radio. To simulate shadow fading, the holder was then instructed to rotate 360 degrees with the drone in a time interval of approximately 10 seconds. The effects of shadow fading on received signal strength at a 5 meter distance can be seen in Figure 4.10.

As the drone rotates and gets fully blocked by the body (at 5.8 seconds) the signal can be seen to attenuate over 20 dB. This deviation in signal can lead to error in the calculated distance in the path loss model and therefore increased error in accuracy as well.

### 4.1.4 Simulated 2D Positioning CRLB and Analysis

With our path loss model, we are able to use the 2D CRLB equation provided in the background section to calculate our standard deviation at any given (x,y) point. This standard deviation, similar to that of ranging, gives us the minimum accuracy our localization system can have in 2D at a certain (x,y) point. To visualize this accuracy, we created our theoretical 2D CRLB by calculating our CRLB error bound at every point in a 2D plane and plotting it. The resulting CRLB for our system is shown in Figure 4.11:

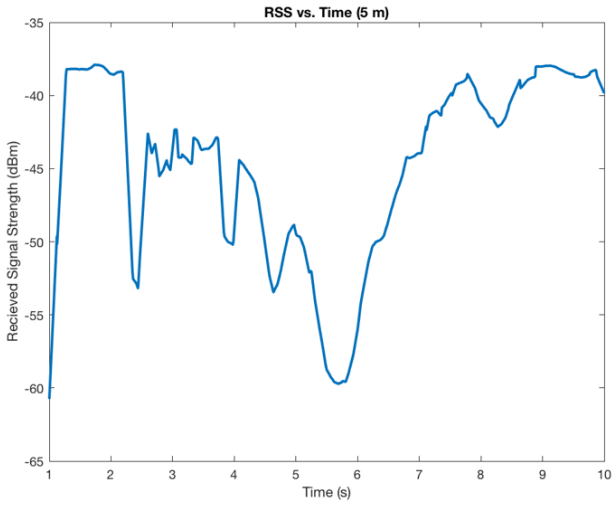


**Figure 4.10:** Shadow Fading RSS vs Time. Shows a dramatic decrease of approximately 15 dBm due to shadow fading throughout the entirety of the rotation.

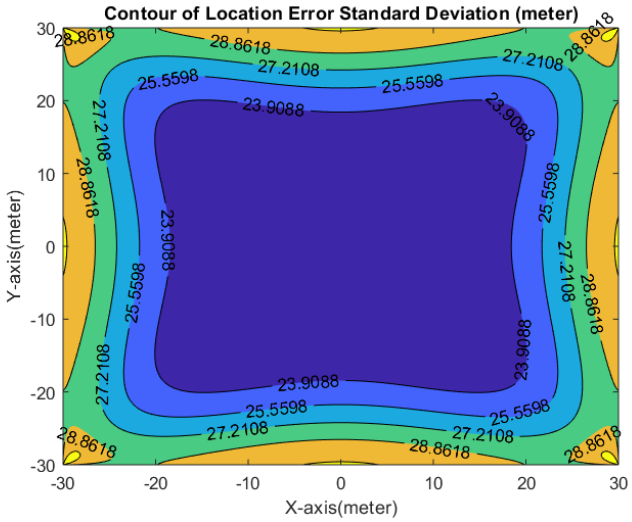


**Figure 4.11:** Simulated 2D CRLB. CRLB contour plots show the theoretical standard deviation of error given a certain location in the theater of localization. Transmitters were placed at four equidistant corners for this simulation.

In 4.11 contour plot, the standard deviation values are calculated given four radios, each placed at the corner of a 60 x 60 meter area. The plot shows higher deviation values at the edges and of the area, where it is at its longest distance for some of the radios, and at its lowest in the middle where all of the radios are generally close to the drone. The highest deviation from this plot was recorded at 28.86m and the lowest deviation was recorded at 23.91m.

### 4.1.5 Simulated 3D Positioning CRLB and Analysis

Like 1D and 2D, the path-loss model was used calculate the CRLB values at given (x,y,z) points. To visualize this 3D CRLB, we developed software that would calculate the CRLB at different height levels and stack them on top of each other, with different colors depicting the different levels of accuracy.

In Figure 4.12, the orientation of the four radios are all the same, with each being at the corner of a 30m x 30m plane. It can be seen that higher error can be found in the lower and upper height regions of the 3D space, while the middle heights have greater accuracy. To further examine the accuracy of the localization system we also created a more detailed plot that only look at specific heights to be able to see the CRLB characteristics closer.

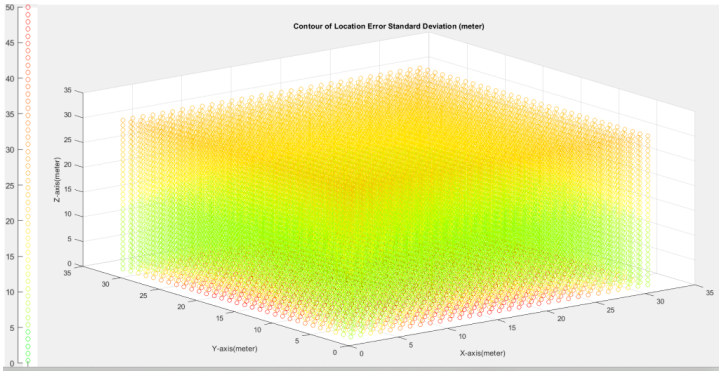


**Figure 4.12:** Simulated Full 3D CRLB. Reference points are arranged at equal height in a quadrilateral formation. The standard deviation of measurement error increases with height.

In Figure 4.13 it can be seen how the different heights affect the CRLB of the localization system. As the height goes up, so does the CRLB value of location error. At the different heights on the graph: 10, 15, 20 and 25 meters they all can be seen to have the similar characteristic of the 2D CRLB plot. At the edges near the individual radios, the CRLB is seen at its highest due to the distance away from the other 3 radios, where in the center the CRLB drops creating a smooth depression in the middle.

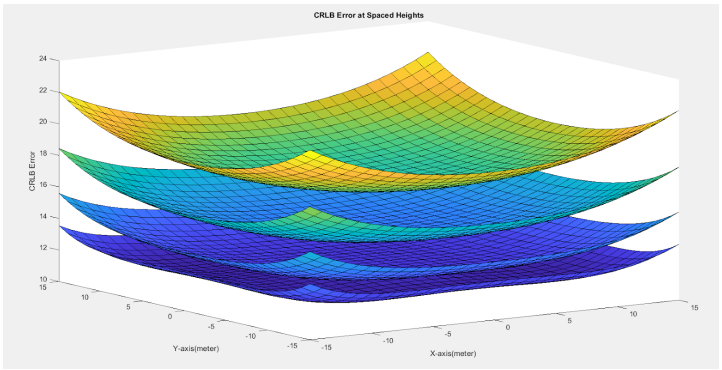


**Figure 4.13:** Multi Layered CRLB Plot. This figure concisely shows the direct proportionality of standard deviation of error with respect to drone altitude..

From the full 3D graph along, we saw the lower levels of height to return suspicious values of CRLB as we expected the CRLB values to start

low and increase along with the height, through further investigation we discovered that the inaccuracy of this value comes from the calculation of the CRLB function, specifically in the inverse portion of the calculations. When looking at how the position of the drone effects the CRLB in lower height conditions, the figure below illustrates the CRLB error at different height levels. CRLB at low height levels can be seen to be higher towards the center and lower towards the edges. In addition the lower height levels show higher CRLB values and as the height rises, the error calms down and flattens out until we get the normal CRLB of the higher height conditions. In the graph the top plot is the low level height at 2 meters and as the plots go down the height rises up to 10 meters.

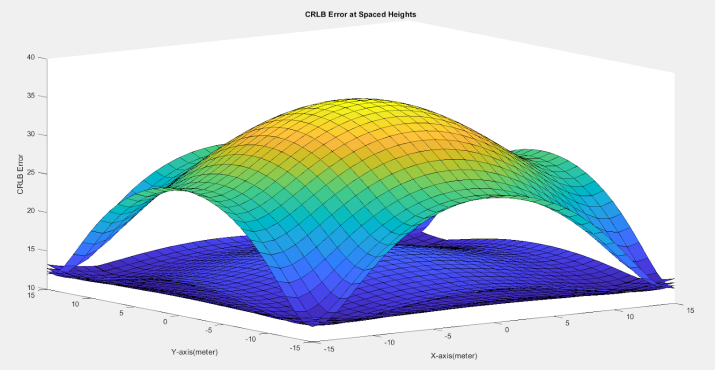


**Figure 4.14:** Multi Layered CRLB Plot Under Low Height Conditions. This plot shows a tighter range of heights closer to the ground. This illustrates the high standard deviation of error close to the ground.

### 4.1.6 Experimental Distance Measurement Error

Distance Measurement Error (DME) of the localization system is assessed using multiple RSS readings from different distances. By creating DME plots, an understanding of where there is error in our system at different distances is established.

Figure 4.15 displays an enormous amount of samples with an error of approximately -10 m when ranging from 30 yd. There are also samples which are scattered around -10 m in either direction, but mainly extending towards 0 m error. This error could be caused by multiple factors, such as multipath effects, signal interference, and shadow fading. The DME is taken into account while observing localization error of the system.

The DME is expected to increase as distance between transmitter and receiver increases. As shown in Figure 4.16, samples are situated mainly between error values of -10 to -20 m when measurements are taken at 60 yd. There is also a spike of error at -35 m, which can be explained by receiving an unwanted transmitted signal. Therefore, only the main spike in the figure was taken into consideration.

Figure 4.17, describes three different setup configurations and the probability of error associated with them. The three configurations are comprised of the radios set up in three different sized squares, 100 square meters, 400 square meters, and 3600 square meters. The smallest square creates the least probability of error at greater distances between drone and receiver. As seen, the probability of error is 100 percent when the drone is 100 m from the nearest arbitrary receive, which is expected. However, when the

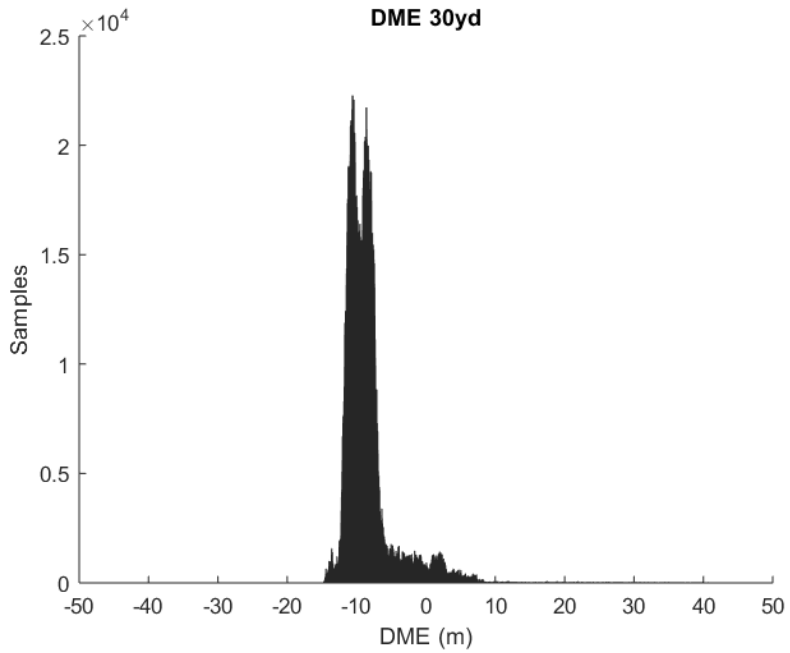


**Figure 4.15:** DME of the system with 30 yards between transmitter and receiver. The second peak on the left-hand side is caused by signal dropouts from the analog front-end

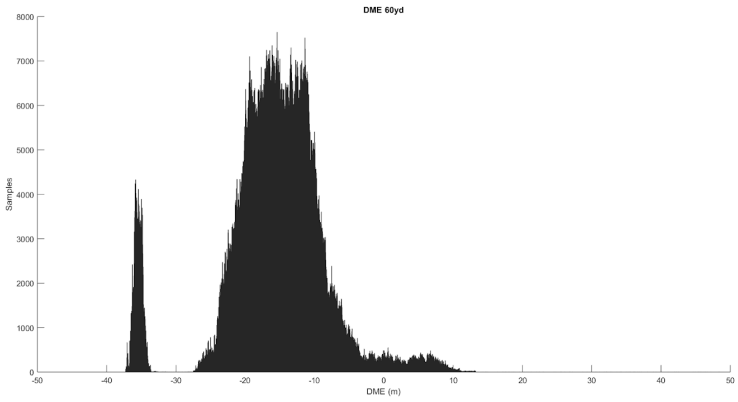


**Figure 4.16:** DME with 60 yards between transmitter and receiver. The second peak on the left-hand side is caused by signal dropouts from the analog front-end

square is very large at 3600 square meters, the probability of error is 100 percent only at a distance of 30 m between drone and receiver. As setup configurations become less compact, the probability of error increases for smaller distances. With proper analysis of Figure 4.17, the setup configuration for the localization system to cover larger distances while preserving performance was analyzed.

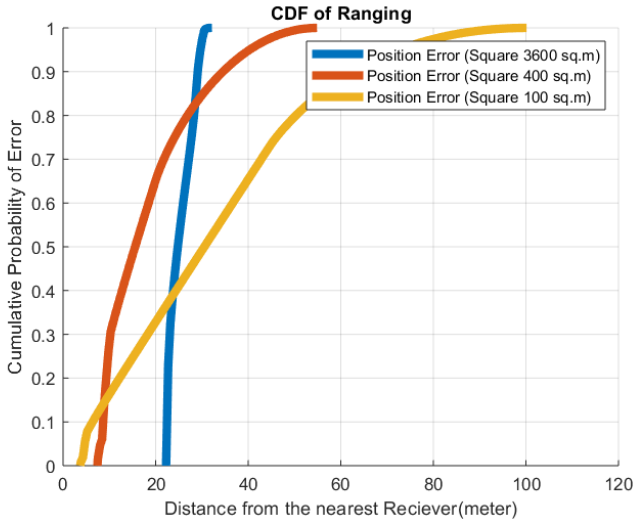


**Figure 4.17:** Probability of Error From Different Ranging Distances. As shown, if the square covers a small area, there is a lower probability of error for greater distances from the receivers. For the largest square, The probability of error is almost constant no matter the distance from nearest receiver.

## 4.2 2-DIMENSIONAL SIMULATED AND THEORETICAL LOCALIZATION RESULTS

The system was implemented successfully in 2D and 3D localization. The team first simulated 2D and 3D localization and then performed real-time localization in 2D space. The system's performance was quantified by the reduction in DME by averaging RSS over a period time prior to calculating RLS to derive location.

### 4.2.1 2-Dimensional Simulated Results and CRLB Comparison

Three USRPs were used in 2D localization. The radios were positioned in an equilateral triangle. The drone was positioned at the center of the equilateral triangle. Recorded RSS power calculations at 30 yards were used as simulated input. To gather real-time 2D data, the localization system is set up in the manner shown in Figure 4.18. The distance between each radio and the drone when it is situated in the center of the triangle is 30 m.

The drone was positioned at (0,0) m. Anchor points were chosen for the three radios . The three radios were positioned at corners of the equilateral triangle at coordinates (0,17.2) m, (15,-8.66) and (-15,-8.66) . The results of the system was plotted over the a 2D CRLB plot. The CRLB plot reflected our derived path-loss model. The CRLB plot and localization system assumes an alpha = -1.988, P0 (dBm) = -35.476 and standard deviation of shadow fading of 4.5124 (dBm).

The red dots in Figure 4.19 are estimated drone position results. The estimated position results have a DME mean of $\mathbf{.9378}\boldsymbol{m}$ and a standard deviaton of $\mathbf{1.47674}\boldsymbol{m}$.

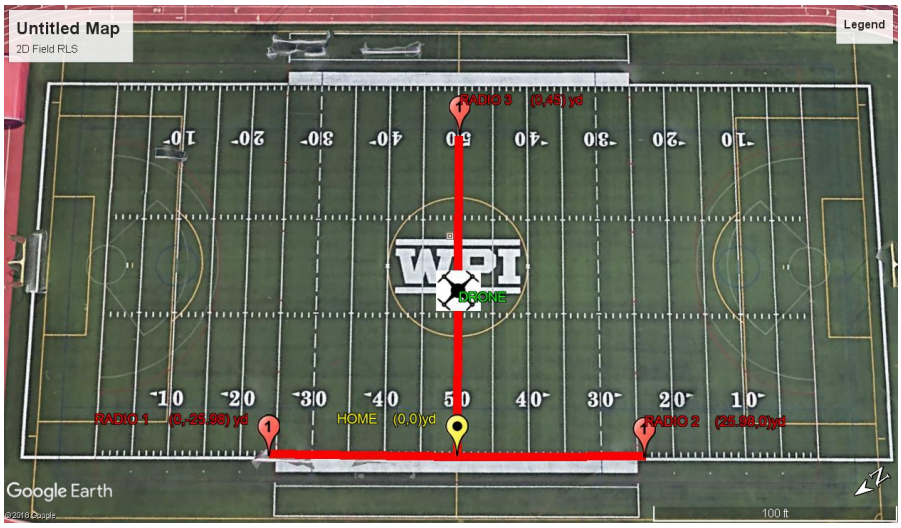


**Figure 4.18:** Setup Configuration with Labeled Axes. This is an aerial map view generated with Google Earth to show exact positioning of each node in the system.

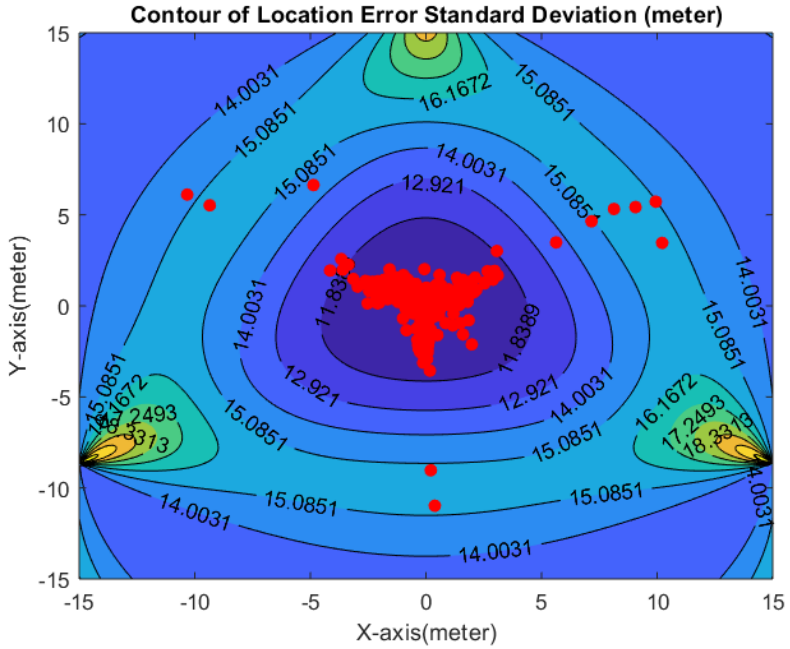


**Figure 4.19:** 2D Localization Comparison with CRLB. Real-time localization results are overlaid on top of the CRLB contour plot.

### 4.2.2 Real-Time 2D Localization Results

As expected, testing in real-time produced more error than the simulated tests. Figure 4.20 displays the drone localization results when the drone is located in the center of an equilateral triangle. Despite the increased amount of measurement error, a majority of the points were located in the center of the triangle. Factors that may have played a role on this testing day could be high signal interference and a greater amount of noise while recording. Despite this, the drone localization system still proved its functionality in

2D. In comparison with the 2D CRLB plot, there is the least amount of measurement error in the center of the plot with the radios set up to create an equilateral triangle. In 2D real-time, most of the points were located in the center of the plot, which shows the accuracy of the system in 2D. Figure 4.20 Real-Time 2D Localization Results demonstrates and validates the systems functionality.

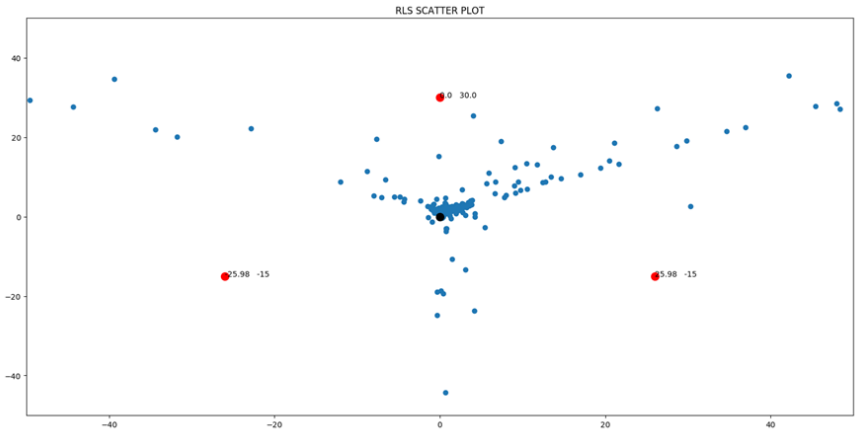


**Figure 4.20:** Real-Time 2D Localization Results. The drone was flown at 3 meters altitude in the center of three equilaterally spaced USRP radio receivers.

## 4.3 3-DIMENSIONAL SIMULATED AND THEORETICAL LOCALIZATION RESULTS

In this section, the effectiveness of the localization system in 3D will be discussed via simulation and real-time results. Previously recorded data will be used in order to generate 3 dimensional plots of the system. In 3D, a total of five software-defined radios used. The system in 3D will be compared to the 3D CRLB generated plots for validation.

### 4.3.1 3-Dimensional Localization Setup and Theoretical Result

To create the greatest accuracy within the system, three of the radios are located at ground level, with the z-axis equal to 0, and the two remaining radios are located in the air at an arbitrary positive z value. In the simulations, the two remaining radios are set to heights of 60 meters. The three ground radios and the two airborne radios are perpendicular with each other to create a minimal amount of measurement error. In theory, the computation of the Recursive Least Squares (RLS) localization algorithm should place the drone directly in the center of the system setup, as shown in Figure 4.21. The drone is situated at x and y axes at 0, and at a z height of 30 meters.

Our system averaged 1472 RSS measurements prior to using RLS to derive location. Our 3D localization system yielded a DME mean of $\mathbf{3.2430}\boldsymbol{m}$ and a standard deviation of $\mathbf{1.2644}\boldsymbol{m}$. The result of our system is shown in Figure 4.21.

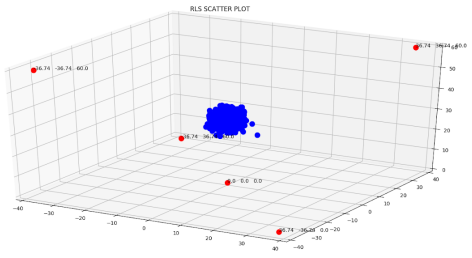

**Figure 4.21:** 3D Simulated Result with Averaging 1472 RSS. This value is chosen because it provides the best localization results without sacrificing much computation time.This is the deployment accuracy of our system.

### 4.3.2 Effects Of RSS Averaging

The system's performance was quantified by the reduction in DME by averaging RSS samples prior to using RLS to derive estimate drone positions. The following figures show the effects of 1 RSS sample averaged, 500 RSS samples averaged, and 1000 RSS samples averaged.The DME Mean and DME standard deviation for each simulation is given in Table 4.2:

| Number of RSS Samples Averaged | DME Mean | DME STD |
|---|---|---|
| 0 | 25.5367 | 12.7854 |
| 500 | 4.9678 | 2.0228 |
| 1000 | 3.7489 | 1.4820 |

**Table 4.2:** Effects of Averaging on Positioning Error

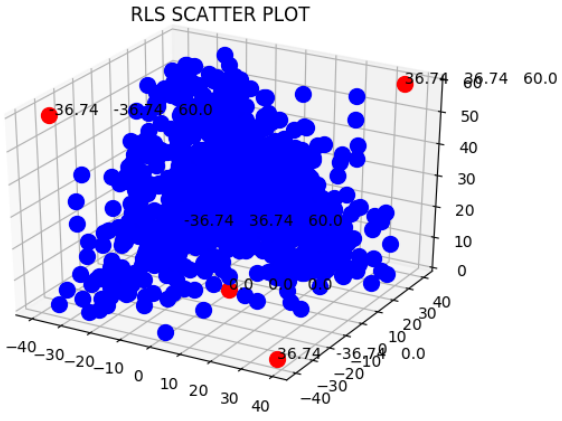


**Figure 4.22:** 3D Localization Averaging 1 RSS Sample.

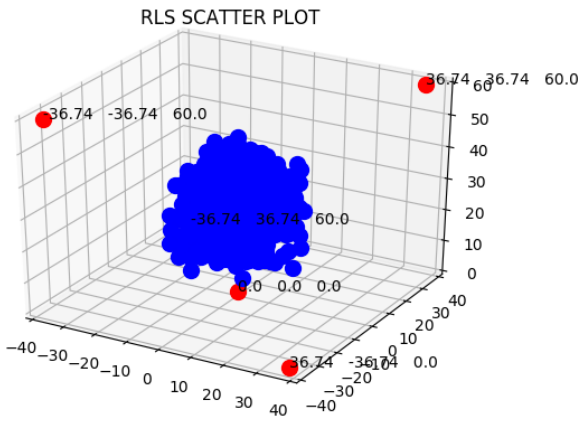


**Figure 4.23:** 3D Localization Averaging 500 RSS Samples.

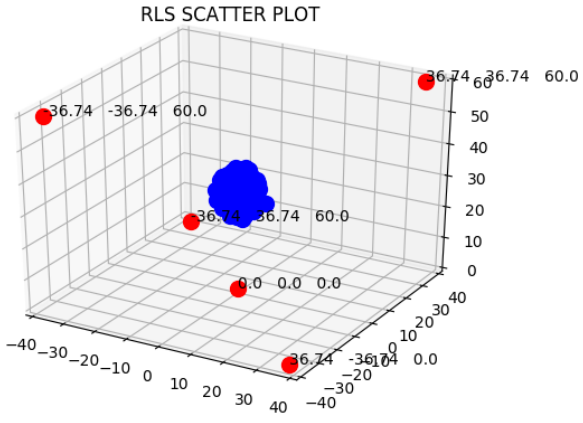


**Figure 4.24:** 3D Localization Averaging 1000 RSS Samples.

Figure 4.25 can be fit to the following equation where x is the number of RSS Samples.

$$DME = 11.7e^{(-0.01256x)} \tag{4.2}$$

The system's accuracy and performance increased by averaging more RSS samples prior to using RLS to estimated drone positions.

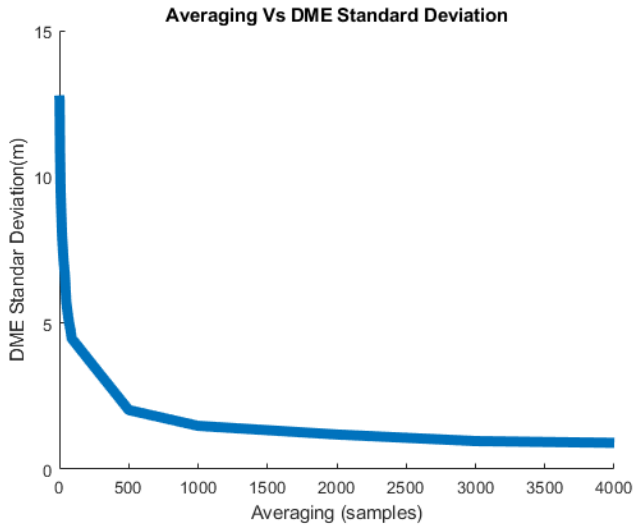


**Figure 4.25:** Averaging vs DME Standard Deviation. DME standard deviation drastically decreases approximately between 0 and 500 averaged samples, and continues to decrease as sample averaging increases.

# 5 CONCLUSION AND FUTURE RECOMMENDATIONS

In an effort to detect and mitigate the threat of unwanted drones, our team designed a RSS-Based 3D localization system utilizing software-defined radio. This project focused on demonstrating proof of concept for RSS localization utilizing software-defined radio. The benefit of an extensively configurable system such as this is that link-layer and waveform-specific changes can be made to the radio platform without having to design additional hardware. The more attractive benefit for the user is that all operations can happen in real time. From a localization engineer standpoint, this allows the capture of vast amounts of data in a short amount of time to feed positioning engines. Our system leveraged averaging for increased precision and accuracy. Our 2D localization system yielded a DME mean of $\mathbf{0.9376}\boldsymbol{m}$ and a standard deviation of $\mathbf{1.4674}\boldsymbol{m}$. Our 3D localization system yielded a DME mean of $\mathbf{3.2430}\boldsymbol{m}$ and a standard deviation of $\mathbf{1.2644}\boldsymbol{m}$. Utilizing software-defined radio also validated that accuracy and performance increased by averaging more RSS samples prior to using RLS to derive location. Additionally, we conclude RLS is limited to the variation in the positioning of radios. For example, 3D localization outdoors with RLS requires high elevation of multiple radios making deployment of system like ours logistically challenging. Future additions to this system would be design of a cognitive radio system capable of calculating path loss in real-time. The ability to calculate distance power gradient in real-time and the variance of shadow fading has the potential to drastically reduce DME. The system we designed can be used as a platform to design rapid deployment systems for adversarial UAV localization. The platform has the ability to develop different location systems using different ranging techniques and estimation algorithms.

# BIBLIOGRAPHY


[1] Li, C., Li, Y., Trian, Z., Weekes, S. and Pahlavan, K. (2019). Design and performance evaluation of a localization system to locate unwanted drones by using wireless signals - IEEE Conference Publication. [online] Ieeexplore.ieee.org. Available at: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8326226 [Accessed 20 Mar. 2019]

[2] PAHLAVAN, K. (2019). INDOOR GEOLOCATION SCIENCE AND TECHNOLOGY. [Place of publication not identified]: RIVER Publishers

[3] Pahlavan, K. and Krishnamurthy, P. (2013). Principles of wireless access and localization. 2013, John Wiley & Sons

[4] Collins, T., Getz, R., Pu, D. and Wyglinski, A. (2018). Software-defined radio for engineers

[5] FAA Aerospace Forecast. (2017). FAA, Forecasts and Performance Analysis Division (APO-100), pp.39- 45.

[6] Pahlavan, K. (2019). UDP identification and error mitigation in toa-based indoor localization systems using neural network architecture - IEEE Journals & Magazine. [online] Ieeexplore.ieee.org. Available at: https://ieeexplore.ieee.org/document/5165323 [Accessed 20 Mar. 2019]

[7] Walsh, N. Gallón, and E. Perez, "The August plot to kill Maduro with drones," CNN, 14-Mar-2019. [Online]. Available: https://www.cnn.com/2019/03/14/americas/venezuela-drone-maduro-intl/index.html. [Accessed: 15-Apr-2019].

WPI