Analysis of the Position-Dependent Error in FTM RTT Indoor Navigation

by

David E. Houle Jr.

B.S. Electrical Engineering and Computer Science
Massachusetts Institute of Technology, 2017

SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING AND COMPUTER SCIENCE
AT THE
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2021

Signature of Author: .....
Department of Electrical Engineering and Computer Science
May 13, 2021

Certified by: .....
Berthold Klaus Paul Horn
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by: .....
Katrina LaCurts
Chair, Master of Engineering Thesis Committee

Analysis of the Position-Dependent Error in FTM RTT Indoor Navigation

by

David E. Houle Jr.

Submitted to the Department of Electrical Engineering and Computer Science
on May 13, 2021 in Partial Fulfillment of the
Requirements of the Degree of Master of Engineering in
Electrical Engineering and Computer Science

ABSTRACT

Fine time measurement (FTM) of the round-trip time (RTT) of a signal between an initiator (smartphone) and a responder (Wi-Fi access point) provides a promising method for indoor positioning. Accurate indoor positioning is a requirement for a wide range of applications, such as asset tracking, indoor navigation, and contact tracing. Unfortunately, the error of reported FTM RTT distance measurements has been shown to have a standard deviation that ranges from 1-2 meters in ideal setups. A major FTM RTT error source was discovered and coined as the "position-dependent error". This error is heavily depend on the position of an initiator relative to a responder, with the reported measurement fluctuating by meters from an initiator position change of millimeters. Using an Android app and a CNC machine for 2D and 3D positioning, these unusual error properties are explored in depth through experimentation. This experimentation includes evaluating the position-dependent error in both the spatial and frequency domains when varying the test setup, using different smartphones and Wi-Fi access points, and changing the bandwidth and central frequency of the Wi-Fi access points. Possible causes of the position-dependent error are analyzed, such as inaccurate time of arrival or super-resolution algorithms, a dependence on received signal strength, and clock instability. In the end, recommendations for error amelioration are made, and the future of FTM RTT is discussed.

Thesis Supervisor: Berthold Klaus Paul Horn
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

I would like to thank my MEng thesis supervisor, Berthold K.P. Horn, for all his work and guidance throughout the program and for his help in the execution of this project. He has been especially supportive as we have been researching remotely during the COVID-19 global pandemic, and I cannot thank him enough for his dedication to the pursuit of knowledge through these unprecedented times.

Contents

List of Figures

Chapter 1

Introduction

Accurate indoor positioning systems have a wide range of potential applications such as asset tracking, indoor navigation, robot control, augmented reality, and contact tracing. Unfortunately, due to signal loss through building materials, GPS does not provide sufficient position accuracy. Instead, other possible indoor positioning systems have been proposed, using techniques such as received signal strength, fingerprinting, angle of arrival, and Fine Time Measurement of Round-Trip Time. Ultimately, Fine Time Measurement of Round-Trip Time shows promising results and warrants further exploration.

Fine Time Measurement of Round-Trip Time (FTM RTT) was first specified in the 2016 version of IEEE 802.11 Wi-Fi standard, otherwise known as IEEE 802.11mc [4]. FTM RTT measures the time that it takes a signal to travel round-trip between a smartphone and a Wi-Fi Access Point (AP). The distance from the AP to the smartphone is calculated based on the signal's propagation in air. Unfortunately, the errors of these FTM RTT distance measurements have been found to have a standard deviation of 1-2 m, even in favorable setups, which may limit potential applications [5].

This research explores one of the major contributions of the FTM RTT error, which has been coined the "position-dependent error" by Berthold Horn [5]. This position-dependent error depends on the position of the smartphone, the frequency and bandwidth of the AP, and other factors. It has been shown that small millimeter

movements of the smartphone in one dimension result in large meter jumps in distance measurements. While the position-dependent error has been discovered, it has yet to be viewed in more than 1D and fully analyzed.

Data for this research was collected using an Android app that was synchronized with the motion of a controllable CNC machine. This CNC machine was used to position the smartphone over a maximum volume of 0.7 m x 0.7 m x 0.08 m with a resolution of 0.075-0.13 mm.

Using the CNC machine, this research explores the position-dependent error in both 2D and 3D, while also evaluating how changes in the experimental setup affect the error. These changes include using different smartphones, using different Access Points, changing the bandwidth and frequency of the AP's signal, and varying the amount of multipath propagation of signals in the scene. Analysis will then be done to try and understand the possible causes of the position-dependent error and how it can be ameliorated.

In Chapter 2, we provide a background of Wi-Fi FTM RTT and discuss related work. In Chapter 3, we display the results of various experiments that were run to better understand the position-dependent error in 1D, 2D, and 3D. In Chapter 4 we dive into the possible sources of the position-dependent error, while in Chapter 5 we explore some ways to lessen the effect of the error. Finally, in Chapter 6 we summarize information about the position-dependent error and draw conclusions from what was discovered. Additionally, we discuss next steps and future work.

Chapter 2

Background and Related Work

Indoor positioning is a widely studied field, since there are a vast range of applications. A few of the current techniques used for positioning are received signal strength, fingerprinting, angle of arrival, and Fine Time Measurement of Round-Trip Time [6].

The received signal strength (RSS) method uses a path loss function [7]. In ideal circumstances or free space, this method has potential, as signal strength should vary as $1/d^2$, where d is the distance between the AP and smartphone. Unfortunately, non-line of sight conditions, interference, and other factors result in insufficient accuracy [3]. Some research is being done to increase the accuracy of RSS only based methods [8].

Fingerprinting maps RSS values from *several* source APs over a desired area [7]. This method requires a lot of work up front to map the space as RSS values depend heavily on position and environment. Additionally, it is necessary to remap an area if any AP is moved, making it less than ideal. Even with its limitations, there is still a substantial amount of research being done with the fingerprinting method by combining it with various other positioning techniques [7, 9, 10, 11].

By calculating the angle of arrival of a signal from a smartphone, APs can pinpoint where the smartphone is located [12]. High angular resolution is necessary, which requires many antennas at longer distances, as angular resolution is inversely proportional to the number of antennas. This method also requires position determination to be done by the AP instead of the smartphone and limits its current

applications.

Additionally, there are many indoor positioning methods attempting to use various sensors on the smartphones to understand the user's position and orientation. Some of these methods are dependent on line of sight paths from the smartphone to external sensors, the smartphone's camera for object detection, or the smartphone's inertial measurement units for pedestrian dead reckoning [6, 13, 14, 15, 16, 17]. These types of methods may have limited applications as users may act inconsistently with their smartphone or block its camera view. Another potential issue is that new external sensors may be needed such as ultrasonic sensors, which would require a large infrastructure change.

Fine Time Measurement of Round-Trip Time (FTM RTT) measures the round-trip time of signals between an AP and a smartphone. It was first included in IEEE 802.11mc in 2016, but has only been widely available on the Android platform since 2018. One issue with using Wi-Fi APs is that in general they are not frequently replaced. As a result, many of the APs in use today do not support FTM RTT as specified in IEEE 802.11mc. Because of these drawbacks, FTM RTT research is still in the earlier stages. Experimentation has shown measurement errors with a standard deviation of 1-2 m [5].

One source of FTM RTT error that has been explored is the offset error. Specific combinations of APs and smartphones produce a different, but consistent, offset that needs to be calibrated to improve accuracy. The same AP and smartphone pair will also require offset calibration for each available channel that an AP may switch to during operation. Calibrating for this offset error could require a lot of work upfront, since some AP offsets range from -10 m to 10 m.

Overall, research into the FTM RTT error has yet to provide sufficient accuracy for some applications. It has been shown that averaging many measurements at a specific location does not make significant improvements in overall accuracy. This is most likely due to a major source of error in FTM RTT that was recently discovered. As mentioned in Chapter 1, Berthold Horn [5] coined this the "position-dependent error". He found large, repeatable changes in the measurement error as the smartphone

position varied over a single dimension. The position-dependent error has been shown to be much larger than any measurement noise, and it is worth determining the root cause. Up to this point, no one has analyzed it in depth. In fact, some FTM RTT research is being done without acknowledging that this error even exists [18]. That is why this research focuses on determining the cause of the position dependent error, so that it can be more widely understood and accepted.

There are a few possible causes of the position-dependent error that will be discussed in detail in Chapter 4. These are time of arrival algorithms used for estimating when a signal reaches a receiver, super-resolution algorithms used for gaining a higher signal resolution than attainable at a given signal bandwidth and sampling rate, received signal strength which seems to have some correlation to FTM RTT measurements, and internal clock stability which may produce errors if it is unstable.

Once the distance from a group of APs has been calculated, triangulation of the smartphone is required. If the smartphone is confined to a single floor, the problem can be formulated in 2D, whereas if there are multiple floors, it becomes 3D. There are some deterministic approaches to positioning such as using a least squares minimization combined with search. Since the AP measurements are noisy, there are typically a range of smartphone position candidates with varying probabilities. Thus, there are plenty of stochastic approaches to smartphone position determination such as Kalman filtering, a particle filter, and Bayesian grid update [9, 11, 19]. Since the AP's measurement noise is shown to be non-Gaussian and dependent on position, many of the stochastic model assumptions do not hold. Bayesian grid update, which uses a combination of an observation and transition model, has encouraging results [20].

Chapter 3

Measuring the Position-Dependent Error

This chapter covers various ways that the FTM RTT error has been measured through experimentation. It starts by laying out the possible error sources, and then dives into the position-dependent error. The position-dependent error experiments include using the Google Pixel 4 smartphone with the Google Wi-Fi APs, comparing Round-Trip Time and received signal strength, examining spatial frequency content, testing dense and open-space setups, comparing Google Pixel 4 and Google Pixel 3 smartphones, comparing Google Wi-Fi APs and ASUS APs, and varying the APs' frequency and bandwidth.

3.1 FTM RTT Error Sources

As discussed in Chapter 2, the 1-2 m standard deviation of the FTM RTT error could stem from various sources. Each of the error sources could contribute in part to the large overall error, but this section will rule out some of these hypotheses.

The first potential error source is simply measurement error or noise. By leaving the smartphone and APs in static positions, Fig. 3-1 shows repeated FTM RTT measurements with a standard deviation of 0.15-0.3 m for the Google APs and a standard deviation of 0.45 m for the ASUS AP. While there was a line of sight between

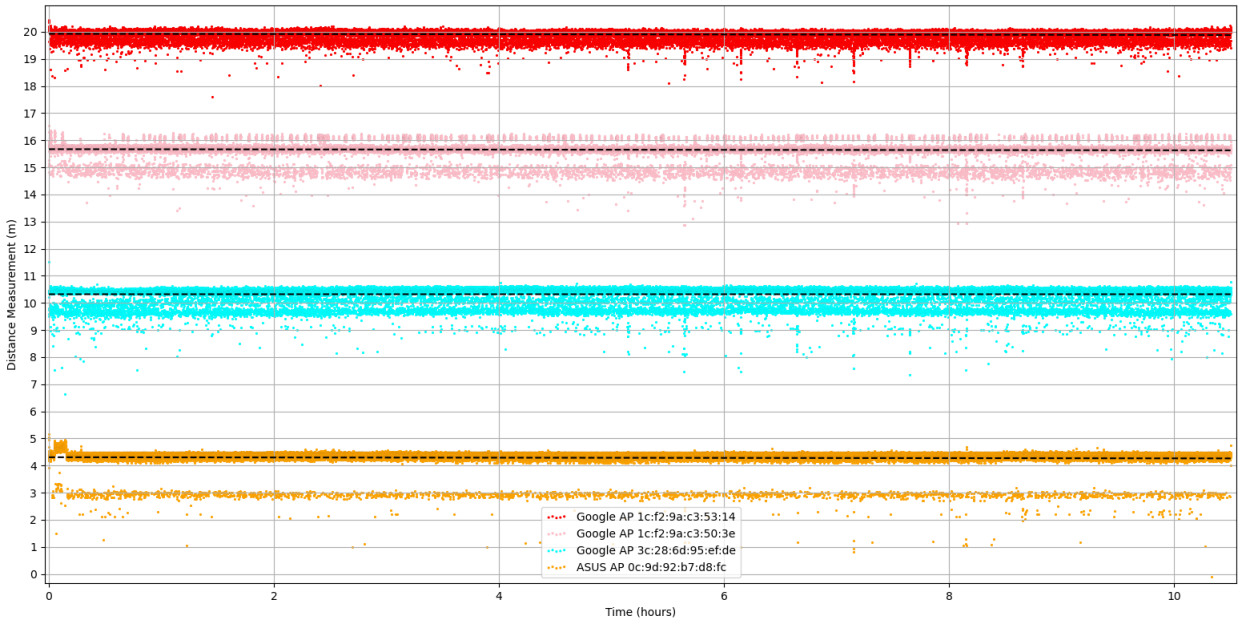


Figure 3-1: Repeated FTM RTT measurements for various APs. Measurements were offset vertically to avoid overlap from different APs.

the smartphone and APs, these measurements were collected in a less than ideal, dense and cluttered setup, where multipath propagation was prevalent.

While this amount of measurement error or noise has some contribution to the overall error standard deviation that can be seen in the ranging measurements, by taking the median of several measurements, the standard deviation of the FTM RTT measurements shrinks even further. Fig. 3-2 shows the results from a sliding window that calculated the median of 5 consecutive FTM RTT measurements from the same data collection. In this case, the standard deviation decreased to 0.08-0.15 m for the Google APs and 0.15 m for the ASUS AP. Thus the measurement error or noise has a relatively small contribution to the overall error which has a standard deviation of 1-2 m in favorable setups.

Additionally, these measurements show that there is very little time component to the FTM RTT error. The magnitude of the error at a single position stays constant over hours of testing. The slight exception to this in Fig. 3-1 and Fig. 3-2, between hours 5 and 9. At almost exactly 30 minutes apart, there is a sudden drop in the FTM RTT measurement distance by 1-2 m. Because the error can be seen at identical

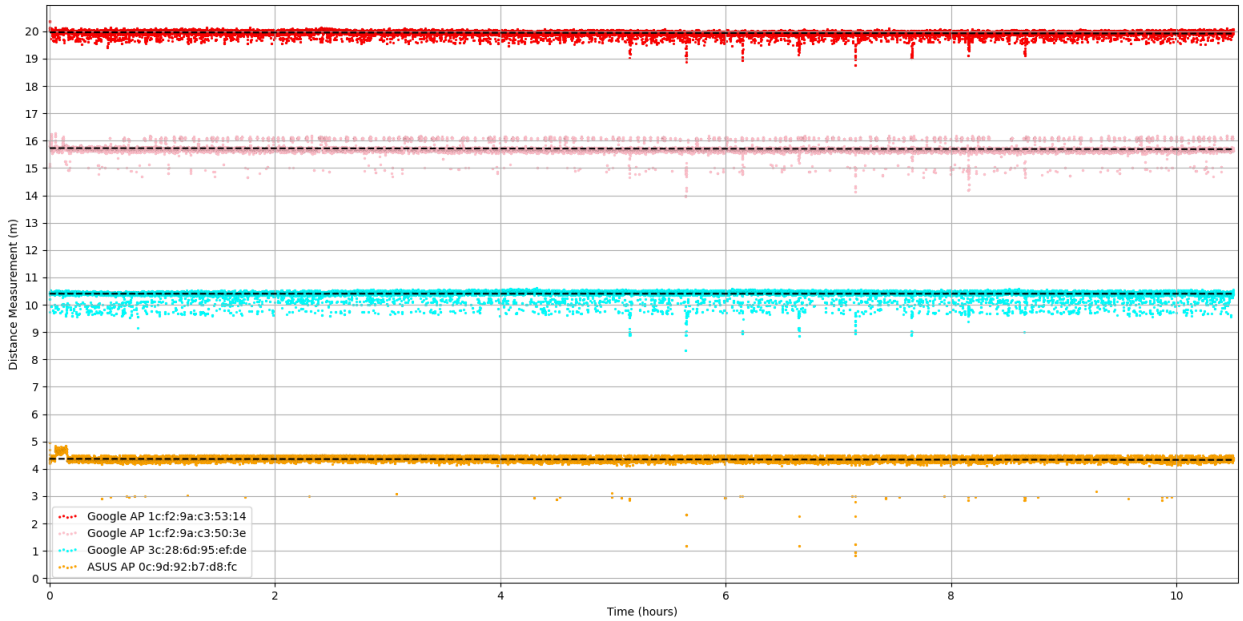
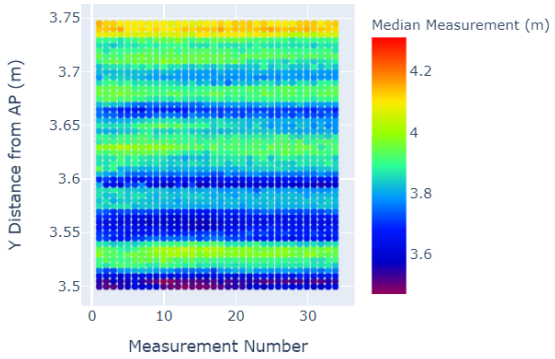


Figure 3-2: Median of repeated FTM RTT measurements for various APs. Each point represents the median of 5 consecutive FTM RTT measurements, which were obtained using a sliding window from the same data collection as Fig. 3-1. Measurements were offset vertically to avoid overlap from different APs.

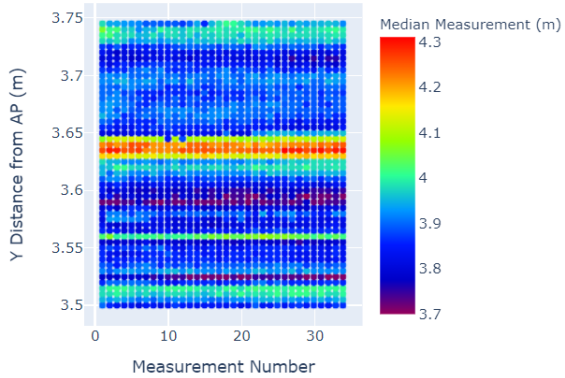
times in the Google Wi-Fi APs and the ASUS AP measurements, it does not appear to be related to the mesh networks communicating. Instead, it may be a result of the furnace turning on in consistent intervals. Overall, the effect is minimal on the standard deviation of the measurements and is also infrequent.

To reinforce the absence of a time dependent error, Fig. 3-3 shows a 1D collection of 50 positions that was repeated 35 times consecutively. Each colored point on the graph represents the median smartphone FTM RTT measurement at a given distance from the AP. Since positions are repeated, if there were a time component to the error, you would expect that the color would be inconsistent as you change the measurement number but remain at a constant Y distance from the AP. In this case, the color is consistent as you increase the measurement number, showing the absence of a time component of the error, and the calculated standard deviations support that claim. At all 50 Y distances from the AP, the standard deviation of the 35 measurement numbers only varied from 0.02-0.2 m, 0.01-0.06 m, and 0.02-0.2 m for Google APs 1c:f2:9a:c3:50:3e, 1c:f2:9a:c3:53:14, and 3c:28:6d:95:ef:de, respectively. Since these

(a) Google AP 1c:f2:9a:c3:50:3e



(b) Google AP 1c:f2:9a:c3:53:14



(c) Google AP 3c:28:6d:95:ef:de

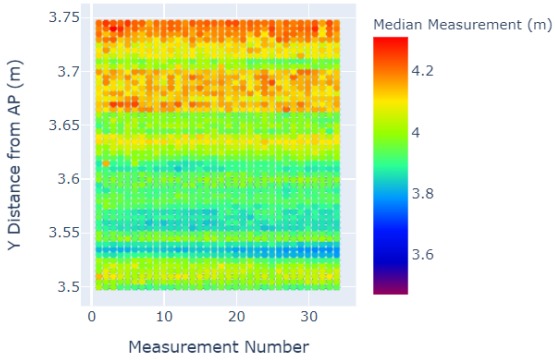


Figure 3-3: 3D plots for each Google AP exploring any time-dependent error. Vertical axis: 1D distance between the AP and smartphone at 50 different distances, each separated by 5 mm. Horizontal axis: repeated collection of those 50 measurements 35 times. Color Bar: the median value of about 20 FTM RTT measurements at any given distance.

measurements were taken over the course of 5 hours, this again shows that any time related error is relatively small or nonexistent when compared to the total FTM RTT error, which has a standard deviation of 1-2 m in favorable setups.

While the error has little to no influence from time and noise, there is still a very large error overall. Running various tests shows that as the smartphone moves across several positions, the reported FTM RTT measurements vary greatly. This can be seen in a single dimension in figure Fig. 3-4. As the distance between the APs and smartphone increases linearly in 1 mm increments, the reported distance value fluctuates by up to 5 m in the worst case. As mentioned in Chapter 2, this overwhelming

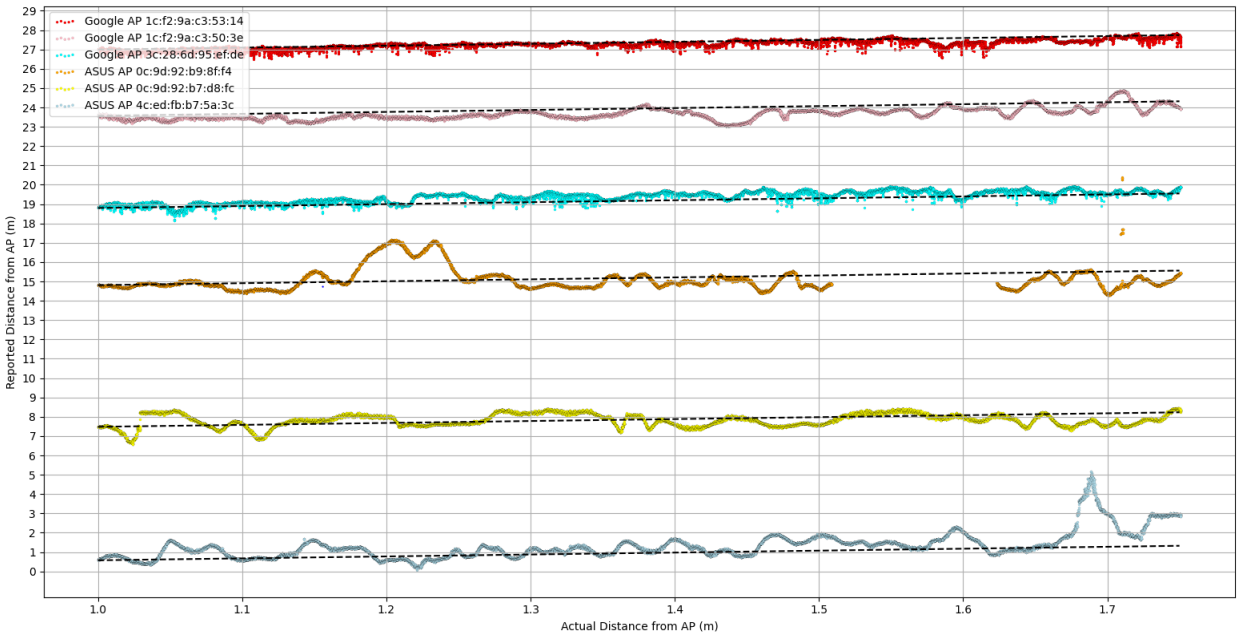


Figure 3-4: 1D position-dependent error result. The overlaid dashed black lines represent the actual distance from the smartphone to the APs.

position-dependent error warrants further investigation. In the following sections, the position-dependent error will be explored in multiple dimensions.

3.2 Google Pixel 4 and Google Wi-Fi AP

Before introducing the position-dependent error, it is important to know the devices and setup that was used. The experiments were conducted with a Google Pixel 4 smartphone in combination with 3 Google Wi-Fi APs, whose MAC addresses are 1c:f2:9a:c3:50:3e, 1c:f2:9a:c3:53:14, and 3c:28:6d:95:ef:de. Each AP's signal had a central frequency that was set to 5210 MHz and a bandwidth of 80 MHz. Those will be the standard signal settings for all experimentation unless otherwise noted. The FTM RTT measurements were collected at 19,600 positions over a 2D, 0.7 m x 0.7 m area with increments of 5 mm between each position. Fig. 3-5, Fig. 3-6, and Fig. 3-7 display the results.

While each Google AP produces a slightly different pattern, the textures of the 2D space are remarkably similar. As the Y distance from the AP increases, some

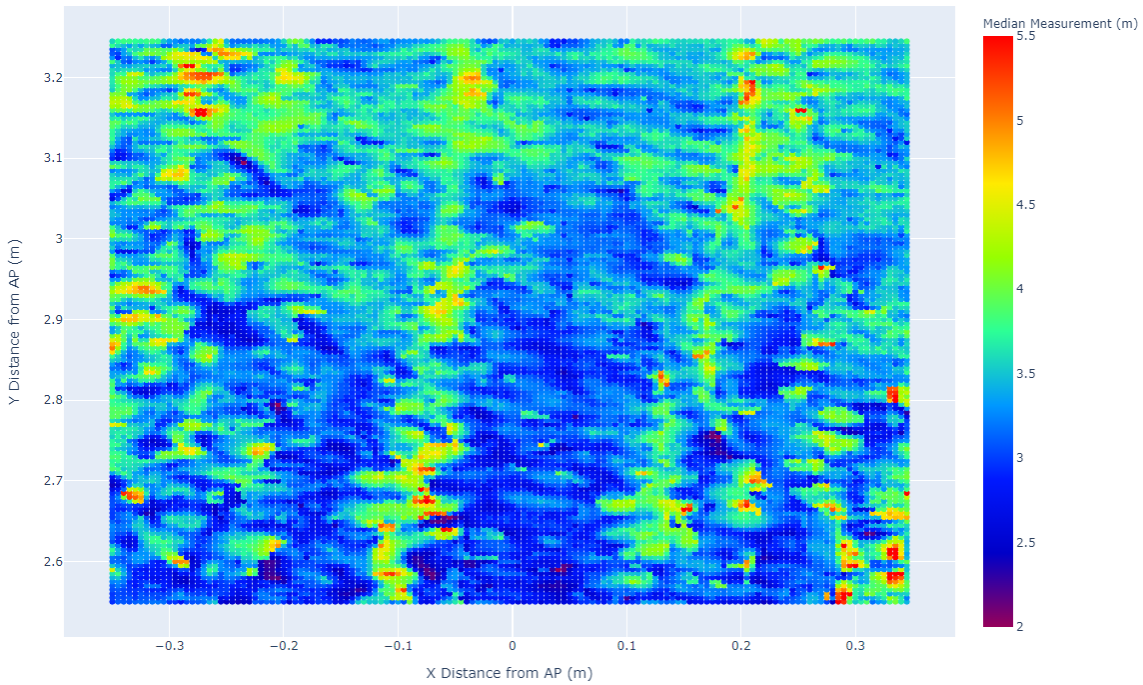


Figure 3-5: 2D position-dependent error plot using Google Pixel 4 and Google Wi-Fi AP 1c:f2:9a:c3:50:3e. The AP was positioned at (X, Y) = (0, 0). Color Bar: Median reported FTM RTT measurement at a given position.

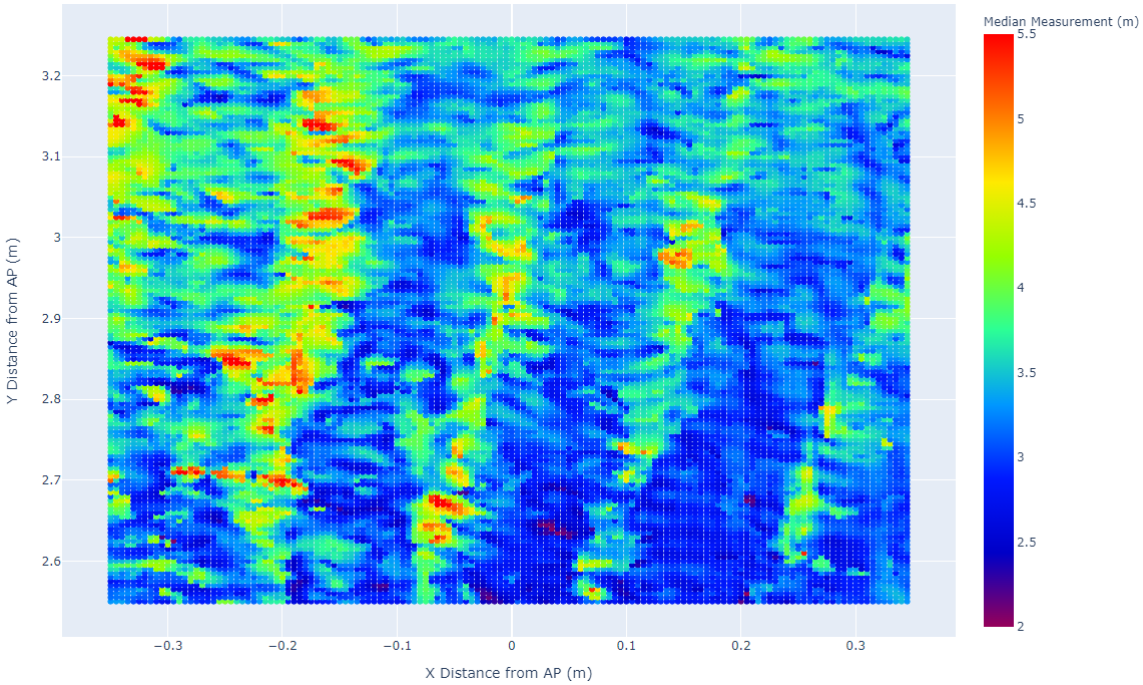


Figure 3-6: 2D position-dependent error plot using Google Pixel 4 and Google Wi-Fi AP 1c:f2:9a:c3:53:14. The AP was positioned at (X, Y) = (0, 0). Color Bar: Median reported FTM RTT measurement at a given position.

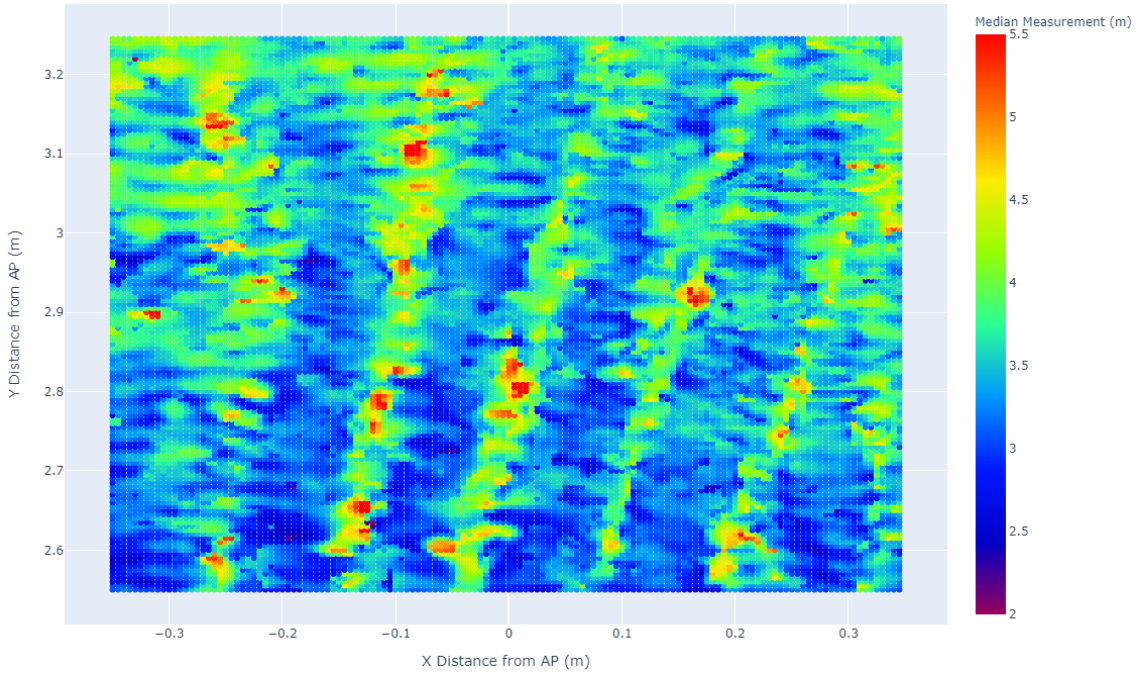


Figure 3-7: 2D position-dependent error plot using Google Pixel 4 and Google Wi-Fi AP 3c:28:6d:95:ef:de. The AP was positioned at (X, Y) = (0, 0). Color Bar: Median reported FTM RTT measurement at a given position.

vertical, smaller wave textures appear, with varying widths of around 0.025 m. As the X distance from the AP increases, some horizontal, larger wave textures can be found, with varying widths of around 0.1 m. These horizontal, larger wave textures are wider and have a greater range of reported measurements when compared to the vertical, smaller wave textures. Additionally, these horizontal, larger wave textures seem to move at a slight angle relative to the AP. The frequency of these wave textures will be explored in more depth in Chapter 3.4.

The position-dependent error has proven to span all 3 dimensions through experimentation. Fig. 3-8 shows FTM RTT measurements when varying the position in the Z dimension with a fixed Y position of 2.9 m. Unfortunately, the machine used for smartphone positioning was constrained to 80 mm vertically, so the results show a limited picture. Even with the limited picture, the familiar wave textures appear, showing that the error extends to 3D.

The visualizations of FTM RTT measurements in two and three dimensions start to provide a better picture of why the position-dependent error may exist in the first

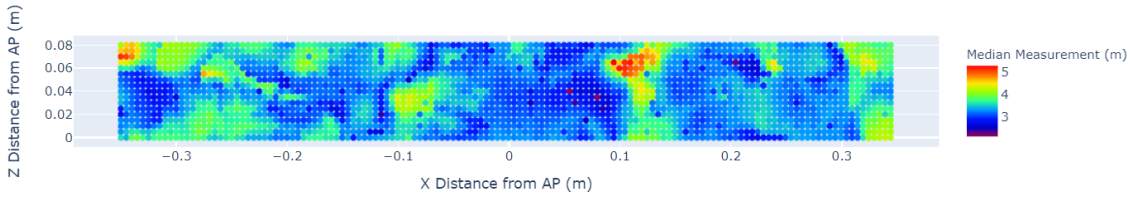


Figure 3-8: Extending the position-dependent error to 3D by varying the X and Z dimensions using Google Pixel 4 and Google Wi-Fi AP 1c:f2:9a:c3:53:14. The AP was positioned at (X, Y, Z) = (0, 0, 0). The smartphone was positioned at Y = 2.9 m. Color Bar: Median reported FTM RTT measurement at a given position.

place. Similar to the 1D case of the position-dependent error, moving the Google Pixel 4 smartphone by only a few centimeters can result in a large jump of up to 3m in reported FTM RTT measurement. The 2D plots show that there is much more structure and consistency to the error than could be seen in the 1D case. The following section will continue to explore the 2D position-dependent error by comparing the measurement plots to the received signal strength of the smartphone.

3.3 Comparison of the Round-Trip Time vs. Received Signal Strength

Since Google's exact signal time of arrival algorithms for FTM RTT measurements are unknown, comparing the 2D results against other parameters may shed some light on how they are actually calculated. The first comparison that can be done with the received signal strength (RSS) of the smartphone. Along with each estimated FTM RTT measurement, the Android API outputs the RSS value measured at the smartphone's antenna. Fortunately, this allows us to display the results of FTM RTT and RSS for the exact same collection at the exact same time.

The resulting plots of RSS are from the same data collection from Chapter 3.2 above. Fig. 3-9, Fig. 3-10, and Fig. 3-11 correspond exactly to Fig. 3-5, Fig. 3-6, and Fig. 3-7. The figures have a strikingly similar structure, again with vertical, smaller wave textures as the Y distance from the AP increases, and with horizontal, larger wave textures as the X distance from the AP increases. Unfortunately, directly

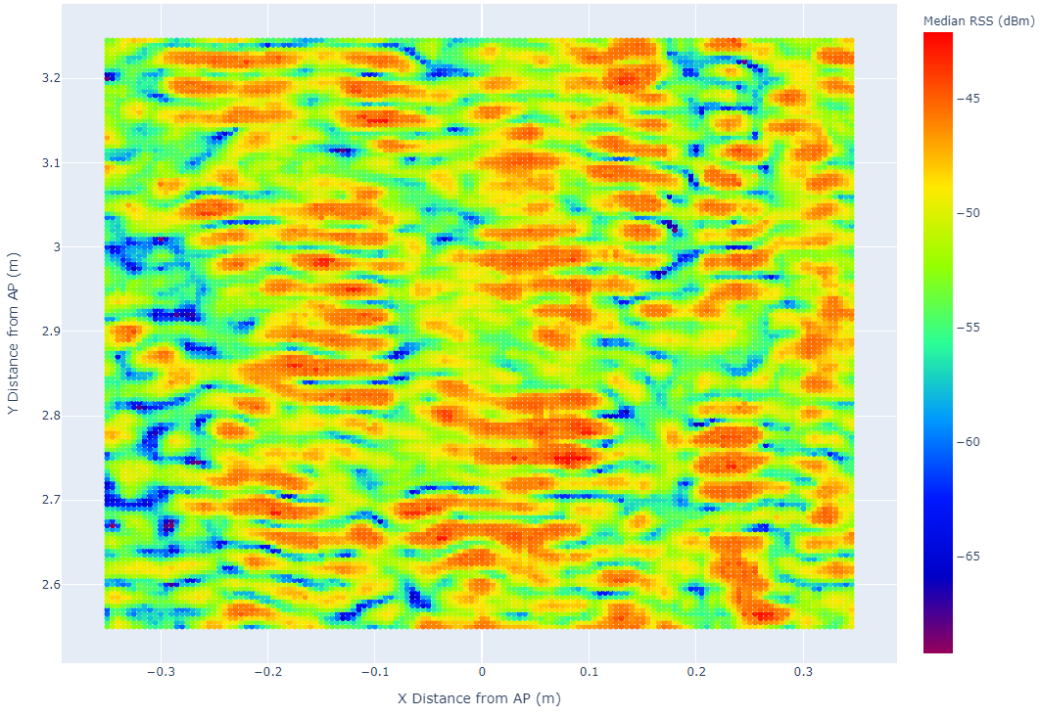


Figure 3-9: 2D RSS plot using Google Pixel 4 and Google Wi-Fi AP 1c:f2:9a:c3:50:3e. The AP was positioned at (X, Y) = (0, 0). This plot is from the same data collection as Fig. 3-5. Color Bar: Median reported RSS at a given position.

calculating the Pearson correlation coefficient between each RSS and FTM RTT plot results in small values of 0.005-0.02, meaning that the measurements themselves have little correlation. This does not mean that there is not some sort of relationship between RSS and FTM RTT. Chapter 3.4 will explore the frequency analysis of the FTM RTT and RSS plots, since the wave textures may correspond to similar frequencies.

Looking at the similarity between the FTM RTT and RSS plots can start to provide some insight into how the position-dependent error varies. RSS may fluctuate due in large part to path loss, or the reduction in power density as the signal propagates through space. Path loss has various causes such as free-space loss from natural expansion of the radio wave, diffraction path loss due to obstruction, and absorption path loss from medium that are not transparent to electromagnetic waves [21]. In this case, multipath propagation of signals, where the transmitted AP signal traveled along non-line of sight (NLOS) paths to the smartphone, are most likely the culprit

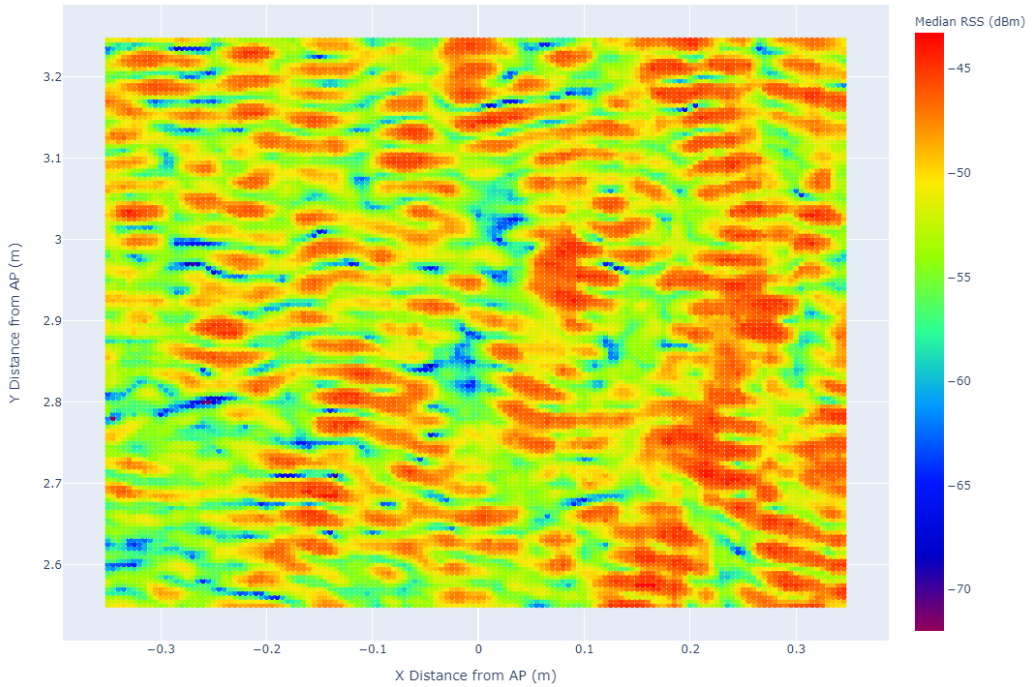


Figure 3-10: 2D RSS plot using Google Pixel 4 and Google Wi-Fi AP 1c:f2:9a:c3:53:14. The AP was positioned at (X, Y) = (0, 0). This plot is from the same data collection as Fig. 3-6. Color Bar: Median reported RSS at a given position.

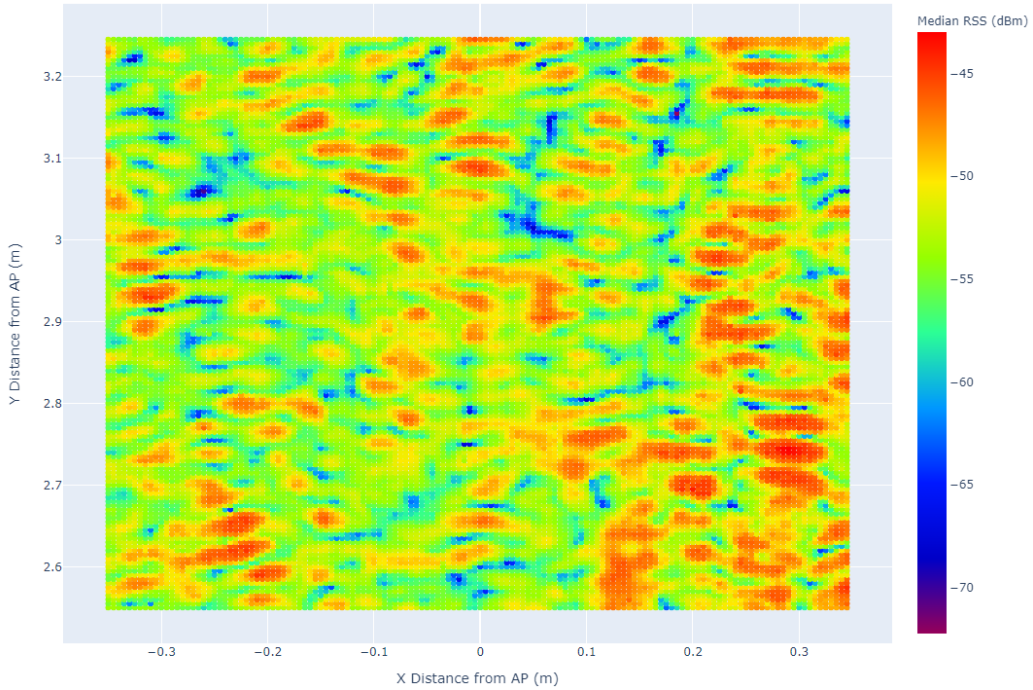


Figure 3-11: 2D RSS plot using Google Pixel 4 and Google Wi-Fi AP 3c:28:6d:95:ef:de. The AP was positioned at (X, Y) = (0, 0). This plot is from the same data collection as Fig. 3-7. Color Bar: Median reported RSS at a given position.

of the fluctuating RSS. This diffraction path loss results in lower RSS than expected for a signal through air. Because the RSS and FTM RTT 2D textures were so similar, and since the diffraction path loss due to multipath propagation of signals may be strongly influencing the RSS, it is fair to say that the multipath propagation of signals could also be strongly influencing the FTM RTT position-dependent error.

3.4 Analyzing the Spatial Frequency Content

Unfortunately, the spatial domain can only provide us with a certain amount of intuition. We can visually see that there are similarities in the texture between the RSS and FTM RTT 2D plots, but since the Pearson correlation coefficient was close to 0, it is hard to extract any specific relationships between the plots in the spatial domain. Fortunately, since each RSS and FTM RTT plot had a texture that was visually made up of similar waves at seemingly the same frequency, looking at the frequency domain of the plots can provide more insight into the content of the waves.

The spatial domain of the image can be transformed into the frequency domain through a 2D Discrete Fourier Transform (DFT). In order to do so, the image was first preprocessed with a Bartlett windowing function where the pixel values gradually reduce to zero at the edges. This is done so that there are no discontinuities in the periodic image, which would distort the Fourier spectrum [22]. There are still discontinuities in the first derivative, however. This results in the frequency domain components decaying as $1/f^2$. The frequencies can be shifted so that the lower frequencies are in the center of the image, with higher frequencies towards the edges. Finally, the Power Spectral Density (PSD) can be calculated by squaring the resulting frequency domain image.

The result of this DFT and processing is shown in Fig. 3-12. By looking at the PSD plot of the FTM RTT and RSS measurements, similar hot spots show up in the area of $(u, v) = (0, \pm 0.035)$. These hot spots represent the dominant frequencies that can be seen in the original spatial domain images. The frequency domain value of $0.035\,(1/\text{mm})$ represents a wave every $28.5\,\text{mm}$ in the spatial domain. This lines up

visually with the vertical, smaller wave textures that can be seen in the original RSS and FTM RTT images, where there are about 3.5 vertical waves per 0.1 m.

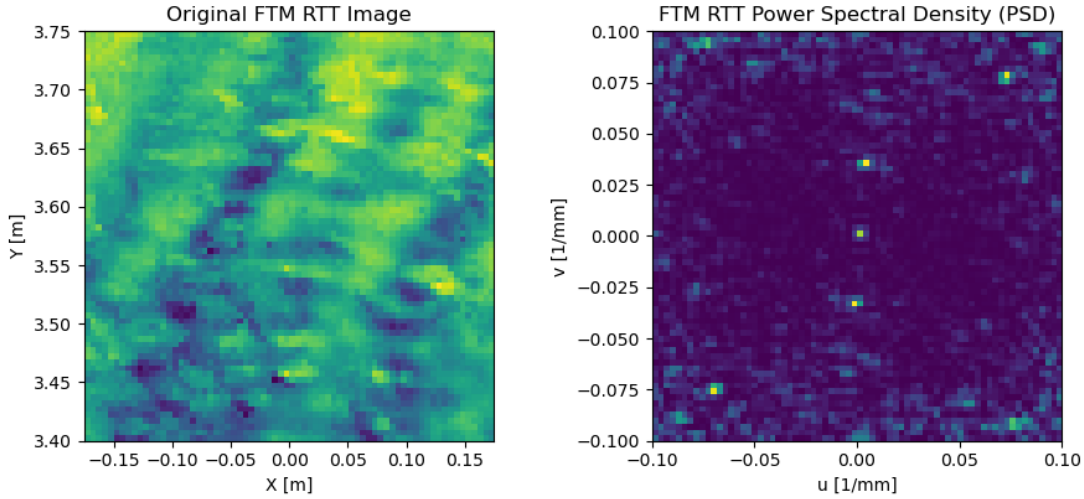
Additionally, these hot spots seem to be related to the wavelength of the signal. The frequency of the signal was 5210 MHz, which corresponds to a wavelength of 57.5 mm in air. The vertical waves were calculated to be 28.5 mm, or almost exactly half the value of the wavelength. In Chapter 3.8, we will dive deeper and see that the waves in the image actually scale with the frequency of the signal. Overall, this relationship between wavelength, RSS frequency, and FTM RTT frequency shows that the algorithms used to calculate FTM RTT may be influenced by similar phenomena. It is likely that the algorithms are heavily dependent on line of sight paths and minimal multipath propagation of signals.

It is important to note that while the frequency hot spots consistently appeared in the RTT image, they did not always appear as strongly for the FTM RTT image as they did in Fig. 3-12. Fig. 3-13 shows the resulting transformation from the same data collection as Fig. 3-6. Note that the data was collected over a larger area, so the resolution of the images is greater. These results show the RSS frequency domain has a hot spot in the same position as before, but the FTM RTT frequency domain does not show such a strong hot spot. Nevertheless, in every case that a hot spot appeared in the FTM RTT frequency domain, it was always strongest at the identical position of the RSS frequency domain and at the vertical position of one half the wavelength of the signal.

3.5 Varying the Testing Setup

Another way of learning more about the position-dependent error, is to study multiple test setups. In this case, we will explore the difference between a dense and an open-space testing setup. In the dense setup, there was a line of sight between each AP and the smartphone, but the measurements were taken in a small room with less than a meter of space between the walls and the testing devices. In the open-space setup, there were several meters between the walls and the testing devices, although

(a) FTM RTT spatial and frequency images with frequency hot spots from Google AP 1c:f2:9a:c3:53:14



(b) RSS spatial and frequency images from Google AP 1c:f2:9a:c3:53:14

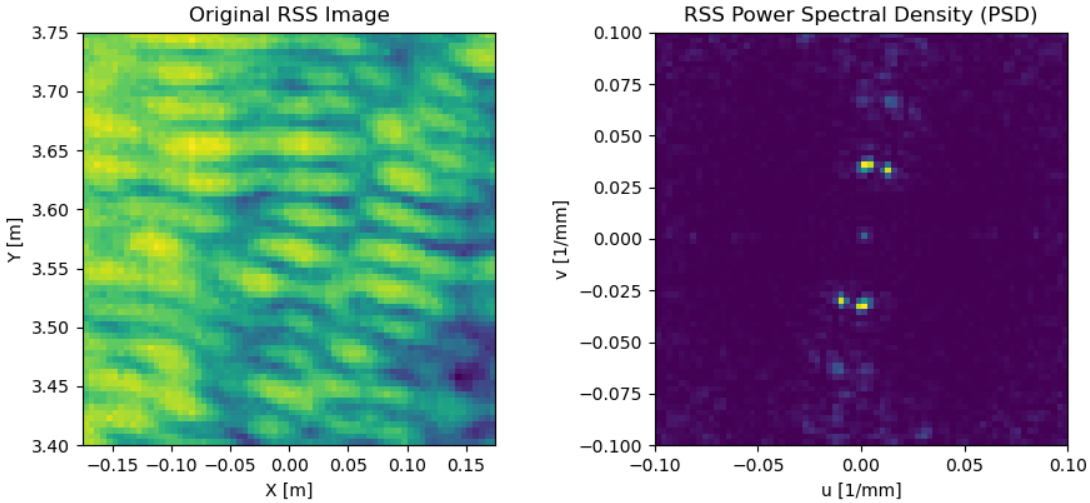
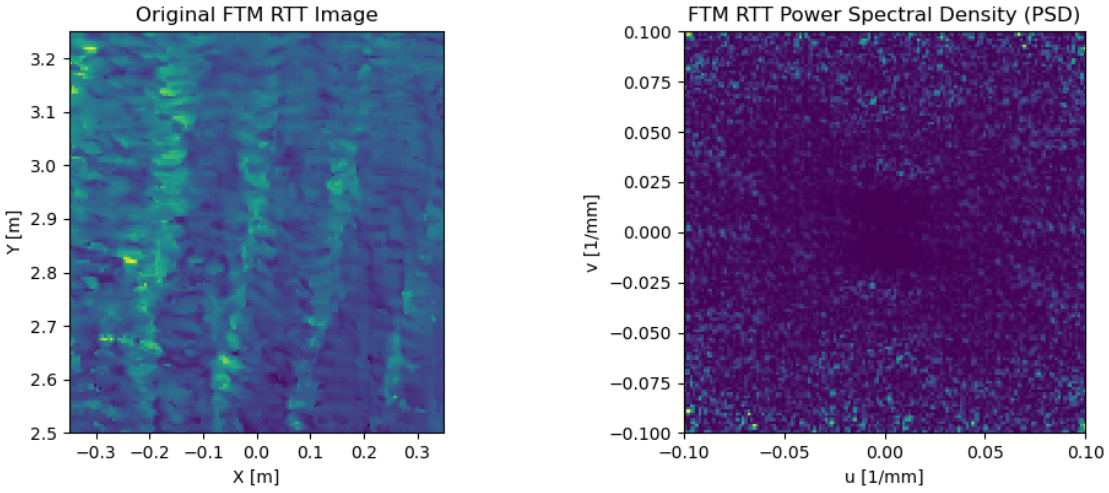


Figure 3-12: Calculated PSD showing the frequency domain of the RSS and FTM RTT measurements from the same data collection using Google AP 1c:f2:9a:c3:53:14. Sub-figure (a) shows the spatial and frequency domains of the FTM RTT measurements, while Sub-figure (b) shows the spatial and frequency domains of the RSS measurements. In (a) and (b), the hot spots in the frequency domain show up in the same position.

(a) FTM RTT spatial and frequency images without frequency hot spots from Google AP 1c:f2:9a:c3:53:14



(b) RSS spatial and frequency images from Google AP 1c:f2:9a:c3:53:14

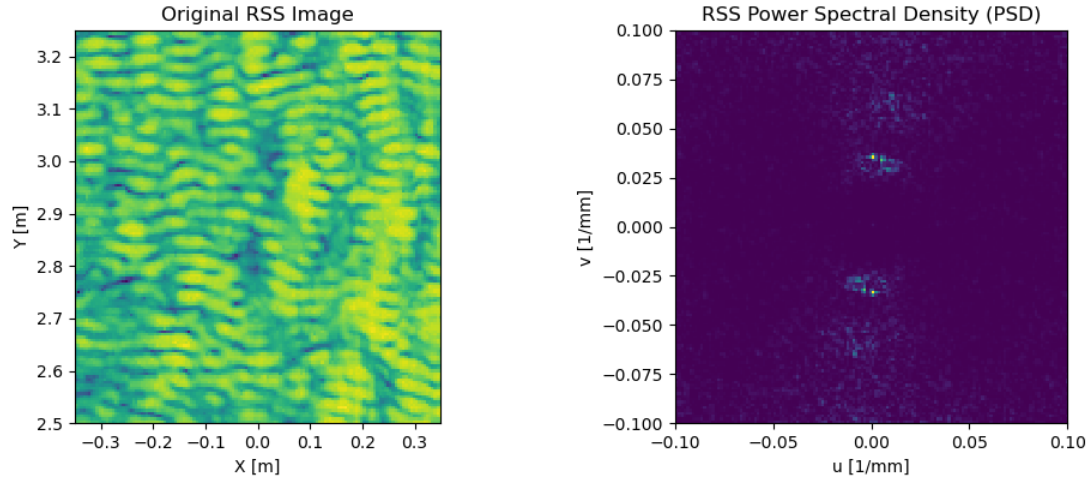


Figure 3-13: Calculated PSD showing the frequency domain of the RSS and FTM RTT measurements from the same data collection using Google AP 1c:f2:9a:c3:53:14. Sub-figure (a) shows the spatial and frequency domains of the FTM RTT measurements, while Sub-figure (b) shows the spatial and frequency domains of the RSS measurements. Sub-figure (a) shows limited hot spots in the frequency domain compared to the separate data collection in Fig. 3-12.

the devices were only raised 0.25 m above the concrete ground.

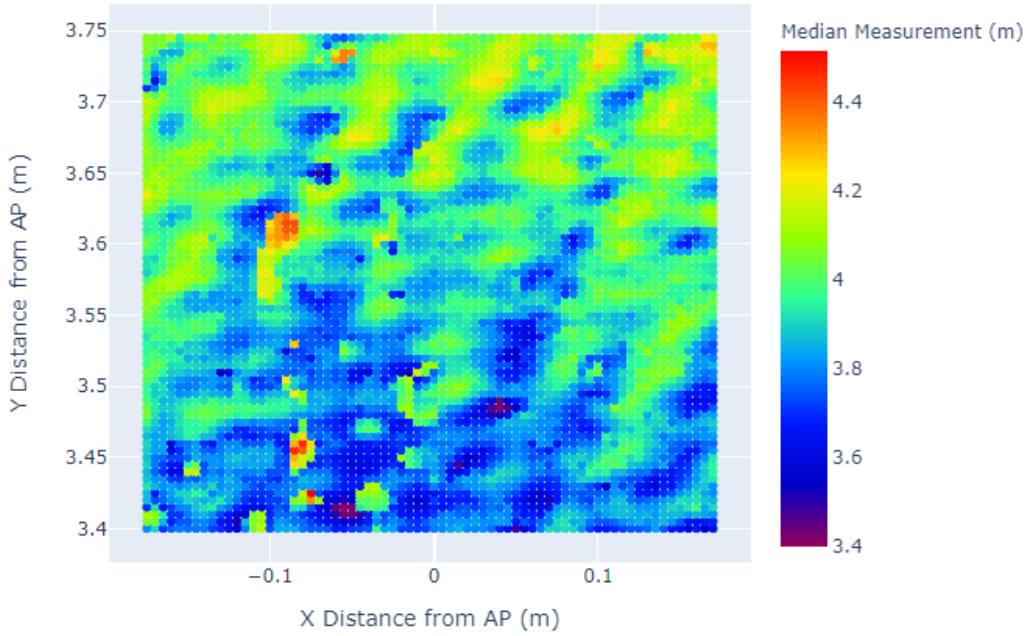
Fig. 3-14 shows examples of FTM RTT measurements with open-space and dense setups. In both plots, similar wave textures appear. In the open-space setup shown in Fig. 3-14a, we can see that the reported FTM RTT measurements scale better with Y distance from the AP, and that the peaks and troughs of the waves are less extreme. The color bar shows that the median FTM RTT measurements only vary by about 1 m over this 0.35 m x 0.35 m area, with most of the variance occurring linearly with Y distance. On the other hand, the FTM RTT measurements in the dense setup shown in Fig. 3-14b do not scale as well with Y distance from the AP since the distance measurements are dominated by the waves with high peaks and deep troughs. The color bar shows that the median FTM RTT measurements vary a lot more than in the open-space, by 2.5 m in the 0.35 m x 0.35 m area, with most of the variance occurring in the wave textures.

These results further support the hypothesis that multipath propagation of signals is playing a large role in the strength of the position-dependent error. In the open-space setup plot, while the position-dependent error still exists, it is much less severe than in the dense setup plot where multipath propagation of signals would have been prevalent. Additionally, the similarities of the wave structure between the tests emphasize the periodicity of the position-dependent error, where the FTM RTT measurement algorithms for estimating time of arrival are somehow dependent on the central frequency of the signal, as one half of the wavelength mentioned in Chapter 3.4.

In both cases of the dense and open-space setups, the RSS plots still have wave textures. Shown in Fig. 3-15, this reiterates that the signal received by the smartphone has varying strength, not only in the distance from the AP, but also by the environment. Similar to the FTM RTT measurement plots, the different between the peaks and troughs in the open-space setup RSS plot seems to be smaller than in the dense setup. Thus, there still seems to be some correlation between the frequencies seen in the RSS and FTM RTT measurements.

For future work, it would be interesting to see the FTM RTT measurement plot

(a) Open-Space Setup FTM RTT Measurements with Google AP 3c:28:6d:95:ef:de



(b) Dense Setup FTM RTT Measurements with Google AP 3c:28:6d:95:ef:de

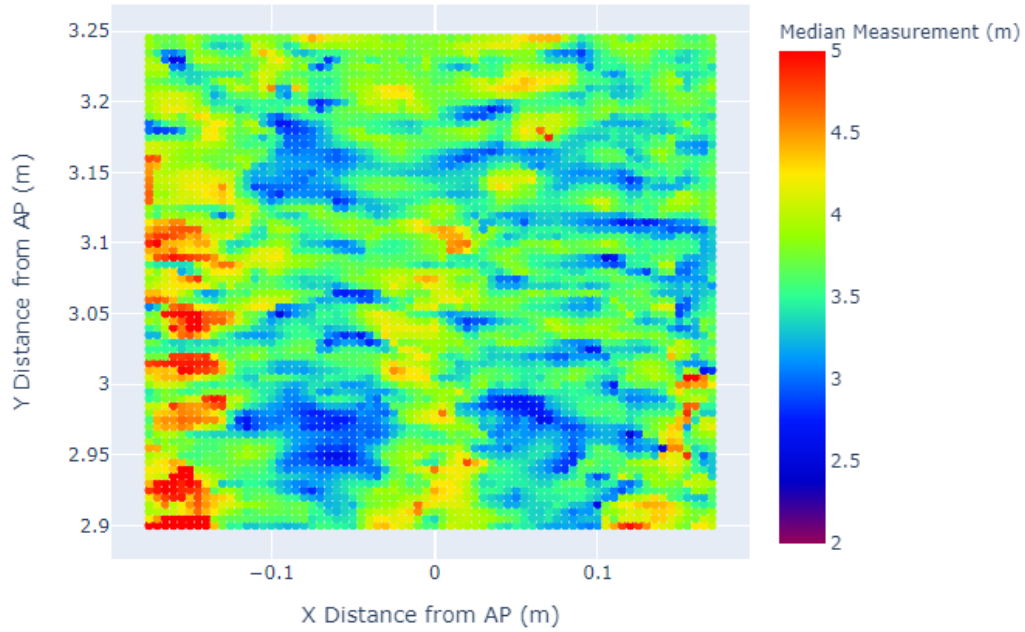


Figure 3-14: 2D position-dependent error plots using Google Pixel 4 and Google Wi-Fi AP 3c:28:6d:95:ef:de in an open-space (a) and a dense (b) environment setup. In both cases the AP was positioned at (X, Y) = (0, 0). While the plots start at different Y distances from the AP, they cover the exact same amount of area. Note the difference in range of measurement values between the setups. Color Bar: Median reported FTM RTT measurement at a given position.

in an extremely open-space setup, such as an anechoic chamber. In this setup, there would be almost no multipath propagation affecting the time of arrival estimation algorithms. Would this produce a flat plot where the FTM RTT measurement only varies with distance, or would there still be some periodicity to the plot?

3.6 Comparing Smartphones: Google Pixel 4 and Google Pixel 3

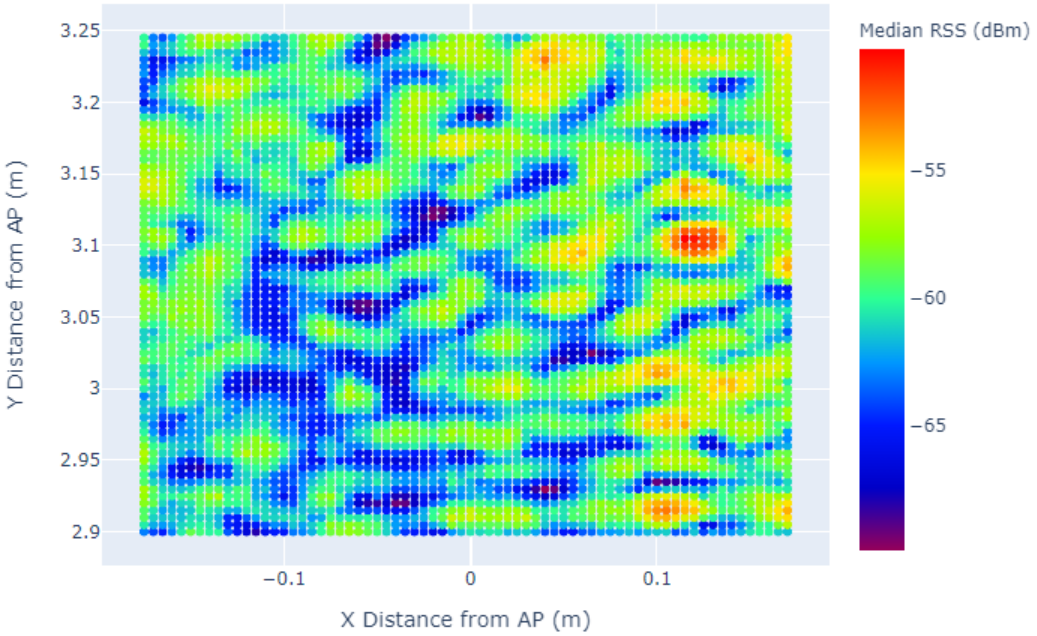
When trying to understand the position-dependent error and applications of indoor positioning systems, it is important to compare results of varying devices that may have different hardware or software. Understanding the differences in performance can empower us to make better recommendations going forward. In this case, the Google Pixel 3 and Google Pixel 4 smartphones are both running on the same software versions, but may have different hardware.

Fig. 3-16 shows the difference between a Google Pixel 3 and Google Pixel 4 open-space data collection where the rest of the setup was identical. Again in both cases, the results have textured waves. In the case of the Google Pixel 3 in Fig. 3-16a, the peaks and troughs are much larger than in the case of the Google Pixel 4. Additionally, the FTM RTT measurements rise and fall more abruptly with quick transitions to the peaks and troughs with the Google Pixel 3. The range of values in the Google Pixel 3 case is about 2.5 m, whereas the range is only 1 m in the Google Pixel 4 case shown in Fig. 3-16b.

These Google Pixel 3 errors are exacerbated in a dense setup. Fig. 3-17 shows the dense setup results, where the range of measurements is 3.5 m with even larger peaks and deeper troughs. Those measurement ranges were wider and less consistent than the Google Pixel 4 in the dense setup, shown in Fig. 3-14b.

The Google Pixel 3 data collections do show similar results to the Google Pixel 4 in the analysis of the frequency domain and the RSS. Fig. 3-18 shows the similar RSS structure, while Fig. 3-19 shows the similar frequency domain with hot spots in

(a) Open-Space Setup RSS Measurements with Google AP 3c:28:6d:95:ef:de



(b) Dense Setup RSS Measurements with Google AP 3c:28:6d:95:ef:de

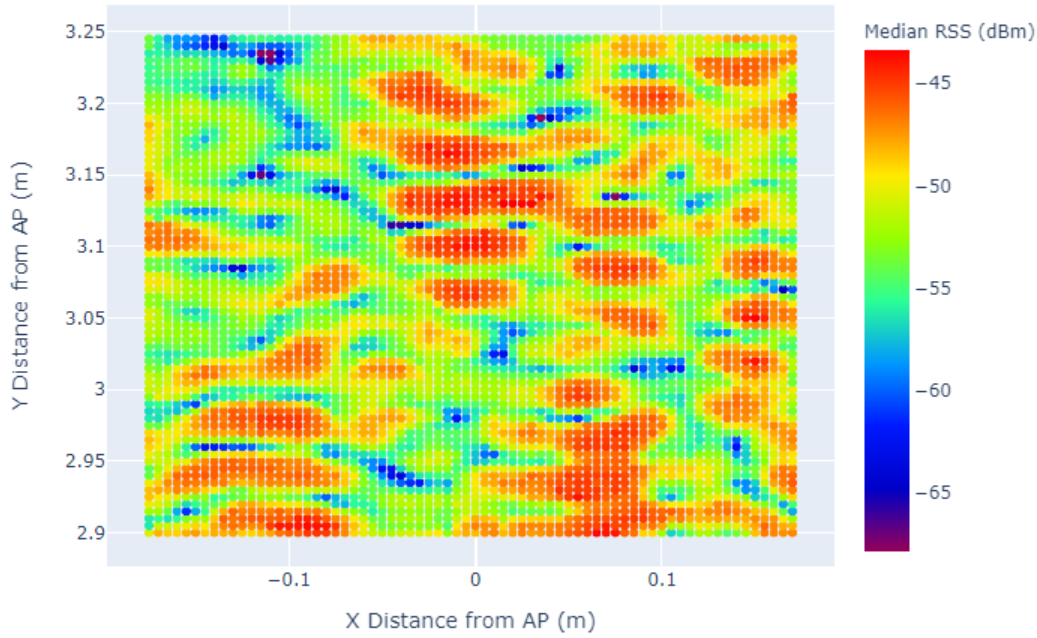
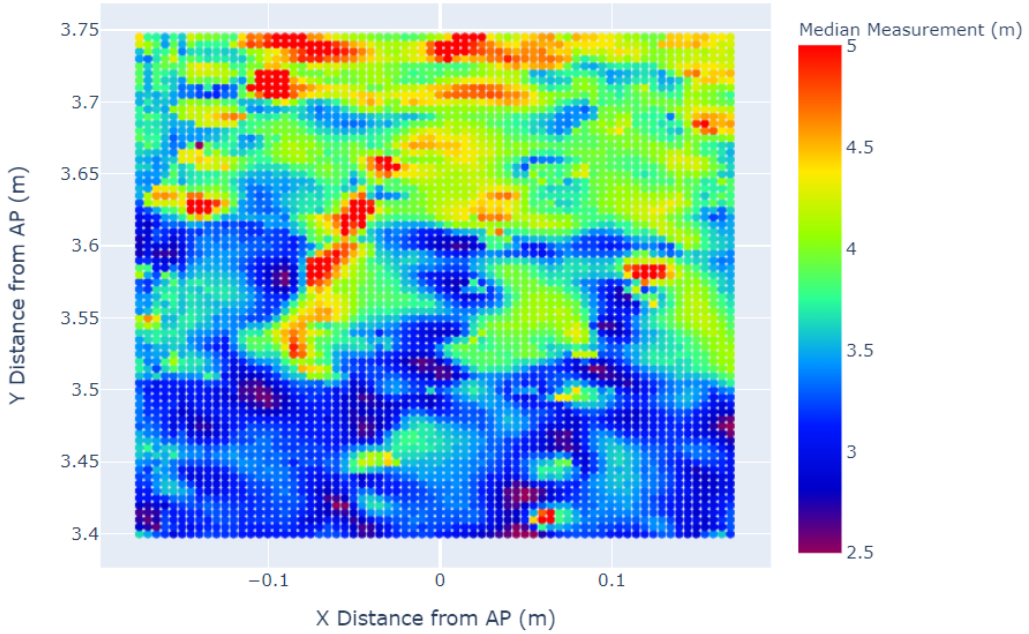


Figure 3-15: 2D position-dependent error RSS plots using Google Pixel 4 and Google Wi-Fi AP 3c:28:6d:95:ef:de in an open-space (a) and a dense (b) environment setup. In both cases the AP was positioned at (X, Y) = (0, 0). While the plots start at different Y distances from the AP, they cover the exact same amount of area. Data is from the same AP and collection as Fig. 3-14. Color Bar: Median reported RSS measurement at a given position.

(a) Pixel 3 FTM RTT Measurements with Google AP 1c:f2:9a:c3:53:14



(b) Pixel 4 FTM RTT Measurements with Google AP 1c:f2:9a:c3:53:14

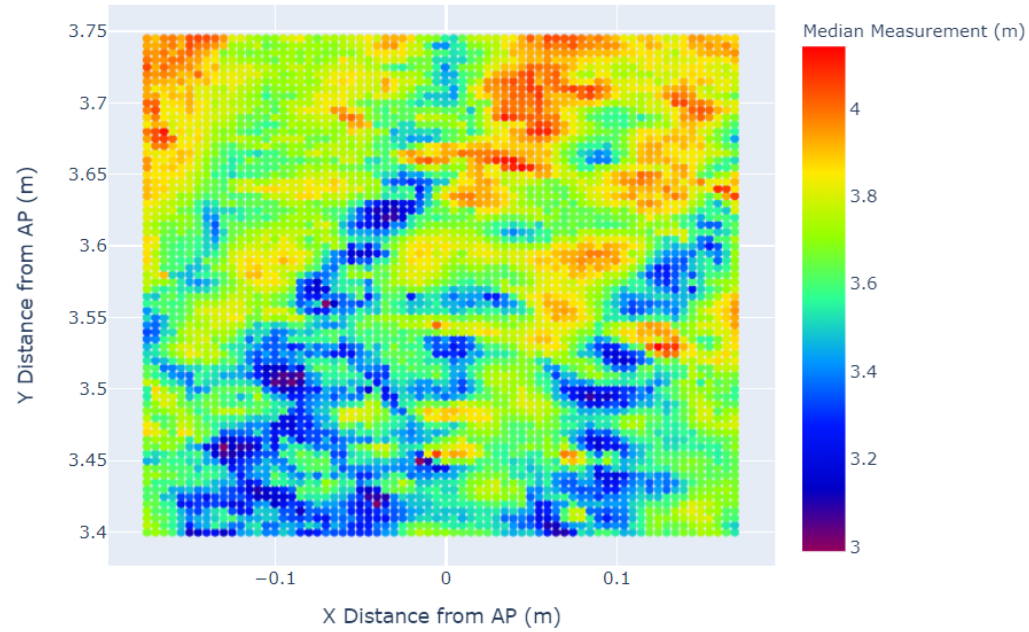


Figure 3-16: 2D position-dependent error plots using Google Pixel 3 (a) and Google Pixel 4 (b) with the Google Wi-Fi AP 1c:f2:9a:c3:53:14 in an open-space environment setup. In both cases, the AP was positioned at (X, Y) = (0, 0). Color Bar: Median reported FTM RTT measurement at a given position.

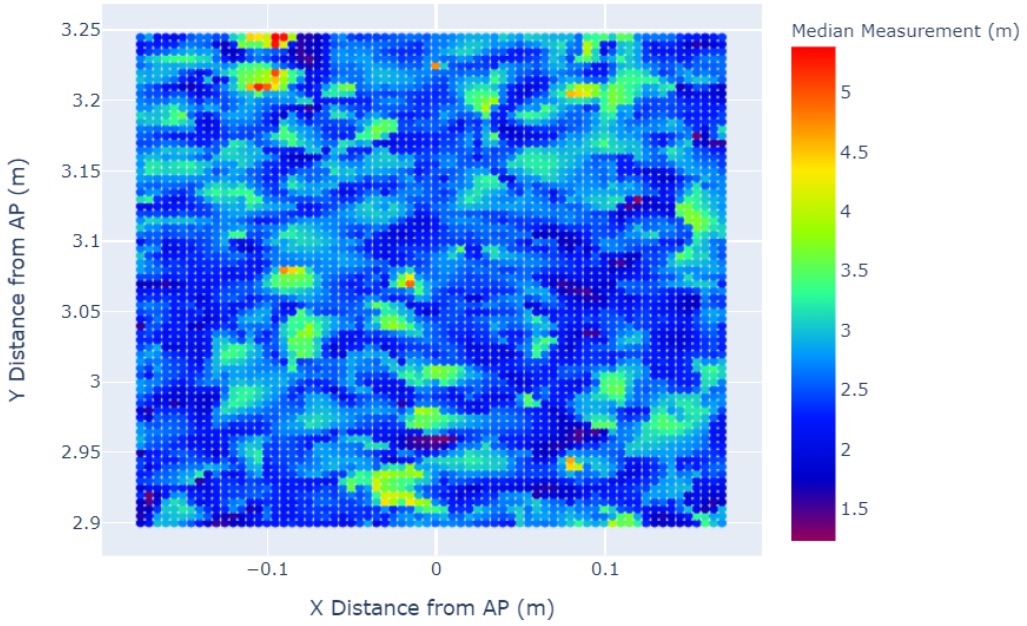


Figure 3-17: 2D position-dependent error plot using Google Pixel 3 and Google Wi-Fi AP 1c:f2:9a:c3:53:14 in a dense environment setup. The AP was positioned at (X, Y) = (0, 0). Color Bar: Median reported FTM RTT measurement at a given position.

the same position as the Google Pixel 4 collection shown in Fig. 3-12a. The similar Google Pixel 3 and Google Pixel 4 frequency results would be expected, as the Google Wi-Fi AP was using the same 5210 MHz central frequency for both collections. In the end, while the Google Pixel 3 results have similar structure to the Google Pixel 4 results, the position-dependent error is significantly larger in the Google Pixel 3.

3.7 Comparing Access Points: Google Wi-Fi and ASUS

Once again, it is important to understand the differences between the devices used for FTM RTT measurements. After looking at different smartphones, the next step is to evaluate different Access Points. Specifically, the comparison will be done between Google Wi-Fi APs and ASUS APs. Unlike the comparison between the Google Pixel 3 and Google Pixel 4, where the software versions were the same, the Google Wi-Fi and ASUS APs have different hardware and software.

Fig. 3-20 shows an example of an ASUS AP from the same data collection as the Google AP in Fig. 3-16b, where all else was equal. While the familiar wave texture

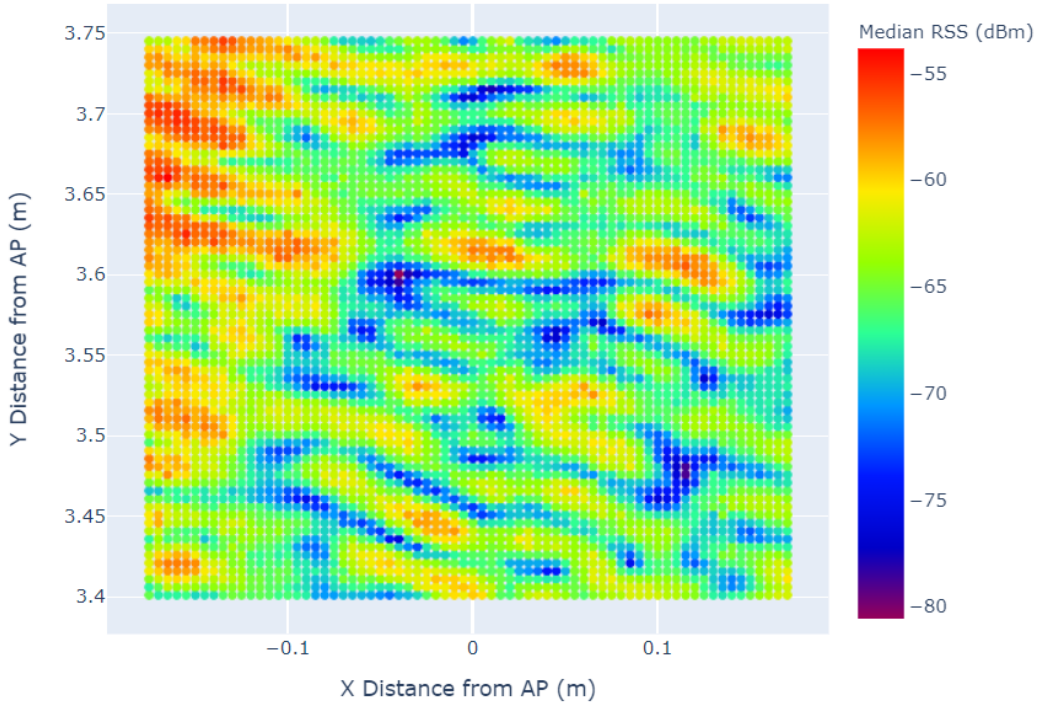


Figure 3-18: 2D position-dependent error RSS plot using Google Pixel 3 and Google Wi-Fi AP 1c:f2:9a:c3:53:14 in an open-space setup, from the same data collection as Fig. 3-16a. The AP was positioned at (X, Y) = (0, 0). Color Bar: Median reported RSS measurement at a given position.

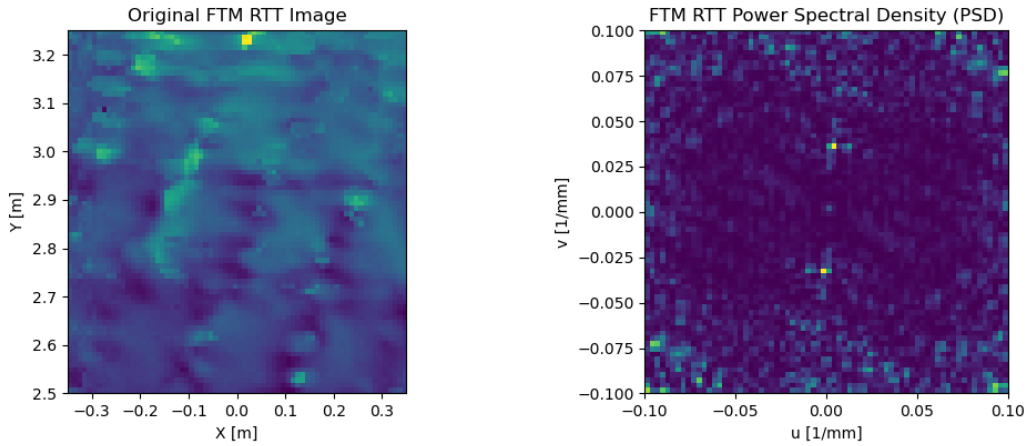


Figure 3-19: Results of the DFT of the Google Pixel 3 FTM RTT measurements using Google AP 1c:f2:9a:c3:53:14 in an open-space setup, from the same data collection as Fig. 3-16a.

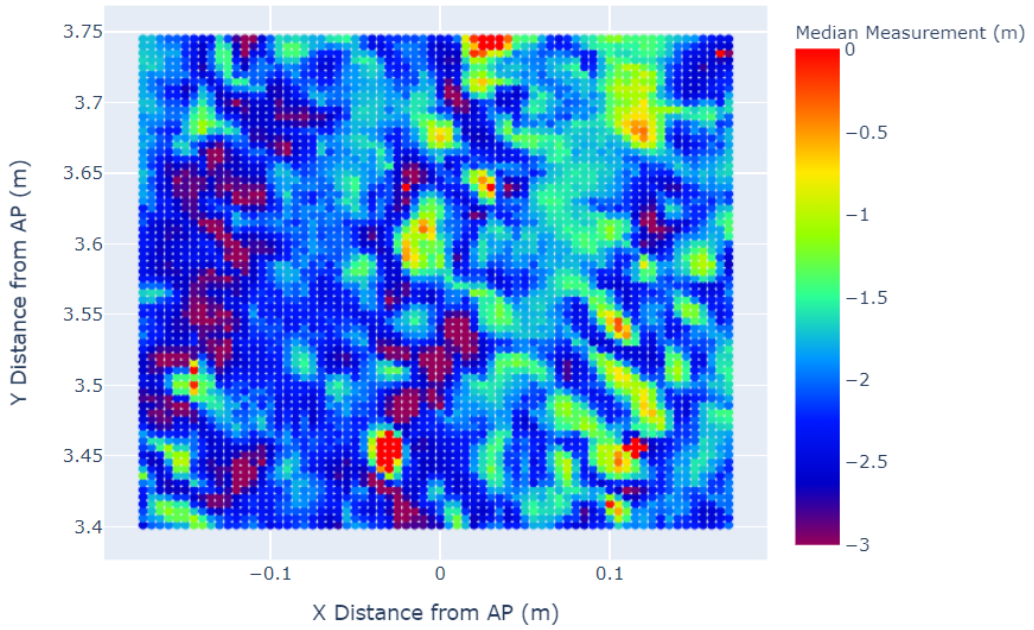


Figure 3-20: 2D position-dependent error plot using Google Pixel 4 and ASUS AP 0c:9d:92:b9:8f:f4 in an open-space setup, from the same data collection as Fig. 3-16b. The AP was positioned at (X, Y) = (0, 0). Color Bar: Median reported FTM RTT measurement at a given position.

can be seen, there are a few major differences between the resulting plots. First, the ASUS AP color bar measurement ranges from −3 m to 0 m. Different combinations of smartphones and APs will have varying degrees of offsets that must be calibrated. In the case of the Google APs, the offset is close to 0 m, while it is clear that the ASUS AP offset is closer to −5 m. Additionally, the ASUS AP peaks and troughs of the texture are more extreme compared to the Google AP in the same open-space setup, which follows with the large range of FTM RTT measurements that can be seen. Lastly, the FTM RTT measurements do not seem to obviously increase linearly with distance in this 0.35 m x 0.35 m data collection.

The ASUS AP analysis of the RSS and frequency domain again proves to be different from that of the Google APs. Fig. 3-21 shows the RSS measurements from the same data collection as Fig. 3-20. In this case, the familiar wave texture appears, but varies in size and shape throughout the plot. Looking at the frequency domain of the FTM RTT measurements in Fig. 3-22, hot spots rarely appear. When they do, they are subtle and have inconsistent amounts of rotation.

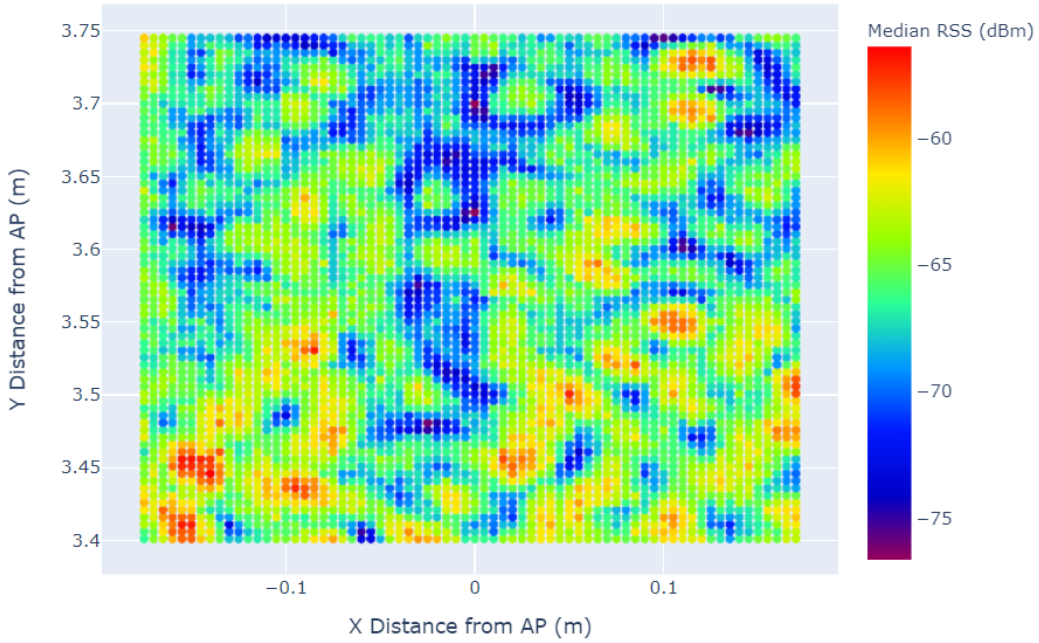


Figure 3-21: 2D position-dependent error RSS plot using Google Pixel 4 and ASUS AP 0c:9d:92:b9:8f:f4 in an open-space setup, from the same data collection as Fig. 3-20. The AP was positioned at (X, Y) = (0, 0). Color Bar: Median reported RSS measurement at a given position.

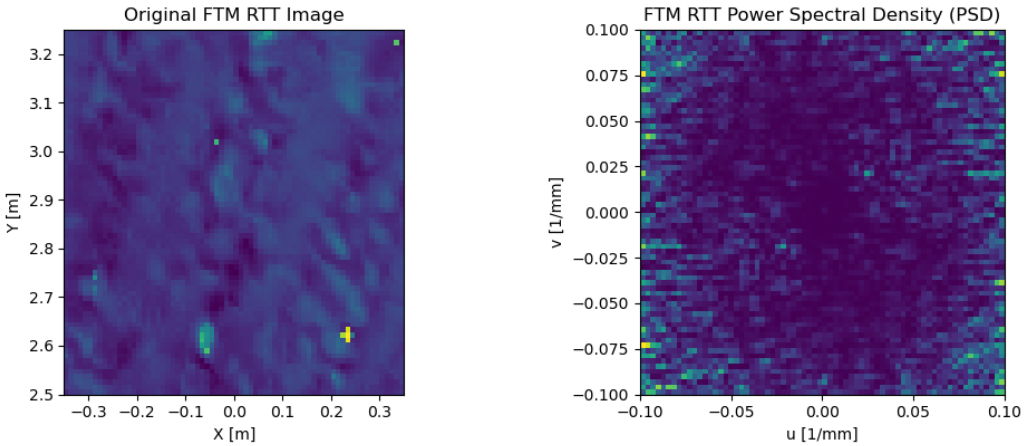


Figure 3-22: Results of the DFT of the Google Pixel 4 FTM RTT measurements using Google AP 0c:9d:92:b9:8f:f4, from the same data collection as Fig. 3-20.

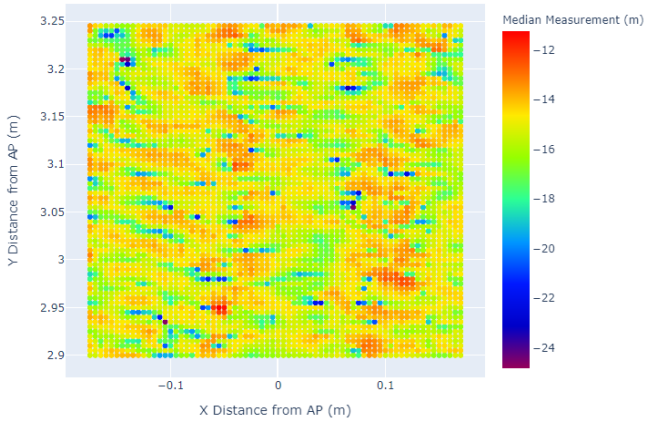
There are a few reasons that the ASUS APs may perform inconsistently compared to the Google APs. First of all, the Google Pixel smartphones and Google APs may be developed and tested together, whereas the ASUS APs may have been tested on other smartphones. Additionally, a major difference may be that the ASUS APs do not have continuous software updates, while the Google APs do. Over time, the updates on the Google APs may make a dramatic improvement in performance as more testing is done and changes are made. Finally, while the ASUS APs are capable of performing FTM RTT measurements, they do not advertise the ability to do so in the beacon frame. As such, improving the performance of the FTM RTT measurements for a device release may be more of an afterthought.

3.8 Varying the Access Point's Bandwidth and Frequency

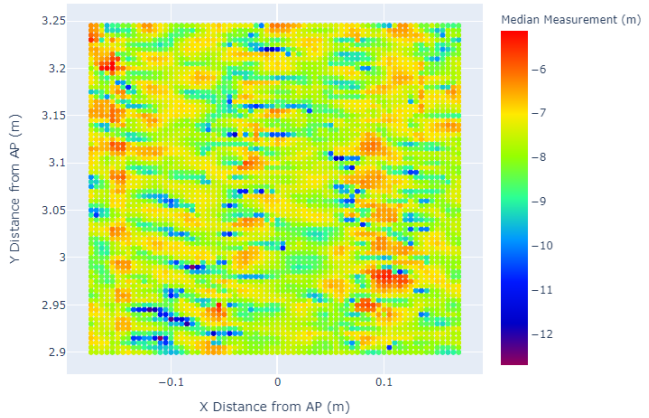
Another way to better understand the position-dependent error is to record FTM RTT measurements while varying the AP signal's frequency and bandwidth. To start, we will fix the central frequency of the signals and vary the bandwidth, and then we will fix the bandwidth while varying the central frequency. Unfortunately, the Google Wi-Fi APs do not allow the user to change the bandwidth of the signal, so all of the varying bandwidth experiments were run using the ASUS APs with the Google Pixel 4 smartphone.

Fig. 3-23 shows the comparison between 20 MHz, 40 MHz, and 80 MHz signal bandwidths. These test were run consecutively, only changing the bandwidth of the signal to note the difference in scales between them. The results show almost identical images. The only difference is the scale of the color bar representing the range of FTM RTT measurement distances. The range of values clearly decreases as the bandwidth increases. The 20 MHz bandwidth plot displays a range of measurements of around 12 m, the 40 MHz bandwidth plot displays a range of measurements of around 6 m, and the 80 MHz bandwidth plot displays a range of measurements of around 3 m.

(a) ASUS AP 4c-ed-fb-b7-5a-3c 20 MHz Bandwidth



(b) ASUS AP 4c-ed-fb-b7-5a-3c 40 MHz Bandwidth



(c) ASUS AP 4c-ed-fb-b7-5a-3c 80 MHz Bandwidth

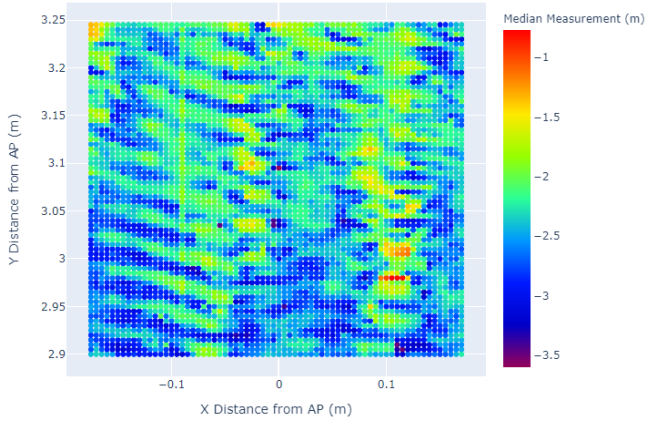


Figure 3-23: 2D position-dependent error plots using Google Pixel 4 and ASUS AP 4c-ed-fb-b7-5a-3c at varying bandwidths of 20, 40, and 80 MHz. The AP was positioned at (X, Y) = (0, 0). Note the decreasing range of measurements as the bandwidth increases. Color Bar: Median reported measurement at a given position.

Clearly, the bandwidth of the signal has a strong effect on the accuracy of the measurements. One way of ameliorating the position-dependent error would be to increase the bandwidth of the signal as much as possible. We would expect that doubling the bandwidth would cut the position-dependent error in half. Unfortunately, current devices are limited to a maximum of 80 MHz bandwidth. Another possible drawback has to do with the resolution of the signal. The theoretical expected accuracy of an 80 MHz signal is only 3.75 m round trip with a clear line of sight [5]. This expected accuracy can be increased with different forms of interpolation and super-resolution algorithms to produce finer resolution. Because it is likely that the Google smartphones, ASUS APs, and Google APs use interpolation or super-resolution algorithms, it is not clear exactly what the effect of increasing the bandwidth above 80 MHz would have on the algorithms. Presumably, an increased bandwidth would continue to increase the resolution of the measurements, but outside of this experimentation it may not be linear. Doubling the bandwidth may not result in exactly double the accuracy without explicit knowledge of the super-resolution algorithms used.

Next, by instead fixing the bandwidth of the signal to 80 MHz, we can vary the central frequency of the signal. Fortunately in this case, the Google APs are able to switch channels and experiments can be run with central frequencies of 5210 MHz and 5775 MHz. Fig. 3-6 shows the FTM RTT measurements using Google AP 1c:f2:9a:c3:53:14 with central frequency 5210 MHz. Fig. 3-24 shows the same dense setup over a smaller area, but instead using the 5775 MHz central frequency. The results look similar again, with the wave textures appearing. There are not obvious differences between the two setups in the spatial domain, so instead we will look to the frequency domain.

Fig. 3-12 shows the frequency domain using Google AP 1c:f2:9a:c3:53:14 with central frequency 5210 MHz. As mentioned in Chapter 3.4, the hot spots on the PSD plots appear in the area of $(u, v) = (0, \pm 0.035)$. The frequency domain value of $0.035\,(1/\text{mm})$ represents a wave every 28.5 mm in the spatial domain. Once again, at a central frequency of 5210 MHz, the wavelength of the signal is about 57.5 mm

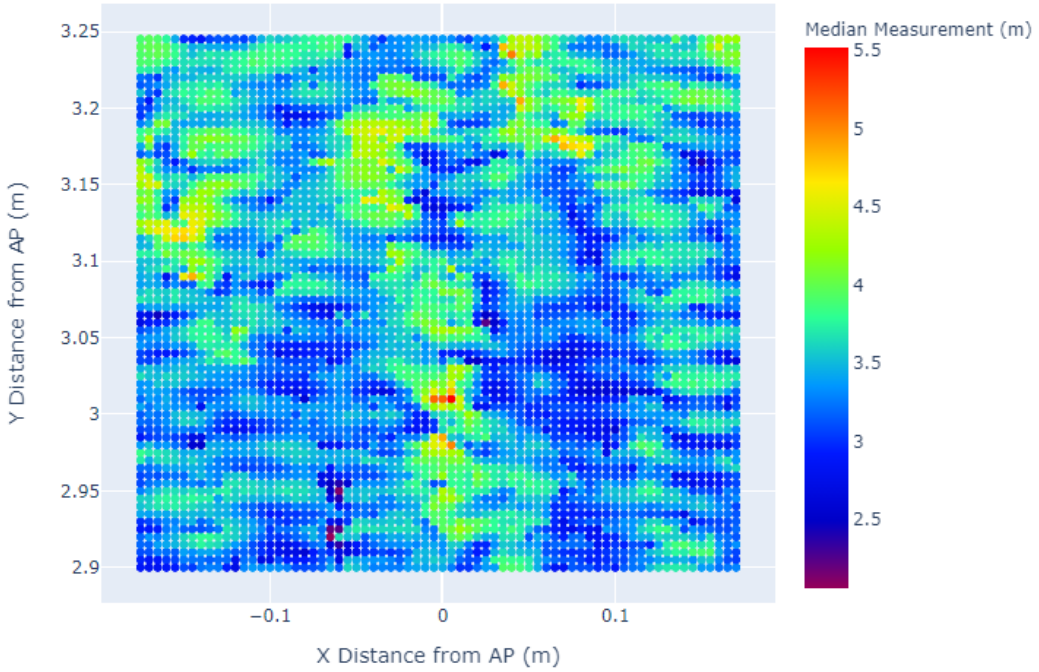
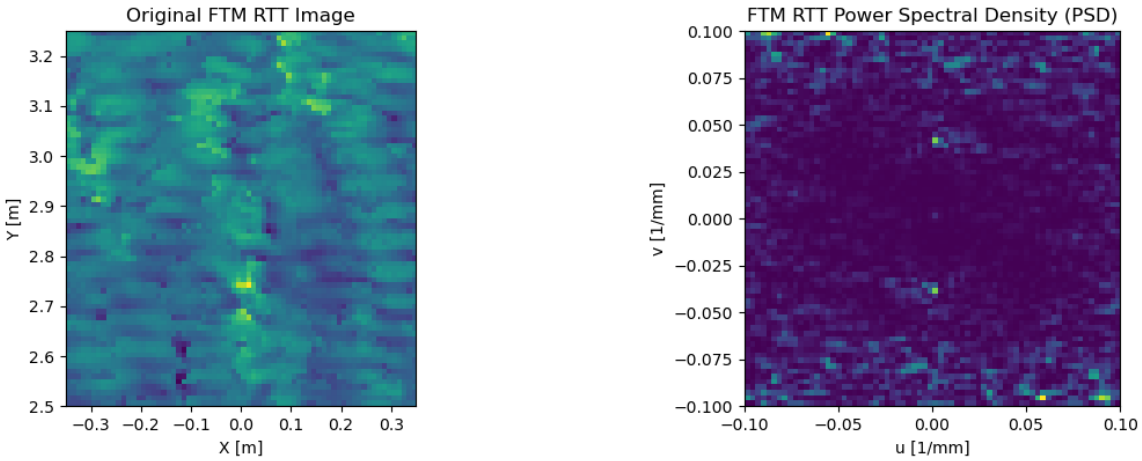


Figure 3-24: 2D position-dependent error plot using Google Pixel 4 and Google AP 1c:f2:9a:c3:53:14 with central frequency of 5775 MHz. The AP was positioned at (X, Y) = (0, 0). Color Bar: Median reported measurement at a given position.

or almost exactly double the dominant frequencies shown in the frequency domain. Fig. 3-25 shows the frequency domain using Google AP 1c:f2:9a:c3:53:14 but instead with central frequency 5775 MHz. With this central frequency, the wavelength is about 51.9 mm. Looking at the PSD plot, the hot spots appear in the area of $(u, v) = (0, \pm 0.04)$. The frequency domain value of 0.04 (1/mm) represents a wave every 25 mm in the spatial domain. Again, this value is almost exactly half of the wavelength of the signal.

Putting the results together, we can see that the central frequency is inversely proportional to the size of the wave textures in the spatial domain. As the central frequency increases, the wave textures decrease in size, following one half of the wavelength of the signal. With a high enough central frequency, the size of the wave textures could decrease to a point where they are blended together depending on how the graph or plot is discretized. Research has shown that using the weighted average of measurements at several different central frequencies can double the accuracy of indoor positioning [5].

(a) FTM RTT spatial and frequency images from Google AP 1c:f2:9a:c3:53:14 with 5775 MHz central frequency



(b) RSS spatial and frequency images from Google AP 1c:f2:9a:c3:53:14 with 5775 MHz central frequency

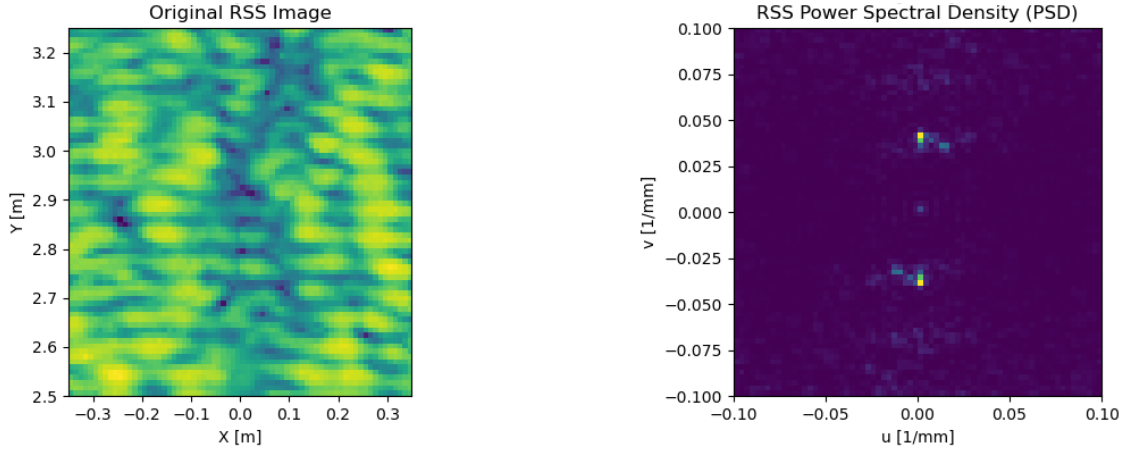


Figure 3-25: Results of the DFT of the RSS and FTM RTT measurements from a data collection using the same Google AP 1c:f2:9a:c3:53:14 at 5775 MHz central frequency. Sub-figure (a) shows the spatial and frequency domains of the FTM RTT measurements, while Sub-figure (b) shows the spatial and frequency domains of the RSS measurements.

Chapter 4

Possible Causes of the Position-Dependent Error

This chapter covers possible components of the position-dependent error in FTM RTT measurements, including inaccuracies that may be a part of the calculations. It focuses mainly on different ways that the time of arrival of a signal is estimated and methods for increasing resolution. Specifically, this chapter explores time of arrival algorithms, super-resolution algorithms, received signal strength calculations, and clock stability.

4.1 Time of Arrival Algorithms

Time of arrival (ToA) algorithms are crucial for estimating distance and position using FTM RTT. Round-Trip Time approximations of a signal through a known medium are heavily dependent on how the arrival times are calculated. Fortunately, there are plenty of algorithms available to estimate a signal's ToA. Before running the ToA algorithms, interpolation or sub-sampling between points may be performed for better resolution. Unfortunately, these ToA algorithms may not be robust to multipath propagation of the signal.

In general, indoor environments are cluttered with walls and other obstructions, and the signal may not have a direct line of sight between the AP and the smartphone.

Additionally, the signal may have many objects to reflect off of. As a result, there may be multiple paths that signals can take and still arrive at a receiver at around the same time as the direct path. Because of multipath propagation, the received signal will be the sum of these reflections, each with a slightly different ToA. The ToA algorithms may rely on undisturbed leading-edges or peaks of the received signal, and thus may provide inaccurate timing estimates in the presence of multipath propagation. Two of the major ToA algorithms that we will explore are peak-tracking and leading-edge projection that use the reconstructed correlation diagram pulse of a signal otherwise known as the correlogram [1]. In the case of these practical devices, we will use bandlimited correlogram, since the devices do not have unlimited bandwidth.

In order to calculate the correlogram using the autocorrelation function, a device receives a pseudorandom noise (PN) code from another transmitting device. The PN-codes have two important properties. First, the autocorrelation peak must be sharp and maximal upon synchronization, or a time shift equal to zero. Second, the autocorrelation must be minimal and very close to zero for any time shift different than zero [23]. Knowing these properties, in order to find the correct time shift, the autocorrelation function can be computed at equally spaced offsets. The most accurate time shift will be chosen as the output with the largest peak. The resulting correlogram is shown in Fig. 4-1 for both the unlimited bandwidth and bandlimited cases.

The peak-tracking algorithm can estimate the location of the bandlimited correlogram's peak using approximate curve fitting with a quadratic or Gaussian model. For the quadratic model, three sample points are required. The first sample point, G_0, is chosen as the point closest to the peak with maximum RSS. Then two points should be chosen that are 1 unit left (G_-) and right (G_+) of G_0. Using the assumed quadratic model and the sample points G_-, G_0, and G_+, we can solve for the coefficients a, b, and c to find the equation of the curve with the following [24]:

$$y(x) = ax^2 + bx + c \tag{4.1}$$

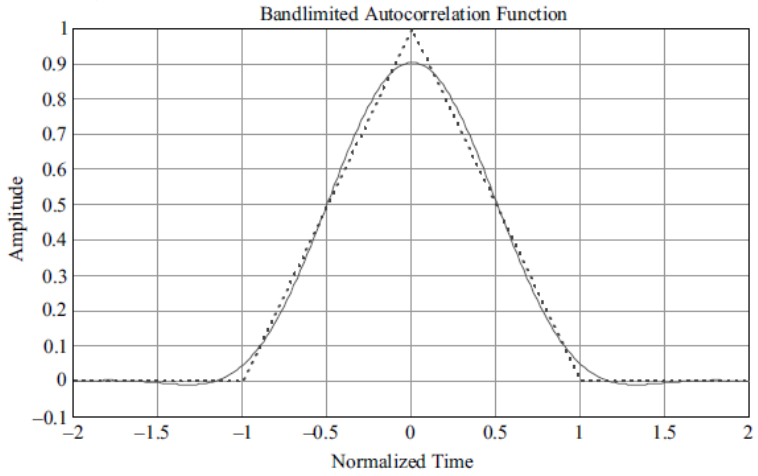


Figure 4-1: Correlogram resulting from the autocorrelation of a PN-code. The dotted line represents the ideal correlogram with unlimited bandwidth, and the solid line represents the bandlimited correlogram. The bandlimited correlogram would be expected from the practical devices used in this research. Figure taken from from the book, *Ground-Based Wireless Positioning* [1].

Once we have the desired quadratic equation, we can find the position of the peak by setting the derivative of the above equation equal to 0. This gives us the location of the peak as:

$$x = \frac{-b}{2a} \tag{4.2}$$

An example of this quadratic model is shown in Fig. 4-2. In the absence of multipath propagation, the bandlimited correlogram's peak can be accurately calculated using this method. In the presence of multipath propagation that is only slightly delayed from the first arriving signal, the peak may be shifted. Since this multipath propagation will be delayed relative to the first arriving signal, the sample point most susceptible to a change in amplitude is G_+. Assuming there is no interpolation or sub-sampling, with a bandwidth of 80 MHz and sampling rate of 80 MHz, sample points would be separated by 12.5 ns, which equates to 3.75 m traveled at the speed of light. Making a large assumption that G_- is near the start of the rising edge of the correlogram, G_+ would be delayed by two sample points or 25 ns from the start of the first arriving signal. That means that a delay of at most 25 ns would start to

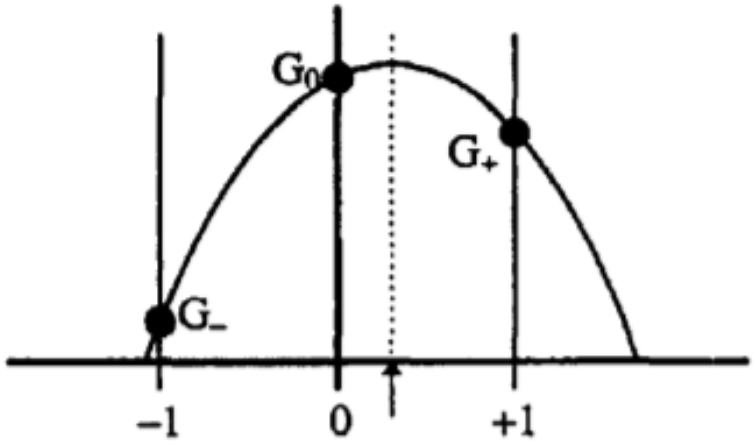


Figure 4-2: Quadratic model of the bandlimited correlogram used in a peak-tracking ToA algorithm. G_0 should be chosen as the maximum RSS sample point with G_- and G_+ as the sample points chosen 1 unit left and right of G_0, respectively. The position of the peak can then be determined by solving Equations 4.1 and 4.2. Figure taken from patent US 6,408,109 B1 [2].

affect the amplitude of G_+. Therefore, a path difference between the first arriving signal and a signal due to multipath propagation of about 7.5 m or less could influence the amplitude of the chosen sample points and change the estimated quadratic model, producing errors in the ToA calculations. Without the strong assumption of the position of G_-, the multipath delay that may affect the amplitude of G_+ could be even larger. Ultimately, the accuracy of the peak-tracking algorithm may be heavily dependent on the amount of multipath propagation of signals present as well as their time delay.

The leading-edge projection algorithm takes advantage of the fact that the signals due to multipath propagation will be delayed relative to the first arriving, line of sight signal. By looking only at the leading-edge of the bandlimited correlogram at the receiver before the peak, it is less likely that multipath propagation has influenced the waveform when compared to the peak-tracking algorithm. The leading-edge projection algorithm takes two points of the bandlimited correlogram. The initial point, P_1, can be chosen as the first sample that is above a certain threshold amplitude, αA, where α may be around 0.15, but can be adjusted based on SNR, and A is the bandlimited correlogram amplitude or max RSS value of the sample points. The second

sample point, P_2, can be chosen at a time delay of τ from the first sample, but is still expected to be on the leading-edge. To simplify the computation, the algorithm assumes that the bandlimited correlogram behaves like the ideal correlogram (a triangle) in in the middle section of the leading-edge. Provided that the two sample points are within the middle section of the leading-edge, the ToA can be calculated by projecting the line from P_2 through P_1 to intersect with the time axis [1]. This will provide an estimate of exactly when the signal was first received. This method is displayed in Fig. 4-3.

While this method may be more robust than the peak-tracking it is still subject to large fluctuations if the signals due to multipath propagation have a small delay relative to the first arriving signal. For example, we can assume a small value for α and that there is no interpolation of the data, with τ being equal to the distance between two adjacent samples. Again, with a bandwidth of 80 MHz and sampling rate of 80 MHz, sample points would be separated by 12.5 ns which equates to 3.75 m traveled at the speed of light. In this case, a delay of just over 12.5 ns for a signal that results from multipath propagation would start to affect the amplitude of P_2. That would mean that a path difference between the first arriving signal and multipath propagation of about 3.75 m or less could start to produce errors in the ToA calculations.

In both the peak-tracking and leading-edge projection algorithms, we can see that there is an inverse relationship between the bandwidth and path difference that begins to affect the ToA algorithm. As the bandwidth and sampling rate increase, the time between sample points for the algorithms decreases. Then as the time between points decreases, the maximum path difference between the first arriving signal and multipath propagation that would be to able to affect ToA estimation will also decrease. The peak-tracking detection algorithm would still be more susceptible to delayed multipath propagation than the leading-edge projection algorithm, because the peak-tracking points will always be centered around the peak of the correlogram, which is already delayed from the start of the rising edge of the correlogram. This delay again allows multipath propagation with a larger path difference to influence

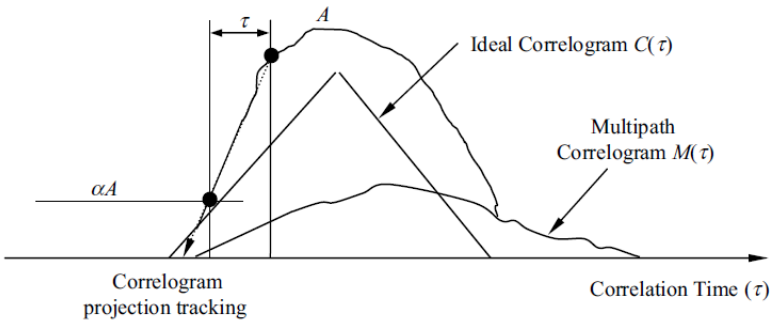


Figure 4-3: Example correlogram used in a leading-edge projection ToA algorithm. A is the amplitude of the observed correlogram. The two points should be chosen on the leading-edge of the correlogram, the first point, P_1, with an amplitude greater than or equal to αA, and the second point, P_2 at a time delay of τ from P_1. The line between P_2 and P_1 will be extended to cross the time axis to determine original ToA. Figure taken from from the book, *Ground-Based Wireless Positioning* [1].

the amplitude of the peak-tracking sample points chosen.

It seems likely that the APs and smartphones used in this research take advantage of ToA algorithms similar to what was discussed. We saw in Chapter 3 that in the dense setups where multipath propagation of signals was prevalent, large fluctuations were seen in the FTM RTT measurements. These large fluctuations line up with what we would expect with the presented ToA algorithms with small multipath time delays. Additionally, these algorithms have low complexity, which makes them ideal for devices like APs or smartphones that are running on battery and may have limited processing power [1].

Google Wi-Fi APs and Google Pixel smartphones use the Qualcomm chipset. Another clue that these simple ToA algorithms may be used in the devices lies within the publicly available Qualcomm patents. In patent US 7.474,994 B2 titled "System and Method for Wireless Signal Time of Arrival", a peak-tracking ToA method is described that again models the bandlimited correlogram of an incoming waveform with a quadratic equation using three samples points [24]. Once again, this method may suffer from signals that result from multipath propagation that corrupt the correlogram when their time delay is small.

Unfortunately, these ToA algorithms can be very sensitive to multipath propaga-

tion, which may start to explain the source of the position-dependent error. It was shown that similar algorithms are likely involved in FTM RTT measurements, and possible solutions are discussed in Chapter 5.

4.2 Super-Resolution Algorithms

As mentioned in Chapter 3, the expected accuracy of a signal at 80 MHz bandwidth is only 3.75 m round trip with a clear line of sight [5]. The maximum bandwidth is set by Wi-Fi standards and may be limited by a device's hardware. The 2013 version of IEEE 802.11 Wi-Fi standard, 802.11ac, increased the maximum bandwidth of a channel to 80 MHz [25]. Most recently, the 2021 version of IEEE 802.11 Wi-Fi standard, 802.11ax, increased the maximum bandwidth of a channel from 80 MHz to 160 MHz, but most hardware does not yet support it [26]. With the current maximum bandwidth for our devices of 80 MHz, the sampling rate of 80 million samples per second returns a sample every 12.5 nanoseconds. Since the wireless signal travels at the speed of light, about 300 000 000 m/s in air, we would expect a sample every 3.75 m. Clearly, this resolution is not fine enough for most indoor positioning applications. That is where super-resolution algorithms come in. They provide a way to increase the resolution beyond the sampling rate, by interpolation between points providing a clearer picture of the incoming waveform. With a higher resolution, it is more straightforward to find the peak of the incoming signal and thus the signal's ToA. There are several different super-resolution algorithms based on representative subspace methods, such as MUSIC, ESPRIT, and Matrix Pencil [27, 28, 29].

MUSIC or Multiple Signal Classification estimates the frequency content of a signal or autocorrelation matrix using an eigenspace method. The first step of the actual algorithm is to calculate the covariance matrix of the data. The next step is to compute the eigendecomposition of the covariance matrix in order to estimate a signal subspace and a noise subspace. MUSIC exploits the fact that noise eigenvectors that compose the noise subspace are orthogonal to the signal vectors [30]. MUSIC then uses the noise subspace to compute the pseudospectrum in order to locate the

support of the signal, where it is nonzero on the time axis [31].

This method makes a few assumptions that may not always be exact in real applications. First, it assumes that a signal vector, x, consists of p complex exponentials in the presence of Gaussian white noise. The number of components, p, is required to be known in advance. Fortunately, the underlying model of the channel is a weighted sum of spaced out impulses, and we do know how many pulses there are. If these components were not known, there are methods for estimating them [32]. MUSIC also assumes coexistent sources to be uncorrelated, which again may limit accuracy in practical applications. Finally, in order to ensure that the noise subspace is identified correctly, the eigenvalues must all be large with respect to the variance of the noise, σ^2. At relatively high SNR, MUSIC is an effective algorithm as long as assumption on the signal, noise, and measurement model are satisfied [31].

Like MUSIC, ESPRIT (estimation of signal parameters via rotational invariant techniques) is a technique to determine parameters of a mixture of sinusoids in a background noise. Again, the knowledge of the number of sinusoids embedded in the noise is required. While similar to MUSIC, ESPRIT is more efficient [33]. The ESPRIT algorithm is also a subspace algorithm, and it computes the eigendecomposition of the covariance matrix of the measured data to estimate a signal subspace and a noise subspace. Unlike MUSIC which uses the noise subspace, the ESPRIT algorithm uses the signal subspace, and frequency content can be estimated using a pseudo inverse method [34]. One advantage of ESPRIT is that it does not depend upon the size of the temporal window, whereas MUSIC has a lower frequency resolution with a shorter length of the time window [30]. Since FTM RTT is estimating range several times per second, the size of the temporal window may be limited, making the ESPRIT method ideal over MUSIC.

The Matrix Pencil super-resolution algorithm uses a generalized pencil-of-function, also known as the matrix pencil method, to obtain the exponents of a sum of complex exponentials. The generalized pencil-of-function is defined as follows. If two functions, $g(t)$ and $h(t)$ are combined on a common interval with scalar parameter Ψ such that $f(t, \Psi) = g(t) + \Psi h(t)$, then $f(t, \Psi)$ is called the pencil of $g(t)$ and

$h(t)$ parameterized by Ψ. When $g(t)$, $h(t)$, and Ψ are selected appropriately, the pencil method of matrices contains important features that are helpful in extracting the poles of the system in one step process [28]. More specifically, the Matrix Pencil super-resolution algorithm considers $L+1$ linearly independent vectors of the discrete signal y, $[y_0, y_1, ..., y_L]$. They are arranged into "information" vectors $Y_1 = [y_0, y_1, ..., y_{L-1}]$ and $Y_2 = [y_1, y_2, ..., y_L]$, where the poles are the generalized eigenvalues of the matrix pencil, $Y_2 - zY_1$ [35]. In the absence of noise, the Matrix Pencil algorithm is able to recover the signal components exactly. Once again, the Matrix Pencil algorithm is also stable in the presence of noise, provided the noise level is not too large [36].

While the advantages of these super-resolution methods in increasing the resolution of a signal are important, there is a moderate to high computational and storage cost [37]. This can make them less than ideal in smartphones and APs which may be powered by a battery and have limited processing power, which means that there is no guarantee that these algorithms are actually used in practice. If these algorithms are implemented on the devices, all of the necessary assumptions may not hold, which could be the root cause of the position-dependent error. Using super-resolution algorithms that provide more sample points that are less accurate in practice would create errors that propagate into larger errors during the ToA estimation. In the case that these super-resolution algorithms are not implemented for FTM RTT measurements, it would be worth experimenting with them to see if they could increase the resolution of an incoming signal in various environments, and thus improve the accuracy of its arrival time estimate.

4.3 Received Signal Strength

As mentioned in Chapter 2, received signal strength and fingerprinting are major areas of research for indoor positioning, but suffer drawbacks as RSS varies indirectly with distance in scenarios with multipath propagation. With that being said, it was shown in Chapter 3 that there is some correlation in the frequency domain of the

RSS and FTM RTT measurements, so it is worth looking at RSS's relation to FTM RTT and ToA once again.

First of all, there is some research being done to directly combine ToA and RSS for hybrid distance measurements with some promising results [38, 39]. A combination of both may be able to provide more accurate distance measurements, as the strengths of each could be used in conjunction to outweigh the other's weaknesses. If a hybrid method were used for estimating FTM RTT measurements, it would make sense that there is some correlation with the RSS measurements, but not fully correlated as was shown in Chapter 3.

There is some evidence that RSS may be involved in the ToA algorithms used in the Qualcomm chips within the devices chosen for this research. Referencing the same patent as Chapter 4.1, US 7.474,994 B2 titled "System and Method for Wireless Signal Time of Arrival", it mentions using the maximum RSS as one of the possible correlation values to run the peak-tracking ToA algorithm on. That would mean that there could be a strong relationship between the FTM RTT and the RSS measurements. We also saw in Chapter 4.1 that the choice of sample points in both leading-edge projection and peak-tracking was dependent on the amplitude or RSS of the bandlimited correlogram. Unfortunately if RSS and FTM RTT measurements are dependent on each other, we may not be able to solve for the position-dependent error without fixing the deeper problem. That is, RSS measurements do not vary as the expected $1/d^2$ in real world scenarios with multipath propagation, standing waves, and more, where d is the distance between the smartphone and AP. Fig. 4-4 shows the poor fit between RSS and distance, and it also shows the large spread of RSS values at a given distance [3]. Assuming RSS and FTM RTT measurements are dependent on each other, ways to ameliorate the position-dependent error in RSS and FTM RTT are discussed in Chapter 5.

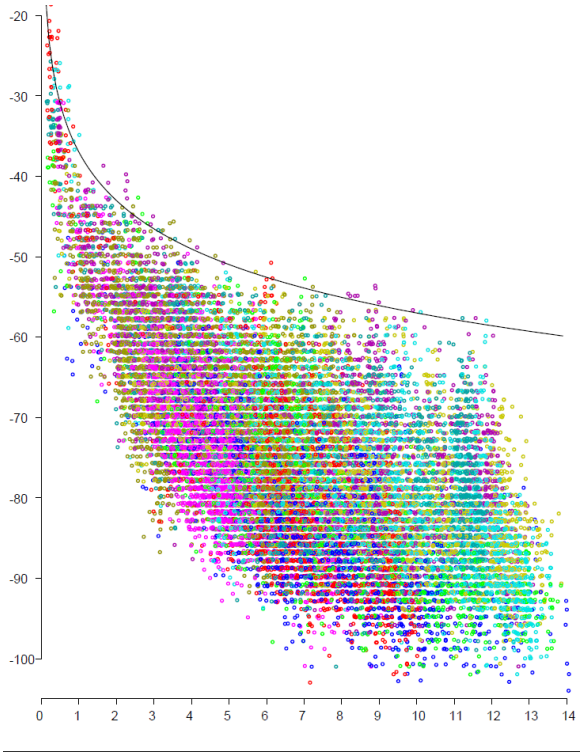


Figure 4-4: Scattergram of 20,000 RSS measurements versus actual distance between the smartphone and AP in a typical three level house. The solid curve corresponds to the inverse square law, which is not a good fit to the RSS measurements. Vertical axis: RSS value in dBm. Horizontal axis: actual distance between smartphone and AP in meters. Data taken from Berthold Horn [3].

4.4 Clock Stability

Since timing is so important in measuring the signal's time of flight, another possible large source of error could be in clock instability of the AP or smartphone. Fortunately, since FTM RTT uses the round-trip time of the signal, the clocks between the devices do not need to be synchronized. With that being said, if the clock on the smartphone were to shift in the middle of the measurement, the change in the resulting estimated measurement could be drastic. Again, since wireless signals travel at the speed of light, about 300 000 000 m/s in air, a clock drift of only 3.3 nanoseconds would result in a 1 m change in expected measurement. Thus, it seems that the accuracy of FTM RTT measurements would be heavily dependent on the precision of the clock over time.

The frequency stability of a clock depends on external conditions like temperature variation, voltage variation, output load variation, and frequency aging. There are several types of clocks, such as crystal oscillators, oven-controlled crystal oscillators, and voltage-controlled crystal oscillators, that may be used in our devices [40]. Basic crystal oscillators, which are the least accurate and stable out of the three oscillator types mentioned, are stable to 10-20 PPM (parts-per million) [41]. The turn around time for a FTM RTT measurement in one device is around 20 µs. That amount of turnaround time equates to $((20/1,000,000) * 0.000020)$ seconds or 400 ps of drift due to any of the external conditions. Multiplying that by the speed of light in air, we find that the clock drift would only shift a measurement by 0.12 m in the worst case. This amount of clock error is small when compared to the meters of error seen in the actual experiments.

Additionally, if the clock used in the devices was unstable beyond the 20 PPM, we would expect that FTM RTT measurements over time would drift. We saw in Chapter 3.1 that the FTM RTT measurements are consistent over hours of experimentation. Additionally, we saw that the position-dependent error was consistent over repeated collection patterns, meaning it is likely that the clock was stable enough throughout all of the data collections. As a result, it does not seem that clock stabil-

ity is a significant component to the position-dependent error.

Chapter 5

Fixing the Position-Dependent Error and Beyond

This chapter covers ways to deal with the possible components of the position-dependent error in FTM RTT measurements. It explores each of the possible causes listed in Chapter 4, and explains how the error may be ameliorated. Additionally, suggestions about desired setups and devices are made. Finally, the future of FTM RTT is discussed. Throughout this chapter, future work is proposed to continue to improve and understand the FTM RTT position-dependent error.

5.1 Fixing the Position-Dependent Error

The first possible cause of the position-dependent error listed in Chapter 4 is inaccurate time of arrival (ToA) algorithms. We saw that multipath propagation of signals could play a large role in disturbing the correlogram, thus lowering the accuracy of the ToA algorithms. In order to ameliorate the error in these basic algorithms, it will be necessary to ignore as much of the multipath propagation in the correlogram as possible. Since the signals resulting from multipath propagation will be delayed relative to the first arriving signal, leading-edge detection algorithms may be the most desirable. It may be difficult to analyze, because we do not have access to the algorithms that the devices use, but future work should include evaluating which

ToA algorithms perform best in open-space and dense setups. Future research should also include exploring new ToA algorithms that are robust to multipath propagation. The results of this work could provide significantly more accurate and consistent FTM RTT measurements.

The next possible cause of the position-dependent error is inaccurate super-resolution algorithms. Super-resolution algorithms can be useful to increase the resolution of a received signal for a device. The signal can then be analyzed to extract the first arriving signal's ToA. Again, since we do not have access to the algorithms of the devices, it may be difficult to evaluate specific super-resolution techniques. Future work should include comparing super-resolution algorithms to find the one that is the most robust to real world scenarios. If super-resolution algorithms are not yet implemented on these devices, it may be due to the high computational and storage costs. When comparing algorithms, these costs must be taken into account. Fortunately, as the batteries and processors of smartphones and APs are upgraded, they will be better equipped to make use of super-resolution algorithms for more accurate FTM RTT measurements.

Another possible cause of the position-dependent error that was discussed in Chapter 4 is the dependence of FTM RTT on RSS. We saw that, in practice, RSS measurements do not follow the inverse square law. To ameliorate an RSS dependent error, we can use the same techniques that are used in RSS based indoor positioning. One of the popular methods mentioned in Chapter 2 is fingerprinting, where RSS measurements from source APs are mapped over a desired area. A similar technique could be explored in future work, mapping FTM RTT measurements from source APs over the same desired area. While this may provide accurate results, FTM RTT fingerprinting would still be subject to the same drawbacks of RSS fingerprinting. These drawbacks include a lot of work up front to map the space, plus the need to remap the space any time an AP is moved. Future work should include evaluating FTM RTT fingerprinting methods and weighing the accuracy improvements versus the implementation weaknesses.

The last potential cause of the position-dependent error that was discussed in

Chapter 4 is the possibility of clock instability. While it was shown that the measurement error due to clock instability would be relatively small in the worst case, more stable device clocks may be necessary as applications grow for indoor positioning that require more accurate measurements. In this case, future work should include evaluating various crystal oscillator categories to be used in FTM RTT devices, especially categories that are robust to temperature variation, voltage variation, and frequency aging.

5.2 Recommendations

Depending on the applications, choosing a Google Pixel 4 smartphone in conjunction with Google Wi-Fi APs may provide acceptable measurement accuracy. This research has shown several reasons for this combination. First of all, Google pushes software updates to both their smartphones and APs that periodically increase the accuracy of FTM RTT measurements. Additionally, the measurement offset between the Google Pixel 4 and Google Wi-Fi APs was shown to be close to 0 m in Chapter 3. With an offset close to 0 m, a time investment is not required to calibrate and re-calibrate a large offset unlike many other APs [42]. Chapter 3 also showed that the combination of the Google Pixel 4 and Google Wi-Fi AP performed the most consistently after experimentation across the various setups. These experiments were conducted in open-space and dense setups with devices including the Google Pixel 3 smartphone, Google Pixel 4 smartphone, Google Wi-Fi APs and ASUS APs. Future work should evaluate other smartphones, including the new Google Pixel 5 smartphone. Future work should also include evaluation of other APs. This includes the new 2020 model of the Google Wi-Fi AP, as this research utilized the 2016 model.

This research has shown that open-space setups are ideal for achieving the highest measurement accuracy. The recommended setup would include as much open-space as possible to avoid the strong position-dependent error due to multipath propagation of signals. Obviously, this ideal setup may not be achievable for all applications. In the case of dense setups, this research has presented the trade-off, that needs to be

taken into account, between the more complex applications of indoor positioning and the lower accuracy due to increased position-dependent errors.

5.3 The Future of FTM RTT

There are some bright spots in the future for Wi-Fi FTM RTT and indoor positioning. As hardware and software are upgraded and updated, the FTM RTT measurements will become more accurate for individual devices. One major change on the horizon is the adoption of Wi-Fi 6 and the 2021 version IEEE 802.11ax Wi-Fi standard [26]. This Wi-Fi standard provides access to 160 MHz bandwidth channels. As we saw in Chapter 3.8, by increasing the bandwidth of the signal, we can also increase the accuracy of FTM RTT measurements. In particular, we saw that doubling the bandwidth of the signal resulted in the scale of the position-dependent error to decrease by about half. Therefore, we might expect that going from 80 MHz channels to 160 MHz channels could immediately double the accuracy of FTM RTT with this new Wi-Fi standard. Future work should evaluate FTM RTT using Wi-Fi 6 enabled smartphones and APs to understand performance enhancements.

Another development in the indoor positioning space is Wi-Fi Aware, also known as Neighbor Awareness Networking (NAN) [43]. This feature provides RTT measurements between devices and is not limited to a smartphone and AP combination. For example, Wi-Fi Aware can estimate the distance between two different smartphones that are separated by up to 15 m. While this feature may not be viable for some indoor positioning applications, it could simplify others. For instance, Wi-Fi Aware could be utilized to ensure that two robots in a warehouse remain a certain distance apart, without knowledge of their exact positions. If you would like to test this out yourself, Google recently released an app onto the Google Play Store called WifiNanScan where you see Wi-Fi Aware in action using smartphones [44].

Additionally, recently Android S has added the ability to do one-way RTT measurements between a smartphone and an AP. In this case, the ToA is calculated at the AP and the smartphone only once, and provides a measurement estimate based

on that round-trip time. Since it only uses one round-trip signal, the one-way RTT measurements do not remove the turnaround time when the signal is being processed in the AP, so large offsets of up to 2700 m have been seen. Calibration is required to remove these offsets. A major advantage of this method is that, in practice, we have seen that all APs respond to the one-way RTT measurement requests, not just APs that advertise FTM RTT capabilities. With all APs cooperating, one-way RTT could drastically reduce the integration costs of indoor positioning when compared to setups which exclusively use APs that advertise FTM RTT capabilities.

Some limited preliminary results provide an error comparison between one-way RTT, NAN, and FTM RTT. For one data collection, the standard deviation of the error for one-way RTT was calculated to be around 1.5 m, whereas a comparable FTM RTT data collection had a standard deviation of the error of around 0.5 m. The standard deviation of the error for NAN was similar to FTM RTT and was calculated to be around 0.65 m in a small test. Future work should include a more rigorous analysis of both one-way RTT and NAN.

Unfortunately, the future of Wi-Fi FTM RTT also has some uncertainty. The next generation Google smartphone, the Google Pixel 6, is dropping the Qualcomm chipset which is used for FTM RTT measurements. Instead, Google is working with Samsung to produce a new, custom designed chipset known as "Whitechapel" [45]. Depending on the priority of FTM RTT within the development of the new chipset, measurement accuracy may be degraded, if FTM RTT is even included as a feature. Future work should evaluate the performance of the Google Pixel 6, once it is released.

Finally, one important future application of indoor positioning, that has come to light in the midst of the COVID-19 global pandemic, is contact tracing. While it may be too late to implement FTM RTT-based contact tracing for this global pandemic, the push for more accurate contact tracing could be a catalyst for the next wave of FTM RTT improvements. As more research is now being dedicated to indoor positioning, Wi-Fi Aware and methods mentioned in Chapter 2 should be better equipped for contact tracing and other related applications during the inevitable next global phenomenon.

Chapter 6

Summary and Conclusions

In this research, we explored the position-dependent error in FTM RTT indoor positioning from multiple perspectives. We first saw that the time and noise components of the total FTM RTT error were small compared to the position-dependent error. We then found that the position-dependent error extends beyond one dimension into 2D and 3D using the Google Pixel 4 smartphone and the Google Wi-Fi AP. By analyzing the frequency domain, we found that the textures of the position-dependent error in FTM RTT and RSS measurements had similar frequency components, and we investigated possible ways to exploit this. We found that the strength of the position-dependent error depends on the particulars of the testing environment, and that increased multipath propagation of signals results in a wider range of errors. We also compared the position-dependent error across devices. We saw that the Google Pixel 4 performed more consistently than the Google Pixel 3 in identical setups, and the Google Wi-Fi APs performed more consistently than the ASUS APs in identical setups. Finally, we saw that the accuracy of the FTM RTT measurements doubled as the bandwidth of the signal doubled, and that the central frequency of the signal affected the texture of the position-dependent error.

After that, we analyzed the possible causes of the position-dependent error and how to reduce them. We studied time of arrival algorithms that can be efficient ways to estimate when a signal is first received, but that are subject to multipath propagation effects. Therefore, leading-edge detection ToA algorithms may be the

most robust to the delayed multipath propagation of signals, and they warrant further evaluation. Super-resolution algorithms provided a way to increase the resolution of an incoming signal, but at a high computational cost. Since these algorithms may not currently be used in the devices, it is worth testing different algorithms in newer, more powerful devices. The possible dependence of FTM RTT measurements on RSS measurements was discovered, and FTM RTT fingerprinting methods were recommended to ameliorate an RSS-based position-dependent error. Finally, clock stability was analyzed. It was shown that, in the worst case of the device's clock instability, the time drift was not large enough to be a significant component of the position-dependent error.

In the end, a recommendation was made to use the Google Pixel 4 in conjunction with the Google Wi-Fi APs. A recommendation was also made for real world applications to use an open-space setups if possible, as the accuracy of FTM RTT measurements will be highest in the absence of multipath propagation. Future work was proposed to evaluate new smartphones and APs, which could provide even more accurate indoor positioning results. Lastly, the bright, but slightly uncertain future of FTM RTT indoor positioning was discussed. As the Google Pixel 6 moves away from the Qualcomm chipset, new experiments will have to be run. Fortunately, the wide adoption of Wi-Fi 6 will provide an immediate accuracy boost with 160 MHz bandwidth channels, that could hopefully spur the next generation of indoor positioning using FTM RTT.

Bibliography

[1] K. Yu, I. Sharp, and Y. J. Guo, *Ground-Based Wireless Positioning*, ch. 4, pp. 77–113. John Wiley & Sons, Ltd, 2009.

[2] W. Silver, A. Garakani, and A. Wallack, "Apparatus and Method for Detection and Sub-Pixel Location of Edges in a Digital Image." `http://people.csail.mit.edu/bkph/courses/EE693B/Patents/US006408109_Silver_Edges_OCR.pdf`, Patent US 6,408,109 B1, June 2002. Accessed: 2021-04-22.

[3] B. K. P. Horn, "Indoor positioning using time of flight with respect to WiFi access points." `http://people.csail.mit.edu/bkph/ftmrtt_RSSI`. Accessed: 2021-04-02.

[4] I. Std., "IEEE 802.11 - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications." `https://standards.ieee.org/standard/802_11-2016.html`, 2016. Accessed: 2021-02-22.

[5] B. K. P. Horn, "Doubling the Accuracy of Indoor Positioning: Frequency Diversity," *Sensors*, vol. 20, no. 5, 2020.

[6] K. A. Nguyen, Z. Luo, G. Li, and C. Watkins, "A review of smartphones based indoor positioning: Challenges and applications," 2020.

[7] G. Retscher, "Fundamental Concepts and Evolution of Wi-Fi User Localization: An Overview Based on Different Case Studies," *Sensors*, vol. 20, no. 18, 2020.

[8] S. Naghdi and K. OâĂŹKeefe, "Detecting and Correcting for Human Obstacles in BLE Trilateration Using Artificial Intelligence," *Sensors*, vol. 20, no. 5, 2020.

[9] X. Wang, X. Liu, Z. Wang, R. Li, and Y. Wu, "SVM+KF Target Tracking Strategy Using the Signal Strength in Wireless Sensor Networks," *Sensors*, vol. 20, no. 14, 2020.

[10] Z. Liu, B. Dai, X. Wan, and X. Li, "Hybrid Wireless Fingerprint Indoor Localization Method Based on a Convolutional Neural Network," *Sensors*, vol. 19, no. 20, 2019.

[11] K. Sung, D. K. â. Lee, and H. Kim, "Indoor Pedestrian Localization Using iBeacon and Improved Kalman Filter," *Sensors*, vol. 18, no. 6, 2018.

[12] *RSS-eye: Human-assisted Indoor Localization without Radio Maps*, (London, England), International Conference on Communication ICC, 2016.

[13] Y. Landau and B. Ben-Moshe, "STEPS: An Indoor Navigation Framework for Mobile Devices," *Sensors*, vol. 20, no. 14, 2020.

[14] T. Feigl, S. Kram, P. Woller, R. H. Siddiqui, M. Philippsen, and C. Mutschler, "RNN-Aided Human Velocity Estimation from a Single IMU," *Sensors*, vol. 20, no. 13, 2020.

[15] M. M. A. Mohammed, C. He, S. Cincotta, A. Neild, and J. Armstrong, "Communication Aspects of Visible Light Positioning (VLP) Systems Using a Quadrature Angular Diversity Aperture (QADA) Receiver," *Sensors*, vol. 20, no. 7, 2020.

[16] W. Zhang, D. Wei, and H. Yuan, "Novel Drift Reduction Methods in Foot-Mounted PDR System," *Sensors*, vol. 19, no. 18, 2019.

[17] G. Zhang, P. Wang, H. Chen, and L. Zhang, "Wireless Indoor Localization Using Convolutional Neural Network and Gaussian Process Regression," *Sensors*, vol. 19, no. 11, 2019.

[18] A. Minetto, F. Dovis, A. Vesco, M. Garcia-Fernandez, A. Lopez-Cruces, J. L. Trigo, M. Molina, A. Perez-Conesa, J. Ganez-Fernandez, G. Seco-Granados, and J. A. Lopez-Salcedo, "A Testbed for GNSS-Based Positioning and Navigation Technologies in Smart Cities: The HANSEL Project," *Smart Cities*, vol. 3, no. 4, pp. 1219–1241, 2020.

[19] B. K. P. Horn, "Recovering Positioning from Distance Measurements - "Multilateration"." `http://people.csail.mit.edu/bkph/ftmrtt_location`. Accessed: 2021-03-22.

[20] B. K. P. Horn, "Observation Model for Indoor Positioning," *Sensors*, vol. 20, no. 14, 2020.

[21] A. Sari and A. Alzubi, "Chapter 13 - Path Loss Algorithms for Data Resilience in Wireless Body Area Networks for Healthcare Framework," in *Security and Resilience in Intelligent Data-Centric Systems and Communication Networks* (M. Ficco and F. Palmieri, eds.), Intelligent Data-Centric Systems, pp. 285–313, Academic Press, 2018.

[22] W. Burger and M. J. Burge, *Digital image processing: an algorithmic introduction using Java*. Springer, 2016.

[23] A. Manikas, "PN-codes, PN-signals and Principles of Spread Spectrum Comms." `https://skynet.ee.ic.ac.uk/notes/CS_2016_7A_PN_codes_PN_signals_and_SSComms.pdf`. Accessed: 2021-04-26.

[24] I. Fernandez-Corbaton, J. Stein, and R. R. Rick, "System and Method for Wireless Signal Time of Arrival." https://patentimages.storage.googleapis.com/9e/84/2f/d59322bc7a8935/US7474994.pdf, Patent US 7.474,994 B2, Jan. 2009. Accessed: 2021-04-02.

[25] I. Std., "IEEE 802.11ac-2013 - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications–Amendment 4: Enhancements for Very High Throughput for Operation in Bands below 6 GHz.." https://standards.ieee.org/standard/802_11ac-2013.html, 2013. Accessed: 2021-05-05.

[26] I. Std., "IEEE 802.11 - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 1: Enhancements for High Efficiency WLAN." https://standards.ieee.org/standard/802_11ax-2021.html, 2021. Accessed: 2021-04-12.

[27] C. Morhart and E. M. Biebl, "High resolution time of arrival estimation for a cooperative sensor system," *Advances in Radio Science*, vol. 8, pp. 61–66, 2010.

[28] T. Jamil, T. Khanzada, S. Memon, A. Ashfaque, and A. Hashmani, "A Novel Method to Implement the Matrix Pencil Super Resolution Algorithm for Indoor Positioning," *Mehran University Research Journal of Engineering and Technology*, vol. 30, 10 2011.

[29] D. Humphrey and M. Hedley, "Prior Models for Indoor Super-Resolution Time of Arrival Estimation," 2009. In Proceedings of the IEEE 69th Conference: Vehicular Technology Conference (VTC), Barcelona, Spain, 26 to 29 April.

[30] John, "A Comparison of FFT, MUSIC and ESPRIT Methods of Frequency Estimation." http://www.raymaps.com/index.php/a-comparison-of-fft-music-and-esprit-methods-of-frequency-estimation, Jul 2020. Accessed: 2021-04-27.

[31] C. Fernandez-Granda, "Super-Resolution." https://cims.nyu.edu/~cfgranda/pages/OBDA_spring16/material/superresolution.pdf. Accessed: 2021-04-26.

[32] E. Fishler and H. V. Poor, "Estimation of the Number of Sources in Unbalanced Arrays via Information Theoretic Criteria," *CoRR*, vol. abs/cs/0501058, 2005.

[33] W. Li, W. Liao, and A. Fannjiang, "Super-resolution limit of the esprit algorithm," *IEEE Transactions on Information Theory*, vol. 66, no. 7, pp. 4593–4608, 2020.

[34] G. V. Tsoulos, *Adaptive Algorithms*, pp. 49–299. Wiley-IEEE Press, 2001.

[35] R. Mohammadi-Ghazi and O. BÃijyÃijkÃűztÃijrk, "Sparse generalized pencil of function and its application to system identification and structural health

monitoring," in *Health Monitoring of Structural and Biological Systems 2016* (T. Kundu, ed.), vol. 9805, pp. 86 – 94, International Society for Optics and Photonics, SPIE, 2016.

[36] S. Chrétien and H. Tyagi, "Multi-kernel unmixing and super-resolution using the Modified Matrix Pencil method," *CoRR*, vol. abs/1807.02862, 2018.

[37] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 7, pp. 984–995, 1989.

[38] R. Kumarasiri, K. Shamaileh, N. Tran, and V. Devabhaktuni, "An Improved Hybrid RSS/TDOA Wireless Sensors Localization Technique Utilizing Wi-Fi Networks," *Mobile Networks and Applications*, vol. 21, 06 2015.

[39] S. Tiwari, D. Wang, M. Fattouche, and F. Ghannouchi, "A Hybrid RSS/TOA Method for 3D Positioning in an Indoor Environment," *ISRN Signal Processing*, vol. 2012, 03 2012.

[40] H. Zhou, C. Nicholls, T. Kunz, and H. Schwartz, "Frequency Accuracy & Stability Dependencies of Crystal Oscillators," tech. rep., Carleton University, 2008.

[41] O. Mancini, "Tutorial Precision Frequency Generation Utilizing OCXO and Rubidium Atomic Standards with Applications for Commercial, Space, Military, and Challenging Environments." `https://www.ieee.li/pdf/viewgraphs/precision_frequency_generation.pdf`. Accessed: 2021-04-11.

[42] B. K. P. Horn, "FTM RTT AP Ratings." `https://people.csail.mit.edu/bkph/other/WifiRttScanX/FTM_RTT_AP_ratings.txt`. Accessed: 2021-04-12.

[43] "Wi-Fi Aware overview." `https://developer.android.com/guide/topics/connectivity/wifi-aware`, 2021. Accessed: 2021-04-20.

[44] "WifiNanScan App." `https://play.google.com/store/apps/details?id=com.google.android.apps.location.rtt.wifinanscan&hl=en_US&gl=US`, 2021. Accessed: 2021-04-20.

[45] B. Schoon, "Weighing the potential pros and cons of Pixel 6 ditching Qualcomm for Whitechapel." `https://9to5google.com/2021/04/04/google-whitechapel-pixel-6/`, 2021. Accessed: 2021-04-12.