Error Analysis of Carrier Phase Positioning Using Controlled Reception Pattern Antenna Arrays

by

Joshua Starling

A thesis submitted to the Graduate Faculty of
Auburn University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Auburn, Alabama
May 6, 2017

Keywords: GPS, Carrier Phase Positioning, CRPA, Jamming, Signal Processing

Copyright 2017 by Joshua Starling

Approved by

David M. Bevly, Chair, Professor of Mechanical Engineering
Lloyd S. Riggs, Professor of Electrical and Computer Engineering
Stanley J. Reeves, Professor of Electrical and Computer Engineering

Abstract

This thesis analyzes and provides algorithms that protect against various carrier phase positioning errors caused by anti-jamming algorithms commonly used to position in a jamming environment. These position errors include effects from the Least-Mean-Square algorithm distorting measurements and the loss of accuracy when a anti-jamming receiver is in the presence of a jammer. While there are a number of different methods in which anti-jamming performance can be accomplished in a GPS receiver, the focus of this thesis is on algorithms that are implemented using multiple antenna, also known as controlled reception pattern array (CRPA) antennas. Carrier phase positioning is a highly accurate position solution, capable of positioning a user down to the centimeter level. However, the increased accuracy also makes the receiver susceptible to new sources of error not typically found when using the ranging code, due to the lower amount of noise on the measurement. Therefore, the effect and magnitude of these errors must be examined.

CRPA anti-jam algorithms attenuate interference and/or strengthen desired signals by leveraging the array's spatial, temporal, and frequency difference between each of the antennas in combination with phase shifting and scaling each antenna's received signal. While this operation can remove or weaken the interference signal, algorithms such as Least-Mean-Square can have a biasing effect on a phase lock loop's estimate of the Doppler frequency due to time variant phase shift applied by the algorithm. Depending on algorithm parameters, the time variant phase shift can cause position drifts of up to a centimeter per minute. The use of a normalization process is shown as a method to remove the phase shift, and therefore the position bias, while not degrading the anti-jam performance.

The second focus of this thesis is the ability of the CRPA anti-jam receiver to maintain carrier phase position accuracy, on the order of centimeters of less, in a jamming environment.

In this experiment, a four element receiver is tested with a wide range of jamming strengths and the standard deviation of Doppler measurements are averaged across all channels to provide an estimate of the accuracy of the carrier phase position solution. Because the tracking loop can be tuned to handle various amounts of noise, the phase lock loop (PLL) bandwidth was varied and individually tested with each jamming strength. At the limits of the CRPA anti-jam receiver's nulling ability, a position solution was computed using a typical receiver PLL bandwidth, a reduced bandwidth, and an extremely low bandwidth with the aiding of an IMU. The results show that the bandwidth must be reduced significantly to maintain carrier phase position accuracy and the tracking loop must be aided by an IMU in order to obtain millimeter accuracy in the high jamming environment.

Acknowledgments

I would first and most of all like thank God for the blessings of my life and for the skills and opportunity to do this work. Secondly, I would like to thank my beautiful bride Katie. I'm deeply thankful for all of your encouragement, support, and patience for all the late nights and weekend work I've had to do to get here. I could not have done this without you. Likewise, I would like to give thanks to my parents for their encouragement and support throughout my entire life. I also owe a great amount of thanks to my adviser, Dr. David Bevly. Thank you for this opportunity and challenging me to do this work. Out of all of my co-workers, I would like to thank most of all Russell Powell for creating an awesome GPS simulation environment. If it wasn't for your work, it would have taken me much longer to finish this program. Lastly, I would like to thank the rest of my co-workers in the GAVLab for your support throughout my time in the lab.

Table of Contents

List of Figures

xiii

Chapter 1

Introduction

1.1 Motivation

High precision GPS is progressively becoming a common technique in all types of positioning and navigation applications. Because GPS has a variety of applications, a GPS receiver could be used for low risk applications, such as agriculture or surveying, all the way to high risk applications, such as autonomous vehicles or military missions. Since high precision GPS applications cover the entire range of risk levels, the GPS receiver must be thoroughly tested to ensure robustness and that the position solution will be correct and meet application requirements. While the applications of the high precision GPS have increased over time, the cost of software defined radios, which easily allow broadcasting at any frequency, has decreased and the crowding around the L frequency band has increased. Controlled pattern reception array, also known as CRPA, antennas provide robust jamming protection by phase shifting and scaling the incoming signals from multiple closely spaced antennas in order to change the effect gain pattern of the array. With the ability to change the gain pattern, signals of interest can be amplified and interference signals can be attenuated to allow for signal tracking in hostile environments. However, the applied phase shifting, scaling, and mixing of multiple signals can distort the carrier phase measurement and cause erroneous positioning. Due to these potential dangers and the robust jamming protection provided by CRPA antennas, the quality of a CRPA receiver's carrier phase position solution must be studied for any possible sources of error as well as research possible improvements to limit the negative effects of anti-jamming on carrier phase positioning. This work explores potential issues that must be addressed for positioning in a hostile navigation environment.

1.2 Prior Work

Since the issue of suppressing noise while strengthening a signal is not specific to anti-jam GPS applications, decades of research has been dedicated to this field, including the specific implementation of multiple antennas. Many of these noise suppression algorithms were able to be converted from fields, such as audio recording or RADAR, to provide GPS anti-jam algorithms a rich literature background. The most straightforward, while not as practical, is the use of deterministic spatial filtering algorithms, such as beam steering and null steering, which leverge the spatial difference between elements. Qamar [17] investigated the performance of various null steering techniques on the bases of null depth, null width, maximum number of placed nulls, as well as computational complexity. In Qamar's work the angle of arrival of the hostile signal was assumed to be estimated using common estimations techniques such as ESPRIT, however, this thesis will focus on anti-jam algorithms that are adaptive and do not require the angle of arrival of the interference signal.

In the field of adaptive anti-jam spatial filtering, various metrics, such as total signal power or signal to noise ratio (SNR), are used to constrain the weighting selection to mitigate the interference source while satisfying the desired metric. Zoltowski [20] developed and tested an optimal solution algorithm designed to minimize the total output power in the time domain using only estimated statistics of the incoming signals. Zoltowski implemented the algorithm such that it could be used as a blind null former or steered to maintain receptivity in the direction of satellites. Both implementations were tested in various jamming environments and were evaluated on receptivity towards the satellites and null depth. Applebaum [2] and Li [14] researched the performance of algorithms that determine the weighting array to maximize the signal to noise ratio and the signal to interference plus noise ratio respectively. In both of these studies, the angle of the arrival of the satellites were used to maintain receptivity such that the respective ratios are maximized. Widrow [19] showed that a recursive error driven algorithm could be used to determine suitable weights for interference mitigation. In the context of GPS, an error driven algorithm is possible due to the tracking

loops maintaining accurate estimates of the received signal such that a desired signal can be generated.

In addition to the spatial only filtering mentioned above, filtering can be done in two domains to increase interference suppression. Using time delay taps to allow for temporal filtering, Frost [11] showed a spatial and temporal filtering algorithm that was constraied to minimize the total output power with a receptivity constraint in a desired direction. Using the spatial and frequency filtering, Moore and Gupta [15] [8] [9] showed the process and performance of minimizing the total output power in the frequency domain. The authors showed that spatial and frequency filtering is comparable to spatial and temporal filtering while also having the option to perform sup-optimal weight determination in order to reduce computational complexity.

While the works mentioned above provide interference mitigation techniques to a GPS receiver, carrier phase measurment integrity is not always maintained when implementing these algorithms. Kim [13] showed position errors due to the group delay bias when using STAP algorithms and carrier phase differential GPS, as well as a process to remove the group delay bias. De Lorzeno [5] of Stanford University deteremined that STAP algroithms and mutual coupling can bias the code and carrier phase measurments and researched the use of a look up table to deterministicly remove the bias. Lastly, Vagle [16] researched the accuracy of the Doppler measurement, phase lock indicator, and the carrier to noise ratio when using algorithms such as power minimization or power minimization distortionless response. These works show that while common CRPA anti-jam algorithms can provide robust protection, the algorithms can also produce measurement errors. The mentioned works are only a small section of the available research done on CRPA algorithms and the effects of using CRPA algorithms when performing carrier phase positioning. This thesis aims to use these prior work in conjunction with the works of this thesis to improve the field of high accuracy GPS positioning with anti-jam capability.

1.3 Contributions

The contributions of this thesis are three-fold. First, a general overview of GPS CRPA algorithms are detailed and explained in concept and process. Second, the work of De Lorenzo is furthered by analyzing the position accuracy and the effect of phase shifting the reference element through unconstrained recursive algorithms, such as the Least-Mean-Square algorithm. Thirdly, the carrier phase position accuracy of a CRPA receiver as a function of jammer strength is evaluated. The evaluation is done by observing the jitter in the Doppler frequency estimate based on jammer strength and phase lock loop bandwidth using the power minimization algorithm for both static and dynamic trajectories. The contributions of this thesis to the research field are as follows:

- Error analysis of unconstrained recursive algorithms while performing carrier phase positioning

- The effect of applying the normalization process while performing carrier phase positioning

- Detailing receiver performance as a function of jammer strength and receiver phase lock loop bandwidth

1.4 Thesis Outline

Chapter 2 provides detail on the GPS signal structure as well as the acquisition and tracking process. This is done in order to gain an understanding of the inner workings of the receiver for the purpose of understanding how algorithms can distort or bias the receiver's measurements and position solution. Chapter 3 gives a brief description of CRPA anti-jam algorithms that fit into the categories of Spatial Adaptive Filtering, Space Time Adaptive Filtering, and Space Frequency Adaptive Filtering. Chapter 4 discusses the simulation environment in which the results of this paper were obtained, as well as show the baseline

performance of the algorithms discussed in Chapter 3. In Chapter 5, the error associated with unconstrained recursive algorithms is evaluated in terms of phase shift and position error. In Chapter 6, the performance of a CRPA receiver using power minimization is evaluated at multiple jammer signal strengths and phase lock loop bandwidths. Lastly, Chapter 7 will go over future work and conclude this thesis.

Chapter 2

Basic Introduction of GPS Signals Processing

The first GPS satellite was launched February 22, 1978 for civilian and military use with what is now known as the L1 signal. Since that day and as of this thesis, the number of GPS signals has gone from two to six and the number of satellites in space has increased form one to thirty-one. While the main purpose of these additional signals are the same as the original two, to position a user, the additional signals were created to give the user designed abilities that were not available when only using the original signals. With the implementation of the military L2 signal, which operates on a different frequency than the original L1 signal, ionosphere correction could be made using the difference in delay between the two signals. Additionally, the L2C signal gives the civilian user access to a signal with an updated code structure, to allow for longer integration time and ionosphere corrections. Lastly, the L5 signal was created with the purpose of better multipath mitigation as well as better ionosphere correction capabilities. The purpose of this chapter is to give the reader a preliminary understanding on the components of the GPS signal as well as the acquisition and tracking process in which a GPS receiver processes the received signal to provide ranging measurements.

2.1 GPS Signal Structure

All six different GPS signals, for the most part, follow the same signal structure. There is a carrier wave which falls in the very high frequency range, a satellite specific ranging code that can be used to determine the satellites in view and provide ranging, and a satellite navigation data message carrying information on satellite and the general constellation. The

modulation of these three components in a single signal can be seen along with basic signal information in Figure 2.1 for the civilian L1 signal.
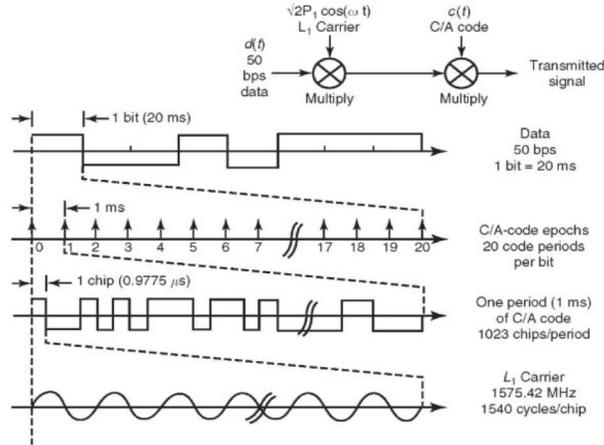
Figure 2.1: GPS L1 Signal Structure [1]

The navigation message is the slowest of the three with a rate of one bit every twenty milliseconds and is encoded onto the ranging code. The ranging code is the second slowest component. For example, the L1 C/A code operates at one complete sequence every millisecond, and is encoded onto the carrier wave. Lastly, the carrier wave is the fastest of all the components and is a sine or cosine wave with a frequency in the L band range of 500 to 1500 MHz. As of the writing of this thesis, there are three different center frequencies at which GPS is broadcast: 1575.42 MHz, 1227.60 MHz, and 1176.45 MHz termed L1, L2, and L5 respectively. These high frequencies give each of the signals a relatively small wavelength, on the order of centimeters, which will allow for high accuracy positioning. On each of the individual frequencies there are two signals: an inphase and quadrature signal. Due to the fact that the inphase signal is broadcast on a cosine wave and the quadrature signal is broadcast using a sine wave, the two signals are 90 degrees out of phase with each other. This 90 degree phase difference and the lack of cross-correlation between the ranging codes on the two signals allows the signals to be transmitted and received without any considerable interference between the two.

7

The ranging code is of significant importance to the GPS signal due to the fact that it plays a large role in the positioning accuracy as well as sets the signal characteristics. The ranging code allows the user to determine what satellites are in view, determine the range, sets the bandwidth of the spread spectrum signal, and allows for the receiving of multiple signals at the same center frequency. For the L1 civilian signal, the ranging code is known as the course acquisition code, or C/A code. The C/A code is a 1023 chip long sequence of ones and negative ones and has a chipping rate of 1.023 MHz, which equates to one full C/A code sequence every millisecond [6]. The special properties of the C/A code and all other GPS ranging codes stem from the fact that all of the codes are Gold codes. The benefit of the Gold codes is the special auto and cross correlation properties of the codes as well as the seemingly random sequence of the codes, hence the alternative moniker of pseudorandom noise (PRN). Gold codes were used for the GPS ranging code because there is very little cross-correlation between two Gold codes [3], as seen in Figure 2.2 which displays the cross correlation between PRN 5 and PRN 15 for all delay combination. Additionally, Gold codes only correlate with themselves in time with less than a single chip offset, seen in Figure 2.3 which displays the autocorrelation property of PRN 19 with itself. Due to the lack of cross-correlation, a GPS receiver can receive and track each of the GPS satellites signals despite each signal being broadcast at the same center frequency.
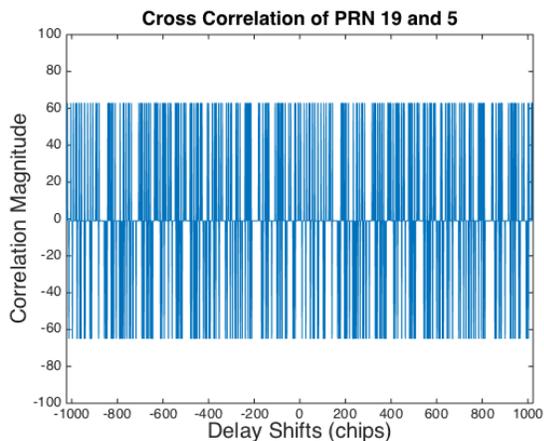
Figure 2.2: Cross Correlation of C/A Codes   Figure 2.3: autocorrelation of C/A Codes
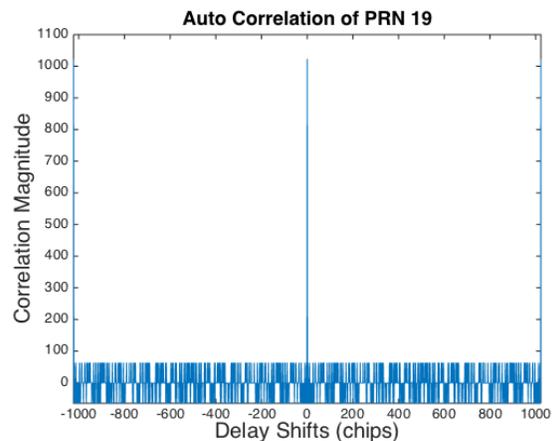
When the ranging code is encoded onto the carrier wave, the spectral content of the carrier signal is changed. In the frequency domain, the signal changes from an impulse at the center frequency to a sinc squared function with a bandwidth equal to that of the chipping rate, which can be seen in Figure 2.4 below. The use of spread spectrum coding allows for increased anti-jamming capability because all of the energy of the signal is no longer at a single frequency value. The benefits to this can be extended even further and seen when comparing the civilian L1 signal to the military L1 signal.
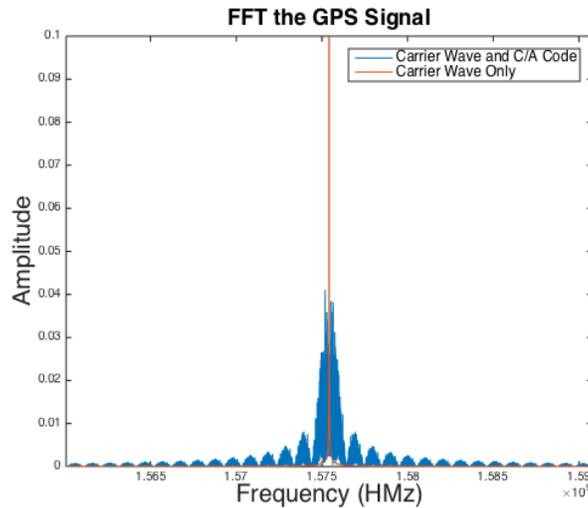
Figure 2.4: FFT of the GPS Signal

The military code uses the P(Y) code which is a combination of two different code sequences. The P code is a publicly available code with a chipping rate of 10.23 MHz, ten times faster than the C/A code, and repeats at the start of every week. The P code is then encrypted with the Y code in order to prevent use from any unauthorized users. Because the P(Y) code has a chipping rate ten times faster than the C/A code, the military signal's bandwidth is ten times larger. This larger bandwidth better protects against narrow band jamming, allowing for a stronger radio frequency interference resistance of around 7 dB compared to the civilian signal [6].

The last component of the GPS signal structure is the satellite navigation message. The navigation message contains the GPS date, GPS time, satellite status, satellite health, orbital

information used to calculate the satellite's position as a function of time, and almanac data which contains the information and status of every other satellite in use. The entirety of a satellites navigation message, also known as a frame, is 1,500 bits, and each frame consist of five subframes each consisting of 300 bits. In the frame, the information is divided up so that subframe one details information on the GPS time and date as well as satellite heath information, subframes two and three contain the ephemeris detailing the satellite's orbital parameters, and subframes four and five contain information on other satellites. Because subframes four and five cannot contain the entire almanac at once, each frame consist of only a twenty-fifth of the complete almanac. To broadcast an entire frame at fifty bits per second takes thirty seconds and all twenty-five frames needed to obtain the entire almanac takes twelve and a half minutes.

2.2 GPS Acquisition

As mentioned in the previous section, a GPS receiver can determine the satellites in view due to the special auto and cross correlation properties of the Gold codes used on the GPS signal. To view the correlation between the received signal and the receiver's generated replica satellite signal at least one complete sequence of the code is needed, which is one millisecond for the C/A code. However, the chip that the received signal started on, also known as the code phase, and the Doppler frequency due to satellite and/or user motion are both unknown. Because of these two uncertainties, a GPS receiver must serially generate a replica code with every possible code delay for every physically possible Doppler frequency then multiply and sum the replica and received signal together to determine if a correlation exist. When this is done, there exist a three dimensional plane of the correlation results, as depicted in Figure 2.5. In the correlation plane one axis represents the chip delay, another represents the Doppler shift, and the vertical axis represents the correlation value. In Figure 2.5, the recorded signal contains PRN 2 which had a Doppler frequency of -2,225 Hz and a code phase of about 79 chips. This is determined by observing that at a code delay of

10

79 chips and a Doppler frequency of -2,225 Hz there is a correlation while every other code phase and Doppler frequency failed to produce a strong correlation. It should be noted that the signal was sampled twelve times faster than the chipping rate which produces a code delay range of 0 to about 12,000 instead of 0 to 1023.
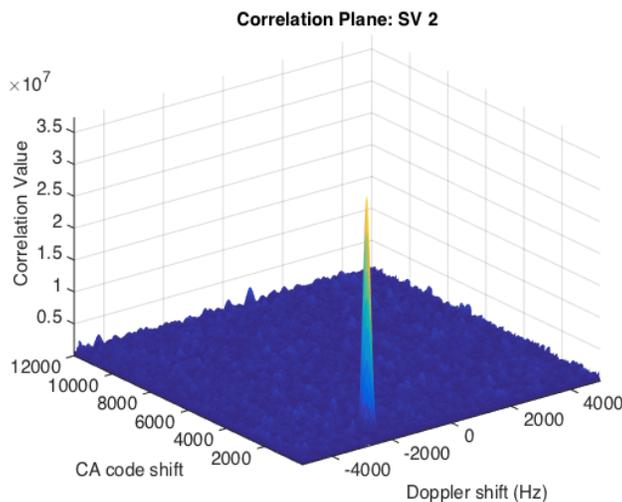
Figure 2.5: Acquisition Plane of PRN 2

2.3   Costas Tracking Loop

Once acquired, the GPS receiver has one signal processing goal: to remove the carrier wave and the ranging code to determine the unknown navigation message. From the acquisition process, the receiver has already determined what satellites are in view, the Doppler frequency of each of those satellites, and the ranging code phase at the start of the data collection. However, the carrier phase is still unknown and the receiver must be able to maintain all of these estimates throughout time to continue to decode navigation data bits. To accomplish these tasks, the GPS receiver uses two control loops: the phase lock loop (PLL) and the delay lock loop (DLL), both depicted in Figure 2.6. The phase lock loop, as its name implies, is designed to maintain phase lock with the received carrier signal such that both the phase and frequency of the received signal are known. The second system, the delay lock loop, is used to maintain a phase lock on the ranging code as it is received.

When these control loops function properly and provide the correct estimate of the carrier frequency, carrier phase, and code phase, the carrier and code can be removed so that the navigation message decoded.
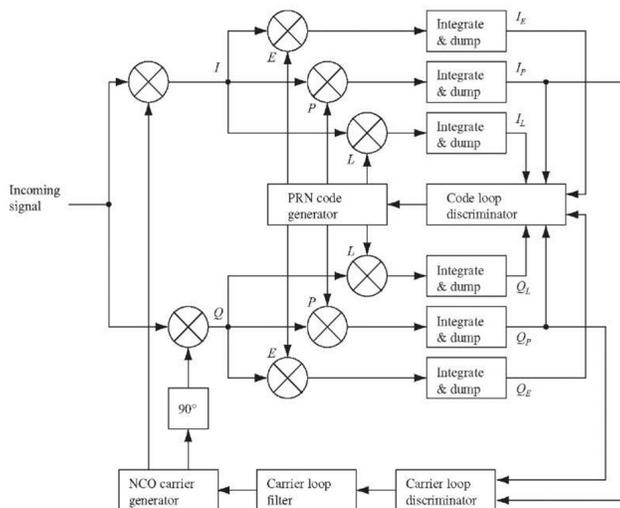
Figure 2.6: Block Diagram of a GPS Tracking Loop [3]

2.3.1 Phase Lock Loop

As mentioned above, the goal of the phase lock loop is to maintain correct estimates of the carrier phase and frequency using a control law that converts phase error to a change in the replica's carrier frequency. The phase lock loop's numerically controlled oscillator, or NCO, then provides a replica signal with the estimate of the signal's frequency and an initial guess of the phase that is mixed with the incoming signal. The mathematical equation for mixing of two sine waves can be seen using Equation (2.1).

$$A\sin(2\pi f_A t) \otimes B\sin(2\pi f_B t) = \frac{(A-B)}{2}\left[(\cos(2\pi(f_A - f_B)t) + \cos(2\pi(f_A + f_B)t))\right] \quad (2.1)$$

If the carrier estimates are correct, the resulting mixed signal should be that of a cosine wave at baseband and a cosine wave at twice the frequency. The mixed signal is then processed

through a low pass filter which removes the higher frequency signal while preserving the baseband signal. Due to the changing bit value of the navigation message, the signal can undergo 180° phase shifts which an ordinary PLL would be sensitive to. To protect the tracking loop, a COSTAS loop is used to provide 180° phase shift insensitivity [3]. There are two arms in a COSTAS loop implementation, an in-phase arm and a quadrature arm. The in-phase arm is created by using the above described process and the quadrature arm is created using the same replica signal at the estimated frequency, except that its phase is shifted forward by ninety degrees. When the estimated phase is correct, the quadrature arm of the tracking loop should have a low correlation to the incoming signal while the in-phase arm will have a high correlation.

In order to use a controller to maintain a phase lock, the phase lock loop must obtain some measurement of feedback to determine when error is present. This is done by using the in-phase and quadrature arm's relative correlation value. Ideally the in phase arm should have a higher correlation value while the quadrature arm should have a correlation value an order of magnitude less than the in phase arm, as seen in Figure 2.7. However, the correlation value of the quadrature arm will increase as the received signal and the in phase signal become out of phase. This loss of phase alignment or drift from phase alignment can occur from a change in the incoming signal's frequency, imperfections in the receiver clock, or a flip in the navigation message data bit which changes the carrier phase.

Using this knowledge, a measurement of phase error in the phase lock loop NCO can be measured by using the in-phase and quadrature prompt value in a discriminator. The typical discriminator used in a GPS receiver is the arc tan discriminator, as seen in Equation (2.2), due to the fact that the discriminator can fully measure error $\pm 90°$ and due to the fact that there is a linear relationship between phase difference and the reported phase error.

$$\phi_{error} = \arctan\left(\frac{Q_P}{I_P}\right) \qquad (2.2)$$

This phase error is then fed through the phase lock loop control law which converts error to a change in the numerically controlled oscillator (NCO) carrier frequency.
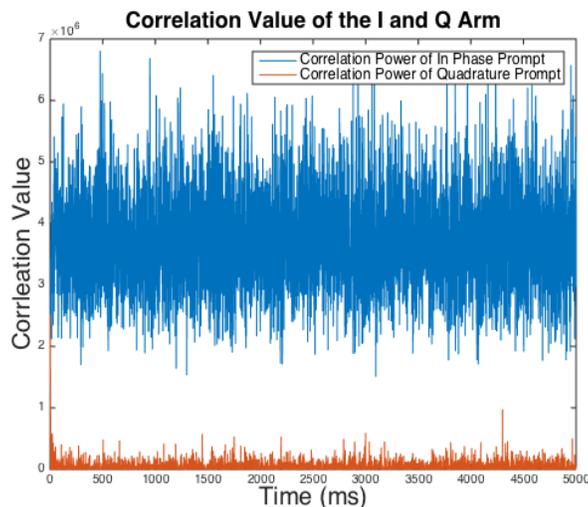
Figure 2.7: Correlation Values of the In-Phase and Quadrature Arms

In a typical receiver, the controller for the phase lock loop is a second order controller with a bandwidth of 20 Hz [3]. The order of the controller primarily is set by the input function type in which the controller must be able to track, such as step, ramp, parabola, etc. These input functions represent changes in the incoming signal's frequency and phase and correspond directly to the user's velocity and acceleration. With knowledge of the application, such as pedestrian, ground vehicle, or aircraft, the suitable phase lock loop controller order can be determined. For static, pedestrian, and ground vehicles applications a second order loop filter is suffcient. With this order of controller, the phase lock loop is able to track step and ramp function like changes in the incoming signal with out steady state error [12].

The controller bandwidth, however, is an engineering trade off between noise filtering and response time. As the bandwidth of the controller is increased, the response time increases which allows for better tracking when the user is dynamic. This dynamic improvement comes at the cost of less noise filtering which will provide a nosier signal. Conversely, if the controller bandwidth is set to a low bandwidth, the controller becomes robust in noisy

environments at the cost of not being able to respond fast enough to track the dynamic signals. In the case that the bandwidth is set too low, the phase lock loop will lose lock and begin to "search" for the signal. With the controller unable to keep up with the change in phase, the phase error will oscillate between positive and negative creating a sawtooth-like estimate of an incorrect Doppler frequency, such as that seen in Figure 2.8.
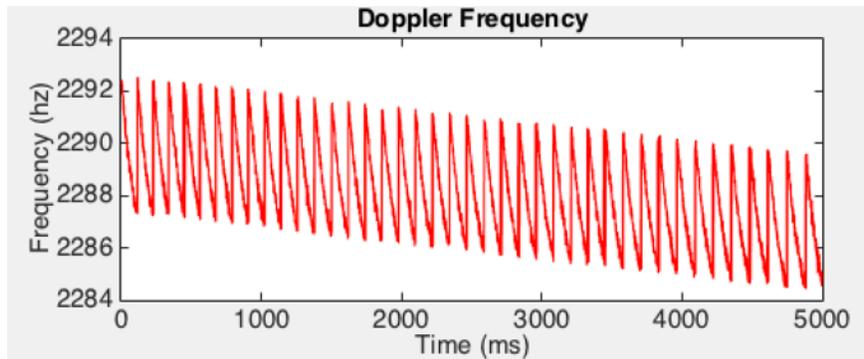
Figure 2.8: PLL Tracking Result Using a Bandwidth of 1 Hz

2.3.2 Delay Lock Loop

The delay lock loop's purpose is to maintain alignment of the incoming ranging code with the receiver's replica of the ranging code. Given an exact time aligned replica, when multiplied together, the two codes should cancel each other removing any phase changes on the incoming signal due to the ranging code bit flips. A single prompt measurement is not enough to track the code phase error because a less than typical amount of correlation or no correlation is ambiguous to whether the replica was either too "early" or too "late". Therefore, the delay lock loop operates using the known ideal correlation function of the Gold code, as seen in Figure 2.9. In the ideal correlation plot, the correlation creates a triangle like shape as the chip delay increases. If the replica is an entire chip early or late, there is little correlation. As the replica gets closer to being promptly aligned, the correlation increases linearly until the replica starts to be misaligned in the late direction. Once the chip delay is in considered "late" direction, the correlation decreases linearly until the replica is an entire chip late.

15
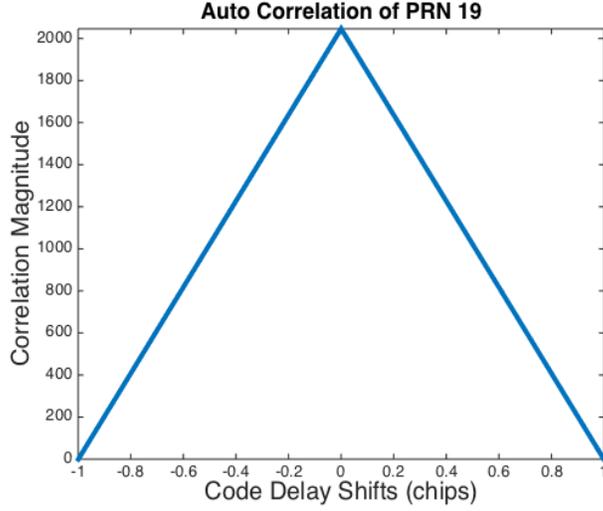
Figure 2.9: Ideal Auto-Correlation Plot of PRN 19

The code phase error is calculated using the correlation value of two time shifted replicas of the PRN code, the "early" and "late" replica. The amount in which the replicas are "early" or "late" can be varied for different applications purposes. Typically half a chip offset from the estimated code phase is commonly used [3]. Using the correlation value of the early and late replicas, the delay lock loop can create a controller that ensures code alignment using a second order controller using the error determined by the normalized non-coherent discriminator in Equation (2.3).

$$e_{code} = \frac{\left(I_E^2 + Q_E^2\right) - \left(I_L^2 + Q_L^2\right)}{\left(I_E^2 + Q_E^2\right) + \left(I_L^2 + Q_L^2\right)} \tag{2.3}$$

When the early and late replica have the same correlation value, there is a zero change input to the delay lock loop NCO. If the late correlator is higher than the early replica, then the delay lock loop NCO will increase the frequency in order to catch up and return to a zero error steady state. The use of the normalized non-coherent discriminator excels because any error larger than a chip saturates, limiting the range of error to plus or minus one, and provides a linear range of error before being saturated [3].

2.4 C/A Code Ranging and Positioning

In order to calculate a user position, the following variables must be known for each satellites: the satellite position, time in which the signal was transmitted, and the time in which the signal was received. The first variable, the satellite position, is determined using the orbital parameters equations detailed in the navigation message that are provided and maintained by the Control Segment. Second, the transmit time is determined by using a combination of the navigation message and the ranging code. In the navigation message there exists a time stamp in the form of the Z-count which provides the GPS time every 1.5 seconds at the start of each subframe. Using the Z-count as the base, the transmit time can be determined by adding the Z-count, the whole number of navigation data bits since the last Z count, the whole number of ranging codes since the last whole navigation bit, and the partial ranging code since the last whole ranging code. With knowledge of the bits per second of the navigation message, duration of the ranging code, and the chipping rate of the ranging code, the transmit time can be determined. Knowing that a chip is transmitted for about a thousandth of a millisecond, that the ranging code has an accuracy of 0.1% of a chip, and assuming the signal travels at the speed of light in a vacuum, the accuracy of the time of flight can be determined to be about one nanosecond, or 0.3 meters. Lastly, the receive time can be determined using the internal clock of the receiver.

Using the time transmitted and received, the range between the user and the satellite can be determined by multiplying the speed of light by the the differences in time, as seen in Equation (2.4).

$$r(t) = c[t_u(t) - t^s(t - \tau)] \tag{2.4}$$

In Equation (2.4), the current receiver time is $t_u(t)$, the transmit time is $t_s(t)$, and the time of flight is shown as τ.

The problem arises that the satellite's and the receiver's clock have no way to agree on the current time and both clocks will drift at different rates. Part of this problem is resolved due to the fact that the clocks inside the satellites are high quality and the drift of the satellite clock is modeled by the Control Segment of the GPS program. These clock models are provided in the navigation message to allow the user to correct the transmitted satellite time. While the use of high end clocks are practical for the satellites, it is not practical in most cases for the receiver's clock. The receiver clock bias effects the calculated range by biasing the range, as shown in Equation (2.5).

$$\rho = r + b_t c \tag{2.5}$$

The receiver clock bias can simply be added as an unknown and solved when determining the user position. This solved receiver clock bias will correct for the error in the receiver clock's time offset error as well as the drift of the receiver's clock over time.

2.5   Carrier Phase Ranging

The basic operating equation of carrier phase ranging can be seen in Equation (2.6).

$$\dot{\rho}(t) = -f_{Doppler}\lambda\Delta t \tag{2.6}$$

This equation calculates the change in the satellite range, or range rate, using the measured Doppler frequency of the signal, the wavelength of the transmitted signal, and the amount of time the range is to be propagated forward. Using the Doppler frequency of the replica signal obtained through the phase lock loop, a high precision range can be recursively updated to provide an accurate measurement of the range. Because Equation (2.6) only updates the range, an initial high precision range must be calculated. The high precision range is calculated by using the LAMBDA Method which is a minimum variance estimator to

calculate the integer ambiguity, N, or the number of whole cycles between the receiver and the satellite, shown in Equation (2.7) [4].

$$\Phi(t) = r + I_\phi + T_\phi + c\delta t + N\lambda + \epsilon_\phi \qquad (2.7)$$

The accruacy of carrier phase positioning is largly limited to how accruactly a receiver can measure the carrier phase. Using the technology and methods currently available, the carrier phase can be measured with the precision of one to five percent of a cycle. Using the L1 signal which has a wavelength of 19.03 cm, gives ranging precision of two millmeters to one centimeter [6]. While the carrier phase is a more accurate measurement, the carrier phase is completely ambiguous to how many cycles of the carrier wave have elapsed. Where the ranging code and navigation message contains the transmit time to provide the time of flight, there is no way to tell if there are a hundred or a million cycles between the user and satellite. Therefore, if the receiver ever loses lock of the signal, the receiver must re-estimate the number of cycles between the user and the satellite. As mentioned previously, this section only serves to be a brief introduction to the topic, not an extensive detailing, and to show how the carrier phase measurements are generated and used to update the user to satellite range. For more information on carrier phase positioning and its implementation see references [6], [7] and [10].

2.6 Chapter Conclusion

This chapter discussed the fundamental basic principles of the signal processing in a GPS receiver. The GPS signal structure was extensively covered in order that GPS signal acquisition and tracking could be fully understood. With an understanding of the GPS signal structure, the mechanics of the tracking loop, such as the phase lock loop and delay lock loop, were discussed. Finally, the process in which ranging measurements, using both

the C/A code and the carrier phase measurement, and the expected accuracy of the two ranging measurements were discussed.

Chapter 3

Anti-Jam Techniques

In the field of GPS, anti-jamming can mean a number of different techniques. Fundamentally, anti-jamming is any technique which will either strengthen the desired signal to be more robust to interference or the attenuation of the received interference. Due to the focus of this thesis on CRPA antenna carrier phase positioning, the content of this chapter will cover anti-jamming techniques that leverage multiple antennas. This chapter will cover the various categories in which individual algorithms can be placed, how individual algorithms operate, and the operational advantages and disadvantages of each algorithm.

3.1 Spatial Adaptive Processing

The first major anti-jam algorithm category to be covered is Spatial Adaptive Processing, also known as SAP. Spatial Adaptive Processing techniques leverage the fact that the received signals between each of the CRPA antenna elements will have, due to their close spacing, a phase difference of less than a cycle. Using the existing phase shift and the use of receiver applied phase shifts and amplitude f, the weighted signals can be combined to provide constructive and destructive interference. Through the phase shifting and scaling of the signals, the effective gain pattern of the antenna can be changed from the ideal isotropic gain pattern of Figure 3.1 to some arbitrary pattern such as of Figure 3.2.

It should be noted that the individual elements of the array are referred to as either a reference element or an auxiliary antenna, where there is a sole reference element and multiple auxiliary elements. The difference between these two types of elements is that the reference element will be the phase center of the array while the signals of the auxiliary elements are the signals that are used to create constructive or destructive interference and
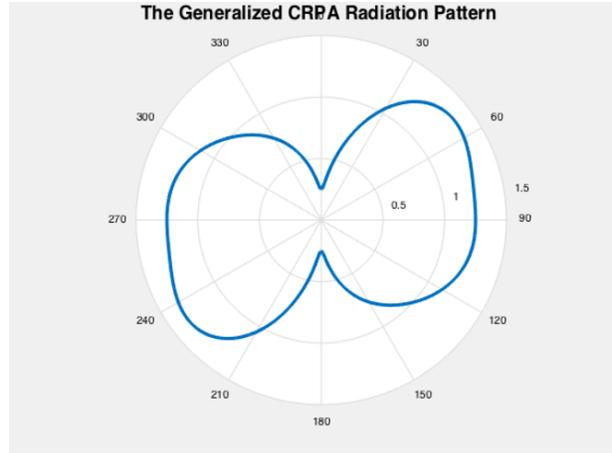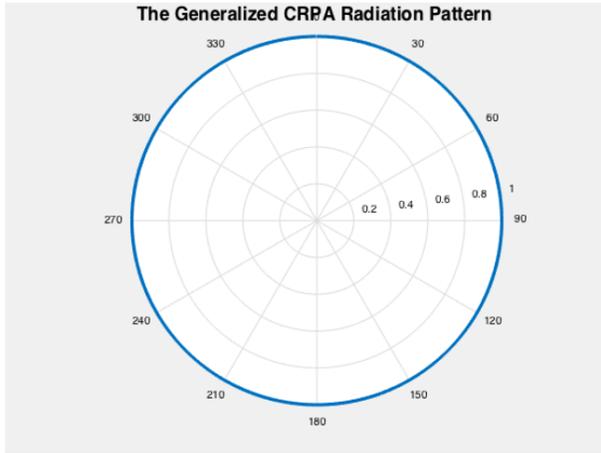
Figure 3.1: Isotropic Gain Pattern of FRPA

Figure 3.2: Isotropic Gain Pattern of CRPA

create the desired gain pattern of the array. The selection of the reference element can be arbitrarily set to any element but can also be set by the geometry of the array, i.e. an element in the absolute center of the array such as the middle element of Figure 3.3.

Figure 3.3: Example of a CRPA Antenna

3.1.1 Beam Steering

Beam steering is a deterministic Spatial Adaptive Processing technique in which all required information must be known a priori. The beam steering weight calculation for an arbitrary element k can be seen in Equation (3.1).

$$W_k = e^{-j\frac{2\pi\hat{r}d_k}{\lambda}} \tag{3.1}$$

There are three variables which must be known in order to perform beam steering: the wavelength of the expected signal, λ, the distance between the element k and the reference element, d_k, and a three dimensional unit vector that points in the desired beam direction, $\hat{r}$, also called the look direction. The design of the beam steering equation is to increase the effective gain pattern in a single direction with no constraint on the remaining receptivity.

The applications for beam steering are instances in which the received desired signal is weak or, if the interference signal is weak enough, a small increase in the desired signal power is sufficient enough to continue tracking. Using this algorithm, the amplitude of the incoming signal can be increased by the number of antennas used. For example, a four element array can increase the receptivity in the desired direction by 6 dB or 8 dB if a seven element CRPA is used and perfectly pointed to the signal of interest. Using Equation (3.1) only a single main beam will be created, with some number of side lobes depending on the array geometry. While a GPS receiver does track multiple satellites simultaneously, each tracking channel can digitally form beams towards a different satellite as long as there is enough computational power to process each channel separately. Using an analog system, only a single beam could be directed with Equation (3.1). The gain pattern of a beam steered array can be seen in Figure 3.4 in which a beam was steered at an azimuth of -45 ° using an element spacing of half a wavelength, or about 9.5 centimeters.

While the beam direction is directly set by the beam steering equation, the beam width is set by the antenna spacing used in the array. As the antenna spacing is increased, the

beam width is decreased. Conversely, as the antenna spacing is decreased the beam width is increased. Using a spacing of half a wavelength, a beam width of roughly 60° is achieved as in Figure 3.4. Figure 3.5 shows that when the element spacing is decreased to a quarter of a wavelength the observed beam width increases to approximately 85°.
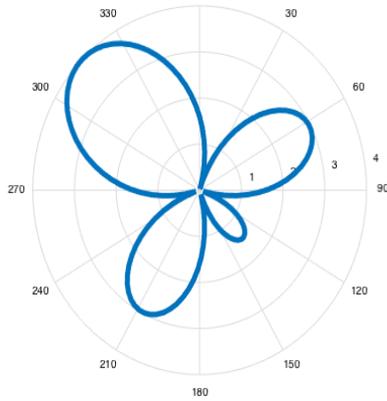
Figure 3.4: Beam Steering Gain Pattern Pointed at -45° Using Half Wavelength Spacing
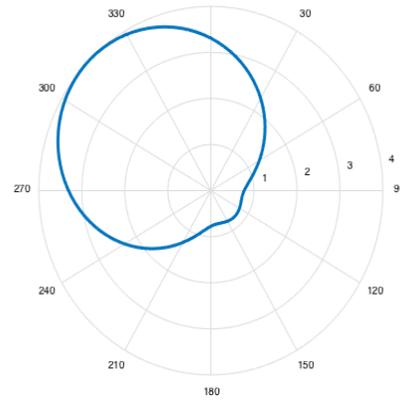
Figure 3.5: Beam Steering Gain Pattern Pointed at -45° Using Quarter Wavelength Spacing

3.1.2 Null Steering

Contrary to beam steering, null steering is designed to decrease receptiveness in a desired direction using the same operating principles as beam steering. While null steering cannot guarantee an increase in a desired signal's power, null steering provides greater attenuation of a single jammer than beam steering provides strengthening of the desired signal. For a four element CRPA receiver, a single null can be placed with roughly -40 dB of attenuation instead of a 6 dB of gain that is possible with the beam steering. The null steering algorithm functions similarly to that of beam steering. The algorithm first phase shifts each of the auxiliary elements to be in phase with the reference element for the desired null direction, as given in Equation (3.2).

$$a_k = e^{-j\frac{2\pi \hat{r} d_k}{\lambda}} \qquad\qquad (3.2)$$

24

Upon phase shifting, the algorithm then scales each of the auxiliary elements so that the combined auxiliary elements are the negative of the reference element using Equation (3.3).

$$b = \begin{bmatrix} 1 & \dfrac{-1}{k-1} & \cdots & \dfrac{-1}{k-1} \end{bmatrix} \tag{3.3}$$

Through the application of null steering, any signals being emitted from the single desired null direction should interfere with one another upon signal combining and cancel each other out using the weight selection of Equation (3.4).

$$W = a \cdot b \tag{3.4}$$

This produces the gain pattern similar to Figure 3.6 where a null was placed at an azimuth of -45° with an element spacing of half a wavelength. Similar to beam steering, the null width can be varied by changing the element spacing. When the element spacing is decreased down to a quarter of a wavelength the null width is increased from 60° to 80° as seen when comparing Figure 3.6 to Figure 3.7.
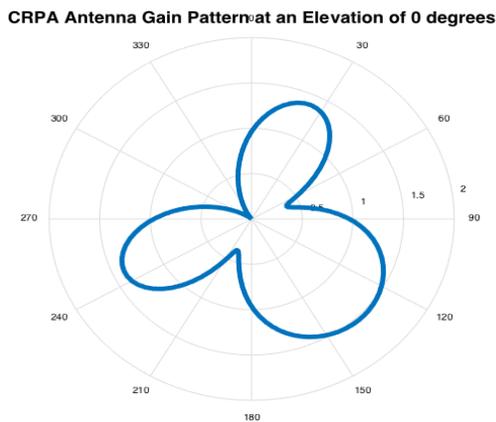
Figure 3.6: Null Steering Gain Pattern Pointed at -45° Using Half Wavelength Spacing
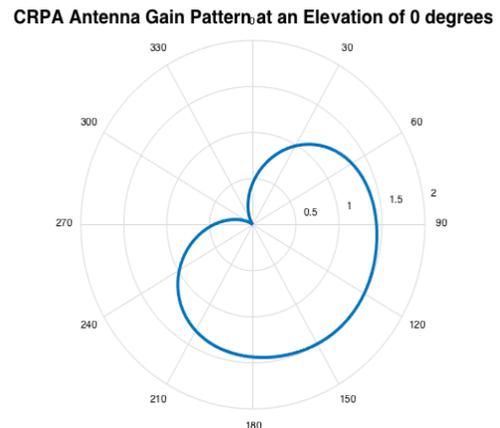
Figure 3.7: Null Steering Gain Pattern Pointed at -45° Using Quarter Wavelength Spacing

Unlike beam steering which can be digitally applied on each of the tracking channels, when forming nulls there is only a finite amount of nulls the algorithm can place, also known

as the system's degrees of freedom. This implementation of null steering can only place a single null in one direction. A SAP implementation of null steering can place up to $K - 1$ null, however this method of null steering is limited to placing all of the nulling ability to a single direction.

A large disadvantage of the null steering technique is that there is no control over the gain pattern besides where the null is placed. Receptiveness to a satellite is therefore uncontrolled and completely dependent upon the geometry, such as antenna spacing and the placement of the desired nulls. Should a satellite be located where a sympathy null is, such as the null in Figure 3.6 at an azimuth of 68°, the satellite's signal will be degraded or will not be able to be used for positioning while null steering is engaged.

3.1.3 Power Minimization

Moving from deterministic algorithms to adaptive algorithms, the next algorithm to be discussed is the power minimization technique. Power minimization is an optimal solution technique which uses no a priori information but uses an estimate of the signal characteristics to determine the optimal weights. The algorithm is designed to, as the name suggests, minimize the total output power of the signal. The algorithm leverages the fact that GPS is received below the thermal noise floor while any jammer will be above the noise floor. Therefore, power minimization should attempt to select weights that removes any jammer or any other strong radio frequency interference while not removing the GPS signal.

The power minimization algorithm can be seen in Equation (3.5) where only two variables are required to compute the weights [20].

$$W_{opt} = -\frac{1}{\delta^T R_{xx}^{-1} \delta} R_{xx}^{-1} \delta \tag{3.5}$$

First, an estimate of the signal characteristics, also known as the autocorrelation matrix (R_{xx}), and second a constraint vector, as seen in Equation (3.6), which is a $1xK$ column vector where K is the number of antennas.

$$\delta = \begin{bmatrix} 1 & 0 & ... & 0 \end{bmatrix}^T \tag{3.6}$$

The use of the constraint vector maintains a unity weight on the reference element so that no phase shifting or scaling is applied to the reference element. The estimation process for the autocorrelation is described in Equation (3.7).

$$R_{xx} = \frac{1}{M} \sum_{m=1}^{M} X_m X_m^H \tag{3.7}$$

In Equation (3.7), X_m is a $Kx1$ column vector of the measured signal from each of the K antennas and M total samples are used in the estimation.

Power minimization contains $K - 1$ degrees of freedom, where K is the number of antennas. Therefore, the more antennas available the more spatial nulls that can be placed. While the use of degrees of freedom is a helpful tool to measure the number of nulls that can be placed, it should be noted that a single jammer can consume more than one degree of freedom if the jammer is exceedingly high powered. Also, it should be noted that two weak jammers geographically placed close to one another could collectively only consume one degree of freedom; therefore, for SAP algorithms degrees of freedom should be thought of as spatial nulls and not jammer specific nulls.

The main advantage to power minimization is that it is a blind beamformer. The algorithm requires no information on the array dimensions, angle of arrival of the desired signal, the frequency of the desired signal or characteristics of the interference signal. This allows for easy implementation even if the array and application specifications are unknown. However, the advantage to power minimization is also its disadvantage. Because the algorithm doesn't require knowledge of the satellite signal's angle of arrival, the algorithm cannot ensure receptiveness to the satellite. This could lead to the degrading of a valid satellite's signal or the nulling of a satellite. Another disadvantage to power minimization is its use of the autocorrelation matrix. Because the autocorrelation matrix will change over time, the

autocorrelation matrix must be re-estimated periodically which increases the computational requirements. This required periodical re-estimation and matrix inversion can possibly limit the number of antennas, depending upon the processing power of the receiver. Also, the use of an estimated, instead of the actual autocorrelation matrix, provides a source of error that can lead to misaligned nulls and the partial acceptance of a jammer signal.

3.1.4 Least-Mean-Square

The next adaptive process is the Least-Mean-Square (LMS) algorithm by Widrow and Stearns [19]. Least-Mean-Square is a recursive error driven algorithm which will form beams or nulls as needed to decrease the error between the weighted input and the desired signal. While in most signal processing cases, generating a desired signal would be an issue as it requires the receiver to have a priori knowledge of the signal it wants to track before it is received, this is not the case for a GPS receiver. Due to the code and carrier NCO, estimates of the incoming carrier wave's frequency and phase as well as the code phase are known and can be used to generate a desired signal. A block diagram depiction of how the algorithm is implemented can be seen in Figure 3.8.
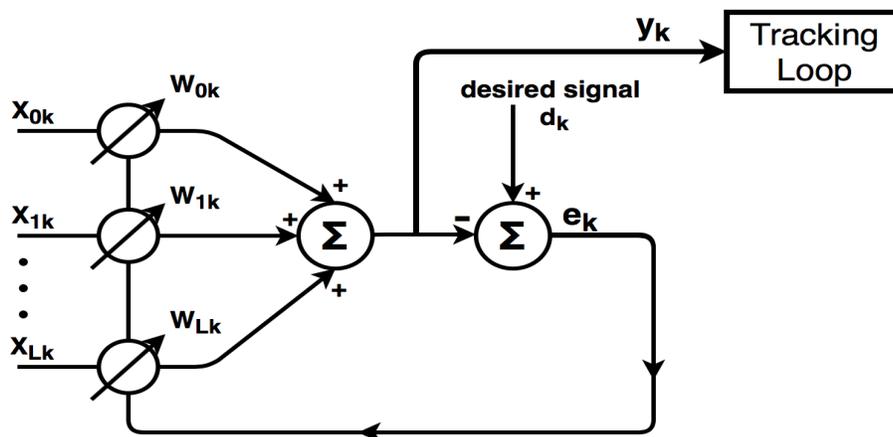
Figure 3.8: Block Diagram of the Least-Mean-Square Algorithm

The algorithm updates the current weights using four variables: μ, X_k, d_k, and W_k, shown in Equation (3.8).

$$W_{k+1} = W_k + \mu X_k (d_k - W_k^T X_k) \tag{3.8}$$

W_k is the current weights as a column vector of length K, X_k is the current measured signal values as a column vector of length K, d_k is the current sample of the desired signal, and μ is the learning coefficient. μ is analogous to the amount of damping the algorithm has as it converges to a solution. Should μ be too large, the algorithm can go unstable causing the weights to increase in magnitude each iteration and approach infinity. By contrast, if μ is too small, the weights will never diverge from the initial weights meaning the algorithm cannot adapt to the environment. Therefore μ must be set to allow for a reasonable response time to obtain the solution but not be unstable. μ can either be hand tuned through trial and error or an appropriate theoretical value can be bounded by using Equation (3.9) where tr represents the trace operation.

$$0 < \mu < \frac{2}{3tr(R_{xx})} \tag{3.9}$$

To show how the LMS algorithm works and the effects of tuning μ, a simple experiment was performed. A sine wave was simulated to be transmitted and received by a linear array where the array was not orthogonal to the source such that there is a phase difference between the two antennas. The desired signal was set to be a cosine wave with same frequency and phase as that of the transmitted signal and the initial weighting was set to be zero on each antenna using a μ value of 0.2. Figure 3.9 shows the weights recursively iterating toward the solution and reaching a steady state solution with some amount of jitter around the solution. Figure 3.10 then shows the desired cosine signal in blue and the weighted sum of the two antennas' signals in orange. The μ was hand tuned to 0.4 so that the response of the algorithm was damped such that there were no oscillations about the solution before

converging. If the μ is increased to 0.93, as seen in Figure 3.11 and Figure 3.12, the weights will converge faster, 50 iterations instead of 200, but suffer from large oscillations before decaying to a steady state solution.
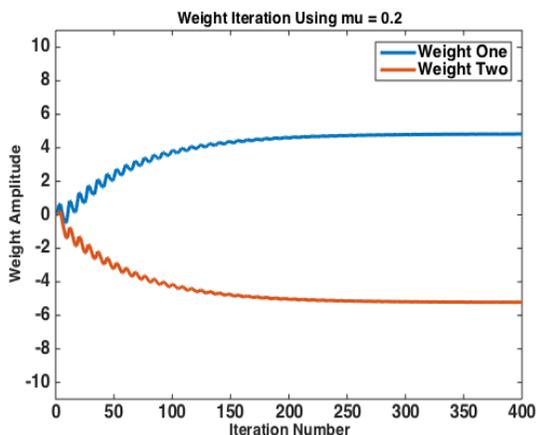
Figure 3.9: LMS Weighting Vs Time for $\mu = 0.2$
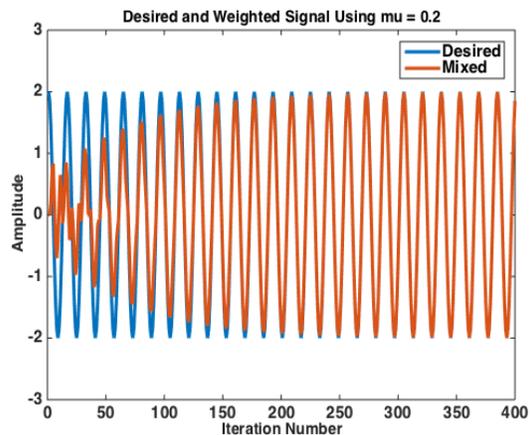
Figure 3.10: LMS Signal Value Vs Time for $\mu = 0.2$
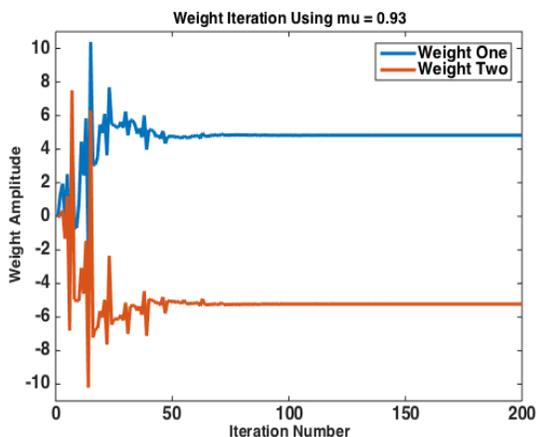
Figure 3.11: LMS Weighting Vs Time for $\mu = 0.93$
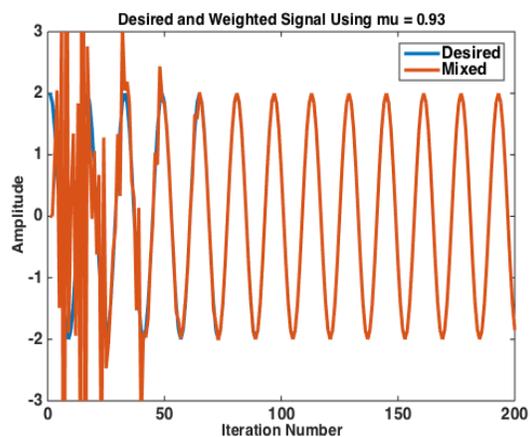
Figure 3.12: LMS Signal Value Vs Time for $\mu = 0.93$

Because the LMS algorithm is a recursive algorithm, the initial weights must be set to some value. A possible initial weighting could be unity weighting for the reference element and zeros for the auxiliary elements or the weights obtained from another algorithm such as beam steering or power minimization. The first option would create an initial gain pattern similar to that of a FRPA antenna while the second option initializes to a anti-jamming

ready gain pattern. It should be noted that the LMS algorithm uses the Newton-Raphson method to determine minimas that minimize the error between the weighted signal and the replica signal. As with other implementations of the Newton-Raphson method, the algorithm is susceptible to local minimums preventing the algorithm from converging to the optimal solution. In the case of initializing the weights to some other CRPA beam steering or null steering algorithm, the LMS may not converge to another solution because the current weight is acceptable and produces little error between the desired and weighted signal. Conversely, in the case of initializing the weights to unity and the remaining weights to zero, the gain pattern may form nulls not directly in the direction of an interference source or form beams not directly at the satellites. This is caused by the fact that the algorithm is designed to minimize error and not maximize the carrier-to-noise ratio of the tracked signal.

To understand how the gain pattern changes over time when using the LMS algorithm, the gain pattern was plotted individually throughout a 104 second long simulation. For this experiment the weights are updated one thousand times a second giving a total of 104,000 weightings arrays for each channel. The initial weights were selected to be a unity weight on the reference element and zeros on the auxiliary elements. This can be verified by seeing that the initial iteration's gain pattern is similar to that of an isotropic FRPA gain pattern. The array's gain pattern was plotted every thirteen seconds, or every 13,000 iterations, and can be seen in Figures 3.13 - 3.21. As it can be seen by comparing Figures 3.13 and 3.14, the algorithm quickly converges towards a solution with beams and nulls starting to be formed. As the algorithm converges toward the solution, the change in the gain pattern slows but is still converging towards the solution as the nulls are deepened, as seen when comparing Figures 3.14 and 3.17. Then, as the error between the desired and weighted signal reaches zero, there is little to no change in the array's gain pattern, as seen when comparing Figures 3.20 and 3.21.

The largest advantage of the Least-Mean-Square algorithm is the recursive implementation of the algorithm. Since the autocorrelation matrix is not required to perform the
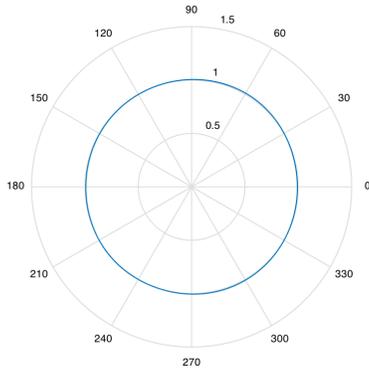
Figure 3.13: LMS Gain Pattern at Iteration
Number 1
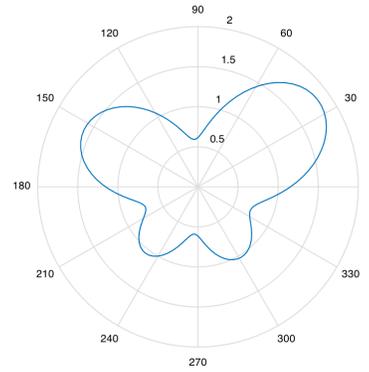
Figure 3.14: LMS Gain Pattern at Iteration
Number 13001
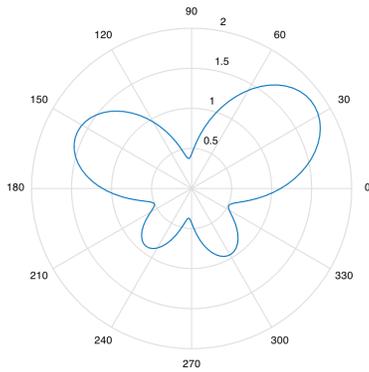
Figure 3.15: LMS Gain Pattern at Iteration
Number 26001
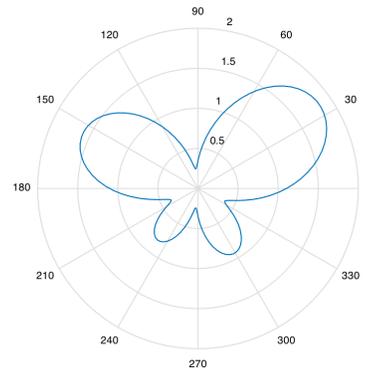
Figure 3.16: LMS Gain Pattern at Iteration
Number 39001

LMS algorithm, the algorithm can run quickly and efficiently. However, due to the recursive nature of the algorithm, if the jammer environment changes, the algorithm will have a transient stage where it is not providing the correct weights to fully null the jammer. Also, should the algorithm lose lock, the validity of the estimates required to create the desired signal become questionable.
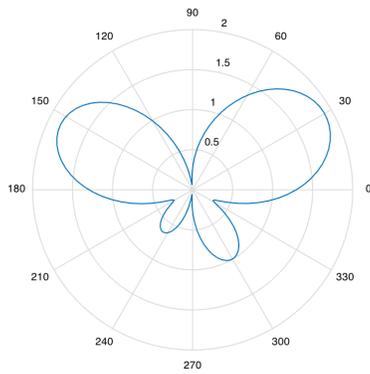
Figure 3.17: LMS Gain Pattern at Iteration Number 52001
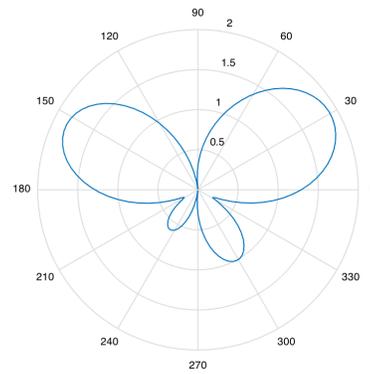
Figure 3.18: LMS Gain Pattern at Iteration Number 65001
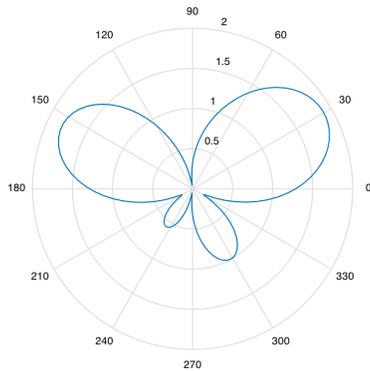
Figure 3.19: LMS Gain Pattern at Iteration Number 78001
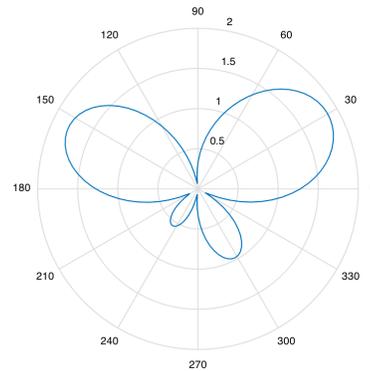
Figure 3.20: LMS Gain Pattern at Iteration Number 91001
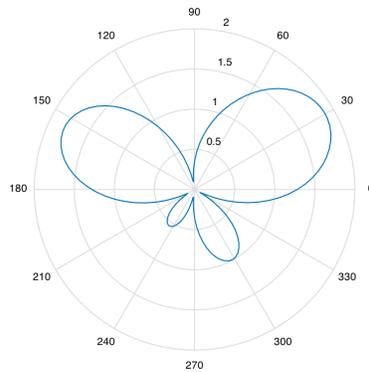
Figure 3.21: LMS Gain Pattern at Iteration Number 104001

3.2 Space Time Adaptive Processing

Space Time Adaptive Processing (STAP) improves upon Spacial Adaptive Processing with the incorporation of time delay tap measurements. These tap measurements are un-weighted measurements from some delay of time passed, as seen in the STAP block diagram of Figure 3.22. With the incorporation of taps, anti-jam algorithms can alter the frequency response of the array as well as the spatial gain pattern of the array. This allows for certain frequency ranges in a direction to be nulled out while not nulling all signals from that direction.
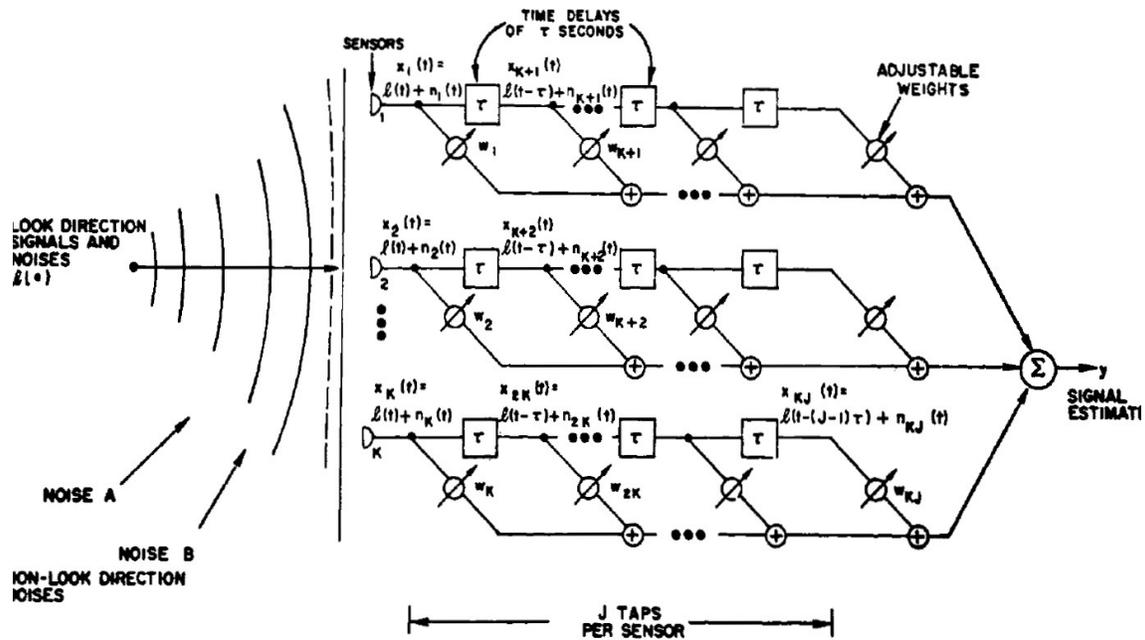
Figure 3.22: Block Diagram of STAP Algorithm [11]

The use of taps also increases the degrees of freedom of the system from K-1 degrees of freedom to KJ-1, where K is the number of antennas and J is the number of time delay taps. However, the KJ-1 degrees of freedom is more of a theoretical upper limits of the possible degrees of freedom. Most algorithms use constraint equations to ensure a linear phase or an all pass response which will decrease the degrees of freedom from KJ-1 to KJ-J. It should also be noted that the bandwidth of the jammer now effects the number of degrees of freedom

consumed. In other words, as the bandwidth of the jammer increases so does the consumed degrees of freedom. For example, a jammer signal whose frequency domain is an impulse only consumes a single degree of freedom while a wide band jammer will consume up to J degrees of freedom [8].

3.2.1 Power Minimization

The power minimization technique can be adapted to use time delay taps in order to perform Space Time Adaptive Filtering. The implementation of the STAP power minimization is very similar to that of the SAP technique, where the only necessary change is to the input vector and the autocorrelation matrix. With the addition of the extra samples, the power minimization algorithm now has more degrees of freedom and the ability to localize jammers in the frequency domain as well as in the spatial domain, allowing for more advanced nulling. However, with the addition of the time delay taps, the computational load is also increased since the autocorrelation matrix size is increased from KxK to KJxKJ. The burden of additional antennas becomes even greater as the the size of the autocorrelation increases by a factor of J for each antenna added.

3.2.2 Least-Mean-Square

Similar to the power minimization technique, the Least-Mean-Square algorithm can also be adapted to Space Time Adaptive Filtering. The block diagram of the STAP Least-Mean-Square algorithm can be seen in Figure 3.23 where the additional measurements are incorporated into the weighting process. While the STAP version of the Least-Mean-Square algorithm does require an increase in computational processing, the increase is not as substantial as the difference between SAP and STAP power minimization due to the lack of autocorrelation estimation in the Least-Mean-Square algorithm. This allows for a greater degree of freedom system and additional nulling localization while not increasing the computational load significantly.

35
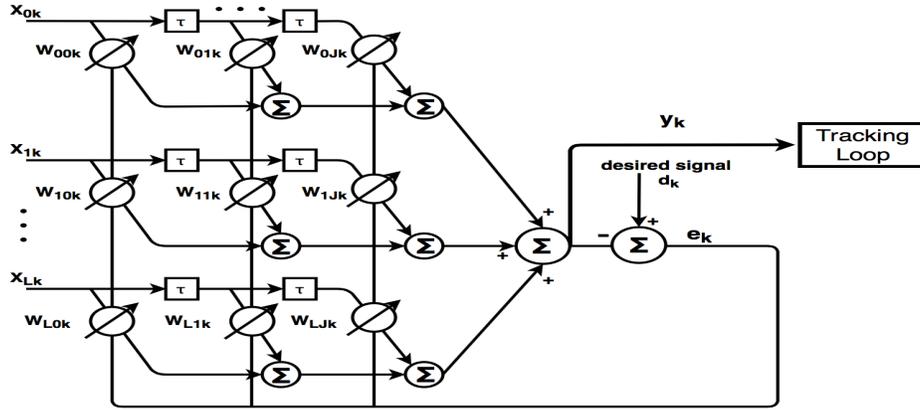
Figure 3.23: Block Diagram of the STAP Least-Mean-Square Algorithm

3.2.3 Minimum Variance Distortionless Response

The Minimum Variance Distortionless Response (MVDR) is a STAP algorithm that was created by Otis Frost as a constrained total output noise power minimizing LMS algorithm that maintains a chosen frequency response in a single direction [11]. Essentially, this algorithm provides the benefits of power minimization while constraining receptiveness towards satellites. MVDR operates using two constraint vectors shown in Equation (3.10) and Equation (3.11).

$$
f = \begin{pmatrix} 1 \\ . \\ . \\ . \\ 1 \end{pmatrix}
\tag{3.10}
$$

$$
C = \begin{bmatrix} c_1 & c_2 & ... & c_J \end{bmatrix}
\tag{3.11}
$$

Equation (3.10) provides the distortionless constraint using a J length column vector and Equation (3.11) is a JxKJ matrix that constrains the weights to satisfy the constraint equations. Using these equations, the optimal solution can be seen in Equation (3.12) which uses

36

Equation (3.10) and Equation (3.11) in conjunction with an estimate of the autocorrelation matrix.

$$W_{opt} = R_{xx}^{-1}C\left[C^T R_{xx}^{-}1C\right]^{-1}f \qquad (3.12)$$

The Minimum Variance Distortionless Response can also be used recursively so that the autocorrelation matrix never has to be estimated, shown in Equation (3.13).

$$W_{k+1} = P\left[W_k - \mu y_k X_k\right] + F \qquad (3.13)$$

Equation (3.13) Instead of using the autocorrelation matrix, the recursive algorithms operate such that it "learns" the statistics of noise arriving from directions other than the direction in which the algorithm is constrained to maintain a beam towards, also called the look direction. Because receptivity is desired in the direction towards the GPS satellites, the recursive algorithm requires the angle of arrival of the satellites .

It should be noted that in order to perform the recursive algorithm the following two equations, Equation (3.14) and Equation (3.15), are used.

$$F \triangleq C\left(C^T C\right)^{-1}f \qquad (3.14)$$

$$P \triangleq I - C\left(C^T C\right)^{-1}C^T \qquad (3.15)$$

These equations are combinations of the constraint vector and constraint matrix and are used to increase the readability of the recursive equation. The full derivation of the optimal and recursive equations can be seen in the original paper written by Frost [11].

The advantage to the Minimum Variance Distortionless Response is that it is a combination of the algorithms previous discussed in this chapter. Where power minimization was not able to maintain a beam on a desired satellite, MVDR constrains the minimization process

to maintain receptivity to a satellite. Additionally, MVDR can be used either as an optimal solution which uses an estimated autocorrelation matrix, similar to that of power minimization, or can be implemented recursively, similar to LMS. However, the implementation of STAP algorithms come at the cost of higher computation complexity where KJ samples are required instead of simply K samples. This computational complexity becomes even more strenuous when the optimal solution method is being used and a KJxKJ autocorrelation matrix must be periodically estimated.

3.3   Space-Frequency Adaptive Processing

Contrary to Spatial Adaptive Processing or Space Time Adaptive Processing where the total output power of the time domain signal is minimized, Space-Frequency Adaptive Processing (SFAP) attempts to minimize the power in the frequency domain of the weighted input signals. Because of the difference in domains in which the algorithm is applied, the anti-jam operating principles are slightly different for SFAP than its time domain counter parts. The largest difference is how the degrees of freedom are consumed. While SAP have a limit on the total number of spatial nulls that can be formed and STAP is limited in the number of spatial nulls that are formed as well as the bandwidth of the interference signal, SFAP is limited on the amount of frequency nulls that can be formed. This limit operates independently of where the interference is emitted from and allows for SFAP to excel in environments where there are multiple jammers that operate at the same frequency. Similar to STAP, the bandwidth of the interference signal effects the degrees of freedom of SFAP where wide band signals will consume more degrees of freedom than a narrow band signal.

The largest benefit of SFAP over STAP is the available option of suboptimal solutions. A suboptimal solution, as the name suggests, provides a slightly degraded anti-jam solution but the computational load of the algorithm is minimized when compared to an optimal STAP or SFAP solution. The load minimization is achieved by calculating weights for each individual frequency bin. The total number of frequency bins is set by the sampling

frequency and number of samples in the block of data used. The width of a frequency bin is determined by the Nyquist frequency, or half of the sampling frequency, divided by the number of samples used. While this requires LxN number of weights to be calculated, the autocorrelation matrix required to calculate the weights is limited to LxL instead of LNxLN, where L is the number of antennas and N is the number of frequency domain bins.

The process in which SFAP is implemented can be seen below [8]. It should be noted that the use of windowing the time domain signal before the transformation to the frequency domain is commonly done when performing SFAP, where the windowed sample value is denoted as ω_n [9]. The purpose for the windowing process is to localize any hostile interference signals to fewer bins to minimize the degrees of freedom consumed by the algorithm when minimizing the total output frequency power. The best windowing methods have been determined to be ones which produce low sidelobes [15]. For this thesis the Blackman windowing technique was used. Other viable windowing techniques that could be used are the uniform and Hamming windowing techniques. In Equation (3.16), the windowed input samples of the lth antenna are converted from the time domain to the frequency domain using a discrete Fourier transformation.

$$\widetilde{X}(l,k) = \sum_{n=1}^{N} \big(\omega_n \cdot X(l)\big) e^{-j\frac{2\pi}{N}(n-1)(k-1)} \tag{3.16}$$

$$l = 1, 2, ..., L$$

$$k = 1, 2, ..., N$$

This process is done using a block of N number of samples and the index k represents the frequency bin.

The autocorrelation for each frequency bin is then estimated using the kth bin from each antenna, as shown in Equation (3.17).

$$\widetilde{R}_{xx}(k)_{li} = \frac{1}{M} \sum_{m=1}^{M} \widetilde{X}_l(k)\widetilde{X}_i(k)^H \tag{3.17}$$

$$l = i = 1, 2, ..., L$$

The frequency domain weights for the kth bin are calculated using Equation (3.18) with the constraint array u as described in Equation (3.19) which is a 1xL column vector.

$$\widetilde{h}(k) = \left(\frac{\widetilde{R}_{xx}(k)u}{u^T \widetilde{R}_{xx}(k)^{-1}u} \right)^{-1} \tag{3.18}$$

$$u = \begin{bmatrix} 1 & 0 & ... & 0 \end{bmatrix}^T \tag{3.19}$$

Then, each of the N frequency bins are weighted and combined, Equation (3.20).

$$\widetilde{y}(k) = \widetilde{h}(k)^T \widetilde{X}(k) \tag{3.20}$$

When each frequency bin has been weighted and combined, the weighted frequency domain signal is transformed back to the time domain, Equation (3.21), to be processed through the GPS tracking loops.

$$y = \frac{1}{\omega_i N} \sum_{n=1}^{N} \widetilde{y} e^{-j \frac{2\pi}{N}(n-1)(k-1)} \tag{3.21}$$

Note that in Equation (3.21) the time domain signal is window compensated with the $\frac{1}{\omega_i}$ operation. This operation is used to maintain the desired signal output level of the array.

3.4   Chapter Conclusion

Multiple antennas receivers have a number of potential algorithms that can provide a GPS receiver with jamming protection. The types of algorithms discussed in this chapter were space adaptive processing, space time adaptive processing, and space frequency adaptive processing. Spatial adaptive processing was shown to leverage the phase differences between the antenna to attenuate or amplify reception in certain directions. Space time adaptive processing added to SAP algorithms with the addition of time delayed samples into the

algorithm. This allowed for both spatial directions and frequencies to be attenuated or amplified. Lastly, space frequency adaptive processing minimizes the total power in the frequency domain by weighting and combining the frequency domain signals.

For each of these categories, multiple algorithms were discussed. Many of these algorithms change the effective gain pattern of the array by phase shifting and scaling the signals from each antenna. These complex weights are determined in order to satisfy some criteria defined by the algorithm, such as minimizing total signal output power or maximizing the signal to noise ratio. The optimal algorithm to be used largely depends on the amount of information known, such as satellite signal's angle of arrival, or the processing power available.

Chapter 4

Simulation Environment

Due to the strict prohibition on transmitting in the L frequency band at which GPS operates, all results for this thesis were obtained using a MATLAB GPS L1 C/A code simulator written by the GAVLab at Auburn University [18] and a modified version of the MATLAB software receiver written by Kai Borre and Dennis Akos [3]. The purpose of this chapter is to describe the basic functionality of the simulation environment, provide baseline positioning performance of the MATLAB software receiver, and show the baseline performance of the CRPA anti-jam algorithms put forth in the previous chapter. All simulations were done using a static trajectory, a second order PLL loop filter, a PLL bandwidth of 20 Hz, and a tracking loop integration time of twenty milliseconds.

4.1 Description of Simulation Environment

The use of the MATLAB simulation environment was chosen for its scenario flexibility and to ensure file synchronization between the data file of each element. The simulation environment supports scenarios such as static trajectories, dynamic trajectories, jamming scenarios, or jamming free scenarios which will be called clean scenarios. The simulator also allows for complete control over the signal simulation. Error sources, such as Ionospheric and Tropospheric delays, can be removed and the noise level on the received signal can be varied. The simulator allows full control over the jamming scenario with the ability to change the quantity of jammers, the location of the jammer, jammer to signal strength ratio (J/S), and duration of the jammer transmission. Currently, the simulation environment allows for impulse, narrowband, frequency sweep, and pulsing jammer waveforms.

4.2   C/A Code Positioning Accuracy

Within the simulation environment, a position solution can be computed using a software receiver or a hardware receiver when using a software defined radio to transmit the simulated signal. This chapter will focus on the baseline performance of the software receiver because the anti-jam algorithms are implemented via the software receiver. Clean scenario simulation results can be seen in Figure 4.1 using the C/A code as the ranging measurement for a static trajectory. The noise on the position solution is quite large with variations in East and North direction of about ten meters and an up direction variation of around twenty meters. While the position solution is very noisy, it can be seen that the software receiver does calculate the correct static position. Using the C/A code, the software receiver calculates a position with the precision of about ten meters in the East and North direction, as shown in the position error plot of Figure 4.2.
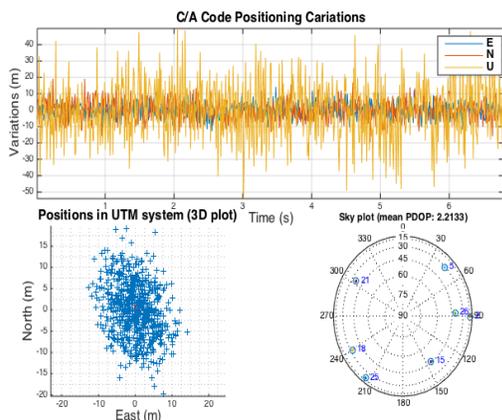
Figure 4.1: Navigation Results Using C/A Code Positioning
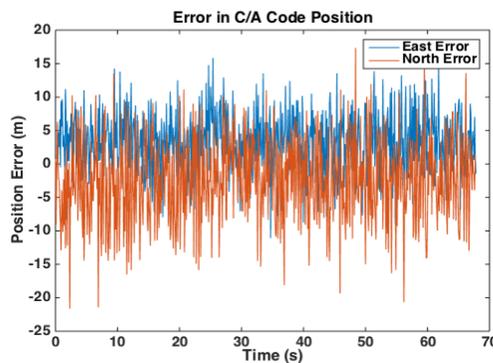
Figure 4.2: Error in Calculated C/A Code Position

4.3   Carrier Phase Positioning Accuracy

Using the carrier phase as the ranging measurement increases the precision of the software receiver's position from meters to millimeters. During carrier phase positioning positioning, the receiver accumulates the number of cycles that have elapsed since the start

of tracking the signal. This provides the change in the range since tracking was initiated. Because of this, the true simulated range was used to initialize the positioning algorithm. The true range is used over typical methods, such as the LAMBDA method that calculate the range between the user and satellite with high accuracy, to limit the possible sources of error that are not tied to the anti-jam algorithms. The position solution and error in the position solution can be found in Figure 4.3 and Figure 4.4 respectively. Using the carrier phase measurement, the accuracy of the position solution goes from several meters when using the C/A code to several millimeters when using the carrier phase measurement.
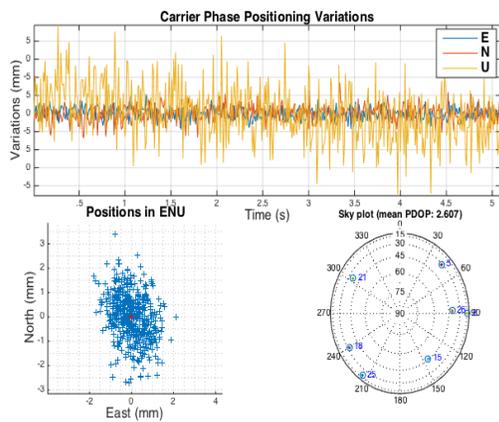
Figure 4.3: Navigation Results Using Carrier Phase Positioning
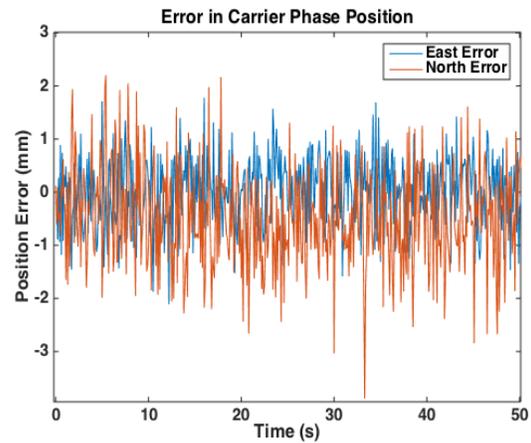
Figure 4.4: Error in Calculated Carrier Phase Position

4.4 FRPA Performance in a Jamming Environment

This section is designed to show performance of a single antenna receiver in a jamming environment to give a baseline performance and to give a frame of reference to the amount of receiver robustness the CRPA anti-jam algorithms provide. Again the receiver settings were set to a phase lock loop bandwidth of 20 Hz, a tracking integration time of twenty milliseconds, and a second order PLL loop filter. The J/S value was increased from 0 dB to 25 dB in increments of 5 dB. As shown in Figure 4.5, the error in the position solution grows

exponentially as the jammer strength is increased. Despite this, the receiver maintains a millimeter level accuracy up to a J/S of 25 dB.

At a J/S of 25 dB, the receiver is no longer able to maintain tracking all of the available satellites, going from seven satellites down to five. The tracking results in the East and North direction of the FRPA receiver at a J/S of 25 dB can be seen in Figure 4.6, where the origin is set as the simulated position. At the 240 ms mark, the effect of the jammer can be seen by the increased noise on the positioning measurement. At 25 dB, the receiver on the edge of maintaining tracking and being unable to maintain constant phase lock due to the amount of noise on the signal. This will results in occasional jumps in position where similar to a cycle slip phase lock is momentarily lost, an inaccurate Doppler frequency is recorded, but then phase lock is re-established in the next measurement. This can be seen at the 420 ms mark where there is a sudden jump in the position solution. At this point one of the tracked signal lost phase lock and reported an inaccurate frequency measurement. The inaccurate frequency measurement leads to an incorrect pseudorange calculation that pushes the position solution from the true position. However, phase lock was re-obtained by the next iteration and the receiver returned to reporting static positioning. At this point the receiver would typically have to re-calculate the high precision range between the user and the satellite to correct for the introduced error. At a J/S of 30 dB, the receiver was only able to track two satellites and therefore unable to calculate a position solution.
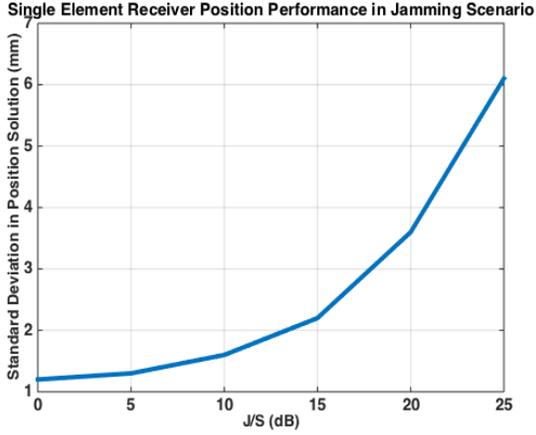
Figure 4.5: Standard Deviation of Position Solution with Various Jammer Strengths
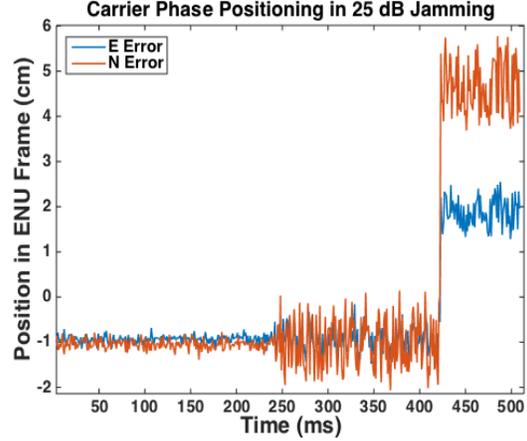
Figure 4.6: Carrier Phase Positioning Using a J/S of 25 dB

4.5 CRPA Receiver Performance in a Jamming Environment

To test the performance of the CRPA anti-jam algorithms discussed in Chapter 3, each of the algorithms was used to compute a carrier phase position solution in a jamming environment of 25 and 35 dB. At 25 dB, the standard receiver was just able to maintain millimeter level accuracy while at 35 dB the receiver could no longer track four satellites and calculate a position solution. The standard deviation of the position results of each of the anti-jam algorithms can be seen in Figure Table 4.1 below for both jamming scenarios.

Table 4.1: CRPA Receiver Positioning Performance

	J/S = 25 dB	J/S = 35 dB
Algorithm	STD of Position Sol (mm)	STD of Position Sol (mm)
FRPA	8.0	NAN
Beam Steering	2.3	6.9
Null Steering	31.7	NAN
Power Minimization	2.0	1.1
Least-Mean-Square	0.691	0.650
MVDR	3.3	11.0
Least-Mean-Square (STAP)	1.3	3.7
Power Minimization (STAP)	1.4	1.4
SFAP	5.87	8.1

For the 25 dB simulation, each of the CRPA algorithms provided a performance improvement over the fixed reception pattern array (FRPA) receiver. The worst performing

46

algorithm was found to be null steering. This lack of improvement from null steering is likely due to the fact that null steering has no control over the receptiveness of the array aside from the pointed null. The lack of control over the array receptivity can lead to degraded performance from sympathy nulls and decreased receptiveness to satellites. In this simulation, the algorithm nulled out the jammer but also three satellites leading to an increase in the dilution of precision (DOP).

The best performance was achieved by the the SAP Least-Mean-Square with a standard deviation of 0.69 mm in the position solution followed by the STAP LMS algorithm. While the STAP LMS algorithm should have performed better than the SAP LMS algorithm because of the better nulling localization capabilities, it is likely that the STAP LMS converged to a less accurate solution than the SAP LMS. Overall the CRPA performance, barring null steering, provided performance similar to that of the FRPA receiver in a 10 dB weaker jammer scenario.

The anti-jam algorithms were also tested in a J/S of 35 dB jamming environment. Note that the receiver with the FPRA antenna could not calculate a position solution in this jamming environment due to the fact that it could only track two satellites. However, all but one CRPA algorithm was able to position with carrier phase positioning accuracy. Null steering was only able to track two satellites and therefore also not able to calculate a position solution. Barring null steering, the other anti-jam algorithms that were tested were able to calculate a position solution during the jamming simulation and maintain the carrier phase level accuracy before and after the jammer was activated.

4.6   Chapter Conclusion

In this chapter the simulation environment and the algorithms discussed in Chapter 5 were evaluated. It was determined that the simulation environment provides an accurate signal generation tool that can be used to calculate a position solution. The software receiver was able to calculate the position solution to the precision of meters when using the C/A

code and to the precision of millimeters when using the carrier phase measurement. These results show that the simulation environment accurately simulates the received signals of a GPS receiver.

The simulation environment was then used to show the positioning performance in a jamming environment of both a standard receiver with a FRPA antenna and receiver using various anti-jam algorithms with a CRPA antenna. It was shown that the FRPA receiver could not position in a jamming environment with a J/S greater than 25 dB. At 25 dB, the FRPA receiver could not track all available satellites and the precision of the position solution was degraded from millimeters to centimeters. The CRPA anti-jam algorithms were shown to have a performance increase over the standard receiver with a FRPA antenna by comparing the precision and availability of the position solution. A jamming environment with a J/S value of 25 and 35 dB was used for the comparison. In both jamming simulations, all but one CRPA algorithms was able to position with millimeter level precision. This shows that the use of CRPA antennas and anti-jam algorithms can provide protection to the receiver in jamming environments in which a standard FRPA receiver cannot obtain a position solution.

Chapter 5

Unconstrained Recursive Algorithm Drifting

The purpose of this chapter is to show how unconstrained recursive anti-jam algorithms can cause erroneous position drifting while performing carrier phase positioning, the magnitude of the position drift, and the effectiveness of the solution. The issue of position drift due to this specific type of algorithm is unique to carrier phase positioning because ranging methods, such as the C/A code, mask the drift with the higher amounts of noise on the measurement.

The Least-Mean-Square algorithm, as discussed previously in Chapter 3 and seen in Equation (5.1), is a recursive algorithm that will update the weights being applied to the incoming samples based on the current error between the weighted and desired signal and the vector of current measured signals from each of the elements, X_k.

$$W_{k+1} = W_k + \mu X_k(d_k - W_k X_k) \tag{5.1}$$

Unlike many of the other algorithms discussed in Chapter 3, the Least-Mean-Square algorithm does not use a constraint vector to maintain a unity weight on the reference element. While the algorithm will still provide anti-jam capabilities, the unconstrained solution could lead to a time variant phase shift on the reference element.

Based off the discussion in Chapter 2, it was determined that one of the internal control laws of the tracking loop is the phase lock loop. In the phase lock loop, the carrier NCO attempts to maintain an accurate replica of the incoming signal's carrier frequency and phase by minimizing the phase error between the the incoming signal and the in-phase prompt replica signal. If any phase error exist, the NCO will change the frequency of the replica signal in order to maintain phase alignment. The replica's frequency will then be used as an

estimate of the incoming signal's Doppler frequency which is used in the pseudorange rate calculation of carrier phase positioning. Due to this cause and effect, the time variant phase shifting from the Least-Mean-Square algorithm can provide erroneous carrier phase position solutions over time as the phase shift on the reference element is continuously being changed.

5.1 Single Jammer Scenario

To test this hypothesis, a jamming environment is simulated in which a receiver with four elements starts off in a clean scenario for thirty seconds. This is followed by a single continuous wave jammer which is turned on for the remainder of the simulation with a J/S of 45 dB. This experiment is designed to determine if the algorithm affects the position solution during the clean portion of the simulation, the transition between the clean and jamming, as well the performance during the jamming portion. The ENU position solution can be seen in Figure 5.1 where the blue dots are the calculated positions during the clean portion, the red dots are calculated positions during the jamming portion, and the green dot is the true simulated position. The error in the East and North directions over time can be seen in Figure 5.2.
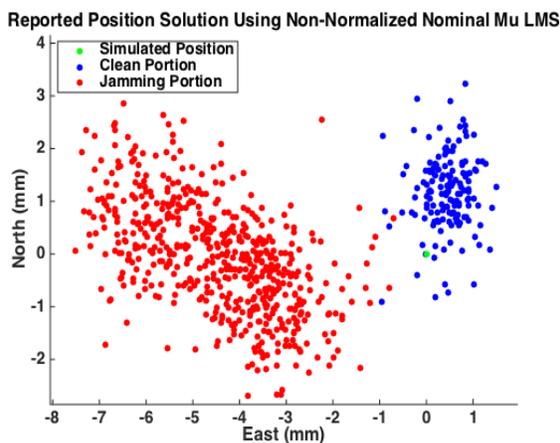

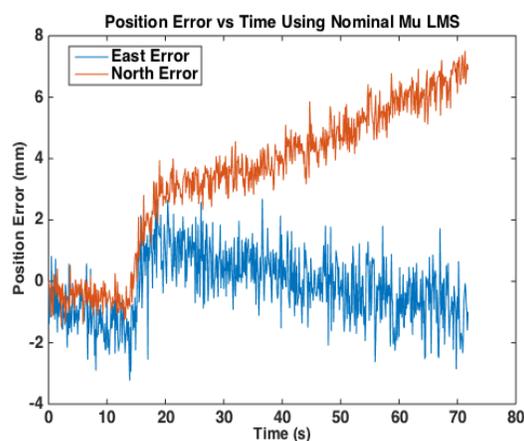Figure 5.1: Single Jammer Navigation Results Using LMS Carrier Phase Positioning, LMS $\mu = 8.5 * 10^{-11}$

Figure 5.2: Single Jammer Error in LMS Algorithm Carrier Phase Position, LMS $\mu = 8.5 * 10^{-11}$

Similar to many of the jammer simulations, the tracking loop loses phase lock momentarily when the jammer is activated around the 18 second mark, causing a small position jump. To show the amount of drift caused by the algorithm, Figure 5.2 is separated into error during the clean portion of the simulation, shown in Figure 5.3, and the error during the jamming portion since the tracking loop regains phase lock, shown in Figure 5.4. Using Figures 5.3 and 5.4, a number of observations can be made. First, the carrier phase position accuracy was not greatly affected and the position solution was maintained on the order of magnitude of millimeters. Second, the algorithm was able to adjust the weights quickly in order to regain phase lock before the estimates of the incoming signal diverged from the received signal. Thirdly, there is a small amount of drift in the clean portion and a slightly larger drift during the jamming portion of the simulation. While the drift is only in the East direction during clean scenario, the jammer portion of the simulation contains 3.5 mm of drift and 4.8 mm of drift in the East and North direction over fifty seconds, respectively. The likely cause of the accelerated drift is the changing phase shift being applied to the reference element from the Least-Mean-Square algorithm.
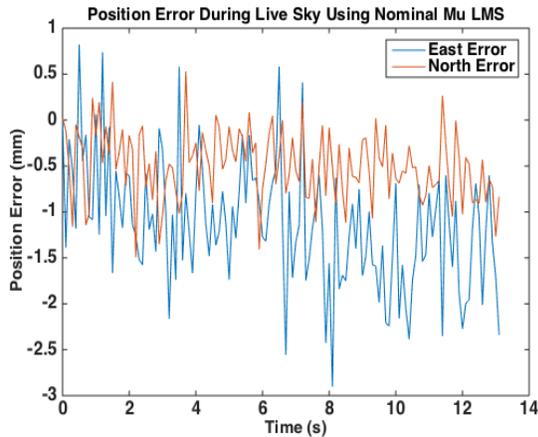

Figure 5.3: Single Jammer Error in LMS Algorithm Carrier Phase Position During Clean Scenario Portion, LMS $\mu = 8.5 * 10^{-11}$
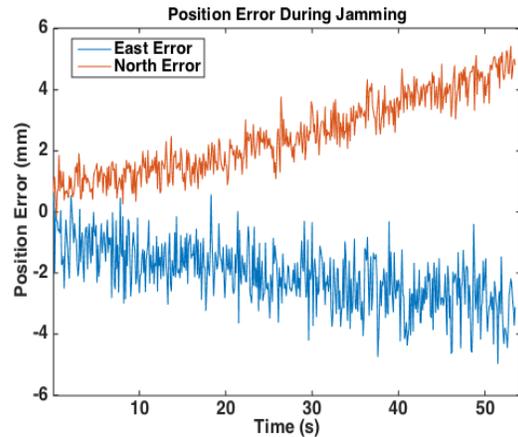
Figure 5.4: Single Jammer Error in LMS Algorithm Carrier Phase Position During Jamming Portion, LMS $\mu = 8.5 * 10^{-11}$

The phase shifts applied to the reference element can be determined with Equation (5.2). Equation (5.2) uses the weighting applied to the reference element and calculating the phase advancement or delay that will occur from the complex weight.

$$\theta = \arctan\left(\frac{imag(W_r)}{real(W_r)}\right) \tag{5.2}$$

The phase shift on each channel throughout the simulation can be seen in Figure 5.5. First, there is a clear difference between the phase shifting during the clean and jamming portions of the simulation. The phase shifting on each channel during the clean portion has a much smaller standard deviation and, compared to the jamming portion, changes much slower over time. Second, it can be noted that there is a large change in the applied phase shift when the jammer is activated. This sudden phase shift show that when the jammer is activated there is a large amount of error between the desired and weighted input signal. The sudden change in phase shift does not effect the position solution due the fact that phase lock is lost at the same time as the sudden change in phase shift. Therefore, when phase lock is re-obtained it is consistently at the new phase shifting value. Lastly, it can be seen that the phase shifting during the jamming portion is much more aggressive than the clean portion.

Figure 5.5: Phase Shift Applied to Reference Element on Each Channel, LMS $\mu = 8.5 * 10^{-11}$

5.2 Multiple Jammer Scenario

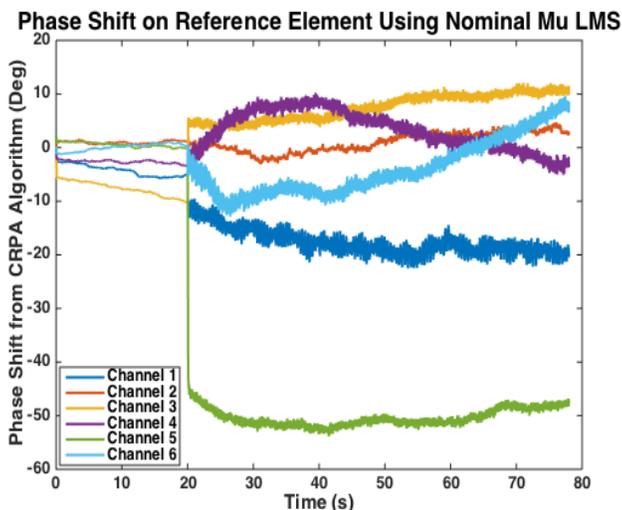While the previous simulation results show that the phase shift does cause a small amount of drift in the carrier phase position solution, the solution was still in the nominal accuracy range of the carrier phase positioning for the duration of the simulation. However, the previous example was a relatively easy nulled jamming simulation where only a single jammer was implemented at a manageable power level. To test the LMS algorithm in a more challenging environment, a second simulation was done with a more aggressive jamming scenario. The second simulation is broken into three different sections. The first section is of a clean environment to show initial performance and lasts for thirty seconds. The second section is of a single jammer (jammer number 1) and lasts for another thirty seconds. Lastly, the single jammer is deactivated and the simulation is finished with a period of thirty seconds where three spatially diverse jammers are activated(jammer numbers 2-4). Each of the jammer positions are shown in Figure 5.6. Three jammers were selected because the receiver is a four element CRPA antenna and the use of three spatial diverse jammers will push the anti-jam algorithm to use more degrees of freedom than a single jammer. The use of this simulation will show an extreme case where the recursive algorithm must transition from one set of spatial nulls to another with a small enough transition time to maintain tracking.

The carrier phase positioning of the second simulation can be seen in Figure 5.7 and the position error in Figure 5.8. It can be seen that when the jammer was activated, the tracking loops lost lock momentarily causing another large jump in the position solution and the error in the East and North position. To show the error drift on the millimeter scale, Figure 5.9 shows the position error in the East and the North direction during the clean portion of the simulation before the loss of lock and the position jump and Figure 5.10 shows the position drift in the East and North directions when the jammer is activated and the tracking loop regains phase lock. The error plots show that the LMS algorithm responded similarly to that of the first test case in which a single jammer was used. While the clean portion does

53
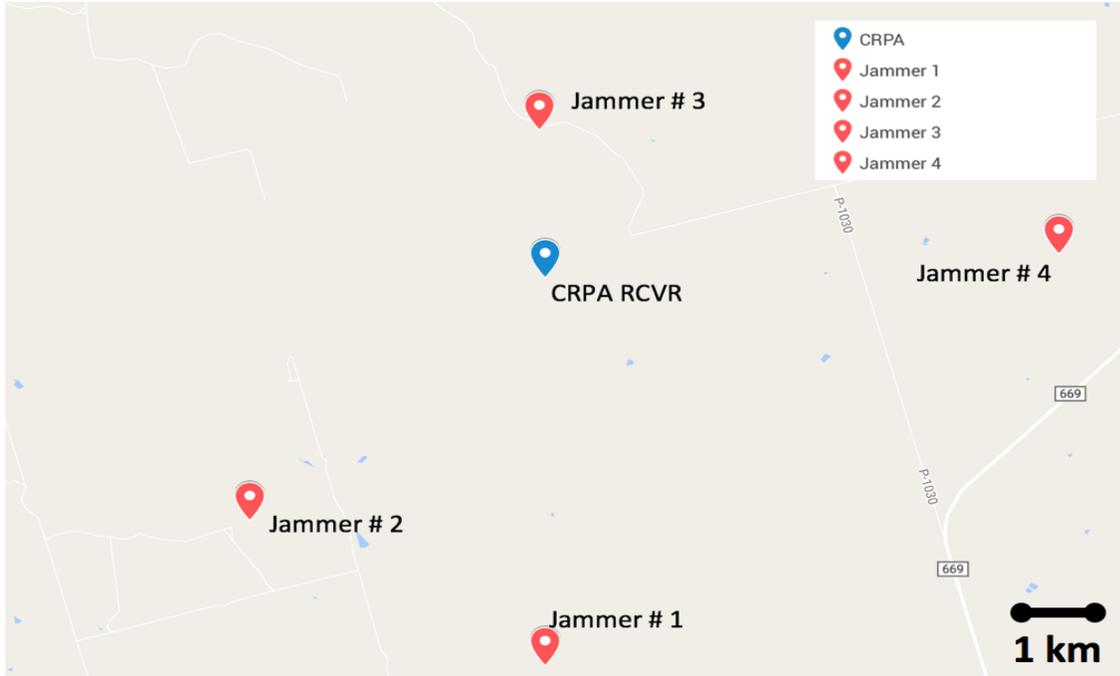
Figure 5.6: Jamming Scenario Map of CRPA Receiver and Jammer Locations

contain a small amount of drift, the drift is accelerated during the jamming portion. During the jamming portion, the drift was able to accumulate to 2 mm in the East direction and 5 mm in the North direction within a minute.
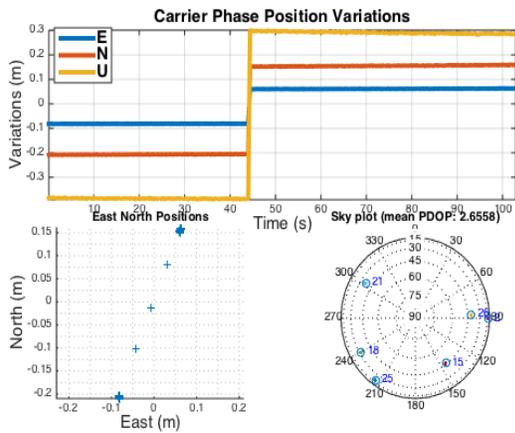
Figure 5.7: Multi-Jammer Navigation Results Using LMS Carrier Phase Positioning, LMS $\mu = 8.5 * 10^{-11}$
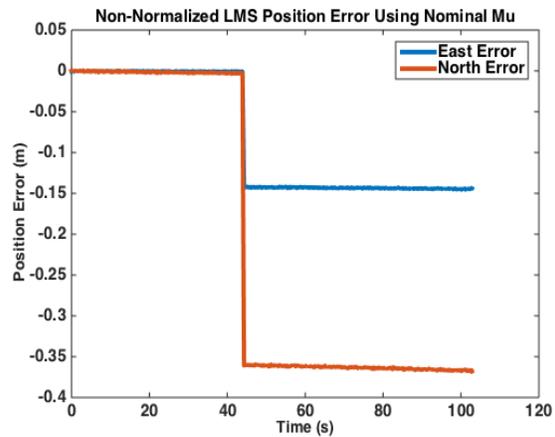
Figure 5.8: Multi-Jammer Error in LMS Algorithm Carrier Phase Position Using $\mu = 8.5 * 10^{-11}$
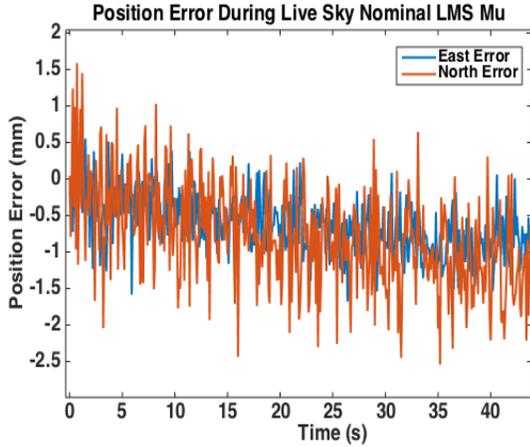
Figure 5.9: Multi-Jammer Error in LMS Algorithm Carrier Phase Position: Clean Portion Using $\mu = 8.5 * 10^{-11}$
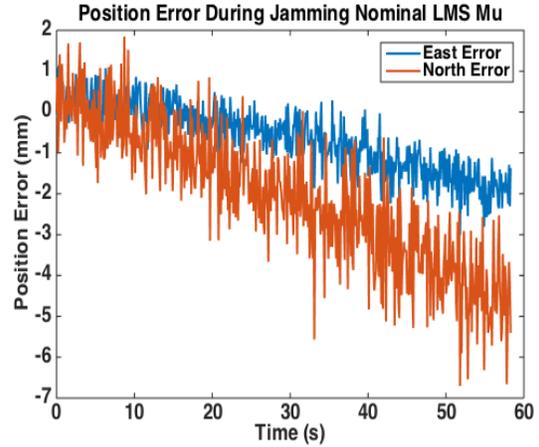
Figure 5.10: Multi-Jammer Error in LMS Algorithm Carrier Phase Position: Jamming Portion Using $\mu = 8.5 * 10^{-11}$

To support the claim that the unconstrained phase shift is causing the position drift, the same scenario was tested using power minimization which implements a constraint vector so that the reference element weight is always unity. The East and North position error during the clean portion can be seen in Figure 5.11 and the East and North position error during the jamming portion can be seen in Figure 5.12. For both section of the testing, the error was contained to about two millimeters of error without any signs of error growth over the simulation. This supports the claim that the time variant phase shift is causes the position drift to occur.

The phase shifting of the reference element is shown in Figure 5.13, where the LMS algorithm failed to converge to a solution on channel two in a timely manner. This corroborates with the results from the tracking loop in which the signal was not tracked once the jammer portion of the simulation was initiated. Once again, there is a sudden change in the applied phase shift at the start of the jamming portion and the applied phase shift dramatically changes as well as an increase in noise of the phase shift. However, the algorithm is able to maintain a constant weighting or phase shift on most of the channels except channels one and three, which can be seen more clearly in Figure 5.14 which only includes tracked channels. In these channels the phase shift can be seen to change over time.
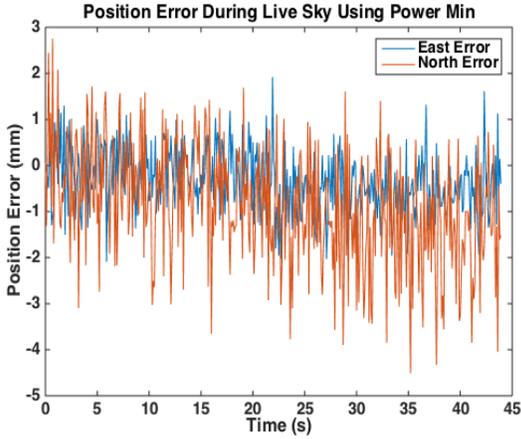
Figure 5.11: Multi-Jammer Error in Power Minimization Algorithm Carrier Phase Position: Clean Portion
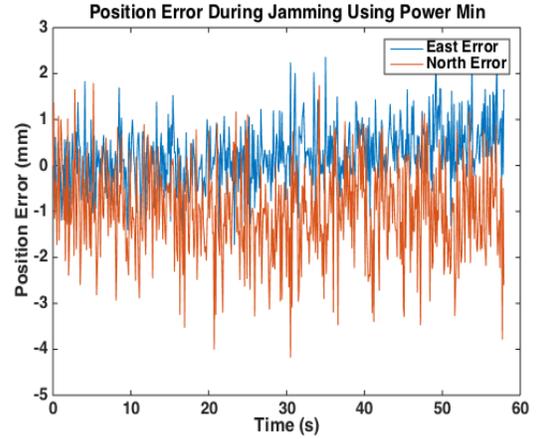
Figure 5.12: Multi-Jammer Error in Power Minimization Algorithm Carrier Phase Position: Jamming Portion
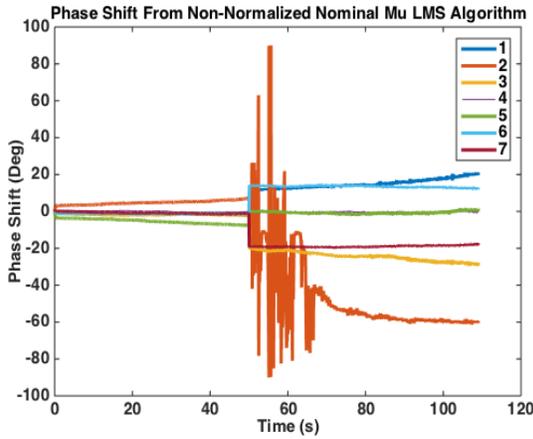
Figure 5.13: Phase Shift Applied to Reference Element on Each Channel, LMS $\mu = 8.5 * 10^{-11}$
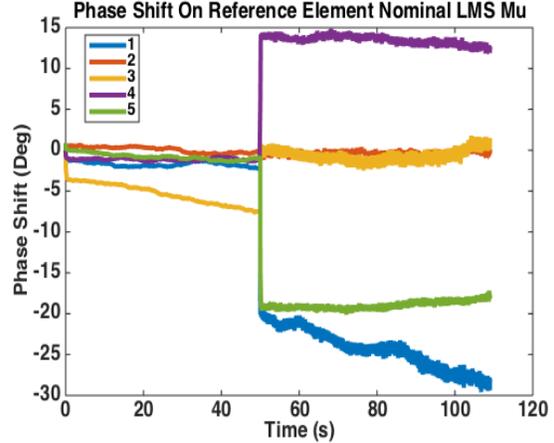
Figure 5.14: Phase Shift Applied to Reference Element on Each Tracked Channel, LMS $\mu = 8.5 * 10^{-11}$

While the previous implementation of the LMS algorithm was successful in nulling out the jamming, the adaptive nature of the LMS algorithm can be tuned for various speeds of convergence that can effect the positioning performance of the CRPA receiver. The trade off variable, as discussed in Chapter 3, is the learning coefficient μ used in the LMS algorithm. As μ is increased, the algorithm will converge faster to a solution producing better beam and nulls, but risk instability. Conversely, μ can be smaller which will help guarantee stability at the sacrifice of response time of the algorithm. To explore this concept, the multiple jammer

56

simulation was processed by the LMS algorithm again with both a larger and a smaller μ value. The μ values in all of the testing can be seen in Table 5.1 below.

Table 5.1: LMS μ Value Used

Small LMS μ Value	$2.5 * 10^{-11}$
Nominal LMS μ Value	$8.5 * 10^{-11}$
Large LMS μ Value	$5.05 * 10^{-10}$

The first test case explored was the larger μ LMS algorithm using a μ value of $5.05*10^{-10}$ instead of $8.5 * 10^{-11}$. The position error during the clean portion and the jamming portion can be seen in Figure 5.15 and Figure 5.16, respectively. While the clean portion does position correctly with no drift, the jamming portion suffers from drifting that is more severe than the previous experiment, accumulating 4 mm of drift in the East direction and 12 mm of drift in the North direction. In a anti-jamming application however, the larger μ has the advantage in a rapidly changing jamming scenario where the algorithm can provide deeper nulls sooner.

Figure 5.15: Multi-Jammer Error in LMS Algorithm Carrier Phase Position: Clean Portion Using $\mu = 5.05 * 10^{-10}$

Figure 5.16: Multi-Jammer Error in LMS Algorithm Carrier Phase Position: Jamming Portion and $\mu = 5.05 * 10^{-10}$

The phase shift applied to each of the tracked channels is provided in Figure 5.17. Due to the larger μ, it should be expected that the LMS algorithm will apply phase shifts with larger variances. This is due to the fact that the larger μ value will not scale down the error

as much as before, therefore allowing larger changes in the weight and applied phase shift between iterations. This expectation is confirmed in Figure 5.17 by examining channels, such as channel three, which varies between phase shifts of -25° to -45° instead of the -20° to -30° observed when μ was set to $8.5 * 10^{-11}$. This increased variation in the applied phase shift corroborates with the increased position drift over time.
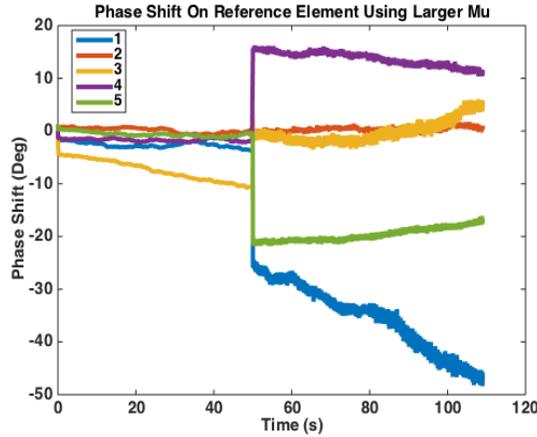
Figure 5.17: Phase Shift Applied to Reference Element on Each Channel, LMS
$$\mu = 5.05 * 10^{-10}$$

The multiple jammer simulation was then processed a third time with a μ value of $2.5 * 10^{-11}$, or about a fourth of the original μ value. For this third test, the μ could not be decreased by the same magnitude that it was increased by in the previous experiment because if the μ is too small, the algorithm will not be trained fast enough to regain phase and code lock on any of the channels. The position error during the clean portion and the jamming portion for the smaller μ value can be seen in Figure 5.18 and Figure 5.19 respectively. As seen in the jamming positioning, the position drift has been significantly reduced.

From the reference element phase shift plot of Figure 5.20, it can be seen that the smaller μ limited the total change phase shift so that after phase lock is regained, the applied phase shift did not vary a considerable amount over time. This lack of a time variant phase shift is what limited the drift from occurring. While the weights and response time were acceptable

for this simulation, the smaller μ algorithm may not be the best solution in all possible applications, even if it does limit the position drift over time.
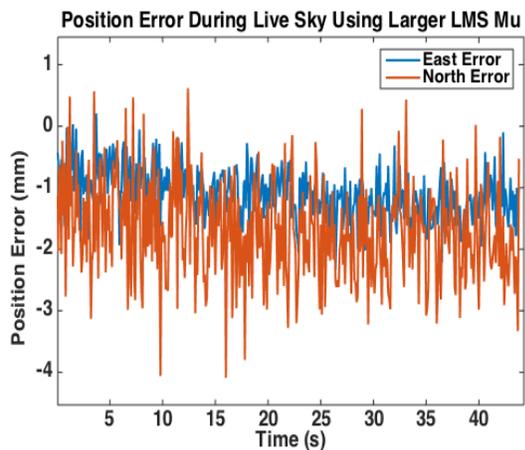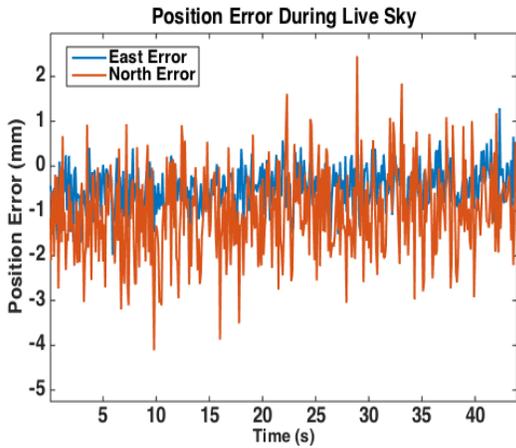

Figure 5.18: Multi-Jammer Error in LMS Algorithm Carrier Phase Position: Clean Portion and $\mu = 2.5 * 10^{-11}$
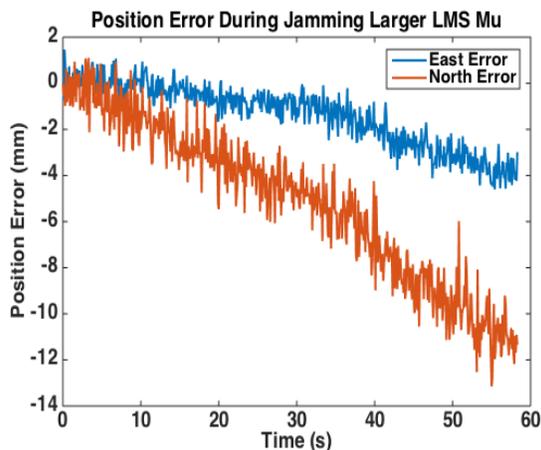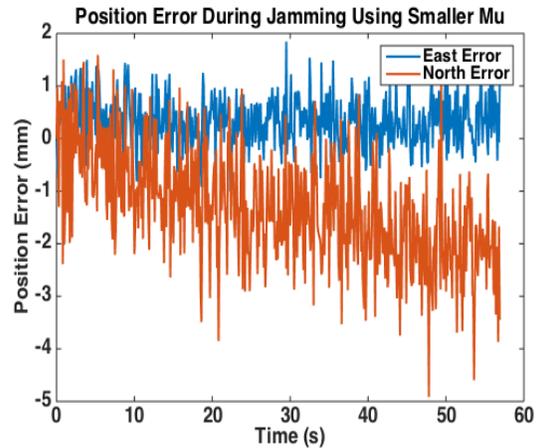

Figure 5.19: Multi-Jammer Error in LMS Algorithm Carrier Phase Position: Jamming Portion and $\mu = 2.5 * 10^{-11}$
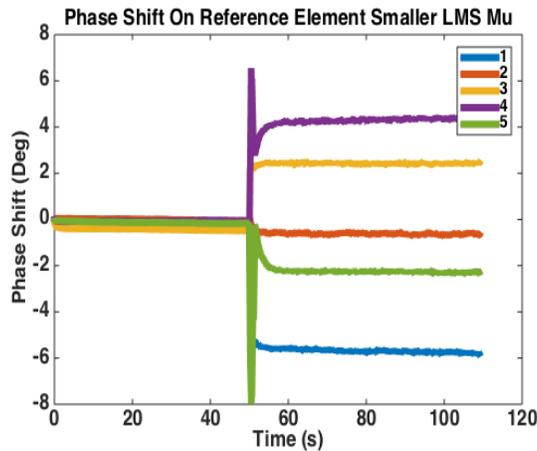
Figure 5.20: Phase Shift Applied to Reference Element on Each Channel, LMS $\mu = 2.5 * 10^{-11}$

5.3 Normalized Least-Mean-Square Single Jammer Positioning Results

The problem of the position drifting when using the LMS algorithm has so far been attributed to the time variant phase shift that is applied to the reference element. All of the weights applied to the antenna signals, however, can be altered and adjusted before

59

application. There exist three options to limit the position error when using the LMS algorithm. First, the algorithm can be constrained such that there is always a unity weight applied to the reference element similar to the recursive MVDR algorithm. However, this method may constrain the solution to a sub-optimal solution. Second, the amount of phase shift can be fed to the phase lock loop and removed when the carrier phase is calculated. Third, the weights can be normalized with respect to the reference element weight [5]. For this thesis, the latter correction is implemented because of it's intuitive and simple correction to the problem.

Through the use of the normalization operation, the effective phase shifting is maintained, and the normalized weights produce the same array gain pattern but with no phase shift being applied to the reference element. The normalization process is shown to not change the effective gain pattern of the array by comparing the gain pattern of a non-normalized weighting gain pattern, shown in Figure 5.21, with the normalized weighting gain pattern, shown in Figure 5.22. Here it can be seen that the two plots are identical. Therefore, the normalization process keep a changing phase shift from being applied to the reference element while not degrading the performance of the LMS algorithm. The normalization operation can be seen in Equation (5.3) where the current weight vector is multiplied by the complex conjugate of the reference weight, W_r, and divided by the absolute value of the reference weight.

$$W = W \frac{W_r^*}{|W_r|} \tag{5.3}$$

Using the normalization operation, the two single jammer scenario was performed again. In the first test scenario, it can be seen in Figure 5.23 that the time varying phase shifting has been removed except for a very small amount due to rounding error. Figure Figure 5.24 and Figure 5.25 shows that a lack of a time variant phase shift improves the position solution and removes the position drift during the clean and jamming portion of the single jammer

CRPA Antenna Gain Pattern at an Elevation of 0 degrees

Normalized LMS Gain Pattern
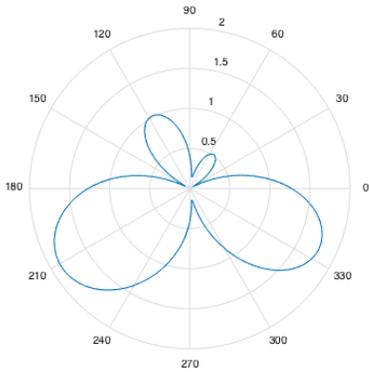
Figure 5.21: Not Normalized Weighting Gain Pattern
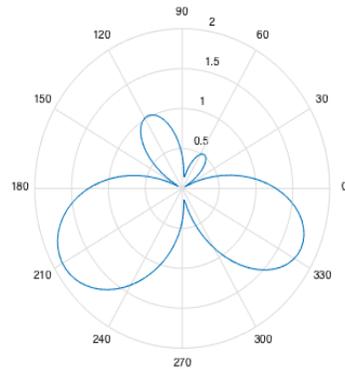
Figure 5.22: Normalized Weighting Gain Pattern

simulation with a LMS μ value of $8.5 * 10^{-10}$. While there is still some error, both the North and East position errors become constant once the tracking loop fully regains phase lock.

Figure 5.23: Normalized Phase Shift Applied to Reference Element on Each Tracked Channel, LMS $\mu = 8.5 * 10^{-11}$
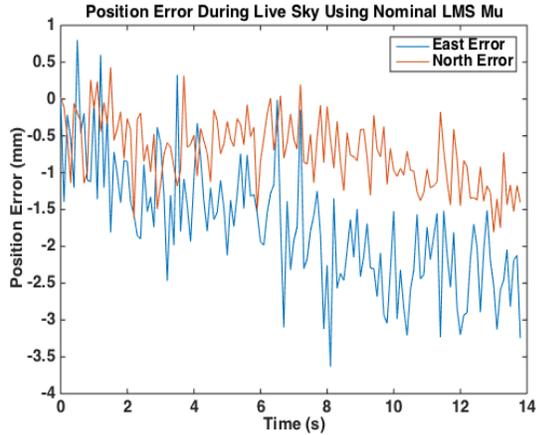
Figure 5.24: Single Jammer Navigation Results Using LMS Carrier Phase Positioning: Clean Portion and $\mu = 8.5 * 10^{-11}$
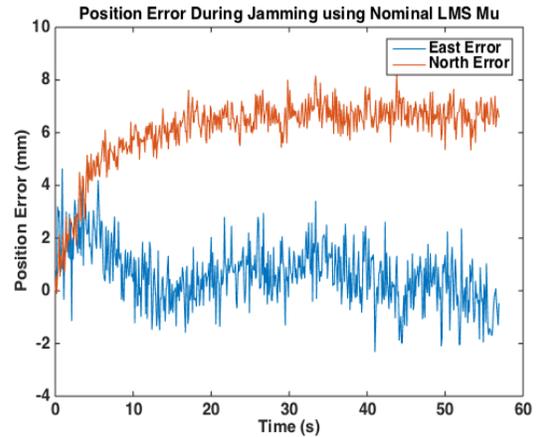
Figure 5.25: Single Jammer Error in LMS Algorithm Carrier Phase Position: Jamming Portion and $\mu = 8.5 * 10^{-11}$

5.4 Normalized Least-Mean-Square Multiple Jammer Positioning Results

The normalization method was also tested for the second simulation where multiple jammer were used. The normalization process was tested with the nominal mu value LMS, where μ was equal to $8.5 * 10^{-11}$, as well as the variants of the multiple jammer simulation where a μ value of $5.05 * 10^{-10}$ and $2.5 * 10^{-11}$ was used. Similar to the first simulation, it is shown in Figure 5.26 that the normalization process succeeded in limiting the phase shift applied to the reference element on each channel, taking rounding error into consideration. Using the normalized weights, the position solutions for both the clean section and the jamming section were calculated and the position error in the East and North direction can be seen in Figure 5.27 and Figure 5.28 using the nominal μ value. It can be seen that the use of the normalization technique mitigates the position drift that was noted when using the non-normalized LMS algorithm. These results are similar to that of Figure 5.29 and Figure 5.30 which show the position error during the clean and jamming portion when using the larger μ value of $5.05 * 10^{-10}$. It should be remembered that as the μ value was increased, the position drift became more severe because the algorithm allowed for larger phase shifts.

However, through the normalizing process, the larger μ solution can have the same amount of position drift as the slower converging solution while having a faster convergence time.
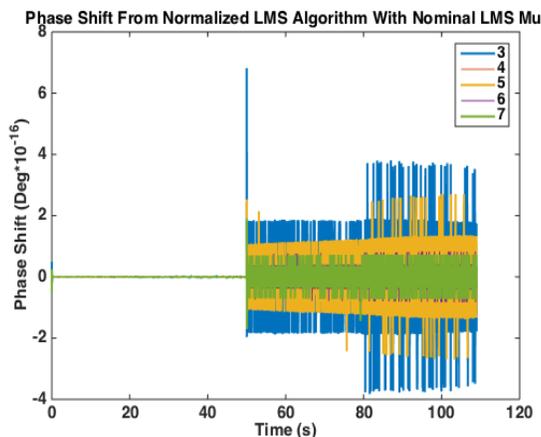
Figure 5.26: Normalized Phase Shift Applied to Reference Element on Each Channel, LMS $\mu = 8.5 * 10^{-11}$
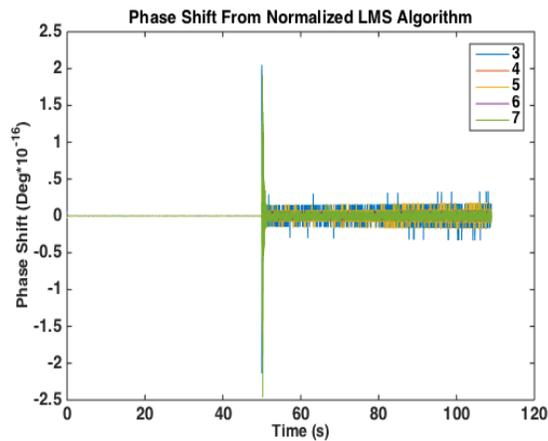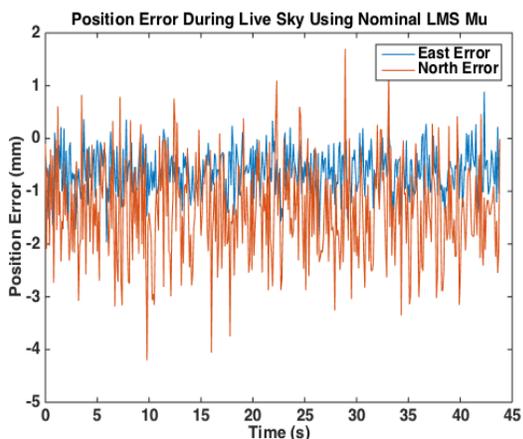

Figure 5.27: Error in Normalized LMS Algorithm Carrier Phase Position During clean Portion and $\mu = 8.5 * 10^{-11}$
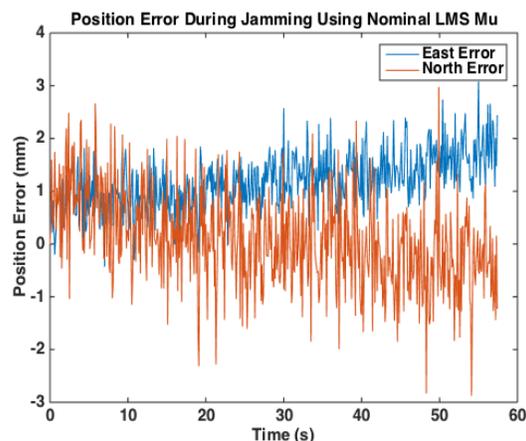
Figure 5.28: Error in Normalized LMS Algorithm Carrier Phase Position During Jamming Portion and $\mu = 8.5 * 10^{-11}$

Lastly, the smaller LMS μ value was tested using the multiple jammer simulation. The position error during the clean and jamming portion can be seen in Figure 5.31 and Figure 5.32. When comparing normalized position error to the non-normalized position error, shown previously in Figures 5.18 and 5.19, it can be seen that the position errors are identical. However, the lack of change when applying the normalization process makes sense because
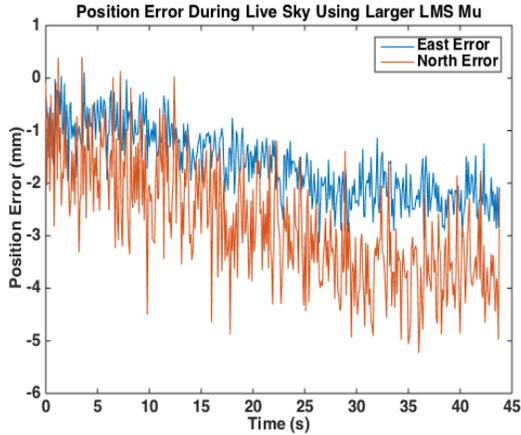
Figure 5.29: Error in Normalized LMS Algorithm Carrier Phase Position During Clean Portion and $\mu = 5.05 * 10^{-10}$
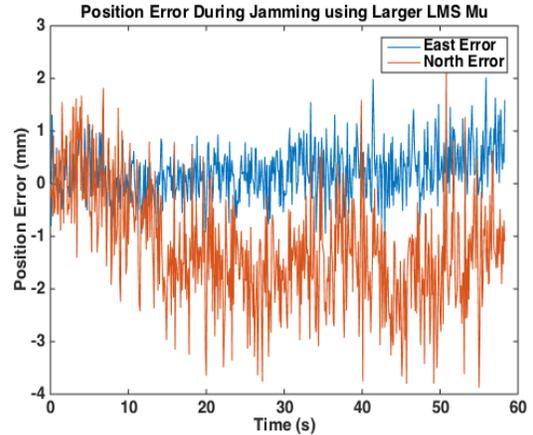
Figure 5.30: Error in Normalized LMS Algorithm Carrier Phase Position During Jamming Portion and $\mu = 5.05 * 10^{-10}$

the change in the phase shift was close to zero before the normalization process was applied. Therefore, there was no drift due to time variant phase shifting to compensate for.
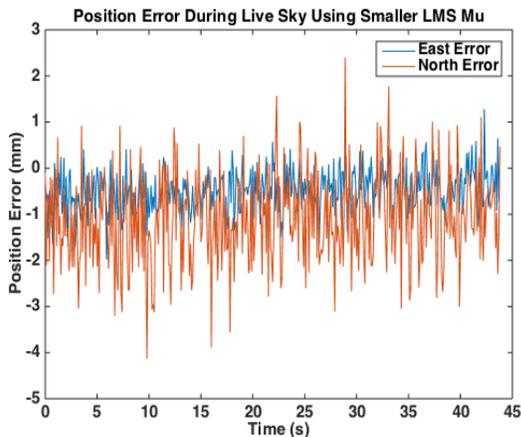
Figure 5.31: Error in Normalized LMS Algorithm Carrier Phase Position During Clean Portion $\mu = 2.5 * 10^{-11}$
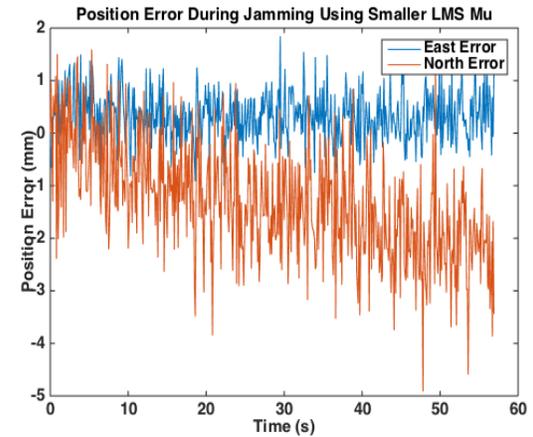
Figure 5.32: Error in Normalized LMS Algorithm Carrier Phase Position During Jamming Portion and Smaller $\mu = 2.5 * 10^{-11}$

5.5   Chapter Conclusion

From the results of the normalized and non-normalized experiments, four things can be observed. First, unconstrained recursive algorithms will cause position drift, on the order

of millimeters. While this drift is small, it appears to be dependent on how quickly the algorithm will change the applied weights, governed by the μ value. Given a large enough μ value and long enough application time, the error potentially could grow to the meter level. Second, the drift appears to happen regardless of the jamming environment and is independent of the severity of the jamming environment. For both the single jammer environments and multiple jammer environments, the drift is roughly the same amount. Third, a low μ value prevents and/or limits the drift due to the fact that the algorithm will not diverge from a solution once the algorithm converges to a solution. Lastly, the weights can be normalized such that no phase shift is applied to the reference element and mitigates the position drift. The normalization process allows for the algorithm to have the advantage of a quick response time when using a large μ value, without the position drift that typically only is achieved with a low μ value.

Chapter 6

Receiver Performance in a High J/S Environment

As the intensity of the jamming environment is increased, the anti-jam capabilities of a CRPA receiver will eventually reach the maximum nulling limit. At the nulling limit, noise will begin to enter the tracking loop and degrade the position solution. To improve the receiver performance in a jamming environment, the bandwidth of the phase lock loop can be lowered, as discussed in Chapter 2. The use of a lower bandwidth is similar to that of using a larger window in a sliding average filter, where the larger window used results in better noise filtering but at the cost of increasing the response time of the filter. While the bandwidth can theoretically be set to any value, the minimum useful bandwidth to the system is limited by the fastest dynamics the controller is required to track. For a GPS receiver the limit of the phase lock loop's bandwidth is set by the dynamics of the user and the satellites being tracked.

6.1 Static Receiver Testing

To test the increased performance provided by changing the phase lock loop bandwidth, a four element CRPA receiver using power minimization was simulated in range of jamming scenarios. Power minimization was selected because of its lack of a prori information required. The jamming environment was simulated to be a single impulse jammer with a J/S value incremented from 30 dB to 74 dB. The incremented range was selected because any J/S value lower than 30 dB has already been determined to be easily nulled by the CRPA anti-jam receiver and it was experimentally determined that J/S value higher than 74 dB causes autocorrelation matrix to become singular causing the power minimization algorithm to break down and the receiver is no longer able to track and obtain a position solution. A

performance evaluation is accomplished by measuring the standard deviation, also referred to as the jitter, of the NCO Doppler estimates on all channels for two seconds of phase locked tracking, or 2000 samples, during the jamming event. Then the standard deviation of each channel is averaged together to provide a single jitter value for that certain J/S value and PLL bandwidth. This jitter generally equates to the position jitter due to the fact that the range rate equation is dependent upon the Doppler measurement of each channel. The results of the J/S value vs. phase lock loop bandwidth experiment can be seen in Figure 6.1.
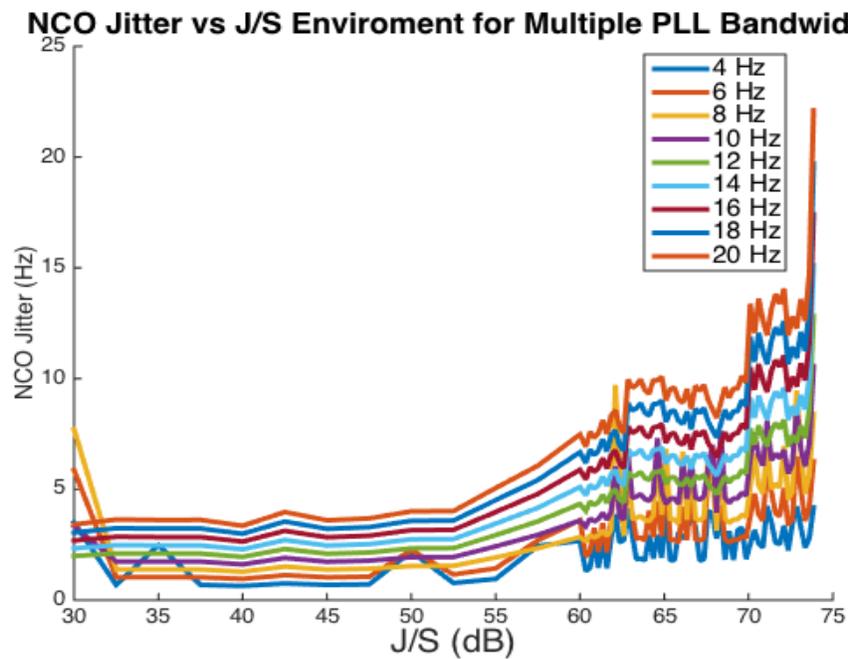
Figure 6.1: Static Testing of Doppler Jitter for Various PLL Bandwidths and J/S Values Using Power Minimization and Four CRPA Antenna

A number of observations can be made using Figure 6.1. First, as expected, as the phase lock loop bandwidth is decreased the performance of the receiver increases linearly. Since the receiver was simulated to be static, the limiting factor of the bandwidth is the satellite dynamics. This limit can be seen at various times with the lower bandwidths, such as 4 or 6 Hz, where there are sudden increases in the NCO Doppler jitter. Second, 8 Hz is roughly the lowest bandwidth that can safely be used in this static trajectory without the risk of spikes in the NCO Doppler estimate due to the satellite motion. Third, it can be

67

noted that there are five distinctly different sections to the receiver performance, regardless of the used bandwidth. The first trend is that of the lower J/S values, roughly between 0 dB and 50 dB. In this section power minimization is fully capable of attenuating receptivity toward the jammer and provide the same level of position accuracy regardless of jammer strength. The next section is that where the receiver performance gradually worsens as the J/S value is increased, roughly between 50 dB and 63 dB. The third section consists of J/S values between 63 and 70 dB and is distinguished from the second section by a sudden spike in the jitter levels at 63 dB. Unlike the previous section, as the J/S value is increased the amount of jitter does not increases linearly. The fourth section of the receiver performance exist between the J/S values of 70 and 73 dB. Similar to the third section, the fourth section differs from the previous section by a sudden increase in the NCO Doppler jitter and remains constant as the J/S value is increased. The last section consists of another sudden spike in Doppler jitter at a J/S value of 74 dB. After this point, the autocorrelation matrix used in power minimization becomes singular for the 20 Hz bandwidth and the process cannot be continued. It should be noted the spikes in Doppler jitter appear to be dependent on the bandwidth used. During these spikes, the receiver that used the larger bandwidths suffered from larger increases in NCO Doppler jitter while the lower bandwidth receiver produced smaller spikes. The observations made from the jitter in the Doppler estimate can also be made using the carrier to noise ratio using Figure 6.2. In the figure, it can be seen that the same trends and sudden jumps can all be seen.

To equate the NCO Doppler jitter into a tangible error value, a position solution was calculated at a J/S value of 74 dB, the highest J/S value experimentally observed to be possible using the implemented receiver using a second order PLL loop filter and a PLL bandwidth of 20 Hz and 7 Hz. The scenario was simulated so that there are fifty seconds of clean data and one hundred seconds where the single jammer is activated. The ENU position solution using a 20 Hz bandwidth can be seen in Figure 6.3 while the error in East and North direction can be seen in Figure 6.4. While the clean portion of the simulation is
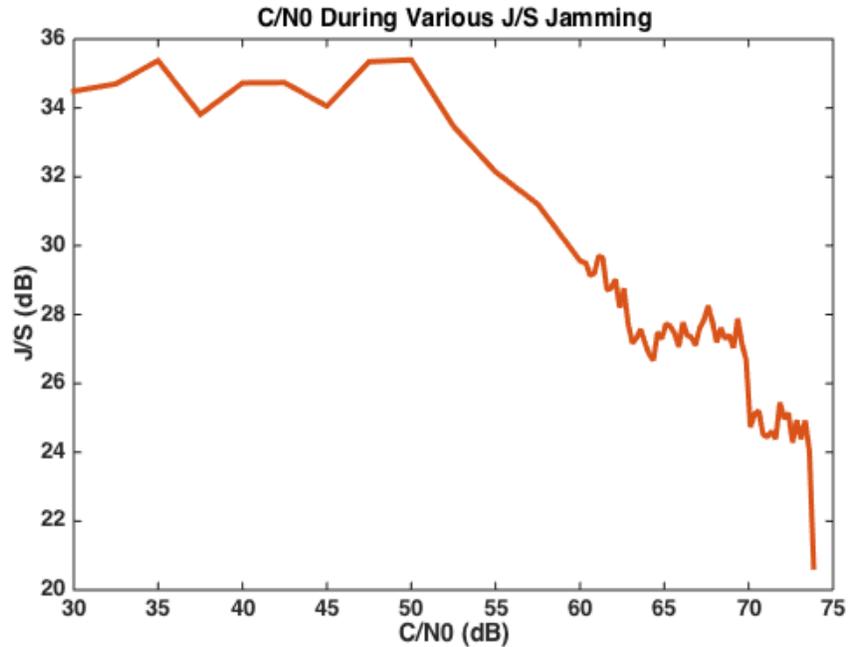
Figure 6.2: Static Testing of C/N0 for Various J/S Values Using Power Minimization and Four CRPA Antenna

able to maintain carrier phase accuracy, at the instant the jammer is activated, the tracking loop is not able to provide a constant position solution. This is due to the large amount of noise in the tracking loop and the jitter in the Doppler estimates. This error brings the accuracy of the carrier phase receiver from millimeters to meters.
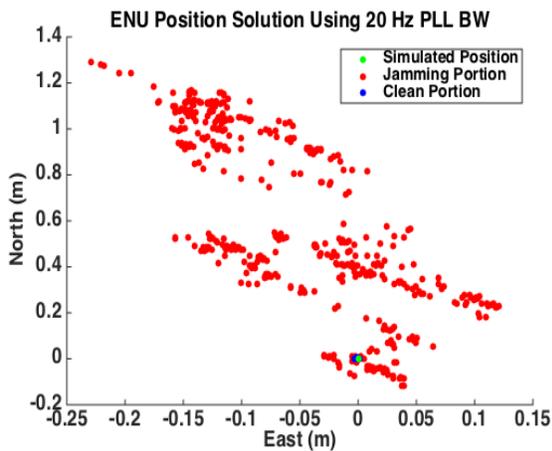
Figure 6.3: ENU Carrier Phase Position Using A PLL Bandwidth of 20 Hz
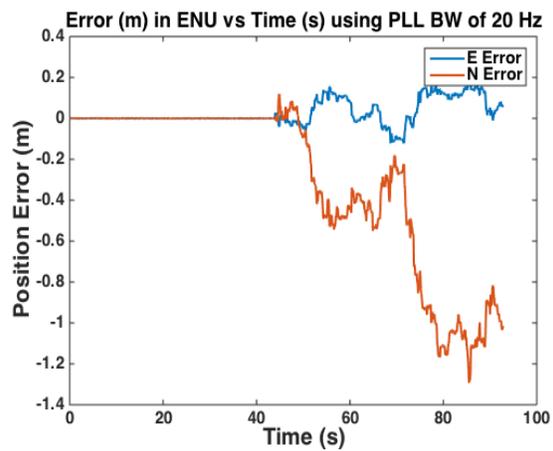
Figure 6.4: Error in Carrier Phase Position Using A PLL Bandwidth of 20 Hz

69

Using a phase lock loop bandwidth of 7 Hz should decrease the jitter of the Doppler estimates by a third, according to Figure 6.1, and therefore provide a much more stable position solution. The ENU position solution using a 7 Hz bandwidth can be seen in Figure 6.5 while the error in East and North direction can be seen in Figure 6.6. As seen, using the lower bandwidth greatly reduced the amount of error when compared to the 20 Hz solution. The software receiver is now able to calculate a static position and a position solution with an accuracy in the magnitudes of centimeters. However, because of the lower bandwidth, this receiver would only be able to be used in static scenarios and not dynamic scenarios.

Figure 6.5: ENU Carrier Phase Position Using A PLL Bandwidth of 7 Hz
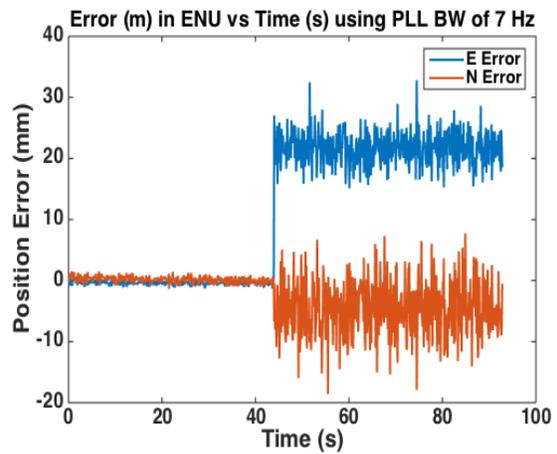
Figure 6.6: Error in Carrier Phase Position Using A PLL Bandwidth of 7 Hz

In order to fully leverage the benefit of a low phase lock loop bandwidth, an Inertial Measurement Unit (IMU) could be added to the system to aid the second order phase lock loop through the use of a tightly coupled tracking loop. An IMU can be used to calculate the user's three dimensional velocity in conjunction with the velocity of the satellite provided by the ephemeris to predict the Doppler frequency of the received signals. This allows the phase lock loop control law's bandwidth to be set without considering the dynamics of the user and satellite motion. Conversly, a third order phase lock loop controller could be used. With a higher order loop filter, the filter bandwidth can be reduced while also improving the dynamic performance of the loop filter at the expense of sensativity to jerk stress [12].

With the omission of the user and satellite dynamics, the next largest dynamic to be considered is that of the receiver clock, which is used to generate the replica signal for the carrier and code wipe off process. The clock dynamics are much lower than the dynamics of the satellite motion [6] and therefore the bandwidth can be lowered in the case of IMU aided signal tracking. The amount in which the bandwidth can be lowered is dictated by the phase noise of the clock and the holdover of the clock, or the amount of frequency drift that occurs to the clock while is not being disciplined. For GPS receivers, the internal clocks could be temperature compensated crystal oscillator (TCXO), oven controlled crystal oscillator (OCXO), or in some cases even use a highly stable atomic clock, such as Rubidium or Cesium. Using TXCO clock the PLL bandwidth can be set between 5 and 12 Hz, for an OCXO the bandwidth can be set to a few Hertz, and for an atomic clock the bandwidth can be set as low as 1 Hz [6]. While the receiver was simulated and processed in software with a perfect clock, allowing the bandwidth to be dropped below a realistic value, a bandwidth of 1 Hz was used show a best case scenario for positioning performance in a high jamming environment.

The final position solution was calculated using a second order PLL filter and a PLL bandwidth of 1 Hz and a simulated perfect IMU to predict each satellite's Doppler frequency. The ENU position solution using a tightly coupled tracking loop with a phase lock loop

bandwidth of 1 Hz can be seen in Figure 6.7 while the error in East and North direction can be seen in Figure 6.8. As seen in the figures below, the use of the 1 Hz bandwidth allows for carrier phase tracking at the millimeter level whereas the typical bandwidth receiver was only able to track with error on the order of meters. The use of the tightly coupled IMU aiding tracking loop also doesn't limit the receiver to be static in order to maintain high accuracy position but allows the receiver to have dynamic trajectories as well.
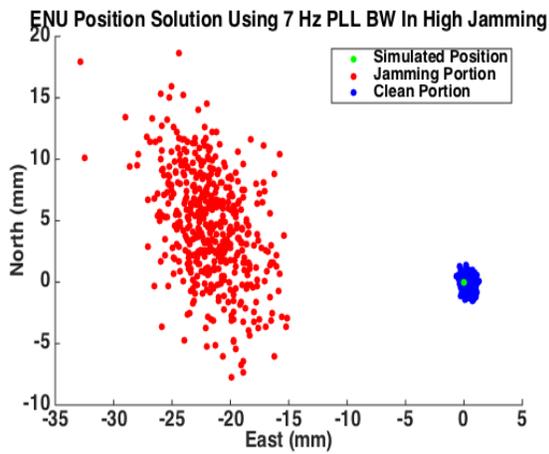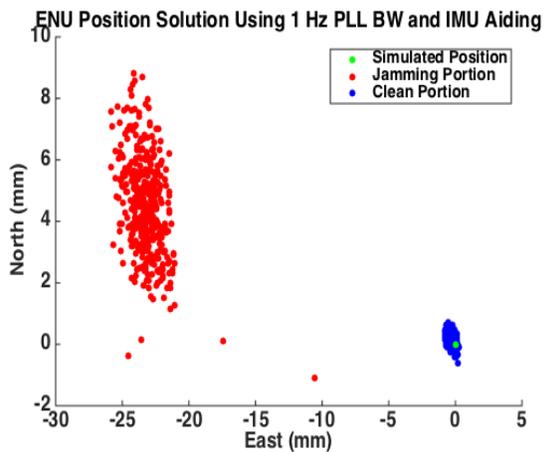

Figure 6.7: ENU Carrier Phase Position Using A PLL Bandwidth of 1 Hz
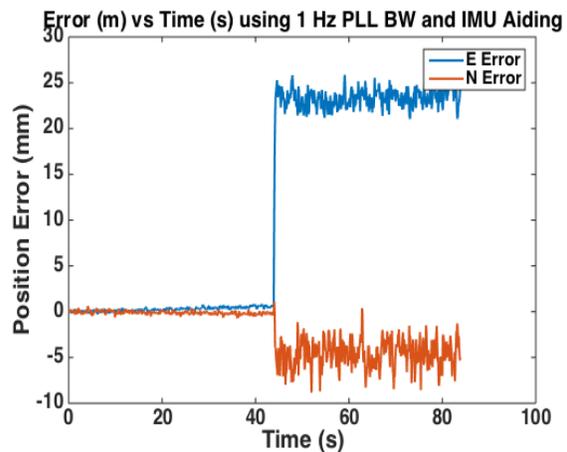
Figure 6.8: Error in Carrier Phase Position Using A PLL Bandwidth of 1 Hz

6.2 Dynamic Receiver Testing

Finally, the experiment was then repeated with single jammer but now using a dynamic trajectory, where the receiver was traveling around 50 mph southbound and the jamming power level was constant throughout the simulation. The NCO Doppler jitter vs J/S value and bandwidth can be seen in Figure 6.9. As expected, the result of Figure 6.9 are similar to that of Figure 6.1 in terms of Doppler jitter for various J/S values. The main difference between the two is the increased quantity and magnitude of jitter spikes for the lower band-width receivers. This is due to the fact that the dynamics of the incoming signal are now a function of the satellite motion and the receiver motion and the lower bandwidth receivers are not fast enough to maintain phase lock. Using Figure 6.9, it can be seen that the lowest

bandwidth possible without the risk of sudden increases in Doppler jitter is around 12 Hz for this particular trajectory. As the receiver moves faster, the lowest possible bandwidth will increase.
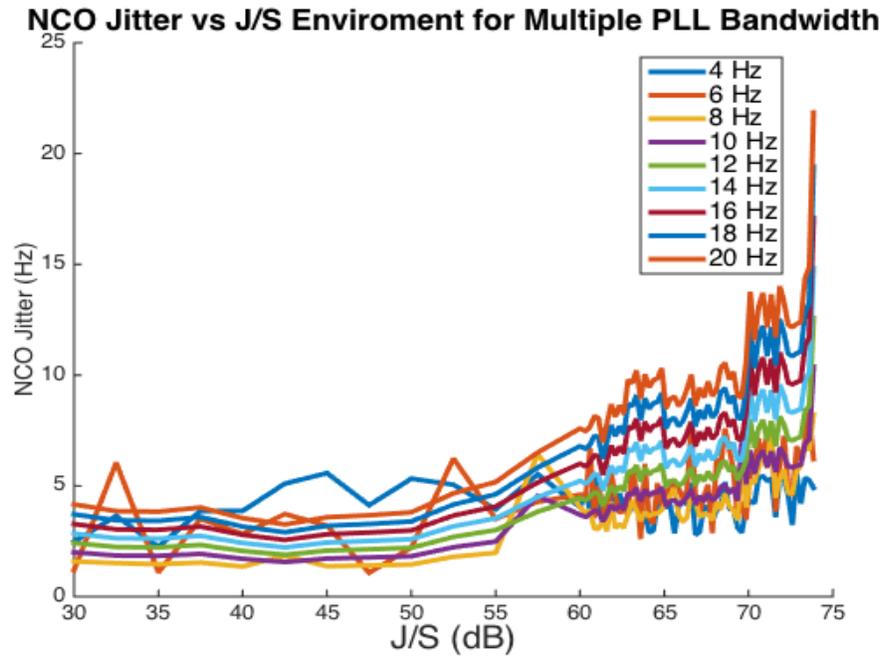
Figure 6.9: Dynamic Testing of Doppler Jitter for Various PLL Bandwidths and J/S Values Using Power Minimization and Four CRPA Antenna

6.3 Chapter Conclusion

In this chapter, experimental data of receiver performance in various jamming scenarios was explored. For these experiments, IF data files for a four element half wavelength spacing CRPA antenna were simulated for a receiver using power minimization. The receiver was repeatedly tested with a static scenario in the presence of a single jammer with various J/S values and with various PLL bandwidth in order to determine the jitter of the Doppler frequency estimate. It was determined that the receiver's calculated position solution will degrade as the J/S value increases, however it is not a linear relationship. The receiver's performance will contain sections in which the jitter does not increase with the J/S. It was shown that the PLL bandwidth has a great effect on the receiver's ability to filter out the

73

noise caused by the jammer but that the PLL bandwidth is limited by the user-satellite dynamics. Finally, the testing was repeated using a dynamics trajectory in which the results corroborated with the results of the static scenario.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

This thesis set out to evaluate the error when performing carrier phase positioning with a controlled reception pattern array (CRPA) antenna receiver. However, when performing carrier phase positioning with CRPA antennas there are a number of areas in which the carrier phase measurement can be distorted. Part of the issue is the physics of the application where closely spaced antennas can bias the carrier phase measurement and cause positioning errors through mutual coupling. Another area of concern is the group delay bias created by STAP algorithms when performing differential GPS. This thesis used a perfect antenna environment simulator in order to examined the error strictly due to Least-Mean-Square algorithm and the error due to noise in the tracking loop by not simulating effects such as mutual coupling. This was done in order to understand the amount of error strictly caused by the applied algorithm.

The error analysis was comprised of two main experiments: errors found when using unconstrained recursive algorithms and the positioning error that occurs as the jamming strength is increased. Using the recursive Least-Mean-Square algorithm, it was found that an unconstrained recursive algorithms cause a small position error that grows over time. It was also determined that the severity of the position drift is independent of the jamming environment but is dependent upon the learning coefficient. While a more aggressive learning coefficient is more desirable in a jamming environment, the error was found to be the largest when using a large learning coefficient. The cause of the position drift was hypothesized to be from the phase lock loop NCO attempting to maintain phase lock with the changing phase shift being applied to the reference element. This claim was validated by comparing

the phase shift applied to the reference element and the severity of the position drift. The error was mitigated through the process of normalizing the weights so that the algorithm does not phase shift the reference element. The use of the normalized weights provides a constant position solution with the ability to have a fast response time. Second, the effect of changing the phase lock loop (PLL) bandwidth on an anti-jam enabled receiver was examined by determining the jitter on the Doppler frequency for various jamming levels. It was shown that the carrier phase position was capable of maintaining phase lock at high jamming levels with the use of lower bandwidth PLL. While the use of a lower bandwidth limits the receiver to be static, extremely low bandwidths can be implemented with the aid of an Inertial Measurement Unit through a tightly coupled tracking loop.

7.2   Future Work

Though this thesis does provide insight into the type of errors that can occur when performing carrier phase positioning with a CRPA receiver, there still exist a number unexplored cases that merit research. While the normalization process does prevent position drift from occurring due to the time variant phase shift, there still exist a small drift even when using the normalization process. This remaining small amount of drift only exist when anti-jam algorithms are being used, therefore the remaining drift is most likely due to another side effect from the recursive algorithm or is a side effect of positioning in a jamming environment. Another topic of interest is testing the error of recursive algorithms with different testing scenarios than what was tested in this thesis. For example, the case in which a STAP LMS algorithm transitions from nulling an impulse jammer to a narrow or wide band jammer to test the weighting transition between frequency nulling. The second testing enhancement would be that of the LMS algorithm's carrier phase position error in a multi-path environment with a dynamic receiver. In this test case while only a single jammer may be present, there will be multiple perceived jammers in which the algorithm will have to quickly adapt to null the interference source and maintain positioning.

In the case of the NCO Doppler jitter testing, future work would possibly include different jamming scenarios and CRPA algorithms. This thesis only tested the scenario in which a single impulse jammer was used with CRPA receiver performing power minimization. It would be of interest to try different quantities of jammers or various jammer bandwidths in combination with different types of algorithms. This would give insight into how each algorithm uniquely performs as the jamming strength is increased. Another future experiment would be the inclusion of mutual coupling into the simulation environment. Mutual coupling is a physical phenomenon that occurs when antenna elements are closely placed together, as in the case of a CRPA antenna, and can change the characteristics of the antenna and provide an additional source of error. It would be of interest to see the comparison of the magnitude of error when compared to error from recursive drifting or the effect of the jammer degrading tracking loop performance.

Bibliography

[1] GLOBAL POSITIONING SYSTEM (GPS), howpublished = `http://what-when-how.com/space-science-and-technology/global-positioning-system-gps-2/`, note = Accessed: 2017-02-17.

[2] Sidney P. Applebaum. Adaptive arrays. In *IEEE Transactions on Antennas and Propagation*, pages 585 – 598. IEEE, 9 1976.

[3] Kai Borre, Dennis M. Akos, Nicolaj Bertelsen, Peter Rinder, and Soren Holdt Jensen. *A Software-Defined GPS and Galileo Receiver - A Single Frequency Approach*. Birkhauser, Boston, Massachesetts, 2007.

[4] Paul de Jonge and Christian Tiberius. *The LAMBDA method for integer ambiguity estimation: implementation aspects*. Delft Geodetic Computing Centre, Delft, Netherlands, 1996.

[5] David DeLorenzo. *Navigation Accuracy and Inteference Rejection for GPS Adaptive Antenna Arrays*. PhD thesis, Stanford University, Stanford, CA, 8 2007.

[6] Per Enge and Pratap Misra. *Global Positioning System - Signals, Measurements, and Performance*. Ganga-Jamuna Press, Lincoln, Massachesetts, 2011.

[7] J.A. Farrell, T.D. Givargis, and M.J. Barth. Real-time differential carrier phase gps-aided ins. In *IEEE Transactions on Control Systems Technology*, pages 709 – 721. IEEE, 2000.

[8] Inder J. Gupta and Thomas D. Moore. Space-frequency adaptive processing (sfap) for interference suppression in gps receivers. In *Proceedings of the 2001 National Technical Meeting of The Institute of Navigation*, pages 377 – 385. The Institute of Navigation, 2001.

[9] Inder J. Gupta and Thomas D. Moore. Space-frequency adaptive processing (sfap) for radio frequency interference mitigation in spread-spectrum receivers. In *IEEE Transaction on Antennas and Propagation*, pages 1611 – 1615. IEEE, 2004.

[10] Christopher R. Hamm, Warren S. Flenniken IV, and David M. Bevly. Comparative performance analysis of aided carrier tracking loop algorithms in high noise/high dynamic environments. In *Proceedings of the 17th International Technical Meeting of the Satellite Division*, pages 523 – 532. The Institute of Navigation, 2004.

[11] Otis Lamont Frost III. An algorithm for linearly constrained adaptive array processing. In *Proceedings of the IEEE Volume No. 8*, pages 926 – 935. IEEE, 7 1972.

[12] Elliott D. Kaplan. *Understanding GPS Principles and Applications.* Artech House Publishers, Norwood, MA, 1996.

[13] Ung Suok Kim. Analysis of carrier phase and group delay biases introduced by crpa hardware. In *Proceedings of the 18th International Technical Meeting of the Satellite Division*, pages 635 – 642. The Institute of Navigation, 9 2005.

[14] Min Li and Feixue Wang. A gnss software receiver beamforming architecture. In *GNSS PNT Symposium*. GNSS PNT Symposium, 2001.

[15] Thomas Dean Moore. *Analytic Study of Space-Time and Space-Frequency Adaptive Processing For Radio Frequency Interference Suppression.* PhD thesis, The Ohio State University, Stanford, CA, 5 2002.

[16] Grard Lachapelle Niranjana Vagle, Ali Broumandan. Analysis of multi-antenna gnss receiver performance under jamming attacks. *MPI*, 11 2016.

[17] Rahim A. Qamar and Noor M. Khan. Null steering, a comparative analysis. In *Proceedings of the IEEE Internation Multitopic Conference*. IEEE, 12 2009.

[18] Charles Powell Joshua Starling and David M. Bevly. A multiple-antenna software gps signal simulator for rapid testing of interference mitigation techniques. In *Proceedings of the ION International Technical Meeting*. ION, 2017.

[19] Bernard Widrow and Samuel D. Stearns. *Adaptive Signal Processing.* Pearson, London, England, 1985.

[20] Michael D. Zoltowski and Anton S. Gecan. Advanced Adaptive Null Steering Concepts for GPS. In *Proceedings of the Military Communications Conference*, page 1214. The Institute of Electrical and Electronics Engineers, 7 1995.

Appendices

.1 Gain Pattern Calculation

While the antenna characteristics and gain pattern of a single antenna receiver are fairly typical and can be determined by the given data sheet, when multiple antennas are combined the receptivity of the array becomes unclear due to it's large dependency on array geometry. The equivalent array gain pattern becomes even more unclear when used with a CRPA anti-jam receiver due to the channel specific phase shifts and scaling applied by the CRPA algorithms to attenuate the reception of interference sources. This section intends to explain the process in which the gain pattern is calculated for a multiple antenna array with the weighting applied by the CRPA receiver.

To calculate the array pattern, the array dimensions must be known a-priori. While the applied weighting does change the gain pattern of the array, the dimensions and geometry of the array largely impact the gain pattern. As the the elements become spaced further apart, the beams or nulls become narrower. As the elements are moved closer, beams or nulls become wider. For these reasons, the array dimensions must be precisely known to show the true array gain pattern.

The gain pattern calculation for a specified azimuth and elevation can be shown in Equation (1).

$$GP(Az, El) = \sum_{k=1}^{K} W_k e^{\left(\frac{j2\pi(p(k,:) \cdot r(Az,El))}{\lambda} \right)} \tag{1}$$

The gain pattern is calculated by summing the total phase shift relative to the reference element for each of the K elements. While the phase shifting and scaling from the CPRA algorithm, W_k is obtained from the tracking loop, the spatial phase shift of the signal relative

to the reference element is dependent upon a number of factors. The first factor is the array dimensions as denoted as p in Equation (1). P is a $Kx3$ matrix which shows the dimension of the array, as shown in Equation (2), where a four element square array is shown with equal spacing symbolized as the variable d.

$$p = \begin{bmatrix} 0 & 0 & 0 \\ d & 0 & 0 \\ 0 & -d & 0 \\ d & -d & 0 \end{bmatrix} \tag{2}$$

In Equation (1), the dimensions of the kth element are dot multiplied with the three dimensional bore sight vector shown in Equation (3).

$$r(Az, El) = \begin{bmatrix} sin(Az)cos(El) & cos(Az)cos(El) & sin(El) \end{bmatrix} \tag{3}$$

The bore sight vector uses the azimuth and the elevation angles that are the desired angles in which the array's gain is to be calculated. To calculate the entire three dimensional gain pattern, the process is to be repeated for each azimuth ranging from zero to three hundred and sixty degrees and elevation angles ranging from zero to ninety degrees.

.2   Jammer Localization Using a Network of CRPA Receivers

While CRPA anti-jam receivers allow a user to position in a hostile GPS environment, any non-AJ receiver will unable to position. The lack of CRPA anti-jam receiver ubiquity is caused by the cost difference between a CRPA receiver and a standard single element receiver as well as size constraints. The increase in cost is due to the extra cost of materials for the additional antennas as well as the increase computing power required to perform STAP and SFAP algorithms. On top of the increase cost, the size of the CRPA anti-jam receiver and additional hardware make it improbable for a dismounted solder to use a CRPA anti-jam receiver. This limits the application of CRPA anti-jam receivers to vehicle platforms, such as

ground vehicles or aircraft. These constraints limit the number of receivers that can position to a fraction of the total number of receivers. To remedy this issue, the concept of using a network of CRPA receivers to locate the jamming source is examined so that the jammer can be located and disabled.

The methods of jammer localization in a local polar frame, Azimuth and Elevation, can be done in a variety of different ways. These methods range from the classical Multiple Signal Classification, also known as MUSIC, to Estimation of Signal Parameters via Rotation Invariance Technique, also know as ESPRIT. While these algorithms have their own advantages and disadvantages, this section desires to implement a jammer localization method using as much of the existing processed data possible from the CRPA anti-jam receiver. Primarily, the proposed system will attempt to determine possible azimuths and elevations to a jamming using the applied weights of the CRPA anti-jam receiver on each channel. As discussed in Appendix Section 1, the effective gain pattern of the array can be calulated using the array's element spacing and weighting applied from the anti-jam algorithm. For example, in Figure 1 it can visually be determine that there are nulls at an azimuth of zero and one hundred and eighty degrees. While it is unclear which null is the desired null and which null is a sympathy null, it can be assumed that at least one of the nulls points in the direction of the jammer. With an series of possible azimuth and elevation toward the jammer, a network of spatially diverse receivers could locate the jammer either on a planer xy coordinate frame with a minimum of two receivers or locate the jammer in an xyz coordinate frame using three or more receivers. This section shows the process in which this localization technique can be performed, sources of error which will degrade the position solution, and the results of a theoretical experiment.

.2.1   Jammer Localization Methodology

To test the jammer localization method purposed in this section, an experimental jammer scenario was created as mapped in Figure 2. Here it can be seen that three spatially
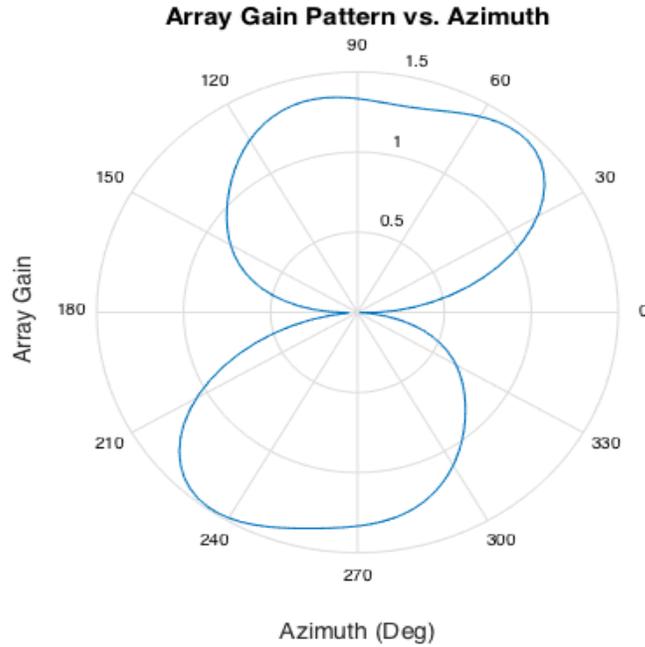
Figure 1: Array Pattern Of CPRA Antenna

diverse CRPA receivers will be used to locate the jammer. The method will attempt to locate a single jammer in a local East, North, Up (ENU) coordinate frame using one of the receivers as the origin. For this experiment the furthest left receiver, receiver one, was used, but the application can be performed using any receiver. Using receiver one as the ENU origin, the coordinates of each of the receivers and the jammer as shown in Figure 3.
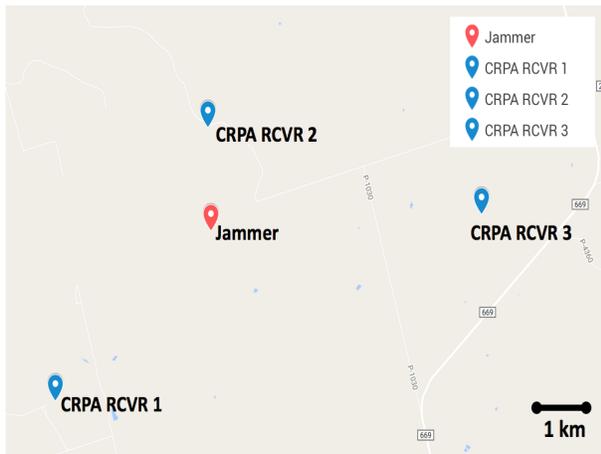

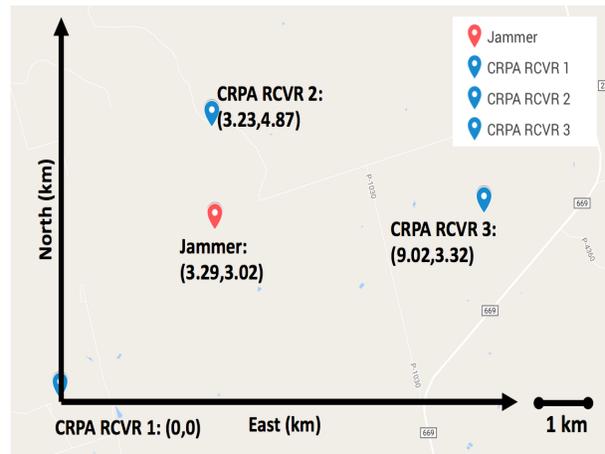Figure 2: Location of Receivers and Jammer

Figure 3: Scenario Map in ENU Frame

To detect possible nulls, first the array pattern must be calculated, as discussed in Appendix Section .1. While Appendix Section .1 calculates the gain pattern for all possible azimuth and elevations, this section will assume the jammer is relatively low to the ground and only the elevation angle of zero degrees is needed. If this assumption is not taken, the principles shown in this section can still be used, but the three dimensional jammer location must be solved instead of the two dimensional solution. Using the calculation of Section .1, the gain pattern of the array with the applied CRPA weights was calculated and can be seen in Figure 4. To detect the nulls, a threshold method is used to flag any angles in which the gain pattern drops below a certain value. For this experiment, a threshold of one was used.
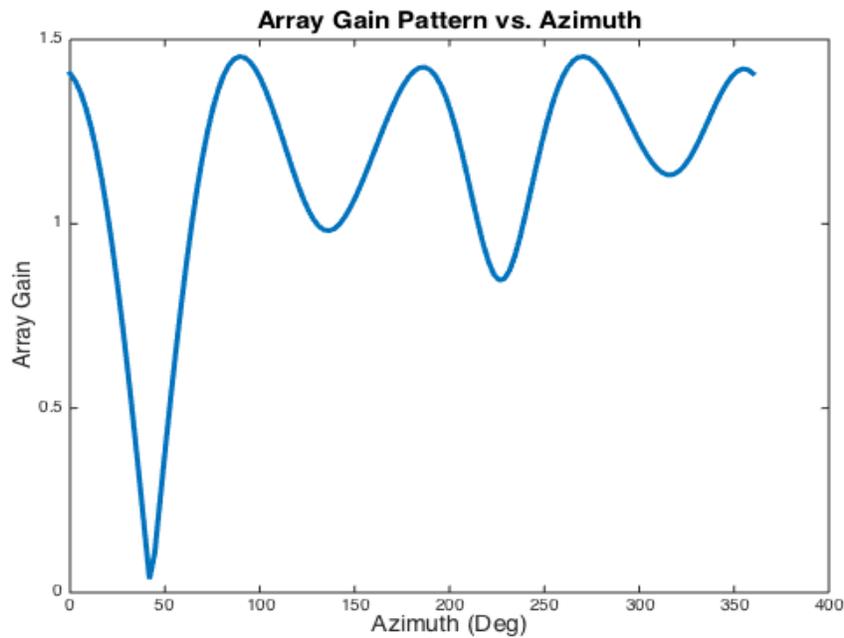

Figure 4: Array Gain Pattern at an Elevation of Zero Degrees

Using the threshold method to flag any possible nulls, the array pattern of receiver one was evaluated for possible nulls, as shown in Figure 5. As it can be seen in the figure, three possible sections where flagged as possible nulls denoted as the non-zero value of the orange line. Due to null being wider than a single degree, a range of sequential values were flagged as possible nulls. To condense these values to a single null, the minimum value of the sequential values were taken as the null angle. This process was repeated for receivers two and three

and the detection results can be seen visually in Figure 6 and Figure 7 or numerically for all of the receivers in Table 1. Each of the receivers were able to determine the true null angle, where the largest error was three degrees, as well as the sympathy nulls.

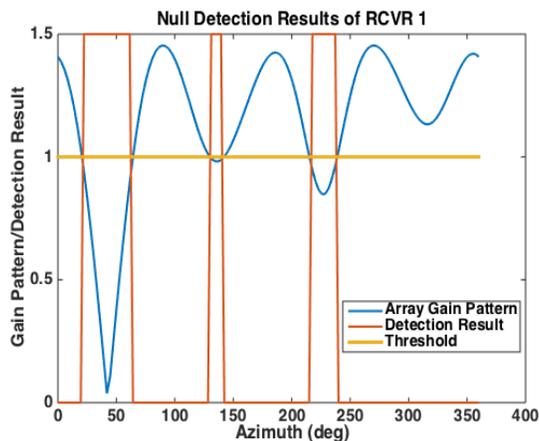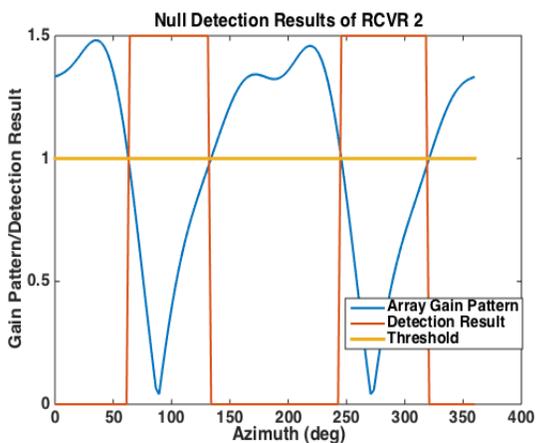Figure 5: Detection Results of Nulls For Receiver 1
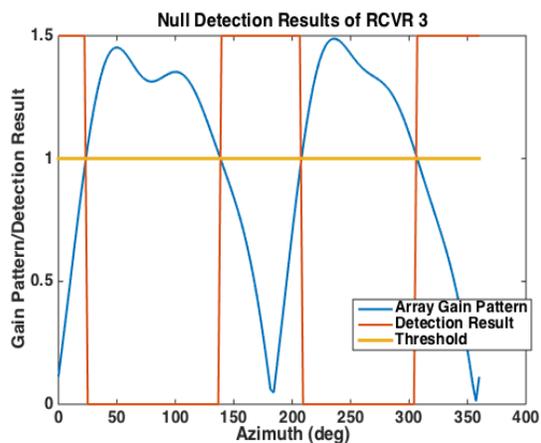

Figure 6: Detection Results of Nulls For Receiver 2

Figure 7: Detection Results of Nulls For Receiver 3

Table 1: Nulling Detection Results

RCVR	Angle to Jammer (deg)	Found Nulls (deg)
RCVR 1	42.58°	41.1024° 131.8110° 182.9812°
RCVR 2	271.8254°	86.4567° 273.5433°
RCVR 3	182.9812°	182.9812° 358.58.27°

To manage the multiple number of detected nulls, the true jammer location is determined using a brute force method to process every possible null angle combination. For the test scenario with the given detection results, there are three receivers and a total of thirty-six different paired solutions. While the number of different combinations does propose an issue, for now assume that the three correct nulls are used and the issue of determining the correct solution will be answered later. To locate the jammer in the ENU coordinate frame, each of the nulls and receiver positions are converted to a linear point slope equation line that pass through the location of the receiver. The null angles are taken to be the slope of the line equation and the point where the line intersects the North axis is calculated using Equation (4).

$$b = N_{RCVR} - atan(\theta_{null})ERCVR \tag{4}$$

In Equation 4, b is the intersection of the North axis, N_{RCVR} is the receiver's North position, E_{RCVR} is the reciever's East position, and θ_{null} is the detected null angle. With the slope and the North axis intersection, the point slope equation can be determined for each receiver as illustrated in Figure 8.
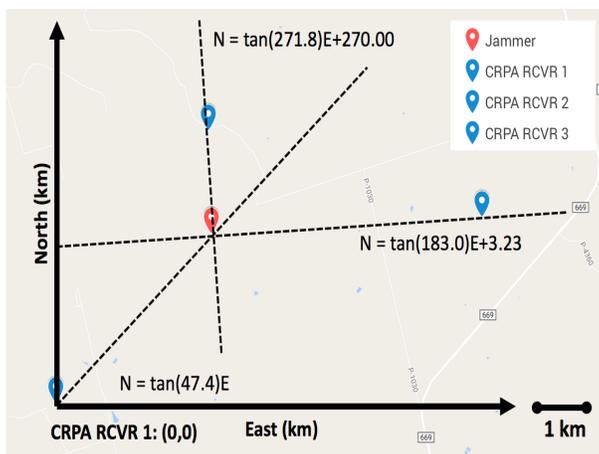

Figure 8: Intersection Point of Null Angles

Because the jammer localization problem was simplified to a two dimensional planner problem, only two equations are required to solve for the jammer's East and North Position.

If the two nulls point toward the jammer, the East and North intersection should be close to the jammer's position. The East coordinate of the intersection of the two lines can be determined using Equation (5).

$$E_{intersection} = \frac{b_1 - b_2}{m_1 - m_2} \tag{5}$$

Simiarly, the North posisiton of the intersection of the two lines can be determined using Equation 6.

$$N_{intersection} = m_1 * \frac{b_2 - b_1}{m_1 - m_2} + b_1 \tag{6}$$

With three receivers there are three different equation combination solutions: receivers one and two, receivers one and three, and receivers two and three. These solutions can be averaged together to form the estimated jammer position solution. However, the method of how to sort through every possible weighting combination has yet to be address. Using a cascading for loop method to process every possible null angle combination, the set of three possible jammer locations for a given set of nulls can be determined. Because incorrect combination of nulls will have largely different intersection points, the true combination of null angles should result in the position solution set with the lowest standard deviation. Each of the jammer position solutions were calculated and the standard deviation of the sets were taken and plotted in Figure 9. Here it can be seen that set four had the lowest standard deviation where the typical standard deviation was about three hundred and fifty times larger than set four. When comparing the average position error in the solution sets, shown in Figure 10, it can be seen that set four has the smallest error as well.

Using set four's average intersection, the results of the localization can be seen in Table 2. The jammer's location was determined to be 3.24 kilometers East and 2.91 kilometers North from receiver one. This means there was an East North error of 40.6 meters and 117.9 meters respectively. Clearly, this method cannot be used to precisely locate the jammer,
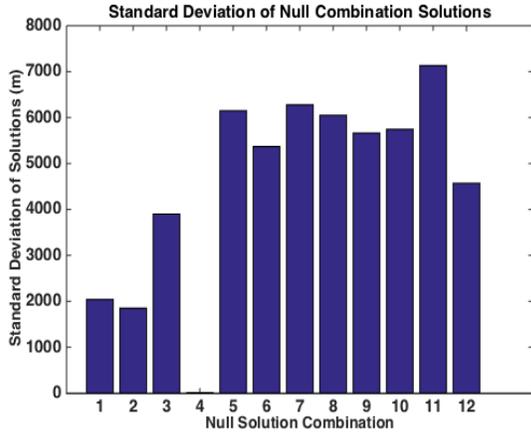
Figure 9: Standard Deviation of Jammer
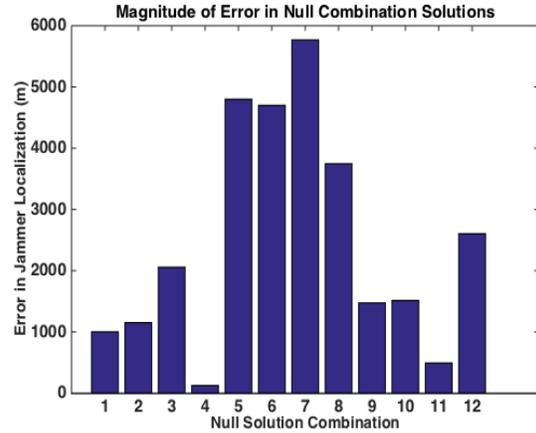Position Solutions

Figure 10: Error of Jammer Position
Solutions

but it can be used to place the jammer in a general area. There also is a possibility for the incorrect null angle combination to be selected because this is not a closed form solution. However, the probability of a false position solution decreases as the total number of CRPA receivers are used. Also the use of comparing the standard deviation between individual solution sets can be used to throw out all erroneous solutions to keep incorrect positions being reported when all null angle combinations are incorrect.

Table 2: Jammer Localization Results

	East (km)	North (km)
Truth	3.29	3.02
Estimated	3.24	2.91
Error	.0406	.1179

.2.2 Jammer Localization Error Sources

While the method described above can determine the position of a jammer, there are a number of different error sources that dilute the accuracy of the solution and areas that may lead to an incorrect position solution. This section intends to address these possible error sources and put forward methods that can be used to limit them. First, the localization of the jammer is highly dependent upon the position solutions of each CRPA receiver. As

expected, the receivers must be able to position in order to create the ENU local coordinate frame. As the uncertainty of in the receiver position increases, so does the uncertainty in the jammer's position. Also, the CRPA receivers must be spatially diverse in order to provide an accurate estimated position. Similar to how bad satellite geometry increases the user's dilution of precision (DOP), bad receiver geometry with respect to the jammer will dilute the position solution of the jammer.

The second source of error is the fact that the true null may not not be directly directly at the jammer. While adaptive algorithms such as power minimization should form a null in the direction of the jammer in order to minimize the total output power, this is not guaranteed when using an algorithm such as Least-Mean-Square. With the LMS algorithm, the gain pattern is changed in iterations such that the error between the desired and the weighted signal is minimized. This could mean that a beam could be pointed towards the satellite without a null towards the jammer or that a null was placed near a jammer but not directly at the jammer in order to obtain the correct amplitude in the mixed signal. Therefore, the algorithm used to determine the weights should be considered when using jammer localization and possibly only use algorithms that minimize the total output power or prioritizing maximizing the carrier-to-noise ratios of the GPS signals. It should also be noted that nulls not directly at the jammer will cause error in the jammer localization that increases as the spacing between the receivers and the jammer increases.

.2.3   Conclusion

In this section, a method of determining the location of a jammer was proposed using the weights from a network of CRPA receivers. An example of the method was done using real gain patterns obtained from a CRPA software receivers in a simulation environment. The possible sources of error was examined that could dilute the solution precision, such as the geometry of the receivers or the type of anti-jam algorithm used. Ways in which to reduce the probability of a false localization was also discussed using additional CRPA

receivers. It was shown that method was able to locate the jammer to a general area but was not able to precisely locate the jammer.