Improving CRPA Anti-Jamming Performance with Virtual Array Integration

by

Gabriel Wiggins

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, 2023

Keywords: GNSS, CRPA, Virtual Array, Genetic Algorithm

Approved by

Scott Martin, Chair, Assistant Professor of Mechanical Engineering
Lloyd Riggs, Professor of Electrical and Computer Engineering
David Bevly, Bill and Lana McNair Distinguished Professor of Mechanical Engineering

Abstract

This thesis explores the concept of integrating a virtual array into current controlled reception pattern antenna (CRPA) technology to increase performance. A four channel virtual CRPA is designed and simulated, and its anti-jamming power minimization performance is compared to that of a uniformly spaced CRPA. It is shown that a virtual array can be integrated into a controlled reception pattern antenna to increase anti-jamming performance of a static receiver tracking the GPS L1 signal. A genetic algorithm is designed to estimate an improved CRPA geometry in an arbitrary signal environment. It is shown that a genetic algorithm is an effective and efficient method of initialization of element position offsets and complex weights. The use of a virtual array enables steering of the antenna elements' phase centers, or points of effective radiation, with solely electronic means. The added flexibility in spatial diversity of the antenna elements create deeper nulls in the array factor, and thus in the overall gain pattern of the CRPA. With deeper nulls, more in-view satellites are acquired. This increases geometric diversity of satellites, which increases position accuracy. It is also shown that in identical jamming configurations, a CRPA employing a virtual array initialized with a genetic algorithm can remain viable in higher J/S environments than a CRPA with uniform spacing.

Acknowledgments

I would first like to thank the family and friends for the love and support I received while completing this work: Clay and Maria Wiggins, Shelby Patti, Breanna and Eric McDaniel, Sam and Madeline McDougal. Without your help, I would not have been able to finish grad school. Secondly, I'd like to thank my grandparents, Larry and Janice Huffstetter, for the continual encouragement and many home-cooked meals during my time in Auburn. Next, I'd like to thank Josh Starling for being an excellent mentor and teaching me about CRPAs and signal processing. Thanks to all the members of the GAVLab for the collaboration and sense of community. In particular, thanks to Christian Moomaw, who was always available to answer a Python or Linux question. Special thanks to every member of the Dead Reckoners Trivia Team and to Moe's Barbecue. No matter what lay in store, I could always count on you to provide 2 hours of fun every Wednesday night, and for that I am grateful. Lastly, thanks to all my advisors: Dr. Scott Martin, Dr. David Bevly, and Dr. Lloyd Riggs, for the advice and encouragement it took to finish this work.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Global navigation satellite systems (GNSS) have quickly developed into a piece of critical infrastructure in the United States and abroad. It is used for navigation for millions of users, ranging from the average consumer to precise military applications in highly dynamic environments. It's also used as a global time standard, keeping other pieces of infrastructure, such as the electric grid and the stock market, in sync. As reliance on GNSS increases, so does the need to protect it. GNSS signals face interference or disruption from a multitude of parties; there are more threats on GNSS signals today than ever before, and the number is only rising. In global conflicts, GNSS disruption is widely used and considered a normal part of warfare tactics [1]. In an event earlier this year, an interference event disrupted navigation signals and prevented aircraft from making safe landings at the Denver International Airport for over 33 hours [2]. Cheap, plug and play jammers designed to prevent employers from tracking company vehicles have exploded in popularity in recent years [3]. This has led to broad efforts to fortify GNSS signals and develop technology to withstand interference. In order to combat the reality of degraded or corrupted navigation signals, anti-jamming and anti-spoofing measures are necessary.

The most effective and sophisticated anti-jamming and anti-spoofing systems employ controlled reception pattern antennas. These antenna arrays act as spatial and frequency

filters to remove interfering signals through beamforming or null-steering methods. CRPAs provide some of the best interference rejection, but have drawbacks. CRPAs generally rely on digital signal processing algorithms which manipulate the individual signal samples from each element. Because CRPA algorithms filter data digitally on a per-sample basis before any other signal processing is done, they can be very computationally expensive. They also require antenna elements to be spaced between a quarter wavelength and half wavelength away from each other, to prevent mutual coupling and grating lobes, respectively. GNSS signals operate in the L–band (1–2 GHz). Thus, for relatively low frequencies such as the L band, arrays can become quite large if many elements are desired. The most straightforward way to increase CRPA capability is to increase the number of antenna elements; however, this also increases footprint, cost, hardware complexity, and computational cost.

The virtual CRPA uses phase center displacement to dynamically steer elements' perceived positions. Under a certain set of conditions, an antenna's phase center, or effective point of radiation, may be moved via electronic means. When placed into an array this results in a real-time reconfigurable array with no moving parts. This makes it possible to select an optimal array geometry to perform an anti-jamming algorithm with. The additional parameter space variability enables deeper and narrower nulls, which increases interference rejection capacity. This is accomplished without increasing computational complexity and without increasing footprint.

1.2 Prior Work

The field of interference suppression using antenna arrays [4, 5, 6, 7, 8] and adaptive processing [9, 4] is mature, with a wealth of resources available. Antenna Analysis and design [10, 11, 7, 12, 13, 14] is also a mature field with rich literature.

Zoltowski [5] investigated an optimal solution to minimize total power received by using covariance estimation. Widrow [9] developed a recursive error-driven algorithm for suppressing interference and maximizing SNR by driving the output to a reference signal using the least mean square algorithm.

Non-uniform arrays are capable of producing the best possible interference rejection in any given scenario. However, traditional arrays cannot be reconfigured as array elements are permanently fixed in place. A virtual array uses phase center displacement to electronically steer elements' phase center, or perceived location. This enables the CRPA to select the optimal geometry for a given scenario without having to physically move any elements.

Pour and Shafai investigated the intentional displacement of antenna phase centers [15, 16]. Mitha and Pour applied phase center displacement to a linear virtual array [17]; this thesis is highly derivative of this work. This thesis assumes a virtual CRPA can be created using similar construction methods as mentioned in [17], and investigates methods of computing optimal array geometry. In addition, this thesis seeks to quantify performance increase from a static-geometry CRPA to a virtual CRPA.

1.3 Contributions

There are three main academic contributions in this work. First, it is shown that anti-jamming performance can be increased by using a virtual array. Next, it is shown that a genetic algorithm can be used to initialize a static virtual CRPA with a more favorable array geometry. Finally, the genetic algorithm is further explored as an initialization technique for the virtual CRPA; various cost functions are tested for viability. These points can be summarized as follows:

- Design and analyze performance of an ideal virtual array CRPA compared to an ideal CRPA with uniform geometry
- Design and analyze a genetic algorithm can be used to initialize a virtual array geometry
- Show an analysis of various cost functions for use in a genetic algorithm associated with the virtual CRPA

1.4 Thesis Outline

This thesis is organized into background material, contributions and experiments, and conclusions/future work. Chapter 2 gives the reader an overview of the GPS Signal structure and basic GPS Signal Processing techniques. Chapter 3 provides an overview of GNSS antennas, including advantageous qualities, typical parameters, and derivations for several common GNSS antennas. Chapter 3 also introduces antenna array theory, which is imperative in anti-jam (AJ) concepts. In Chapter 4, the concept of AJ is introduced, as well as common AJ terminology and algorithms. Deterministic and adaptive approaches are discussed. Spatial processing is compared to space-time processing, and several algorithms are discussed in detail. Chapter 5 discusses virtual array theory, including phase center displacement, forming an array with adaptive geometry, and methods of control. Chapter 5 also discusses how a virtual array is integrated into GNSS anti-jamming and discusses genetic algorithms as a method of initialization. Chapter 6 outlines the simulation environment and experimental methodology. This is followed by a presentation and discussion of results. Chapter 7 discusses conclusions drawn from the experiments and recommendations for extensions of this work.

Chapter 2

GPS Signal Processing

2.1 Overview

In this chapter, an overview of GPS signal processing is given. First, the GPS L1 signal structure is examined. This includes description of binary phase shift key (BPSK) modulation, PRN sequences, and the power spectrum of the modulated signal. Next, serial and parallel signal acquisition are discussed. Next, tracking and demodulation are discussed. Finally, it is shown how to turn observed pseudoranges into a position solution using a least squares approach. The experiments performed in this work all use the GPS L1 signal. This signal is the oldest and most widely used by civilians. However, the principles carry over to other newer signals, even though they have different modulation schemes.

The Global Positioning System allows for real-time calculation of a user's position anywhere in the world. It is made up of three segments: the space segment, control segment, and user segment. The space segment contains the satellite vehicles that are responsible for broadcasting navigation messages. These coded messages make it possible to track navigation signals and use them to produce range estimates to the satellites. The information included in the signals allows the user to compute the satellite's position and velocity, clock correction terms, and information about constellation status. These signals are transmitted on an ongoing basis and are available 24/7 around the world. The satellites do not receive any information back from the user, making GPS a purely passive system [18]. The control segment is responsible for monitoring and maintaining

the satellite constellation. Since satellites will naturally drift from their prescribed orbits, the control segment makes periodic corrections to their flight paths. It also monitors satellite health parameters such as battery and propellant levels, and solar panel status. If needed, the control segment may deactivate or decommission a satellite, or activate spares. It also updates satellite ephemeris and almanacs, the essential data the user needs to calculate satellite position and velocity [18]. The final segment, the user segment, is made up of all user equipment receiving and decoding the GPS navigation messages.

Applications vary, but most users rely on GPS for navigation. This includes position, velocity, and attitude. The other main use for GPS is timing. Since GPS is available across the globe and is one of the most reliable and accurate sources of timing, critical infrastructure and any users requiring precision timing can use the GPS navigation message to discipline a local clock [18].

2.2 GPS Signal Structure

2.2.1 Modulation

The GPS L1 signal is Binary Phase Shift Keyed, and is comprised of 3 elements: carrier, PRN, and data message. The carrier is a sine wave in the L-band with center frequency 1575.42 MHz. The L band was chosen for GPS because it can easily penetrate vegetation and weather conditions like rain, clouds and fog. BPSK signals are used to encode binary data onto the carrier using phase. If a 1 is encoded, the carrier remains in phase; however, when a -1 is encoded, the carrier is shifted 180° out of phase. The first of two binary messages encoded on the carrier is known as the PRN (pseudorandom number) or C/A (coarse acquisition) code. This message helps identify each satellite, and makes it possible to acquire and track the signal. The PRN is broadcast at a rate of 1.023 MHz and is 1023 bits long. Each complete cycle of a PRN, known as a chip, repeats every millisecond. Besides the PRN, the carrier wave is modulated with a data message containing information about the satellite and constellation used for positioning. This

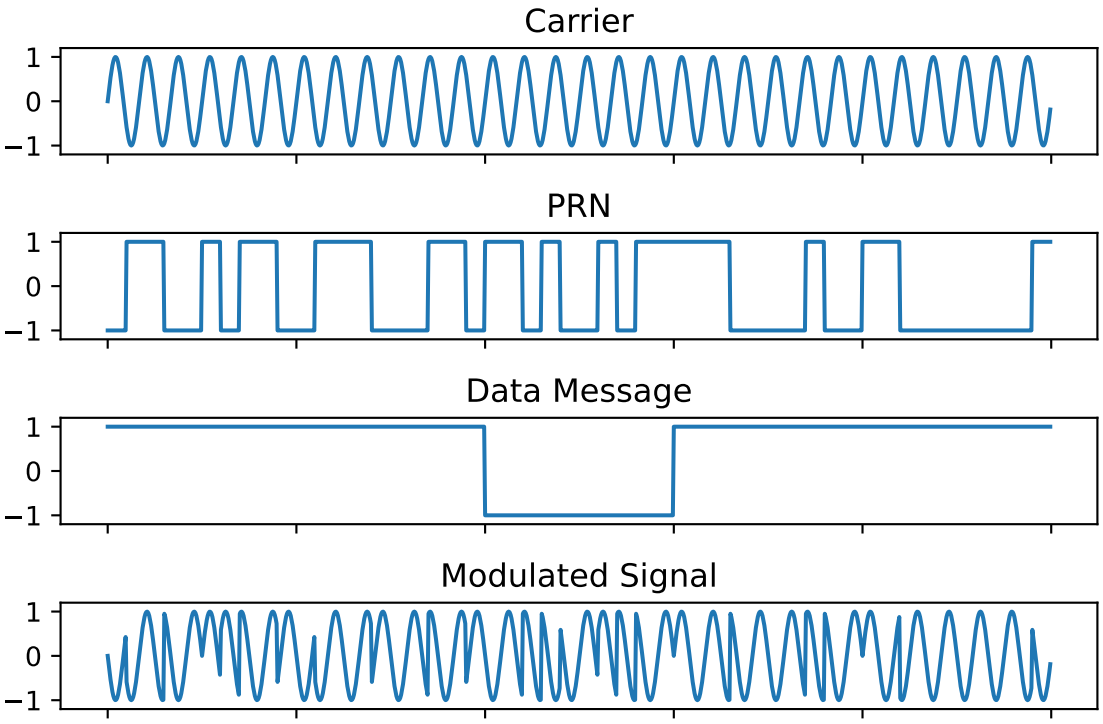

Figure 2.1: BPSK Modulation

message is broadcast at a much slower rate of 50 bps. This modulation scheme can be seen in Figure 2.1.

By modulating the signal, the energy is spread across a broad range of frequencies, resulting in a signal with wider bandwidth. This prevents the possibility of narrowband jamming. Narrow signals such as continuous wave (CW) signals can simply be filtered out. However, it also decreases spectral density. This contributes to GPS signals being received beneath the noise floor at very low power levels. The modulation also allows multiple signals to be tracked at the same time on the same carrier frequency, as discussed in section 2.2.2. GPS L1 C/A is modulated such that the main lobe is approximately 2 MHz wide. This signal is used widely by civilians and the code sequence is publicly available. Other signals, such as GPS L1 P(Y) are spread using a much longer code that spreads power even further away from the center frequency. P(Y) codes are not publicly available and have more interference mitigation capability due to the wider bandwidth; however, they also require receivers that are keyed with the private code and RF front ends that are much wider bandwidth.

2.2.2 Multiplexing

Multiplexing refers to the ability to send multiple streams of information over the same communication link. GPS uses a scheme called Code Division Multiple Access (CDMA) to multiplex navigation signals [18]. Different spreading codes (C/A codes) are used on each satellite. This enables the same carrier frequency to be broadcast from each satellite and for every satellite to broadcast navigation messages simultaneously. This is in contrast to Frequency Division Multiple Access (FDMA) and Time Division Multiple Access (TDMA). FDMA signals are broadcast on different carrier frequencies, and TDMA signals are broadcast in bursts which are separated in time. CDMA has the unique ability to use the same carrier frequency and broadcast all signals simultaneously. This prevents signals from being confused with one another, and allows the receiver to continuously track all available signals.

2.2.3 PRN Sequence

PRNs belong to a larger set of binary sequences known as Gold Codes. These codes have the special properties of only correlating with themselves in one place, not correlating with any other Gold Code, and being predictable and easily duplicated [18]. They are created with linear feedback shift registers, a logic circuit that takes in initial conditions known as taps, and returns a unique repeating sequence. A table of initial taps can be found in the GPS Interface Control Document (ICD) [19]. Examining the autocorrelation properties of PRN 1, it's apparent that it only correlates with itself in one place. This can be seen in Figure 2.2. It also doesn't correlate with any other PRN code. This is essential for isolating individual navigation signals and not mistaking them for others. These sequences are unique to each satellite. They serve as satellite identifiers as well as aid in tracking and pseudorange formation. GPS L1 PRN's are of length 1023 and broadcast at a chipping rate of 1.023 MHz, making each sequence, or chip, 1 msec long. The PRN repeats in perpetuity and allows the receiver to track the navigation signal.

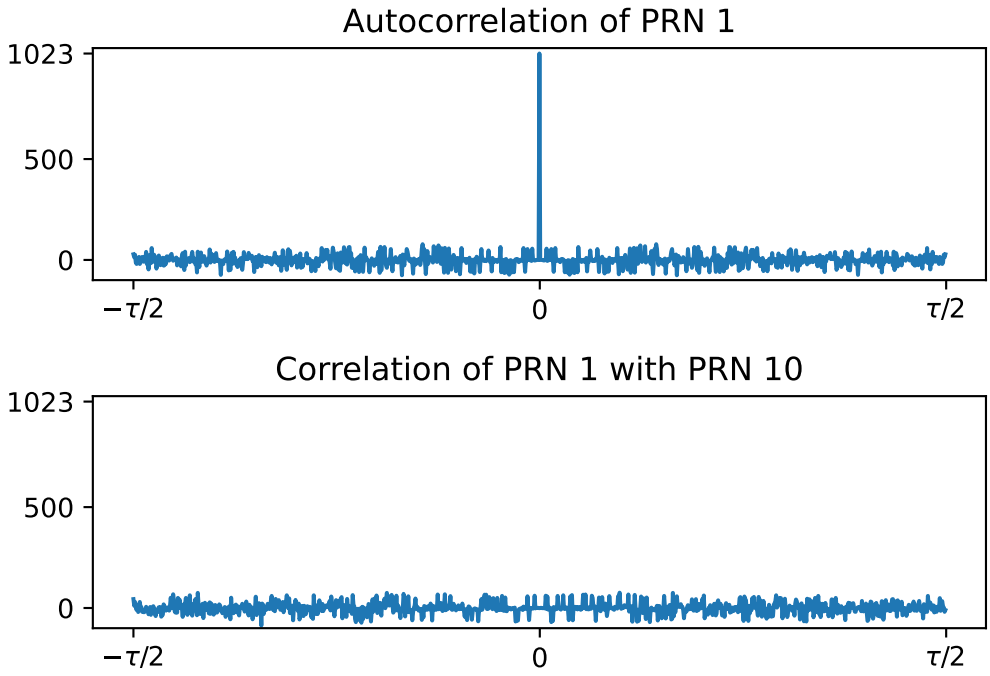

Figure 2.2: Autocorrelation and Cross-Correlation of GPS PRN Sequences

2.2.4 Data Message

The data message on the GPS signal has a data rate of 50 bps, and contains information about the satellite and its orbit. This includes satellite health, time, errors, orbital parameters, etc. The set of orbital parameters, known as the ephemeris, enables the user to calculate the position and velocity of the satellite and the time of transmission of the navigation message. This in turn enables the user to determine the pseudorange to the satellite. The pseudorange is estimated using time difference of arrival. Using ephemeris, the receiver calculates the position of the satellite and ascertains at what time a navigation signal is broadcast. It then estimates the time it takes the signal to reach the receiver, and multiplies by the speed of light to get a pseudorange. Data message contents and organization can be found in the GPS ICD [19].

2.3 Acquisition

For the GPS signal to be useful, it must be tracked and decoded. Since the GPS signal is received beneath the noise floor, it is not possible to track it directly. First, it must be acquired through correlation and integration to get a rough estimate of the Doppler shift and code phase. Acquisition is the first step in tracking and decoding. It is used to kick off the tracking process, where estimates will be refined, and the signal will be decoded. After the signal is mixed down to baseband, acquisition begins with generating a local copy of the PRN in question. Because PRNs only correlate with themselves in one place, a search over the PRN's code phase is necessary. The signal center frequency is known; however, the satellite and user dynamics introduce Doppler shift. Thus, it is also necessary to perform a Doppler search. This turns acquisition into a two-dimensional problem: estimating Doppler frequency and code phase. Once these have been properly estimated, then tracking can begin.

2.3.1 Serial Acquisition

For each channel, the serial acquisition algorithm searches in code phase and Doppler shift sequentially. Since the code phase is 1023 chips long, the code phase axis of the acquisition plane is 1023 long. The Doppler axis typically ranges from -5 kHz to +5 kHz in 500 Hz increments. Since this search is performed sequentially, it requires the correlation to be performed a number of times equal to the number of code offsets multiplied by the number of Doppler shift bins. This is because a correlation is performed for each code phase shift sequentially for every Doppler shift bin.

2.3.2 Parallel Acquisition

Parallel acquisition allows the receiver to simultaneously search through either all Doppler bins, or all code phases. In a software receiver, this can make acquisition a faster operation. Both operations require FFT's, or Fast Fourier Transforms.

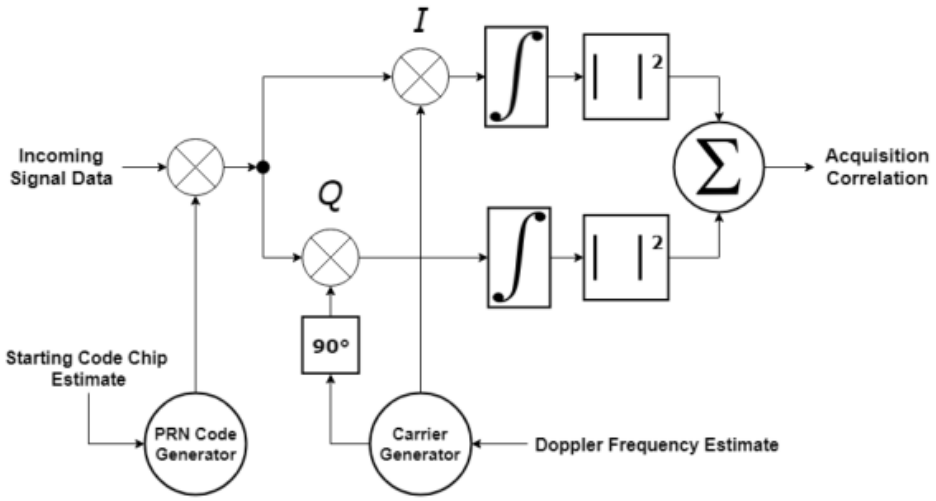

Figure 2.3: Serial Acquisition Algorithm

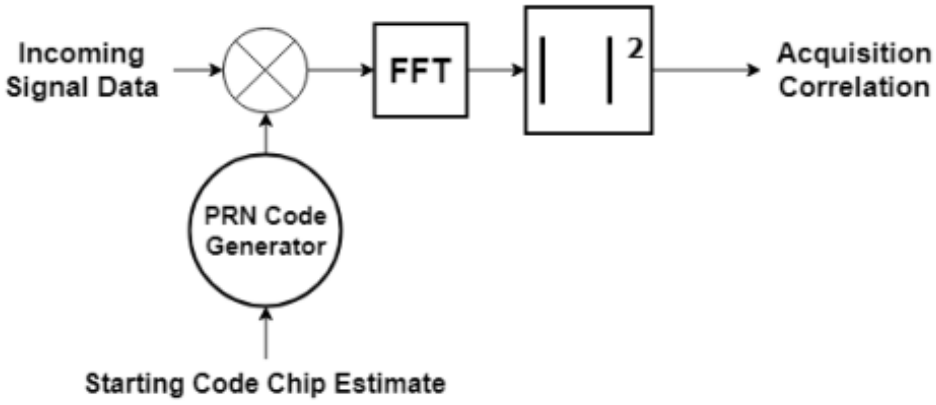

Figure 2.4: Parallel Frequency Search Algorithm

The parallel frequency search algorithm searches all Doppler shift bins at the same time. This is accomplished by mixing the code offset estimate with the incoming signal, then taking the squared magnitude of the FFT. This reduces the number of calculations to the number of code offsets. This algorithm is shown in Figure 2.4.

The parallel code search algorithm first mixes the I and Q (In-phase and Quadrature) components of a locally generated carrier with the I and Q of the incoming signal. These are combined, and the FFT is taken. This is mixed with the complex conjugate of the FFT of each PRN sequence. The result is found by taking the magnitude squared of the inverse FFT of this signal. This algorithm is shown in Figure 2.5.

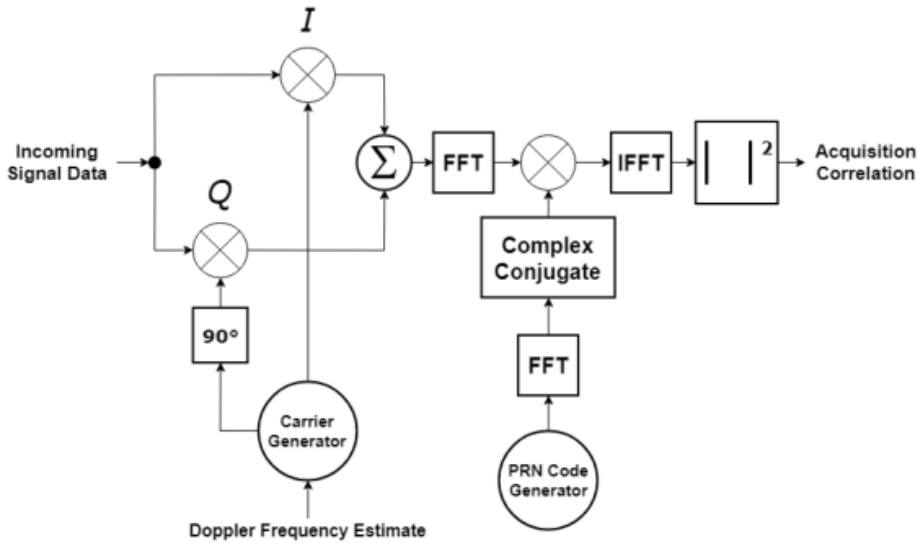

Figure 2.5: Parallel Code Search Algorithm

2.4 Tracking

2.4.1 Phase Lock Loop

After completing acquisition, the receiver has a close estimate of the code phase and Doppler frequency of the incoming signal. The next step is tracking. To generate accurate range estimates and decode the data message of the incoming signal, the code and carrier phase must be accurately tracked. This is because the primary function of the GPS receiver is carrier wipeoff. Carrier wipeoff refers to removing the sinusoidal carrier component of the signal so that only the data message and ranging code are left. The resulting signal is called the baseband signal. Because the data message and ranging code shift the phase of the signal by 180 degrees, the tracking loop must be resiliant to changing phase. In general, any phase lock loop that is insensitive to 180 degree phase changes is called a Costas PLL. The phase lock loop serves the promary purpose of tracking the incoming signal's frequency and phase. First, a replica sinusoid at the frequency given by acquisition results is generated. This frequency is close, but not perfect. The replica is copied and shifted to produce and in-phase arm and a quadrature arm. Both arms are correlated with the incoming signal, and compared via a discriminator to determine phase error. In a functioning PLL, correlation of the in-phase arm will remain high,

and correlation of the quadrature arm will remain low. The Costas PLL discriminator is shown in Equation 2.1 [20]. This term represents the phase error between the replica signal and the incoming signal.

$$\phi_{PLL} = \arctan\left(\frac{QP}{IP}\right) \tag{2.1}$$

2.4.2 Delay Lock Loop

Just as the carrier frequency has a Doppler shift, so does the code frequency. At the time of acquisition, the receiver only has a coarse estimate of code offset. The DLL is the mechanism for refining this estimate and tracking the code accurately. Tracking the code is vital for a GPS receiver as it enables decoding of the navigation message and formation of pseudoranges. The delay lock loop leverages the fact that each PRN has maximum autocorrelation at exactly one point, and will try to align the replica code with this peak. However, one correlation of the replica code with the incoming code is not enough because it creates an ambiguity. It is impossible to tell with only one correlation measurement whether the replica is too early or too late in the code. The autocorrelation of a PRN sequence is shown in Figure 2.6 [6]. The autocorrelation function has a peak in one place, and is linear in the pull-in range of 1 chip behind to 1 chip ahead. Three correlation measurements with the incoming code are created to ensure the precise code shift is observed. These are known as the early, prompt, and late correlators. When the early and late correlators produce the same measurement, it must be true that the prompt measurement is at the maximum value. The error in code phase is determined by a normalized non-coherent discriminator, shown in Equation 2.2 [6]. Using the code error, the receiver will speed up or slow down the replica code to align it with the maximum correlation peak.

$$e_{code} = \frac{(I_E^2 + Q_E^2) - (I_L^2 + Q_L^2)}{(I_E^2 + Q_E^2) + (I_L^2 + Q_L^2)} \tag{2.2}$$

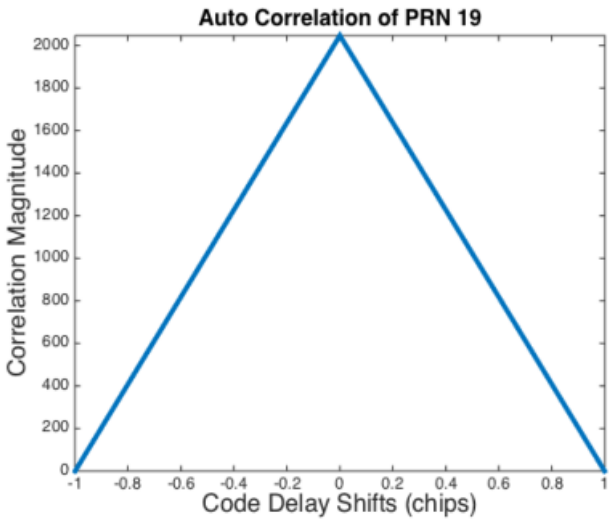

Figure 2.6: Autocorrelation of PRN 19

2.4.3 Obtaining Measurements and positioning

Once the receiver has lock on carrier and code phase and can decode messages, obtaining pseudoranges is relatively simple. GPS pseudoranges rely on measuring the time of flight, or how long it takes a transmitted signal to reach a receiver. The receiver measures the time difference between the known time of transmission from a satellite and time of reception at the receiver. It can be seen from Equation 2.3 that true range r is a function of the speed of light c, time of reception t_u, time of transmission t_s, and time of flight τ [6].

$$r(t) = c[t_u(t) - t_s(t - \tau)] \tag{2.3}$$

The time of transmission is a message available in the GPS data message. The time of arrival is known at the receiver, but it is only an estimate. GPS satellites have very precise clocks capable of holding constant time which does not drift for an extended period. However, even these clocks are prone to drift eventually and must be corrected with the appropriate ephemeris terms. A typical inexpensive receiver clock does not keep time well and is especially prone to bias. Therefore, receiver clock bias must be estimated when performing positioning.

2.5 Errors and Interference

There are many factors which contribute to measurement errors in GPS receivers. The relative geometry of available satellites affects performance, as well as errors induced by the space, control, and user segments. In general, these errors are combined and assumed to be gaussian, as well as independent and identically distributed for each satellite. Combining these errors, the User Equivalent Range Error (UERE) is derived. It can be assumed that satellite UERE is equal to the pseudorange error factor. It represents the magnitude of error on each satellite at any given time. Variance of UERE is determined by the sum of the variance of each individual error source. Total position and timing error is determined by multiplying the geometry factor by the pseudorange measurement error, shown in Equation 2.4 [18].

$$\text{error in GPS Solution} = \text{geometry factor} \times \text{pseudorange error factor} \tag{2.4}$$

Chapter 3

Antenna and Array Theory

An antenna is the device responsible for directing electromagnetic waves into a system. It facilitates the transition from radiation in free space to a cable, waveguide, or other type of transmission line. The antenna is a vital part of a GNSS receiver as it is the first point of contact for a navigation signal. It is especially important because navigation signals are weak in nature, and experience a large amount of free space path loss by the time they pass through the atmosphere and reach the receiver. As the first component in the system, the antenna is responsible for rejecting out of band signals, receiving and amplifying in band signals, and directing them into the receiver. In addition, it is possible to place multiple antennas in an array which may be used for null steering or beamforming. In GNSS, this is referred to as a controlled reception pattern antenna, or CRPA. CRPA's have controllable gain patterns and facilitate spatial and frequency based methods of interference rejection and mitigation. This chapter gives an overview of important characteristics of GNSS antennas and a background on antenna array processing as used in CRPA's.

3.1 GNSS Antenna Analysis

3.1.1 Role of Antennas in a GNSS System

The antenna plays a critical role in a GNSS system. First, it acts as a spatial and frequency filter for the incoming signal. A well designed antenna ensures that no undesirable out-of-band signals are received, and all signals in the desired band are received. It minimizes multipath through gain pattern, choke rings, or other mechanisms [7]. It

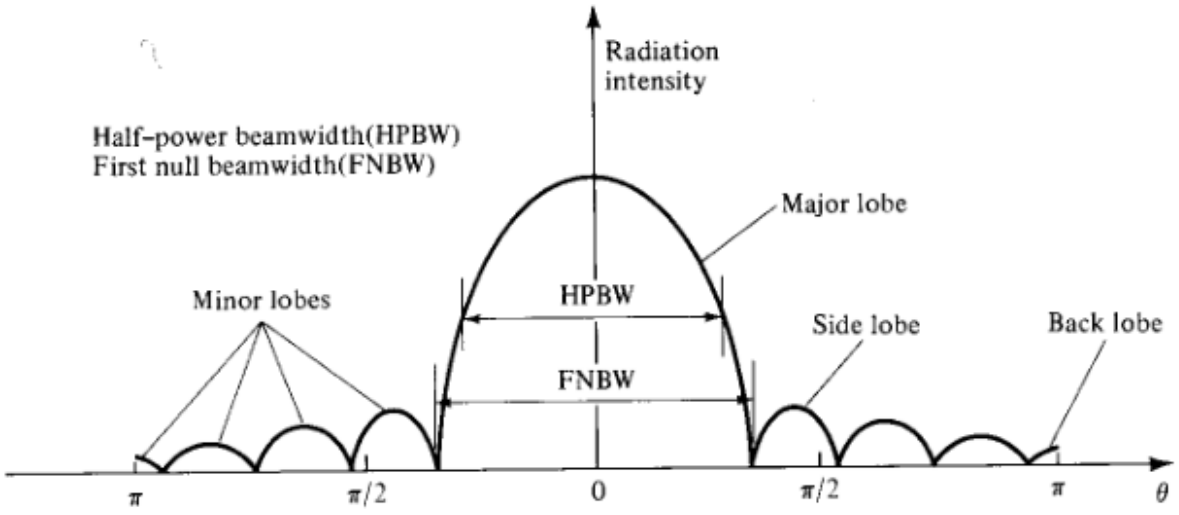

Figure 3.1: Radiation Pattern Parameters

also provides the receiver with the maximum number of available satellites due to a low directional gain and broad radiation pattern [7].

3.1.2 Radiation Pattern Directivity, and Gain

The radiation pattern is one of the fundamentals of antenna analysis. It describes an antenna's relative strength of radiating energy as a function of azimuth and elevation angle. Figure 3.1 shows the various characteristics of a measured radiation pattern [10]. The most important characteristics in the main lobe are the half-power beamwidth and the first null beamwidth. The half-power beamwidth is measured as the width of the main beam 3 dB down from peak radiation intensity. The first null beamwidth is the width of the main beam before reaching a null, or point of minimum radiation intensity. Antenna designers also place high importance on side lobe and back lobe behavior, as these control the amount of radiation in areas other than the main beam.

It is important to note that measurements of an antenna's radiation pattern are measured in the far field. Antennas behave differently depending on how far measurements are recorded from. This is due to the way electromagnetic waves are generated. For an antenna to be in the far field, measurements must be recorded at a sufficient distance for waves to collapse into plane waves. This is dependent on the antenna's size and frequency

of operation. The most common test for the far field condition is $d > 2L^2/\lambda$, where d is distance, L is antenna length, and λ is the wavelength of the frequency of operation.

From the radiation pattern, directivity can be calculated. Directivity measures how directional an antenna is, or how much radiated energy is contained in the direction of maximum radiation compared to the rest of the pattern. Directivity is calculated as shown in Equation 3.1, where $F(\theta, \phi)$ represents the normalized radiation pattern [21].

$$D = \frac{1}{\frac{1}{4\pi}\int_0^{2\pi}\int_0^{\pi}|F(\theta,\phi)|^2 \sin\theta \, \mathrm{d}\theta \, \mathrm{d}\phi} \tag{3.1}$$

Antenna Gain is the ratio of radiation intensity in the direction of maximum radiation to the radiation intensity that would be measured from an isotropic radiator with the same input power [10]. Most commonly measured in dB, this parameter is used more frequently than radiation pattern because it takes into account antenna efficiency, or how much power is radiated compared to input power. Antenna efficiency is shown in Equation 3.2, where ϵ_R is radiation efficiency and P is power. Gain can be related to Directivity as shown in Equation 3.3. Gain can be expressed as a constant or as a function of angle. When expressed as a function of angle, Gain is essentially the radiation pattern measured in units of Gain, or performance relative to an isotropic radiator including losses [21].

$$\epsilon_R = \frac{P_{radiated}}{P_{input}} \tag{3.2}$$

$$G = \epsilon_R D \tag{3.3}$$

For GNSS antennas, the gain pattern must be omnidirectional in the top hemisphere, so signals will be received with equal relative power. This translates to wide half-power and first null beamwidths. It also must have low backlobes for multipath suppression. High efficiency is also imperative, as GNSS signals are already weak and any additional losses in efficiency degrade performance.

3.1.3 Antenna Type

The most common antennas used in GNSS are the microstrip patch and the helix. These antennas can provide the necessary circular polarization and desired radiation pattern of a GNSS antenna.

The microstrip patch antenna is constructed using similar techniques as printed circuit boards. As a result, they are inexpensive to produce, easy to replicate, and can be designed with as simple or complex a shape as necessary. The patch antenna can be fed with a coaxial probe feed, microstrip transmission line, or by proximity coupling [11]. These antennas are suitable for GNSS because they can be made small enough to fit into mobile devices, and are relatively easy and inexpensive to manufacture.

3.1.4 Polarization

Polarization is defined as the orientation of the electric field vector as a function of time. This function is characterized by the polarization ellipse. Polarization can be obtained by calculating the axial ratio of the polarization ellipse, which is shown in Figure 3.2 [22] [10]. Axial ratio is calculated by dividing the radius of the major axis by the radius of the minor axis, shown in Equation 3.4. This quantity is commonly reported in dB, where 0 dB indicates equal magnitude of major axis and minor axis.

$$AR = \frac{OA}{OB} \tag{3.4}$$

A trace of the direction of the electric field as it pertains to polarization can be seen in Figure 3.3 [10]. Axial ratio is commonly defined in terms of dB. Circular polarization results from an axial ratio of about 3dB or less; however, the axial ratio of pure circular polarization is 0 dB (horizontal and vertical sizes of the ellipse are equal). There are two special cases of polarization ellipse. If the axial ratio is infinite, the resulting polarization is linear. If the axial ratio is 0 dB, the resulting polarization is circular.

GNSS signals utilize right-hand Circular Polarization (RHCP). This design choice was made with multipath rejection in mind. Multipath is one of the largest sources of

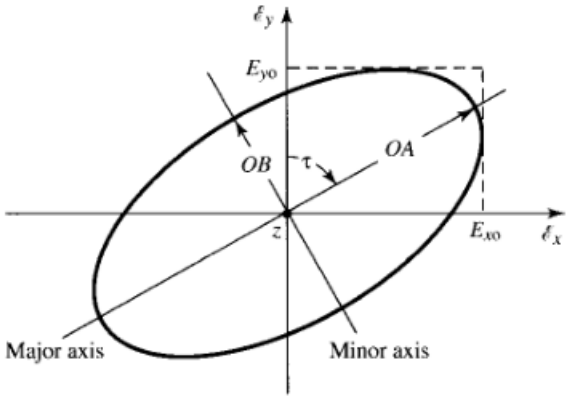

Figure 3.2: Polarization Ellipse

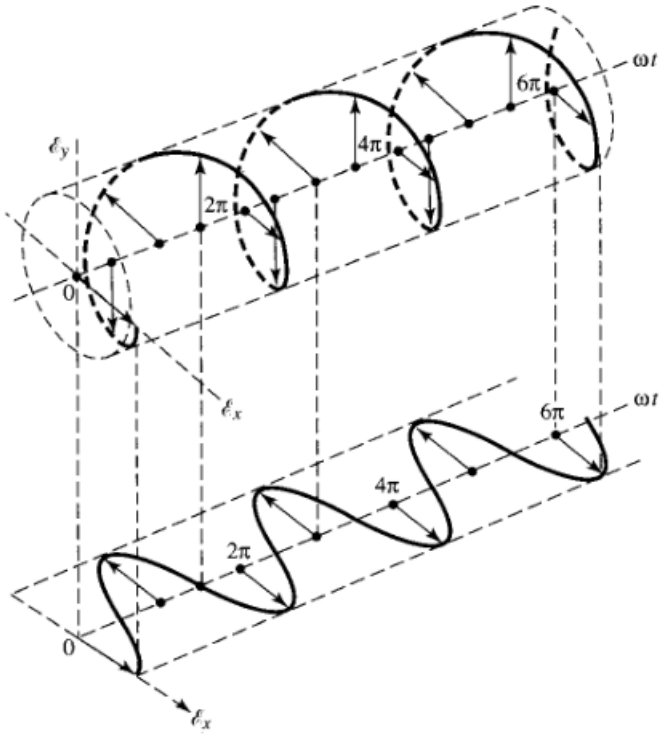

Figure 3.3: Trace of the polarization of an electromagnetic wave

error in GNSS positioning because the typical GNSS receiver uses Time Difference of Arrival to calculate satellite pseudoranges. An error in the received time of arrival of a signal can cause errors of tens of meters or more. The use of RHCP minimizes multipath error because at severe angles of incidence, reflected signals reverse polarity, and therefore have a large component of left-hand Circular Polarization (LHCP) upon reception. Since the polarization mismatch between LHCP and RHCP is 100%, the LHCP component is essentially eliminated. In practice, it is virtually impossible to remove all multipath effects by using an antenna with RHCP, but it is an effective mechanism for reducing multipath effects.

Satellites have constantly changing attitude relative to a user. If they were transmitting linearly polarized waves, the polarization mismatch between satellite and user would also be constantly changing. The use of circular polarization reduces the effects of this polarization fading effect. In addition, linearly polarized waves are more susceptible to the effects of Faraday rotation caused by the Earth's ionosphere. Faraday rotation causes distortion of a wave's polarization and also contributes to polarization fading, which can cause a degradation of reception [23].

3.1.5 Frequency

Antennas are only designed to work over a specified frequency range. In this way, they act as filters, rejecting out of band signals that may cause errors or interference. The frequency range of operation may be narrow, or wide, depending on application. Antennas may even be designed to receive several frequency ranges that are individually quite narrow. This is the case in many multi-band GNSS antennas. Since GNSS signals are very weak, high rejection is desirable in all bands where GNSS signals do not reside. Therefore, if an antenna is designed to receive multiple GNSS signals that are separated from one another in frequency, it may have several narrow bands of operation that are separated from each other.

3.1.6 Bandwidth

The Bandwidth of an antenna describes its range of usable frequencies given specified performance parameters. It is commonly expressed as a percentage, ratio, or range of frequencies [11, 10]. GNSS antennas may have bandwidth as low as 2 MHz to capture most of the power of the L1 signal. However, if the system will be used for receiving multiple bands or P/Y code, it must have higher bandwidth.

High bandwidth can be a desirable characteristic for receiving a wide-bandwidth signal or receiving multiple signals in different bands. However, it also creates vulnerabilities to interference. If an antenna is designed to reject signals outside a narrow bandwidth, only interference in that band can have a significant impact on reception of the desired signal. Conversely, if an antenna has a high bandwidth, there is a much larger range of frequencies where interference can have a negative effect. An antenna is the first line of defense against interference and can be a highly effective filter for out-of-band interference.

3.1.7 Phase Center Stability

An antenna's phase center is the effective point of radiation or reception. However, phase center is not a geometric point, but rather a function. Phase center stability is how much an antenna's phase center drifts. This is largely a function of elevation angle, and is an important parameter to consider in GNSS systems, especially for precision applications. Phase center stability is particularly important when the precise geometry of the array is used in signal processing tasks such as direction finding or beamforming. In such cases, the phase response of each element of the array is analyzed in three dimensions and compiled into an array manifold, which is assumed to be constant. If the array geometry changes slightly due to undesired phase center shifting, the array manifold is no longer accurate and errors may be introduced.

3.1.8 Cross-Polarization

Cross polarization is defined as being the polarization ellipse orthogonal to a reference polarization ellipse [22]. In other words, if the desired polarization if RHCP, then the cross polarization is LHCP. It is desired for GNSS systems that the cross polarization be minimized. If a GNSS antenna has high cross polarization, it is more susceptible to multipath errors. This is because reflected signals such as multipath are reversed in polarization after reflection. This was one of the primary motivators for making the GPS signal circularly polarized.

3.2 Array Theory

An antenna array is a group of antenna elements placed closely together with a common receiver. Since the elements are located at different points in space, there is a phase difference of the incoming signal for each element. The signals from each element constructively or destructively interfere when summed, depending on if they are in phase or out of phase. Each element is controlled by changing its amplitude and phase. This is how the gain pattern is adjusted and beams or nulls are formed.

Antenna arrays are used in most forms of anti-jamming and anti-spoofing. They allow for dynamic adjustment of gain pattern and the ability to point nulls or beams in space. Where one GNSS antenna has a mostly omnidirectional gain pattern in the upper hemisphere, antenna arrays have widely variable gain patterns as a function of several parameters [10]. The 5 parameters under the most direct control of the antenna designer are:

- Geometry (how the elements are configured in space)
- Relative displacement between elements
- Excitation amplitude of individual elements
- Excitation phase of individual elements
- Gain pattern of individual elements

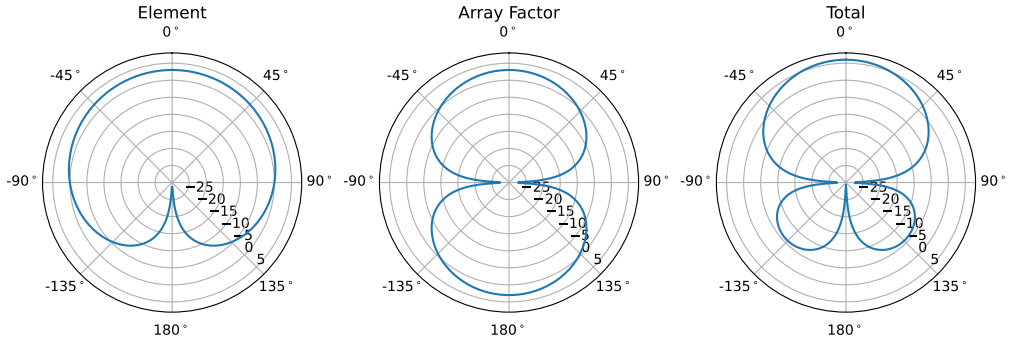

Figure 3.4: Pattern multiplication of a cardioid element pattern with a uniformly excited two element array spaced at $d = \lambda/2$

3.2.1 Array Factor

The array factor is the contribution that element geometry and excitation brings to the overall pattern. It is a way to represent the response of the array as a function. The array factor is a linear combination of the contribution of each element in the array. The parameters that affect array factor are array geometry and the amplitude and phase excitation of each element. Array geometry refers to how the elements are placed relative to each other in space. This is multiplied with the element pattern to achieve the total pattern. To get the total radiated pattern of an antenna array, the element pattern and array factor are multiplied. This is shown in Figure 3.4.

$$total\ pattern = element\ pattern \times array\ factor \tag{3.5}$$

GNSS antennas have an omnidirectional pattern in the top hemisphere, and very low backlobes. Low backlobes are desirable to prevent multipath reflections from entering the receiver. Since GNSS antennas can be considered omnidirectional in the top hemisphere, array analysis for GNSS arrays can be conducted solely with the array factor.

It is convenient to define a generic method of evaluating array factor for an N-element array of arbitrary geomtery as seen in Figure 3.5 [21]. The position of the n_{th} array element is described by Equation 3.6:

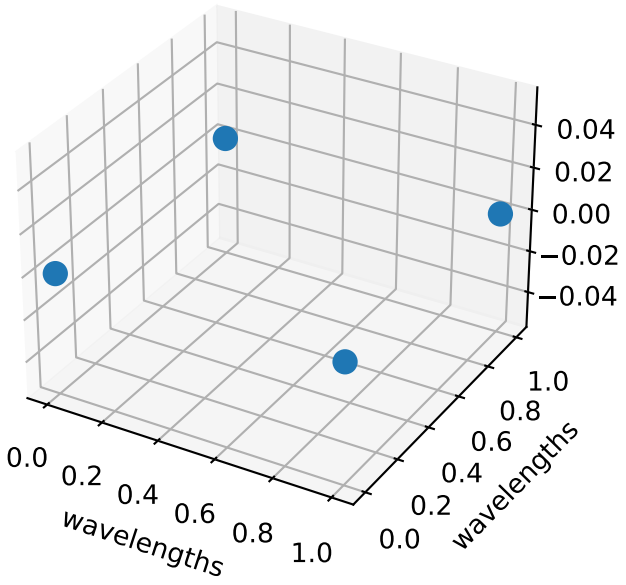

Figure 3.5: Planar Array Geometry

$$d_n = \begin{bmatrix} x_n & y_n & z_n \end{bmatrix} \tag{3.6}$$

and the set of positions of each element is an $N \times 3$ matrix D, where

$$D = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix} \tag{3.7}$$

Antenna arrays leverage spatial phase between each element to achieve constructive and destructive interference. The wave vector k is used to define the magnitude of spatial phase difference of an incoming signal along x, y, and z axes as shown in Equation 3.8.

$$k(\theta, \phi) = \begin{bmatrix} k_x, & k_y & k_z \end{bmatrix} = \frac{2\pi}{\lambda} \begin{bmatrix} cos(\theta)cos(\phi) & sin(\theta)cos(\phi) & sin(\phi) \end{bmatrix} \tag{3.8}$$

The array factor can be written as:

$$AF = \sum_{n=1}^{N} w_n e^{-jk \cdot d_n} \tag{3.9}$$

$$W = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix} \tag{3.10}$$

where W is the weight vector comprised of complex weights for each element. The array output Y can be found via 3.11:

$$Y = W^T X \tag{3.11}$$

where X is the vector of outputs from the antenna elements.

3.2.2 Array Geometry

Array geometry refers to how an antenna elements are configured in space. Linear uniformly spaced arrays are the simplest configuration of an array geometry. However, they can only be scanned in one plane. Since satellites and interference sources are distributed in three-dimensional space, this is not ideal for a CRPA. Due to their ability to point beams and nulls in any direction in 3 dimensional space, planar arrays are the design of choice when selecting a geometry for a CRPA. A visualization of planar array geometry can be seen in 3.5.

3.3 Grating Lobes

Antenna arrays are generally kept beneath a half wavelength apart to avoid the phenomenon of grating lobes. Grating lobes occur because the maximum discernible phase between two antenna elements is half a wavelength. This closely correlates to the Nyquist sampling rule that states a signal must be sampled at least twice the rate of the highest frequency present to avoid aliasing. Grating lobes are, in effect, spatial aliasing and may lead to undesirable effects.

3.4 Non-uniform Arrays

A non-uniformly spaced array is an array with unequal spacing between elements. A non-uniformly spaced array offers the antenna designer another parameter to adjust to achieve the desired characteristics like radiation pattern. The goal may be increased directivity, or in the case of power minimization, increased null depth in the direction of an interference source. It is not often practical to design a non-uniformly spaced array because the environment or requirements may change, and the geometry of the array becomes less effective than the uniform case. This is why arrays are typically uniform; if the needs of the situation change, a non-uniform array could provide worse performance than uniform. The thinned array is a way to combat this undesirable behavior of a suboptimal non-uniform geometry. In a thinned array, an array consists of many more antenna elements than necessary, and only a subset of the array is active at any given time. This provides the benefit of being able to adapt array geometry in real time to adapt to current requirements or signal environment. However, this takes up more space and can require complex feed networks or switch networks to activate only the desired elements.

Another method of changing the effective geometry of a non-uniformly spaced array is via phase center displacement. This phenomenon leverages an electromagnetic property of the stacked patch antenna element and enables variable array geometry [17]. This method is further discussed in Section 3.5.

3.5 Phase Center Displacement

An antenna's phase center, as previously defined in 3.1, is its effective point of radiation. Usually, in antenna arrays, phase center displacement occurs when elements are weighted unequally in magnitude. This is generally considered a nuisance parameter, and its effect mitigated via array manifold calibration or other means [8]. However, recent research has been focused on intentional phase center displacement of individual elements [15, 24, 17]. By intentionally displacing element phase centers, a virtual array with variable perceived

element positions is created. The simplified far-field equations from [17] for a stacked patch in the TM_{11} and TM_{21} modes are shown in Equations 3.12 - 3.13, where (r_{dpc}, θ_0, z_0) is the location of the displaced phase center in polar coordinates. J is the Bessel function of the first kind, and A_{21} is the excitation ratio. This controls the direction and magnitude of phase center displacement.

$$E_\theta = -j\frac{e^{-jk_0r}}{r}e^{jk_0r_{dpc}\sin\theta\cos(\phi-\phi_0)}e^{jk_0z_0\cos\theta}\{[J_0(u_1)-J_1(u_1)]\cos\phi + jA_{21}[J_1(u_2)-J_3(u_2)]\cos2\phi\} \tag{3.12}$$

$$E_\phi = -j\frac{e^{-jk_0r}}{r}e^{jk_0r_{dpc}\sin\theta\cos(\phi-\phi_0)}e^{jk_0z_0\cos\theta}\{[J_0(u_1)+J_1(u_1)]\sin\phi+jA_{21}[J_1(u_2)-J_3(u_2)]\sin2\phi\}\cos\theta \tag{3.13}$$

The fundamental mode of the microstrip patch antenna is the TM_{10} mode. In this mode, and higher order modes, the phase center of the antenna is aligned with its geometric center. However, when two patches of different modes are co-located, the phase center can be displaced based on the excitation ratio of the two patches [16]. This in turn allows the creation of antenna arrays containing variable phase centers. This gives the effect of having variable element positions.

In this work, a virtual array is simulated in which phase centers of individual elements may be displaced $.25\lambda$ toward or away from the center of the array. Different array configurations are possible, but only one configuration is analyzed for simplicity.

3.6 Optimization Using Genetic Algorithm

3.6.1 Overview

The challenge of selecting an optimal array geometry is one of optimization. As the number of variables in an optimization problem increases, the complexity dramatically increases. Gradient descent methods don't typically work well with non-uniform array spacing or other electromagnetic field problems because of the high degree of nonlinearity

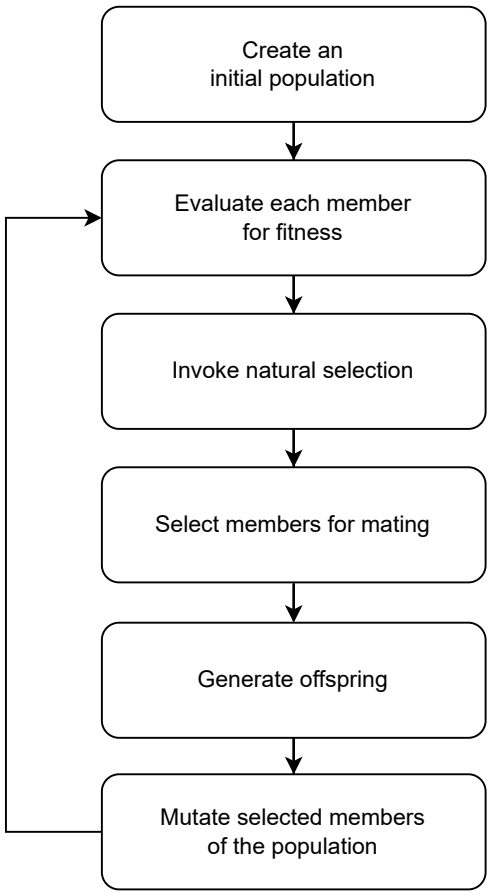

Figure 3.6: Flow Chart of a Simple Genetic Algorithm

[25]. These challenges result in a large, multidimensional search space with many local minima. Therefore, stochastic and statistical methods must be used.

Genetic algorithms (GA) are a class of stochastic based optimization algorithms. They work well in highly nonlinear environments and search spaces with large dimensions [25]. A GA replicates the Darwinian idea of natural selection in genetics. "Survival of the fittest" yields a population that has adapted to its environment by breeding, mutation, and natural selection. The process of a simple genetic algorithm is shown in Figure 3.6.

3.6.2 Initial Population

In a GA, a gene is a simply a variable, or parameter. A chromosome is a collection of genes, all parameters being optimized, and the input into a cost function. A population is a collection of chromosomes. This is shown in Equation 3.14 [25].

$$population = \begin{bmatrix} chrom_1 \\ chrom_2 \\ \vdots \\ chrom_N \end{bmatrix} = \begin{bmatrix} gene_{11} & gene_{12} & \cdots & gene_{1M} \\ gene_{21} & gene_{22} & & \\ \vdots & & \ddots & \vdots \\ gene_{N1} & & \cdots & gene_{NM} \end{bmatrix} \tag{3.14}$$

For this work, each chromosome contains element offsets and weights, shown in Equation 3.15. Element offsets are specified in terms of wavelengths in the interval $[-.25, .25]$. Element weights are complex in the interval $[-1 + e^{-j\pi}, 1 + e^{j\pi}]$. Since the reference element is held constant, there are only 3 weight genes corresponding to elements 2 through 4. The resulting population is an $N \times 7$ matrix, where N represents population size. The initial population is uniformly distributed on the interval of each parameter.

$$population = \begin{bmatrix} offset_{11} & \cdots & offset_{14} & weight_{12} & \cdots & weight_{14} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ offset_{N1} & \cdots & offset_{N4} & weight_{N2} & \cdots & weight_{N4} \end{bmatrix} \tag{3.15}$$

3.6.3 Evaluating Fitness

The next step is to evaluate chromosomes, or population members, for fitness. This is done by passing the genes of each population member into a cost function. For this work, the cost function is defined as the received power of the system given the incoming noisy signal, element offsets, and element weights. This is shown for a cost function f in Equation 3.16.

$$f\left\{ \begin{bmatrix} chrom_1 \\ chrom_2 \\ \vdots \\ chrom_N \end{bmatrix} \right\} = \begin{bmatrix} cost_1 \\ cost_2 \\ \vdots \\ cost_N \end{bmatrix} = \begin{bmatrix} P_{r_1} \\ P_{r_2} \\ \vdots \\ P_{r_N} \end{bmatrix} \tag{3.16}$$

3.6.4 Natural Selection

The next step is to invoke natural selection. This means sorting population members by cost, and only allowing a set number of the population to survive. This work uses a survival rate of 50%, so the result is a new population half the size. Another common method of natural selection is thresholding, where members of the population with a cost below a set threshold are allowed to survive. Thresholding works well when the desired cost is known a priori. Since there is no a priori knowledge of the jamming environment, this work uses survival rate instead of thresholding.

3.6.5 Mate Selection

Next, to replenish the population to full size, members of the population are selected to breed new members. This work uses a common mate selection scheme known as roulette wheen selection. Members are selected for breeding based on their rank in terms of cost. This results in members with higher fitness being selected for breeding more frequently, as shown in Figure 3.7

3.6.6 Offspring Generation

Traits are inherited from one parent or another in a process called crossover. Many variations of crossover are available. This work combines variables of each parent in a random weighting scheme. Each variable of the new offspring is a weighted linear combination of the mother and father. This is shown in Equation 3.17. Here, β is a uniformly distributed 1×7 random vector on [0,1]. The result is a new population member with blended genes of the mother and father.

$$\text{offspring} = \beta\text{mother} + (\beta - 1)\text{father} \tag{3.17}$$

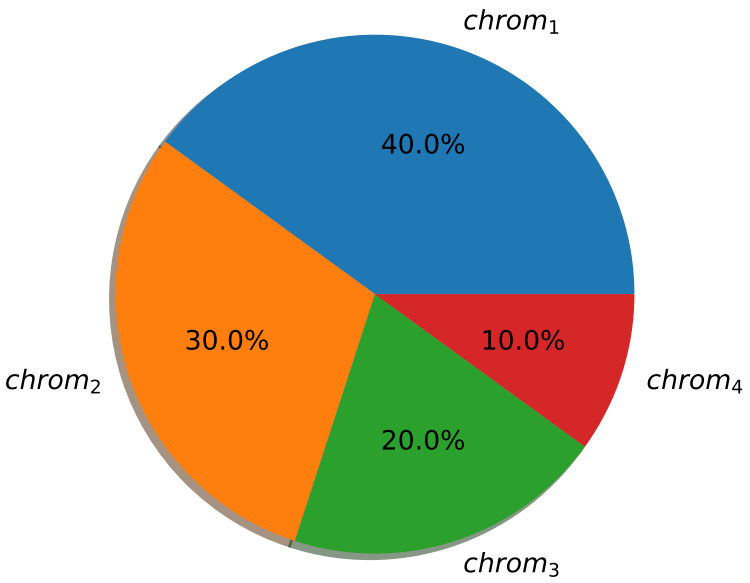

Figure 3.7: Roulette Wheel Selection with 4 Parents

3.6.7 Mutation

After breeding, the population has been replenished. However, the children members will only have blended genes of the parents. If this were the last step, it would be impossible for the search space to be fully explored because genes only fall inside the interval of the parents. In order to keep the population exploring different parts of the search space, mutations are introduced. In general, the GA designer can mutate as many, or as few, members and genes as desired. A high mutation rate implies the search space is more thoroughly explored, but the solution becomes less stable because fewer population members remain centered near the current best solution. More mutations do not always mean better performance or faster convergence. Conversely, a low mutation rate can result in a stale population that doesn't deviate much from the parents. Random genes of random children are mutated at a specified rate. Each gene is randomly perturbed with a random value on its interval as shown in Equation 3.18. This is accomplished by creating a mutation mask with the positions corresponding to parents set equal to 0, and

the positions corresponding to children being uniformly distributed ones and zeros at the specified mutation rate. An example of a mutation mask for a population of 4, with 6 genes, and a mutation rate of .5 is shown in Equation 3.19. This could result in a gene out of bounds of the original interval, so the resulting population is clipped to ensure all genes stay within original bounds.

$$new\ population = population + mutation \times mutation\ mask \tag{3.18}$$

$$mutation\ mask = \begin{bmatrix} m_{11} & m_{12} & \cdots & m_{1M} \\ m_{21} & m_{22} & \cdots & m_{2M} \\ \vdots & & \ddots & \vdots \\ m_{N1} & & \cdots & m_{NM} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix} \tag{3.19}$$

3.6.8 Conclusion

Once mutations have been applied to the population, the iteration is complete, and the resulting population is reevaluated for fitness. There are several options for determining when to stop the GA. Among them are a set number of iterations, specified run time, cost threshold, or if the best solution has not changed in several iterations. Upon completion, the fittest member is selected as the final solution.

Due to the search space having many local minima, it is easy for the GA to find a local 'optimal' solution. However, mutations (uniform perturbations) of offspring keep the population exploring other areas of the search space. A GA is guaranteed to find a solution. However, it will rarely coincide with the true optimal solution, unless initialized with a very large population and allowed to run indefinitely. For this work, a population size and an iteration number of 50 are used. This provides a satisfactory compromise of convergence and computational complexity.

Chapter 4

Anti-Jamming

As mentioned in chapter 1, GNSS signals are susceptible to numerous sources of interference, and because GNSS signals are very weak, they are incredibly sensitive to them. Whether intentional or unintentional, interference sources must be sufficiently mitigated to ensure reliable navigation using GNSS. The CRPA is the most effective method of countering interfering signals as it can act as a spatial and frequency filter. The process of filtering interference out of navigation signals is known as Anti-Jamming. There are many methods of filtering interference, of which several will be discussed in detail in this chapter. This chapter broadly discusses Adaptive Anti-Jam techniques, including spatial and temporal methods and several algorithms implemented to meet these ends. The least mean square algorithm is used most often in this thesis, so it is explored more in depth.

4.1 Power Minimization

Anti-jamming relies on the principle of power minimization. GNSS signals are received beneath the noise floor. This is used as an advantage in anti-jamming because any signals received above the noise floor are assumed to be interference. This particularly useful because it doesn't require any a priori knowledge of interference to be effective. It also prevents the need to differentiate between navigation signals and interference signals. During power minimization, deterministic or adaptive algorithms are applied to signal inputs from the antenna array to drive the received power of the output signal to zero, or as close to the thermal noise floor as possible. This is shown in Figure 4.1.

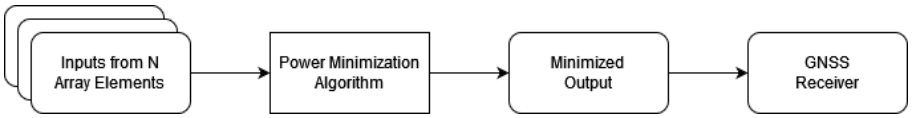

Figure 4.1: Power Minimization Block Diagram

The inputs from an antenna array have spatial diversity. This makes it possible for an AJ filter to control the array factor and point nulls in directions where interference is located. The AJ power minimization algorithm controls the amplitude and phase of each spatially separated input to achieve these ends. This is known as spatial adaptive processing (SAP). However, this type of processing can be extended by applying tap delays to input signals. By taking several consecutive snapshots of inputs from the antenna array, the inputs are separated spatially as well as temporally. This effectively turns the power minimization algorithm into a time domain finite impulse response (FIR) filter of order M, where M is the number of tap delays. This added benefit comes in addition to the spatial processing component of a SAP power minimization algorithm. The extended time and space domain filter is considered to be a space-time adaptive (STAP) filter. This added benefit allows narrowband interference sources to be filtered out while preserving the rest of the spectrum. However, this type of processing is not suitable for wideband interference as the filter is unable to isolate and minimize a wideband interference source. STAP also comes with the added expense of extra computing. If an N channel array is employed in a SAP configuration, the power minimization algorithm has N inputs. When employed in a STAP configuration, the number of inputs increases to $N \times M$, where M is the number of tap delays. This increases the amount of processing power required by a factor of M. Figure 4.2 shows that the STAP model is essentially the SAP model with an FIR filter component on each spatial channel [26].

When discussing the performance characteristics of AJ systems, the term "degrees of freedom" is often cited. Degrees of freedom is loosely defined as the number of signals that can be nulled and enable reliable operation of the GNSS receiver. In general, SAP systems yield $N - 1$ degrees of freedom. For STAP systems, this number is increased to $NM - 1$, where N is the number of antenna array elements and M is the number

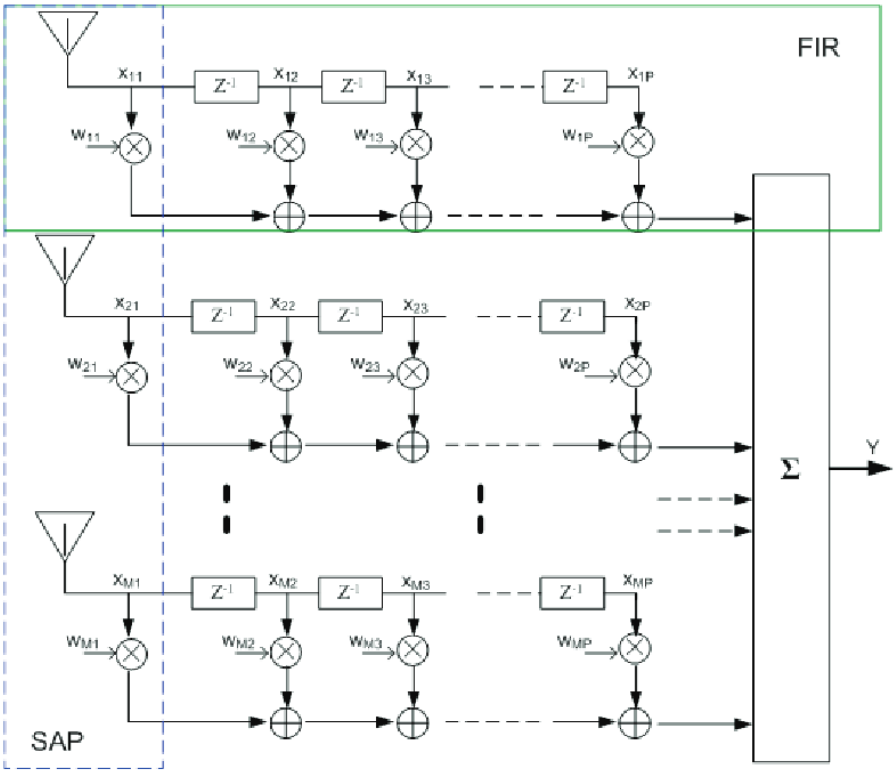

Figure 4.2: STAP Model

of tap delays and the order of the FIR filter [27]. However, signal content and power can affect these numbers. For instance, sufficiently wideband signals render the FIR component of STAP systems ineffective. If a signal is powerful enough, it may consume more degrees of freedom than one. Degrees of freedom is an important metric in evaluating AJ performance but is affected by signal characteristics such as modulation scheme, power, and bandwidth.

4.2 Deterministic Anti-Jamming

Deterministic algorithms for interference mitigation take some a priori knowledge and use that knowledge to produce an array weight vector. The most effective deterministic AJ algorithm is the Minimum Variance Distortionless Response (MVDR) algorithm. It requires no a priori information and relies entirely on observations of the incoming signal. This requires sufficient estimation of the signal covariance matrix, which can be computationally expensive. Thus, the algorithm performance is limited to the accuracy of the

estimated signal covariance matrix. Additionally, the signal covariance matrix must be inverted any time a new weight vector is computed. This computational cost adds up quickly and can be very computationally expensive if new weight vectors are generated often. The algorithm is shown in Equation 4.1 [5]:

$$W_{opt} = \frac{1}{\delta^T R_{xx}^{-1} \delta} R_{xx}^{-1} \tag{4.1}$$

where δ is the 1 x N constraint vector that keeps the weight of the reference element at unity, and N is the number of antenna elements, shown in Figure 4.2.

$$\delta = \begin{bmatrix} 1 & 0 & ... & 0 \end{bmatrix} \tag{4.2}$$

Equation 4.3 shows how to estimate the signal covariance matrix R_{xx} [6]:

$$R_{xx} = \frac{1}{M} \sum_{m=1}^{M} X_m X_m^H \tag{4.3}$$

where X denotes the signal state and X^H denotes the hermitian transpose of X, also known as the conjugate transpose. X, the signal state, is an N x 1 vector of array outputs at a particular time. It can be thought of as a signal snapshot. Since the power minimization algorithm requires an inversion of an N x N matrix each time new weights are calculated, this algorithm can be very computationally expensive.

MVDR operates on the assumption that a good estimate of the signal covariance matrix can be obtained. W_{opt} can only be achieved under the condition that R_{xx} is a perfect estimate. The signal covariance estimate R_{xx} is integrated over m snapshots of the input and is inverted upon completion. This presents an engineering trade-off. If the signal covariance matrix is integrated over a smaller number of snapshots, a weight vector W is available at a higher update rate; however, the estimate may be degraded. Conversely, if the signal covariance matrix is integrated over many snapshots, W approaches W_{opt}, the optimal weight vector representing optimal power minimization. However, in a dynamic environment with mobile interference sources, dynamic waveforms, or a mobile

platform, a long integration period will lead to a stale W. By the time a new W is calculated, the signal environment or platform attitude could change, rendering W obsolete. As with all engineering trade-offs, the system designer must attempt to select an appropriate compromise of accuracy and speed for the particular application.

MVDR can be extended to be a STAP filter as seen in Figure 4.2. However, this can make matrix inversion calculation prohibitively expensive from a computation perspective. R_{xx} is an $N \times N$ matrix in the SAP configuration, where N is the number of antenna array elements. In a STAP configuration, this is increased to $NM \times NM$, where M is the number of tap delays. Thus, increasing the number of tap delays in the signal snapshot increases the number of array elements in R_{xx} by a factor of M^2.

4.3 Adaptive Algorithms

4.3.1 Least Mean Square

Least mean square is a recursive, error driven algorithm detailed in [9]. It is an adaptive linear combiner, with the output of each iteration being the linear combination of the input multiplied by a set of complex weights. It is also memoryless, meaning it is only dependent on the current time. LMS is an excellent choice for many signal processing applications including interference mitigation due to its simplicity. All that is required is the input vector X_k and the desired output d_k at each time. The input vector X_k represents the input at time k:

$$X_k = \begin{bmatrix} X_{0k} & X_{1k} & ... & X_{n-1k} \end{bmatrix} \tag{4.4}$$

and the weight vector represents the set of weights applied to the input at time k.

$$W_k = \begin{bmatrix} W_{0k} & W_{1k} & ...W_{n-1k} \end{bmatrix} \tag{4.5}$$

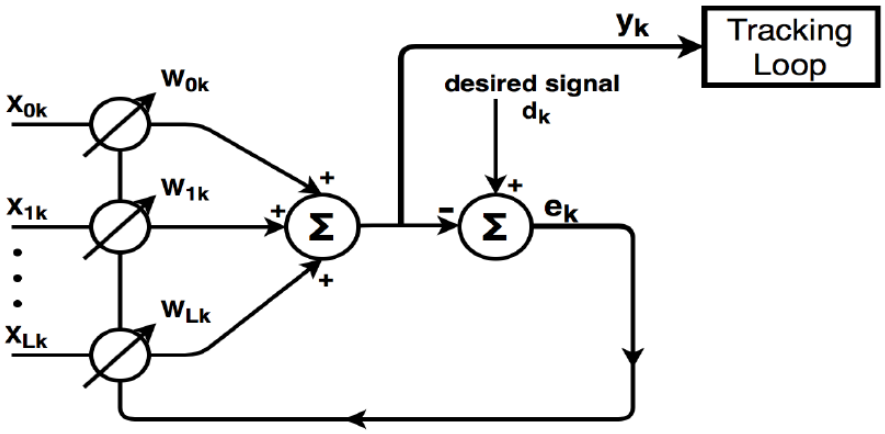

Figure 4.3: Block Diagram of Least Mean Square

The error ϵ_k and output y_k is found at time k,

$$\epsilon_k = d_k - y_k \tag{4.6}$$

$$y_k = XW^T \tag{4.7}$$

and finally, the weight vector W_{k+1} is updated:

$$W_{k+1} = W_k + \mu\epsilon_k^H X_k \tag{4.8}$$

where μ is the gain constant that regulates stability and speed of convergence and ϵ_k^H is the hermitian transpose of the error ϵ at time k. The selection of an appropriate μ is of high importance, as it affects the time required to minimize $E[\epsilon^2]$.

The following is an example of least mean square Implementation of noise cancellation. In this example shown in Figure 4.4, $\hat{X}_k$ is a noisy measurement of the desired signal d. The desired signal d is known. It's clear that the choice of the gain constant is very important as it regulates convergence speed and stability. In the first case, the selected μ is too small. Least mean square does not converge, and the output signal looks much like the noisy input. In the second case, the selected μ is too large, and causes undesirable oscillations. In this case, least mean square is close to being unstable. The third case shows a good choice of μ. It converges quickly while also maintaining stability.

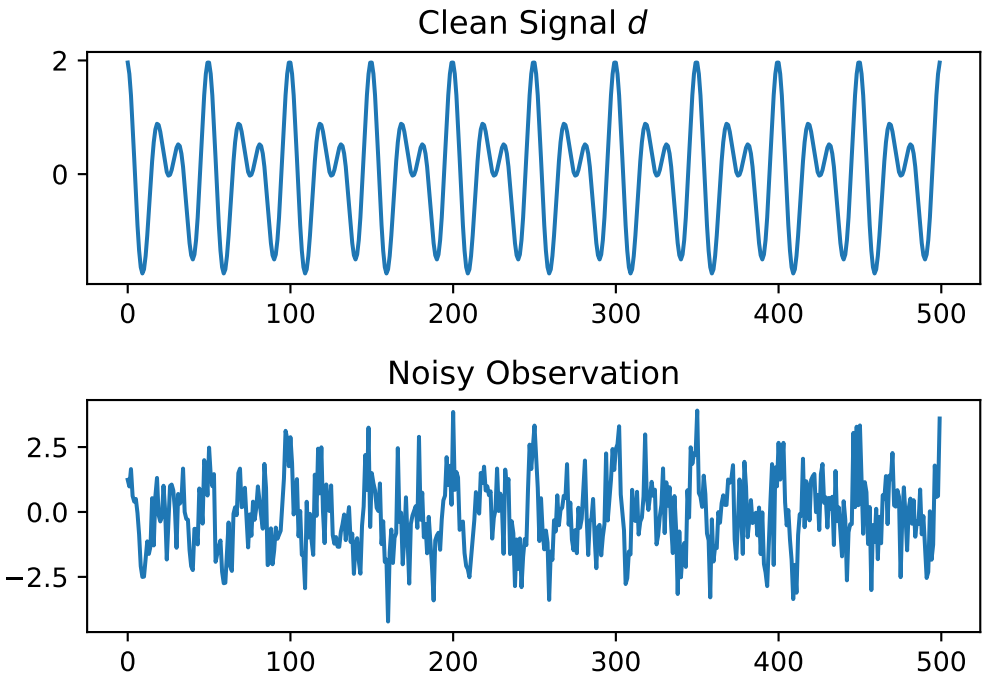

Figure 4.4: LMS Noise Cancellation Example

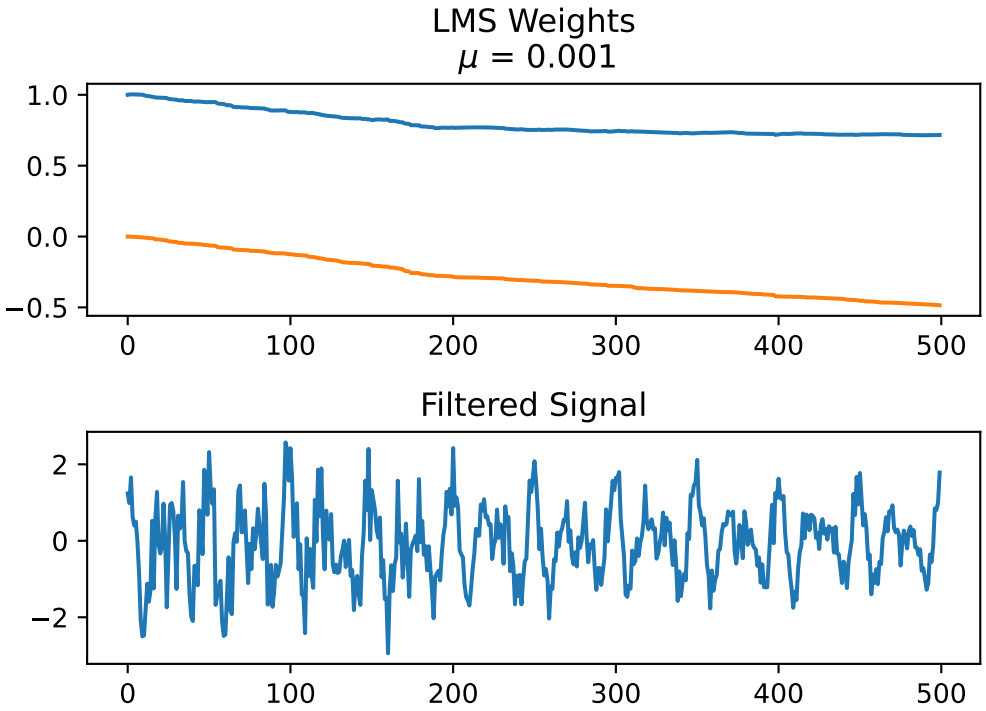

Figure 4.5: LMS Noise Cancellation Output, $\mu = .001$

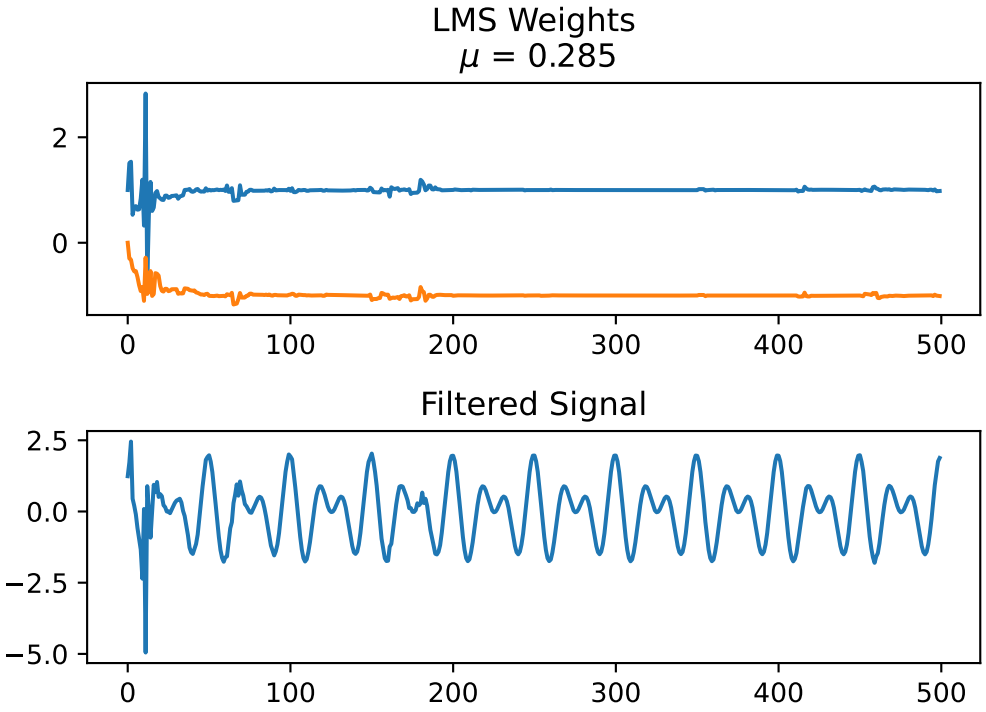

Figure 4.6: LMS Noise Cancellation Output, $\mu = .285$

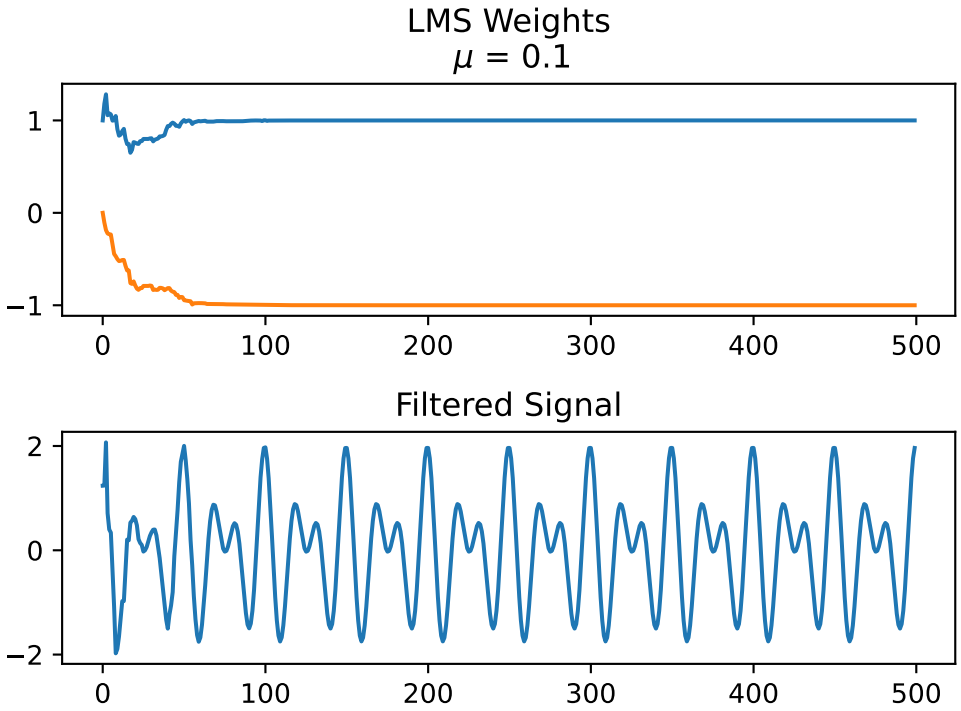

Figure 4.7: LMS Noise Cancellation Output, $\mu = .1$

It is possible to calculate the maximum stable value of μ. This can be seen in Equation 4.9, where tr denotes the trace operation and R_{xx} is the signal covariance matrix. In practice, it is prudent to use a value significantly smaller than μ_{max} as this value can lead to overshoot and undesirable oscillations [9].

$$0 < \mu < \frac{1}{\mathrm{tr}[R_{xx}]} \tag{4.9}$$

The GPS anti-jamming implementation of the least mean square algorithm is similar to the previous simple example, with several nuances. To get the best performance from the LMS algorithm, the spatial and temporal components of adaptive processing must be utilized. The input to the LMS algorithm is now an $NM \times 1$ vector where N is the number of CRPA elements and M is the number of tap delays. Similarly, W takes the same form.

$$X_k = \begin{bmatrix} X_{00} & X_{10} & ... & X_{NM} \end{bmatrix} W_k = \begin{bmatrix} W_{00} & W_{10} & ... & W_{NM} \end{bmatrix} \tag{4.10}$$

Because the goal is power minimization, the desired signal d is 0. Unconstrained, the LMS algorithm will converge to a weight vector of all zeros, as this technically minimizes received power. This introduces the need for a constraint on W. A reference weight must be selected to be held constant at unity to prevent zero-convergence. Figure 4.8 shows a jamming scenario LMS is able to mitigate. The process of convergence can be seen; the array factor begins omnidirectional as the initial weight vector $W_0 = [1\ 0\ 0\ 0]$. As the LMS algorithm approaches convergence, a null is formed in the direction of the interference of 90°.

In Figure 4.9, a scenario with four interference sources at −45°, 30°, 90°, and 165° is shown. It can be seen that LMS has used all available degrees of freedom and is unable to place effective nulls in the required directions. This is an example of the limitations of SAP. At some point, the system runs out of usable degrees of freedom and cannot effectively place nulls on all interference sources.

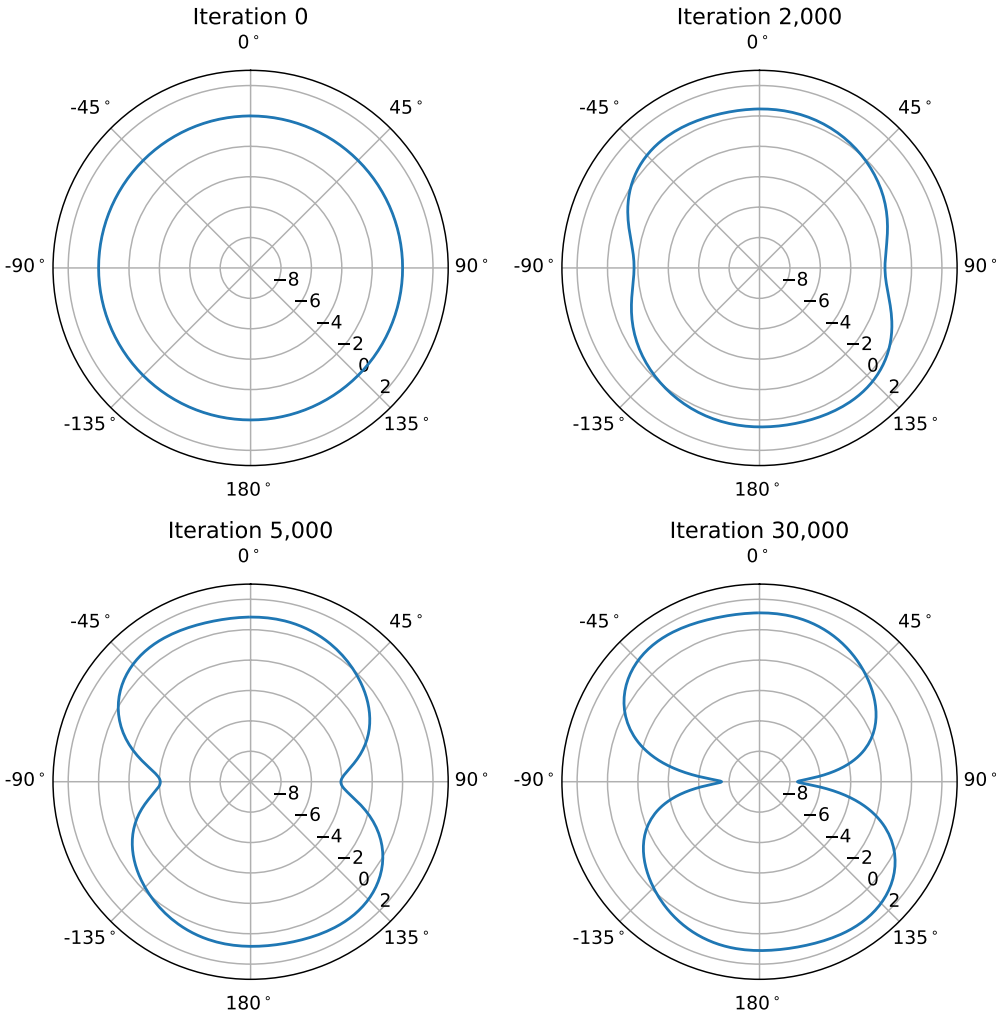

Figure 4.8: Array Factor at various stages of Least Mean Square

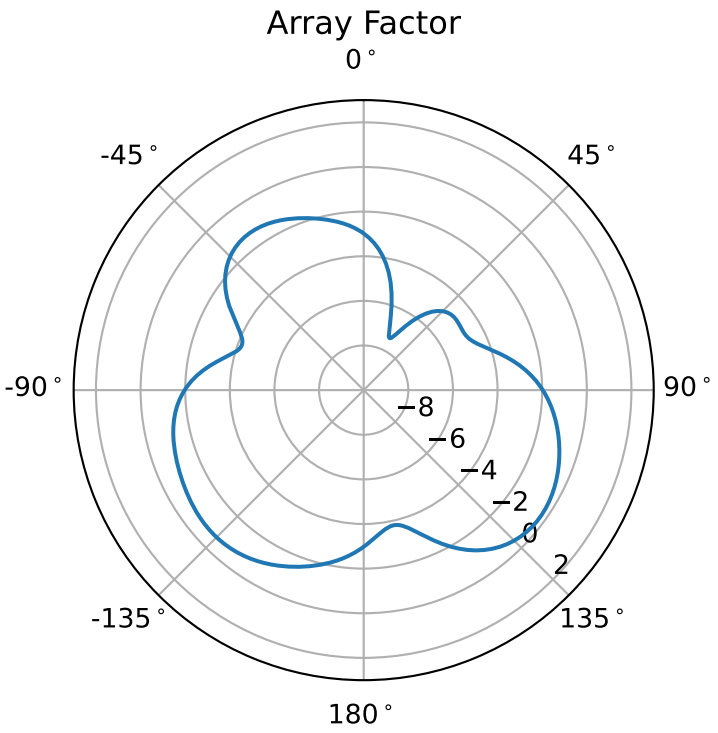

Figure 4.9: Array Factor of LMS Scenario with 4 Jammers

4.3.2 Recursive Least Square

The recursive least squares (RLS) algorithm is similar to LMS in that it is an error driven and recursive. RLS uses a matrix inversion lemma to directly update the inverted signal correlation R_{xx} instead of performing costly matrix inversions. Unlike LMS, RLS is not memoryless, and has a forgetting factor λ. A λ closer to 1 will yield a system with longer memory that is slower to change. The side effect of a system with long memory is suboptimal performance in the presence of a dynamic receiver. The same will be true in a dynamic signal environment. Conversely, a low λ value can lead to excess weight being placed on recent measurements. This can in some cases lead to filter instability. Using recursive Equations 4.11-4.17, an optimal estimate of element weights can be found [28].

$$\text{define } P = R_{xx}^{-1} \tag{4.11}$$

$$P[0] = \delta^{-1} I \tag{4.12}$$

$$W[0] = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T \tag{4.13}$$

$$K[i] = \frac{P[i-1]X[i]}{\lambda + X^H[i]P[i-1]X[i]} \tag{4.14}$$

$$W[i] = W[i-1] + K[i](d[i] - W^H[i-1]X[i]) \tag{4.15}$$

$$P[i] = \frac{1}{\lambda}(P[i-1] - K[i]X^H[i]P[i-1]) \tag{4.16}$$

$$e[i] = d[i] - X^H[i]W[i] \tag{4.17}$$

4.4 Anti-Jam Induced Errors on Carrier and Code Phase

Phased arrays leverage spatial diversity to adjust the overall gain pattern in real time. As seen in previous sections, this comes from placing amplitude weights and phase delays on individual elements. This can result in distortion or displacement of the phase center of the array. For standard positioning, this is not an issue. However, any carrier phase level positioning may be affected, because these solutions rely on finding the integer number of carrier phase cycles between the receiver and satellite. If the phase center has shifted by even a fraction of a wavelength, there could be negative effects on carrier cycle integer estimation.

Chapter 5

Results

5.1 Introduction

In this chapter, several datasets from variations of the GA are introduced and analyzed. A deep dive into several cases is performed to get a more complete picture of system performance. The results section is structured as follows. First, a detailed description of the simulation environment is given. Following are a discussion and analysis of several variations of the genetic algorithm, and the results of monte carlo simulations with these algorithms. A closer look into typical performance is given in Section 5.4. Here, an example is shown, and the system parameters are observed, such as array factor and adapted geometry.

5.2 Simulation Environment

This section goes into detail about the simulation environment used, including which signals are analyzed, how measurements and interference are generated, and how results are computed. The first step is simulating the clean navigation signal. These experiments use the GPS L1 navigation signal simulated at 12.5 complex megasamples per second (CMSps) at 1200 March 1, 2022, in Auburn, AL. At this time, there are 10 satellites in view. To achieve the spatial phase difference for each satellite necessary to simulate CRPA measurements, each satellite is recorded as a separate IQ stream. Each PRN is replicated for each antenna element. The result is 10 channels on each element, or $N \times 10$ total IQ streams, where N is the number of antenna elements. Each PRN on each element

has a unique phase offset, which is calculated using the known satellite geometry derived from the GPS ephemeris message.

To calculate satellite azimuth and elevation angles, satellite Earth centered Earth fixed (ECEF) positions are found from the GPS ephemeris. ECEF is a coordinate system in which the origin is colocated with the center of the Earth. The positive z axis extends from the origin through the geographic North Pole, the positive x axis extends from the origin through the point (0°, 0°), and the positive y axis passes through 90°E longitude in the plane of the equator. Then, satellite positions are rotated into the local receiver north east down (NED) frame. With satellite positions s in the NED frame $s = [s_n, s_e, s_d]$, azimuth and elevation angles can be expressed as:

$$\phi = \arctan\left(\frac{s_e}{s_n}\right) \tag{5.1}$$

$$\theta = \arcsin\left(\frac{-s_d}{\sqrt{s_n^2 + s_e^2 + s_d^2}}\right) \tag{5.2}$$

The spatial phase offset β for the nth element can be calculated as seen in Equation 5.3.

$$\beta = k \cdot d_n \tag{5.3}$$

As seen in 5.4 and 5.5, k is the wave vector and d_n is the nth entry of the element position matrix D.

$$D = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix} \tag{5.4}$$

$$k(\theta, \phi) = \begin{bmatrix} k_x, & k_y & k_z \end{bmatrix} = \frac{2\pi}{\lambda} \begin{bmatrix} cos(\theta)cos(\phi) & sin(\theta)cos(\phi) & sin(\phi) \end{bmatrix} \tag{5.5}$$

After applying the spatial phase offset for each channel on all elements, channels are added together. This represents the composite IQ stream each antenna element would

see in a live sky environment, and the result is one IQ stream per element containing phase coherent IQ from all satellites in view. The next step is setting the noise floor. This is accomplished by adding independent white Gaussian noise to each element with noise power approximately 20 dB higher than satellite signals.

The interference sources are generated using a similar method to the satellite navigation signals. For each interference source, one common IQ stream is generated using filtered white Gaussian noise to achieve the correct bandwidth. This stream is replicated for each element, and the correct spatial phase offset is applied to each stream to achieve a phase coherent interference source across all elements. The only difference between the control array and the virtual array is the adapted geometry. The control array has a standard geometry and spatial phase offsets for each PRN are constant. However, the virtual array has a unique geometry for each scenario, and the spatial phase offsets are slightly different from the control array. Because navigation signals are received beneath the noise floor, they have no effect on the power minimization cost function used in the genetic algorithm. For this reason, satellite signals are not included in the initialization step of finding an optimal geometry. The adapted geometry is solely a function of minimizing the received power cost function, and therefore uses only interference sources to evaluate sample geometries.

The next step is the least mean square space adaptive filter. At this point in the simulation, the control array and the virtual array inputs are comprised of independent thermal noise, phase coherent satellite navigation signals, and phase coherent interference. The only difference in array inputs is the array geometries. The signals for both arrays are processed using the same LMS filter with $\mu = 0.001$. The output of this filter is a single spatially filtered IQ stream for each array. Spatial-only processing is done because interference signals at the center frequency with wide bandwidths are not suitable for STAP methods and in many cases perform worse than SAP methods.

The final step is acquisition comparison. Each array output is used in a parallel acquisition algorithm with a non-coherent integration period of 10 ms. The results are

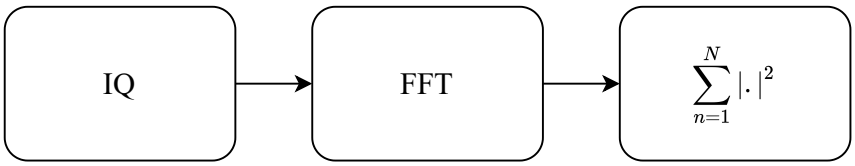

Figure 5.1: Received power algorithm

judged on how many satellites can be acquired from each array output and the null depth at each interference location.

5.3 Experiment Methodology

Genetic algorithms contain stochastic processes that lend themselves to analysis via monte carlo simulation. Since a GA will almost always produce a better solution than the control array geometry, but almost never produce the optimal solution, monte carlo testing is necessary. The monte carlo experiment uses non-unique interference sources uniformly distributed in space. In other words, to preserve the unique noise profile of each interference source, they are seeded identically in every monte carlo run. However, their placement in space is uniformly distributed between runs. This removes geometric effects of some interference source locations performing better than others in a power minimization context. Tests are run with 1-5 interference sources ranging from 20 dB to 55 dB J/S. Since the goal of this work is improved performance without the expense of additional overhead, population size and iteration count are kept within reasonable bounds.

Four variations of the genetic algorithm are analyzed. The first is a continuous encoded GA that takes in no a priori information. Antenna parameters including position offset and weight can take any continuous value within the given range. The cost function used in this GA is received power. Therefore, this algorithm employs power minimization. The method of calculating received power is shown in Figure 5.1

The next variation uses a priori information about present interference sources to aid performance. This GA would be useful in an AJ system with direction finding capability, as it uses known locations of interference sources to aid performance.

In a realistic system, the element position offsets are not continuous variables. Because the magnitude of offset is dictated by excitation ratio of element modes, the precision is dependent on the quality of attenuators used to achieve the desired excitation ratio. To model this behavior, a variation of the GA is analyzed where element phase center offsets are no longer continuous, but calculated in discrete amounts. The element phase center offsets are quantized to the nearest tenth of a wavelength to meet these ends.

The last variation of the GA includes a least-mean-square power minimization in the cost function. In this variation, only the element positions are member variables. The element weights are not included as GA member variables. The element weights are instead calculated by performing least-mean-square power minimization during the cost function step of the algorithm.

5.4 Selected Results

A particular scenario is analyzed to show the effects of the virtual array. In this scenario, 4 interference sources 2 MHz wide are present at $-170°$, $-65°$, $45°$, and $90°$. The adapted geometry selected by the genetic algorithm can be seen in Figure 5.3. The control array is unable to converge to an acceptable solution. This can be seen in the array factor comparison shown in Figure 5.2. However, the virtual array is capable of producing deep, narrow nulls. As a result, the virtual array acquires 4 additional satellites. Shown in Figure 5.4, the virtual array maintains enough satellites to calculate a position fix, while the control array does not.

5.5 Monte Carlo Results

This section shows monte carlo results of all genetic algorithm variants. Tables 5.1, 5.2, and 5.3 show performance averaged across monte carlo runs. The tables show how each cost function performs as a function of the number of active interference sources. Tables

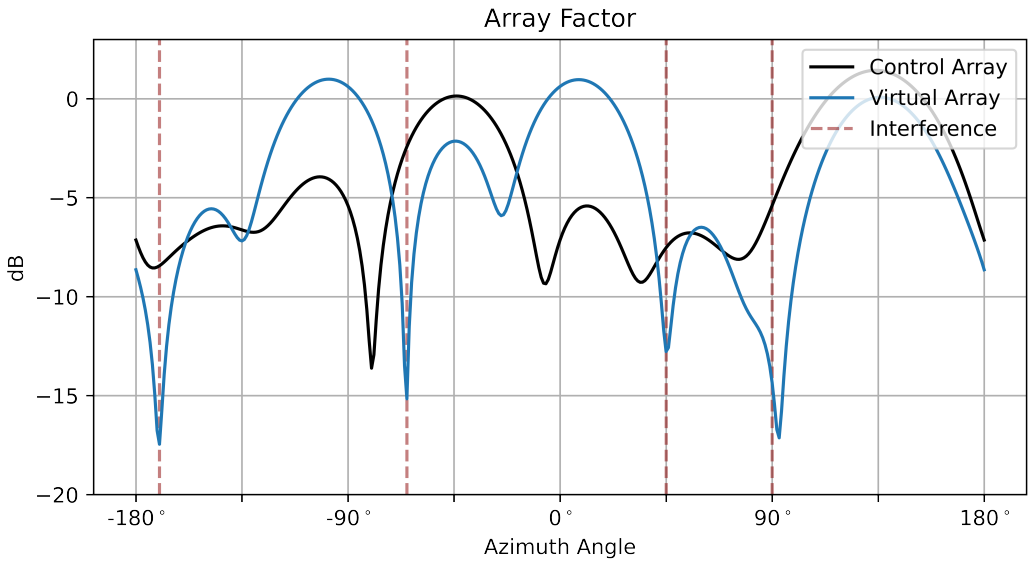

Figure 5.2: Array Factor Comparison

Table 5.1: Received Power Improvement (dB)

	1	2	3	4	5
Power Minimization	-0.004	-1.091	-2.115	-2.956	-2.508
Quantized	-0.004	-1.079	-2.174	-2.257	-2.056
Unequal Weight A Priori	-0.003	-1.070	-2.219	-2.694	-2.514
Equal Weight A Priori	-0.003	-1.008	-2.161	-2.856	-2.460
LMS-in-the-loop	-0.003	-1.138	-2.262	-2.891	-2.487

summarizing monte carlo results are shown in this section. Expanded graphical results for each GA variant can be seen in Appendix A.

The performance of the GA with power minimization cost function can be seen in Figures A.1 and A.2. This configuration is used as a comparison for the remaining variants.

The GA's in previous experiments operate in the continuous domain. Digital signal processing will allow as much precision as needed in the complex weight values. However, it is unlikely that element position offsets offer the same flexibility. The element position offsets should be capable of being described in discrete quantities. This is the purpose of quantization. The element position offsets are quantized to discrete distances in these experiments. These results can be seen in Figures A.3 and A.4.

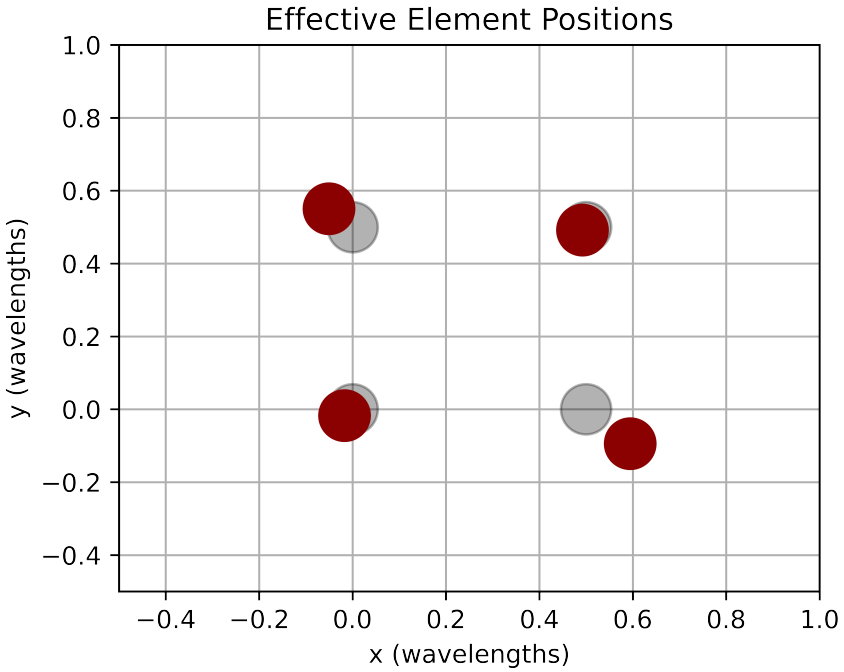

Figure 5.3: Element Position Comparison

Table 5.2: Satellites Acquired Improvement

	1	2	3	4	5
Power Minimization	0.004	0.511	0.889	0.838	0.548
Quantized	0.002	0.482	0.952	0.662	0.532
Unequal Weight A Priori	-0.053	0.489	0.917	0.802	0.490
Equal Weight A Priori	-0.015	0.488	0.908	0.824	0.379
LMS-in-the-loop	-0.028	0.553	0.922	0.872	0.479

In this section, the cost function is adapted to include a priori information. Specifically, the known locations of interference sources is provided to the genetic algorithm to make it more robust and reliable. Two versions of this cost function are analyzed. The first weights received power more heavily at 75%, and interference location 25%. The second weights both factors evenly. The results for the unequally weighted cost function GA can be seen in Figures A.5 and A.6. The results for the equally weighted cost function GA can be seen in Figures A.7 and A.8.

In the final GA variant, the element weights are taken out of the GA selection process; optimal weights for each geometry candidate are calculated during the cost function phase of the algorithm. The results of the power-minimization-in-the-loop GA can bee see

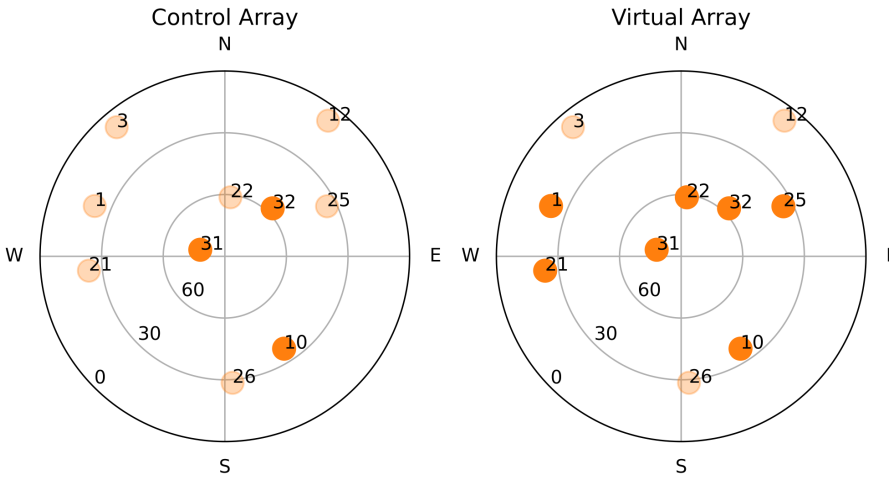

Figure 5.4: Skyplot Comparison

Table 5.3: Null Depth Improvement (dB)

	1	2	3	4	5
Power Minimization	-0.016	-1.795	-2.837	-2.419	-1.741
Quantized	0.009	-1.606	-2.717	-1.682	-1.332
Unequal Weight A Priori	0.101	-1.787	-3.037	-2.399	-1.807
Equal Weight A Priori	0.074	-1.697	-2.880	-2.470	-1.890
LMS-in-the-loop	0.061	-1.837	-2.912	-2.415	-1.747

It can be seen from the monte carlo results that a virtual array initialized with a genetic algorithm can decrease received power and increase degrees of freedom, resulting in more satellites acquired. However, it is also plain that the adapted geometry of the virtual array has a wider standard deviation of satellites acquired. On average, an adapted array geometry will increase performance.

Chapter 6

Conclusion and Future Work

The goal of this work was to validate the concept of using the virtual array in CRPA anti-jamming, and to develop a method of element position initialization. A monte carlo experiment was designed to determine if the adaptive geometry shows a significant improvement from a control geometry. In addition, a simulation environment was developed to test anti-jamming algorithm performance for an arbitrary number of interference sources with an arbitrary array configuration.

In conclusion, it has been shown that the virtual array with adaptive geometry has the potential to increase CRPA anti-jamming performance. In addition, it has been shown that a genetic algorithm is a viable solution to initialize such an array with the appropriate element position offsets. Various cost functions for genetic algorithms were introduced and analyzed, although performance was fairly consistent.

In particular, the results from the quantized cost function are very promising. These data show that even with significant quantization, an adaptive virtual array can outperform a standard array in received power, satellites acquired, and null depth.

6.1 Dynamic Receiver Testing

The GA used to improve CRPA geometry is not designed to run in real time on a dynamic platform. It is designed to initialize a static receiver. The geometry is assumed to remain constant indefinitely. However, receivers are often not static. In the future, work should be devoted to investigating how to transition to a new geometry while the receiver is

experiencing dynamics. A potential solution to this problem would be to set bounds on element weight drift, the error term in the LMS filter, or received power level. Once a value drifts outside a threshold, the receiver runs the GA once again to compute an improved geometry. However, if the receiver is experiencing sufficient dynamics, and the threshold is hit continuously, this would have the same effect of running GA in real time, which is not desirable as it adds overhead.

6.2 Hardware Implementation

The hardware discussed in this work has yet to be successfully implemented. A linear virtual array has been prototyped, but the planar case has yet to be realized [17] As such, certain hardware assumptions are made in this work. Most notably, mutual coupling and phase center instability are ignored. However, these almost certainly will affect performance without sufficient mitigation.

6.3 Direction Finding Integration

If combined with a total AJ/AS system capable of direction finding, this algorithm could potentially become more potent. In the case of the blind GA, it has no a priori knowledge of signal environment. Thus, it uses the best metric available, received power, as a cost function. However, if the receiver has some a priori knowledge of signal, we can modify the cost function to make it more likely of reaching an optimal solution, and less computationally expensive. The new cost function is the array factor at the angle of each jammer. If it is determined that the jammers have unequal power, the cost function can be modified to be a weighted average, taking into account the power level of each.

6.4 Control Array Geometry Analysis

This work only compared the virtual array performance with a standard half wavelength uniformly spaced planar array. Some additional insights may be available by comparing to arrays of different uniform spacing.

6.5 Direction Finding Performance Improvement

In this work, it is shown that virtual arrays can increase CRPA power minimization anti-jam performance. Another use of the CRPA is direction finding; that is, estimating the line of bearing to an interference source. Future work should seek to quantify what performance improvement, if any, is obtained when using virtual arrays for direction finding purposes.

References

[1] T. Hitchens, "Satellite jamming 'normal' by militaries during conflict, not peacetime: State Dept. official," Mar. 2022. [Online]. Available: https://breakingdefense.sites.breakingmedia.com/2022/03/satellite-jamming-normal-by-militaries-during-conflict-not-peacetime-state-dept-official/

[2] D. Goward, "What happened to GPS in Denver?" Sep. 2022, section: GNSS. [Online]. Available: https://www.gpsworld.com/what-happened-to-gps-in-denver/

[3] C. Matyszczyk, "Truck driver has GPS jammer, accidentally jams Newark airport." [Online]. Available: https://www.cnet.com/culture/truck-driver-has-gps-jammer-accidentally-jams-newark-airport/

[4] S. Applebaum, "Adaptive arrays," IEEE Transactions on Antennas and Propagation, vol. 24, no. 5, pp. 585–598, Sep. 1976, conference Name: IEEE Transactions on Antennas and Propagation.

[5] M. Zoltowski and A. Gecan, "Advanced adaptive null steering concepts for GPS," in Proceedings of MILCOM '95, vol. 3, Nov. 1995, pp. 1214–1218 vol.3.

[6] J. Starling, "Error Analysis of Carrier Phase Positioning Using Controlled Reception Pattern Antenna Arrays," Ph.D. dissertation, Auburn University, Apr. 2017, accepted: 2017-04-16T18:52:33Z. [Online]. Available: https://etd.auburn.edu//handle/10415/5601

[7] B. R. Rao, K. McDonald, and W. Kunysz, GPS/GNSS Antennas. Artech House, 2013.

[8] D. S. D. Lorenzo, "NAVIGATION ACCURACY AND INTERFERENCE REJECTION FOR GPS ADAPTIVE ANTENNA ARRAYS," Ph.D. dissertation.

[9] B. Widrow and S. D. Stearns, Adaptive Signal Processing. Prentice-Hall, 1985, google-Books-ID: X74QAQAAMAAJ.

[10] C. A. Balanis, Antenna Theory: Analysis and Design. John Wiley & Sons, Dec. 2015.

[11] W. L. Stutzman and G. A. Thiele, Antenna Theory and Design. John Wiley & Sons, May 2012, google-Books-ID: xhZRA1K57wIC.

[12] "IEEE Standard Definitions of Terms for Antennas," IEEE Std 145-1993, pp. 1–32, Jul. 1993, conference Name: IEEE Std 145-1993.

[13] M. N. Solomon, "Multi-mode patch antenna system and method of forming and steering a spatial null," US Patent US6 252 553B1, Jun., 2001. [Online]. Available: https://patents.google.com/patent/US6252553B1/en

[14] K. Antoszkiewicz and L. Shafai, "Impedance characteristics of circular microstrip patches," IEEE Transactions on Antennas and Propagation, vol. 38, no. 6, pp. 942–946, Jun. 1990, conference Name: IEEE Transactions on Antennas and Propagation.

[15] Z. Allahgholi Pour, "Control of phase centre and polarization in circular microstrip antennas," Ph.D. dissertation, 2006, accepted: 2013-05-17T18:05:50Z. [Online]. Available: https://mspace.lib.umanitoba.ca/xmlui/handle/1993/20779

[16] L. Shafai and Z. A. Pour, "Displacement of phase center location in circular microstrip antennas," Microwave and Optical Technology Letters, vol. 50, no. 10, pp. 2531–2535, 2008, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/mop.23762. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/mop.23762

[17] T. Mitha and M. Pour, "Principles of adaptive element spacing in linear array antennas," Scientific Reports, vol. 11, no. 1, p. 5584, Dec. 2021. [Online]. Available: http://www.nature.com/articles/s41598-021-84874-7

[18] E. Kaplan and C. Hegarty, Understanding GPS: Principles and Applications. Artech House, 2005.

[19] "GPS.gov: Space Segment." [Online]. Available: https://www.gps.gov/systems/gps/space/

[20] T. Watts, "A GPS and GLONASS L1 Vector Tracking Software-Defined Receiver," Ph.D. dissertation, Jul. 2019, accepted: 2019-07-29T19:31:08Z. [Online]. Available: https://etd.auburn.edu//handle/10415/6899

[21] P. Bevelacqua, "The Antenna Theory Website." [Online]. Available: https://www.antenna-theory.com/

[22] "IEEE Standard for Definitions of Terms for Antennas," IEEE Std 145-2013 (Revision of IEEE Std 145-1993), pp. 1–50, Mar. 2014, conference Name: IEEE Std 145-2013 (Revision of IEEE Std 145-1993).

[23] G. J. K. Moernaut and D. Orban, "An Introduction to Bandwidth, Gain Pattern, Polarization, and All That," p. 6.

[24] Z. Allahgholi Pour, "Investigation of parabolic reflector antennas as single- and multi-phase centre virtual antennas," Ph.D. dissertation, Jan. 2012, accepted: 2012-01-13T17:00:23Z. [Online]. Available: https://mspace.lib.umanitoba.ca/xmlui/handle/1993/5080

[25] R. L. Haupt and D. H. Werner, Genetic Algorithms in Electromagnetics. John Wiley & Sons, Apr. 2007, google-Books-ID: ioNrIAXJU_cC.

[26] H. Zhao, B. Lian, and J. Feng, "Space-Time Adaptive Processing for GPS Anti-Jamming Receiver," Physics Procedia, vol. 33, pp. 1060–1067, Jan. 2012. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1875389212014873

[27] Rao, GPS/GNSS Antennas.

[28] A. Bertheussen, "Adaptive Beamforming Using the Recursive Least Squares Algorithm on an FPGA," Jun. 2015.

Appendix A

Expanded Monte Carlo Results

This appendix details results shown in Chapter 5. Monte Carlo results for each GA variant are shown. Satellites acquired and null depth are shown for each variant.

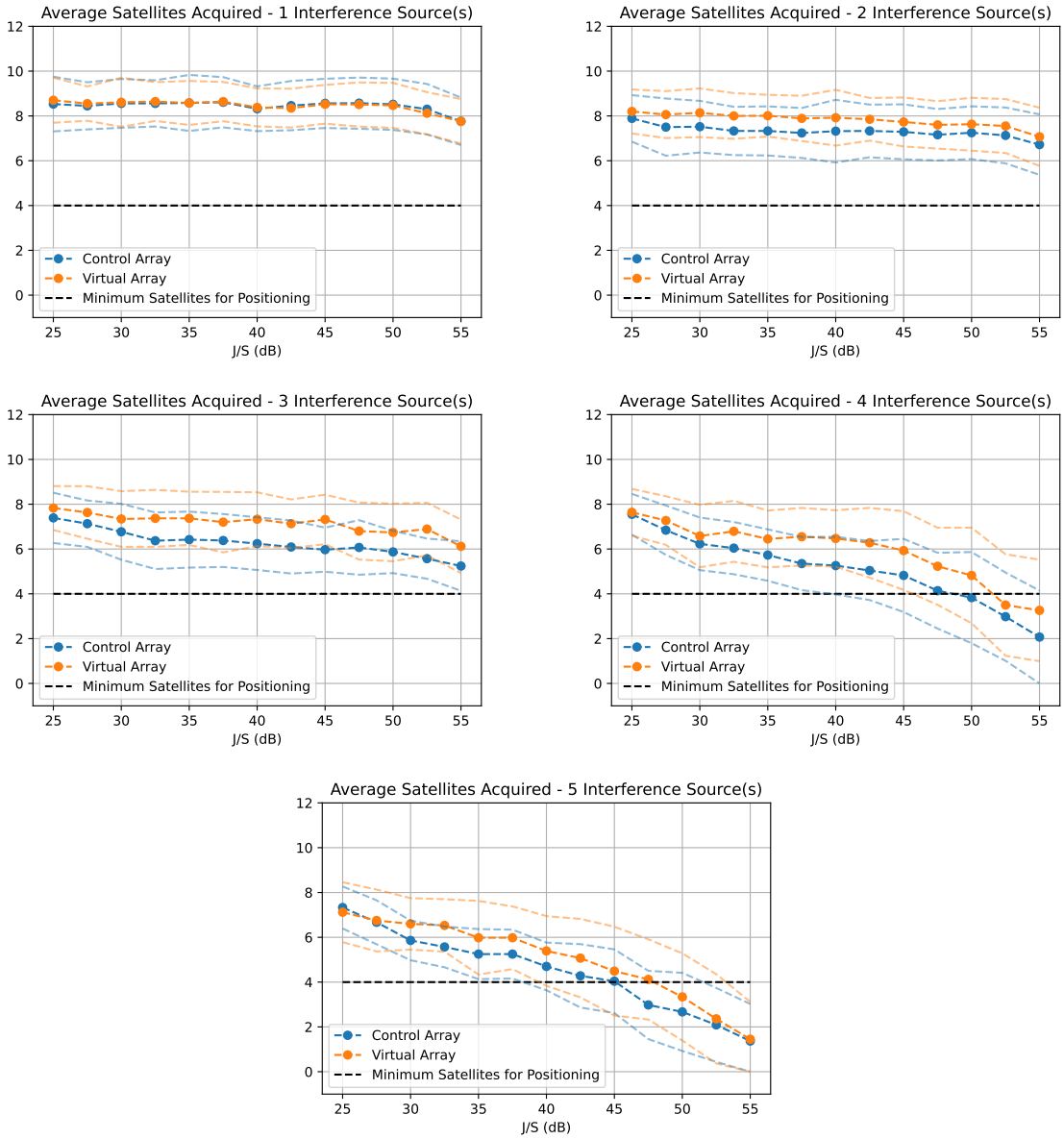

Figure A.1: Power Minimization Satellites Acquired ($\pm 1\sigma$)

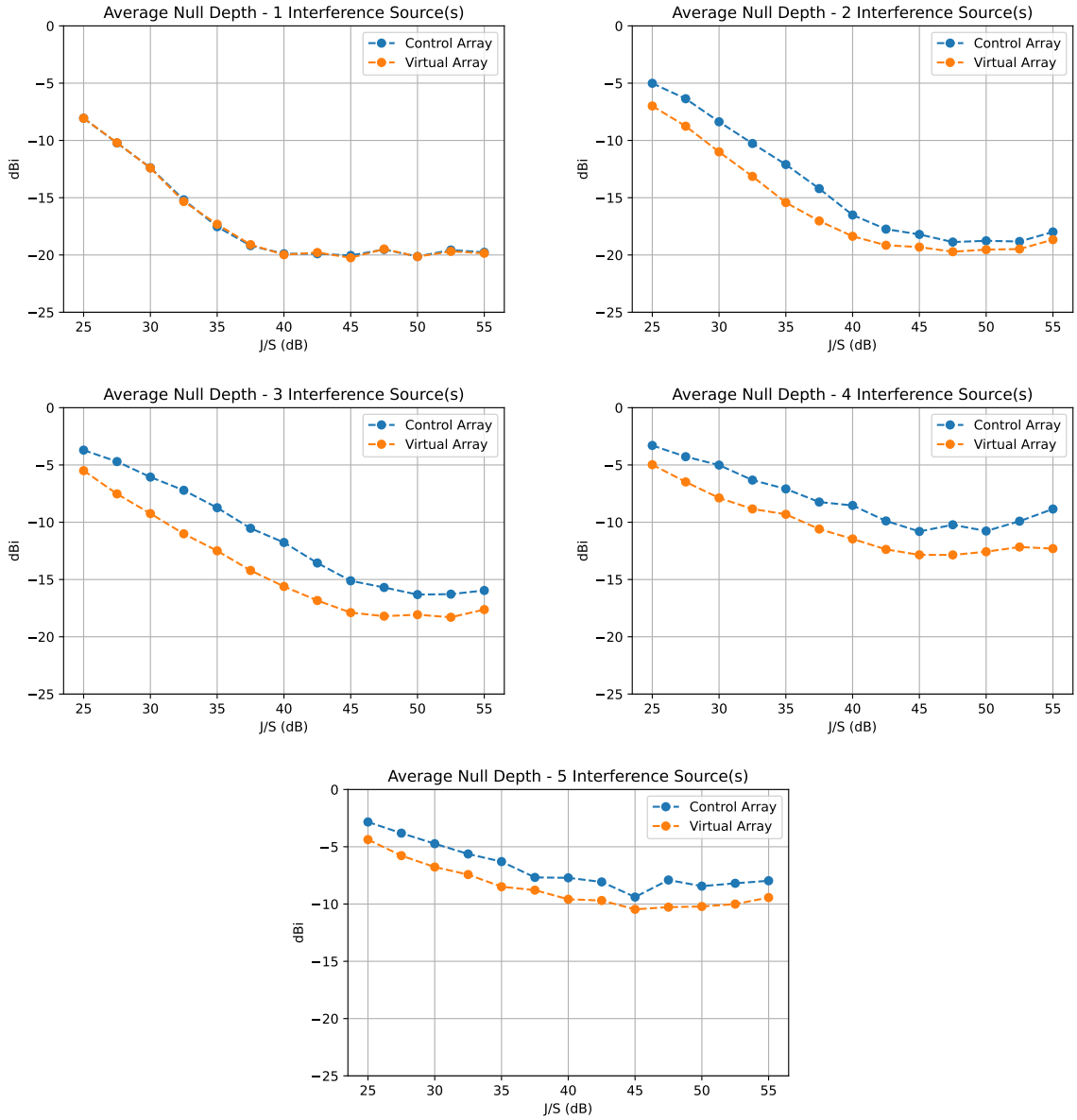

Figure A.2: Power Minimization Null Depth

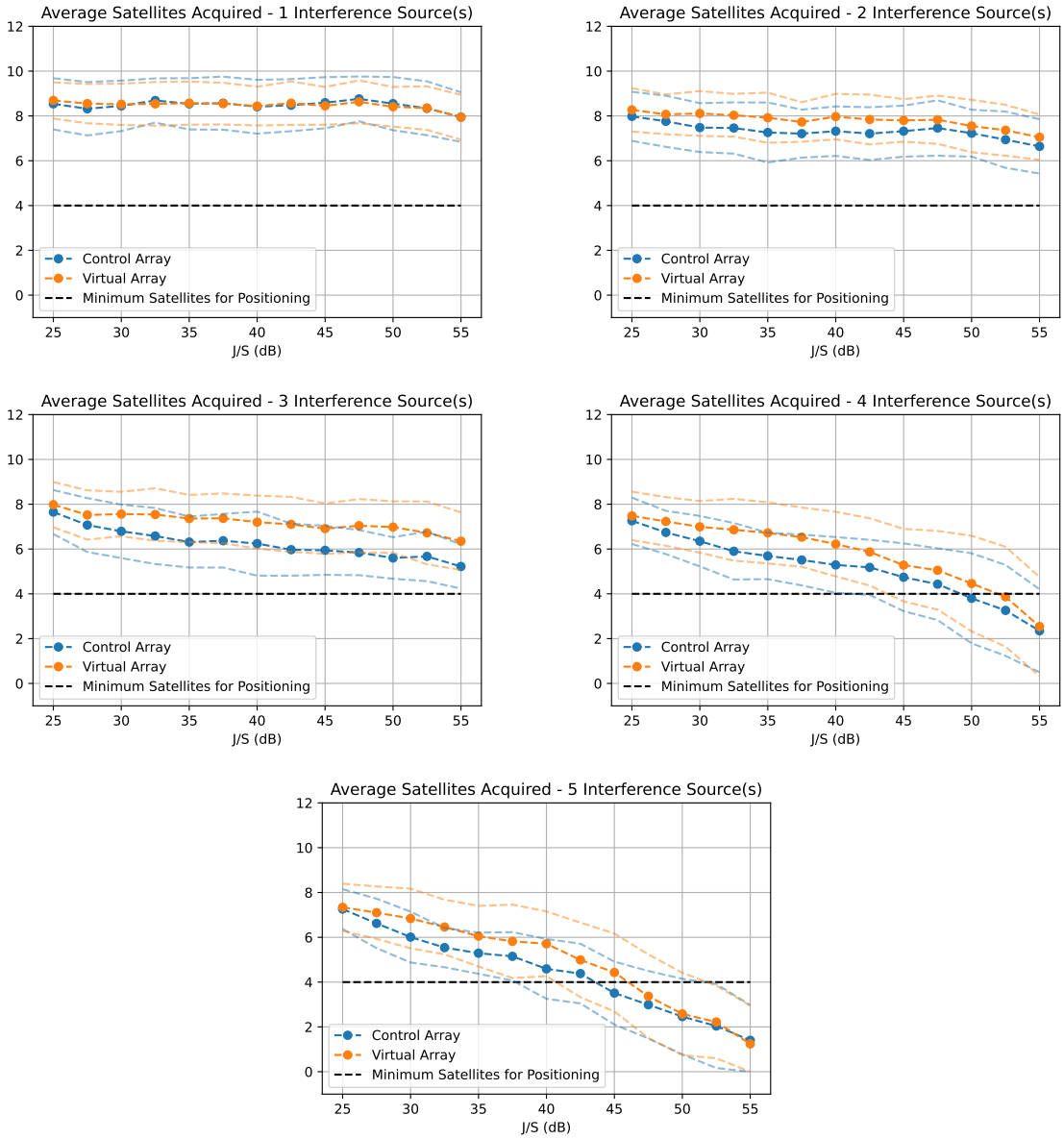

Figure A.3: Quantized Power Minimization Satellites Acquired ($\pm 1\sigma$)

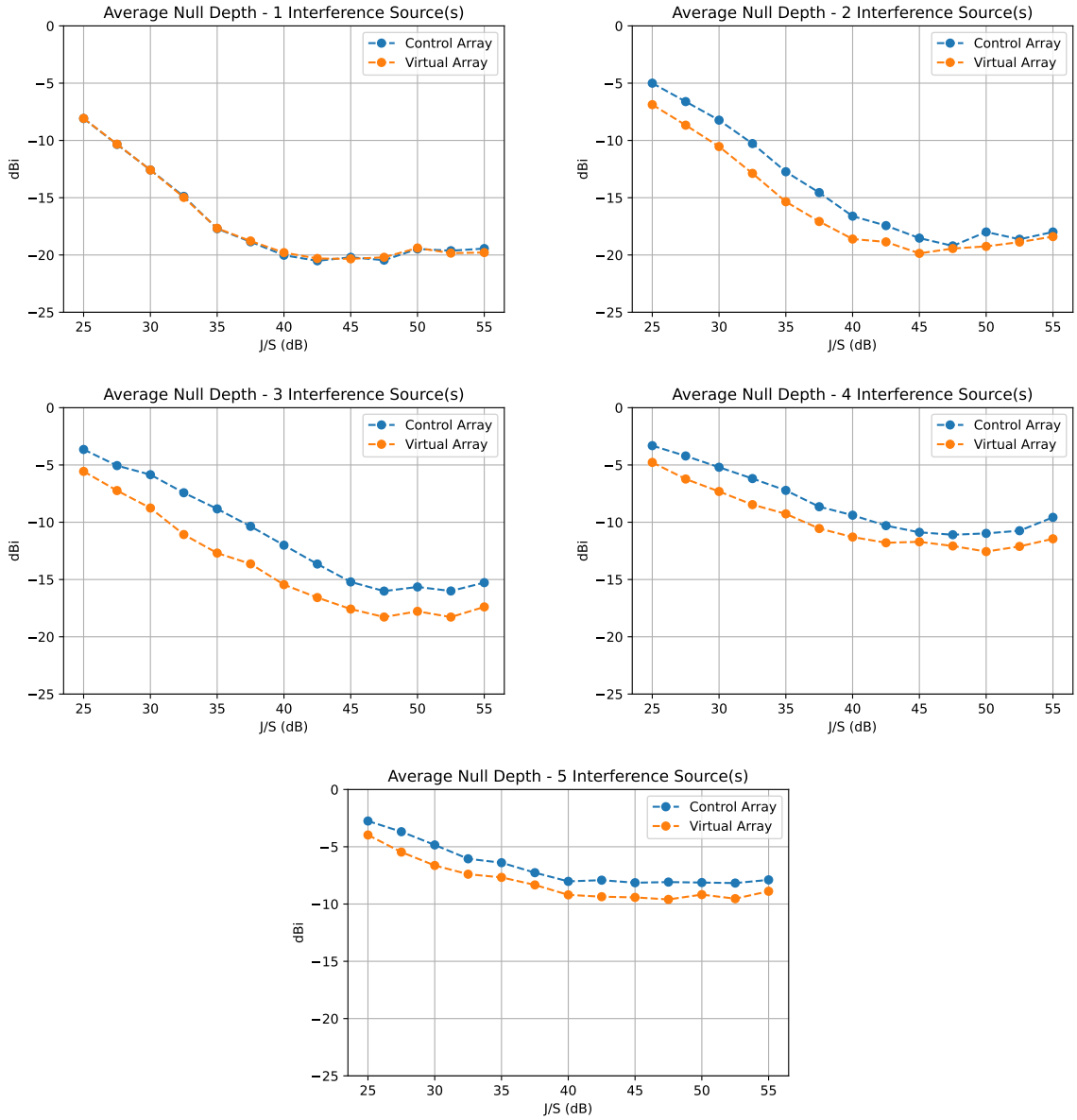

Figure A.4: Quantized Power Minimization Null Depth

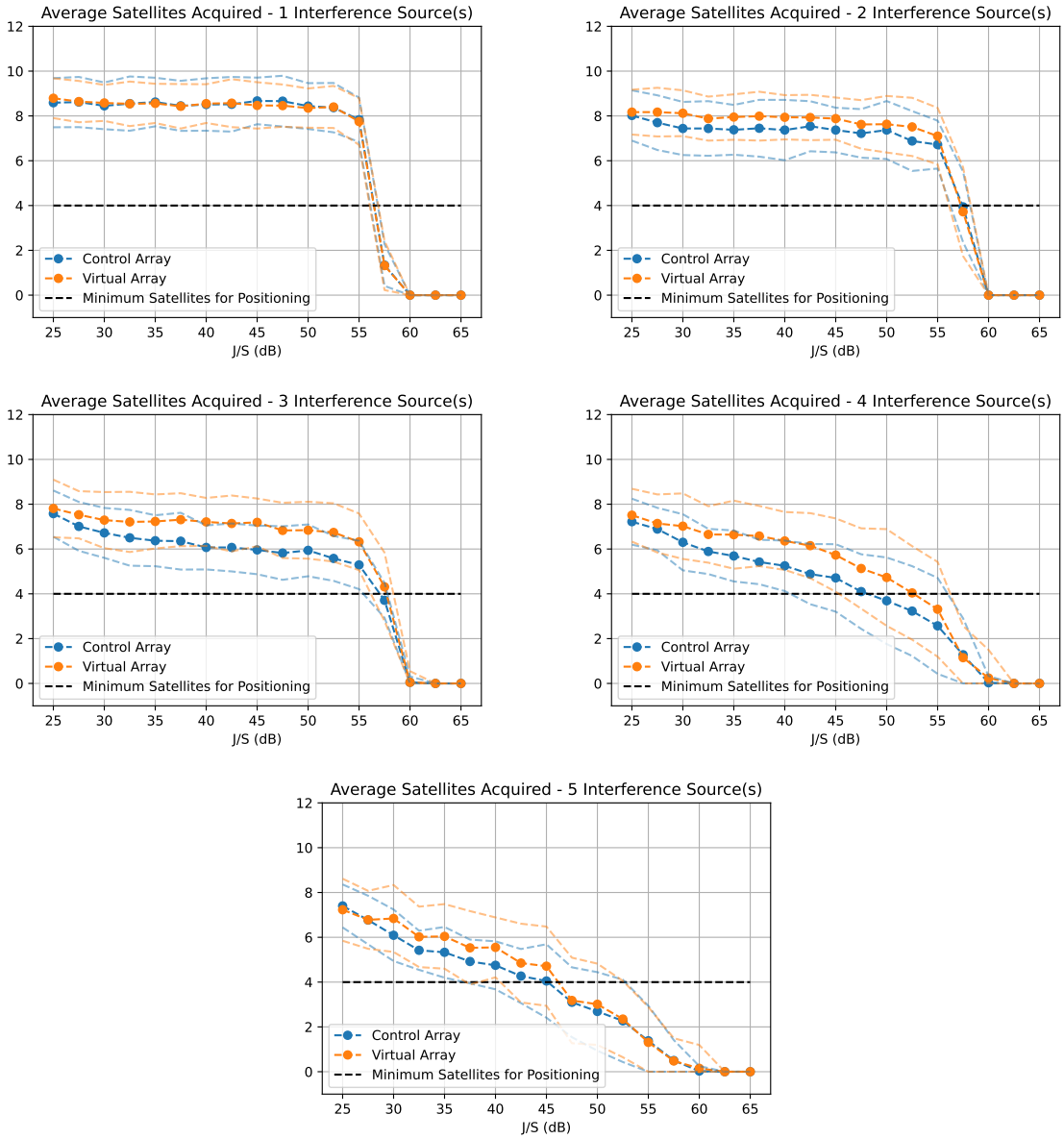

Figure A.5: A Priori 25% Interference Performance ($\pm 1\sigma$)

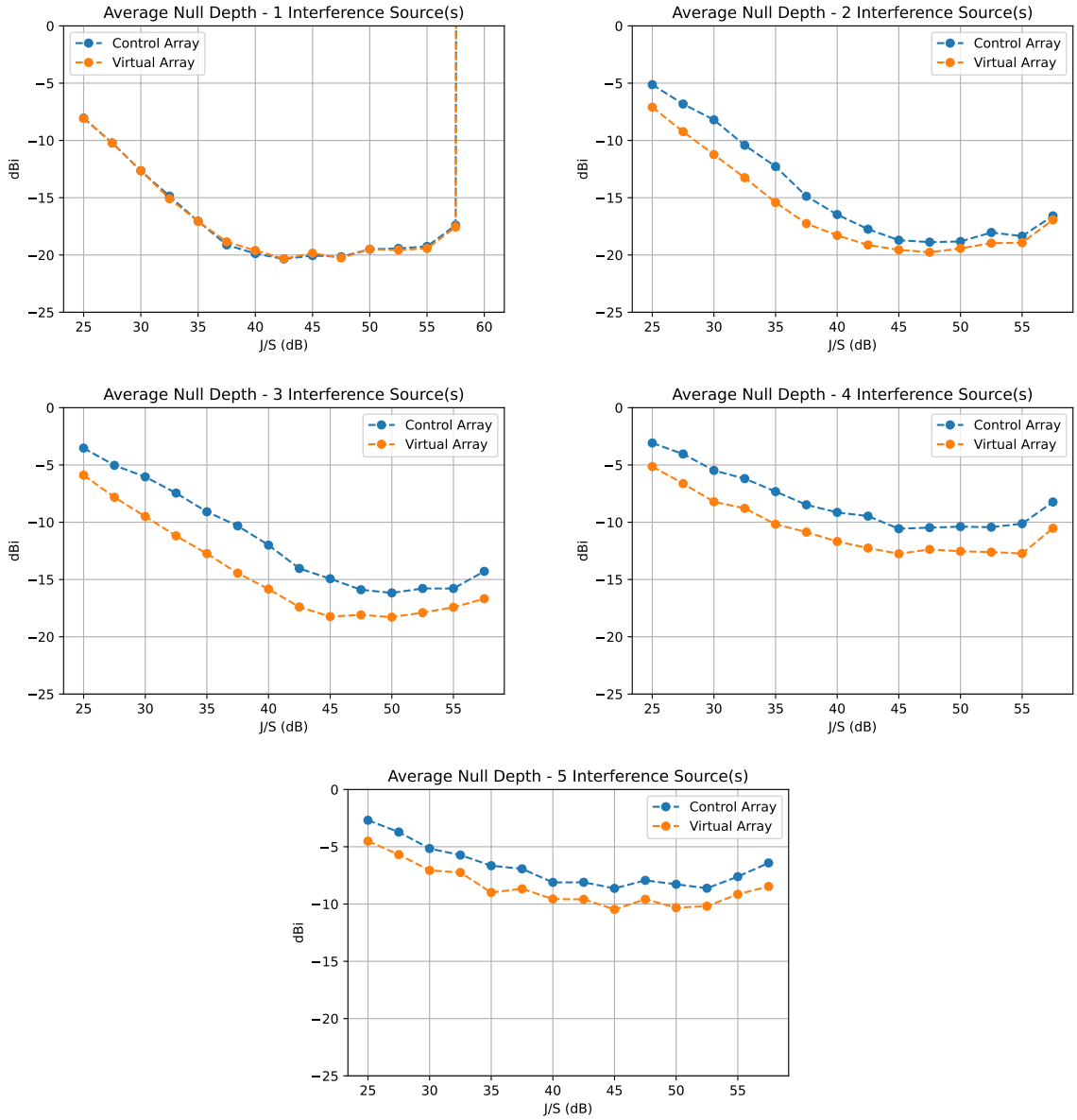

Figure A.6: A Priori 25% Null Depth

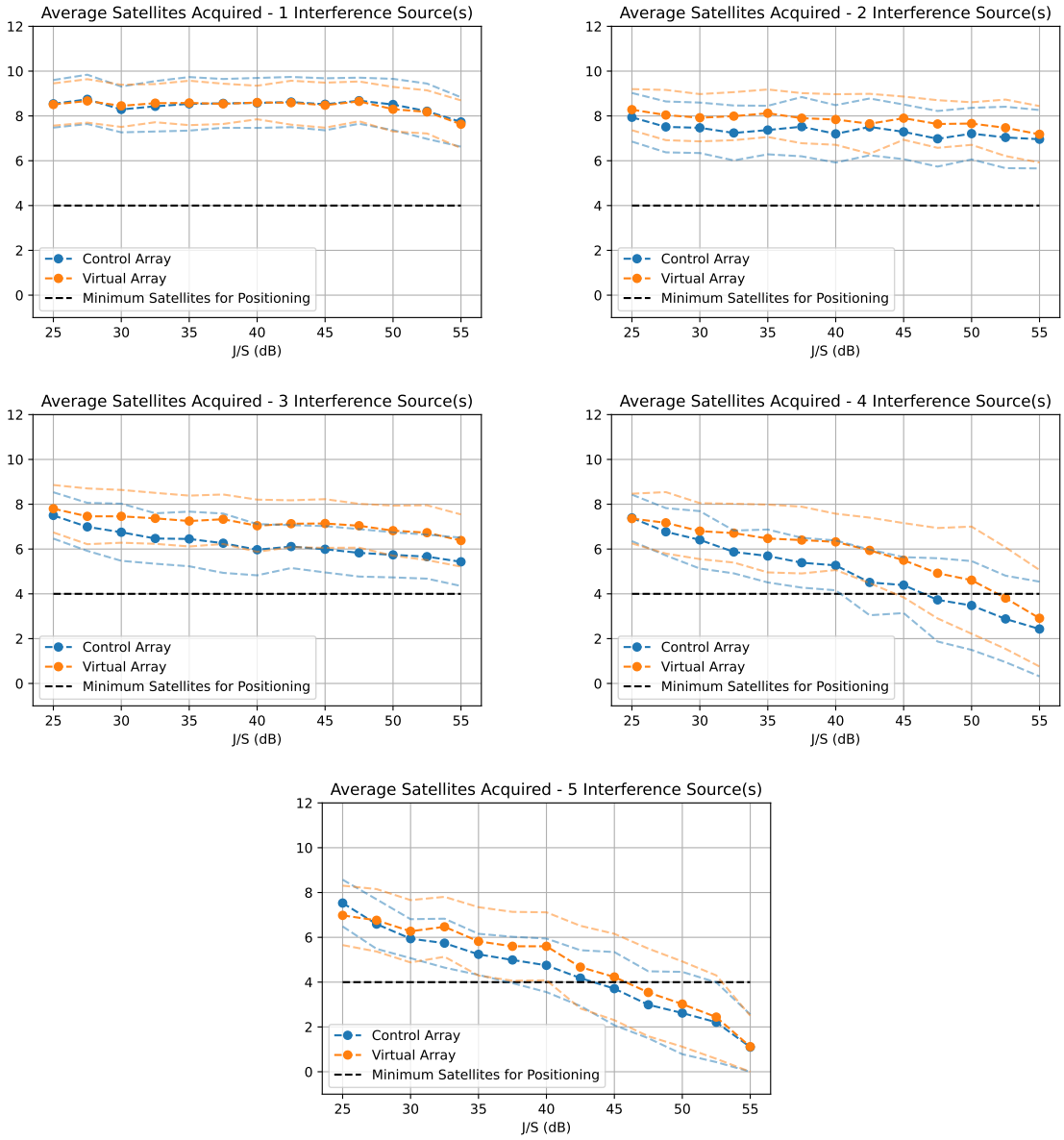

Figure A.7: A Priori 50% Interference Performance ($\pm 1\sigma$)

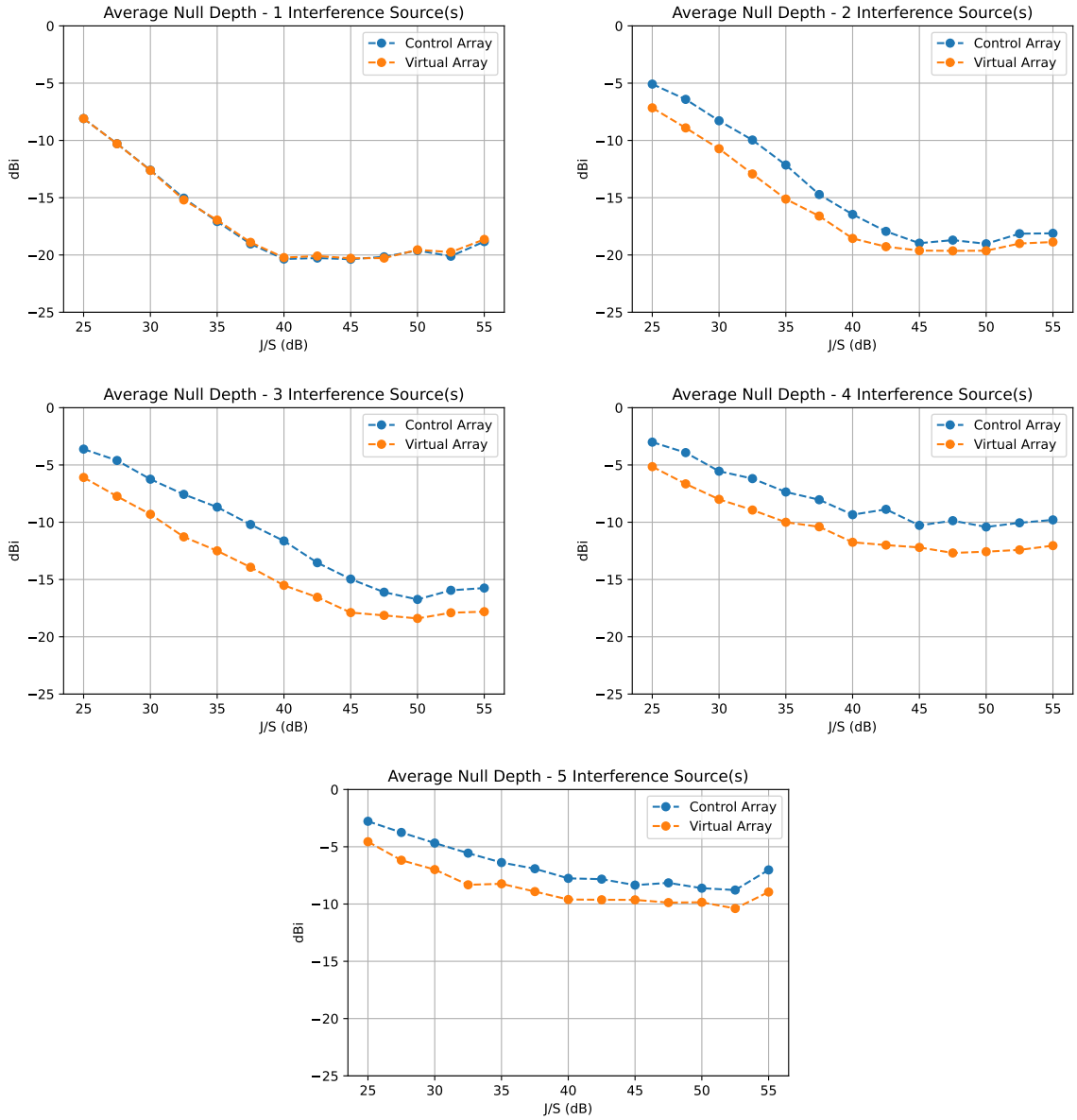

Figure A.8: A Priori 50% Null Depth

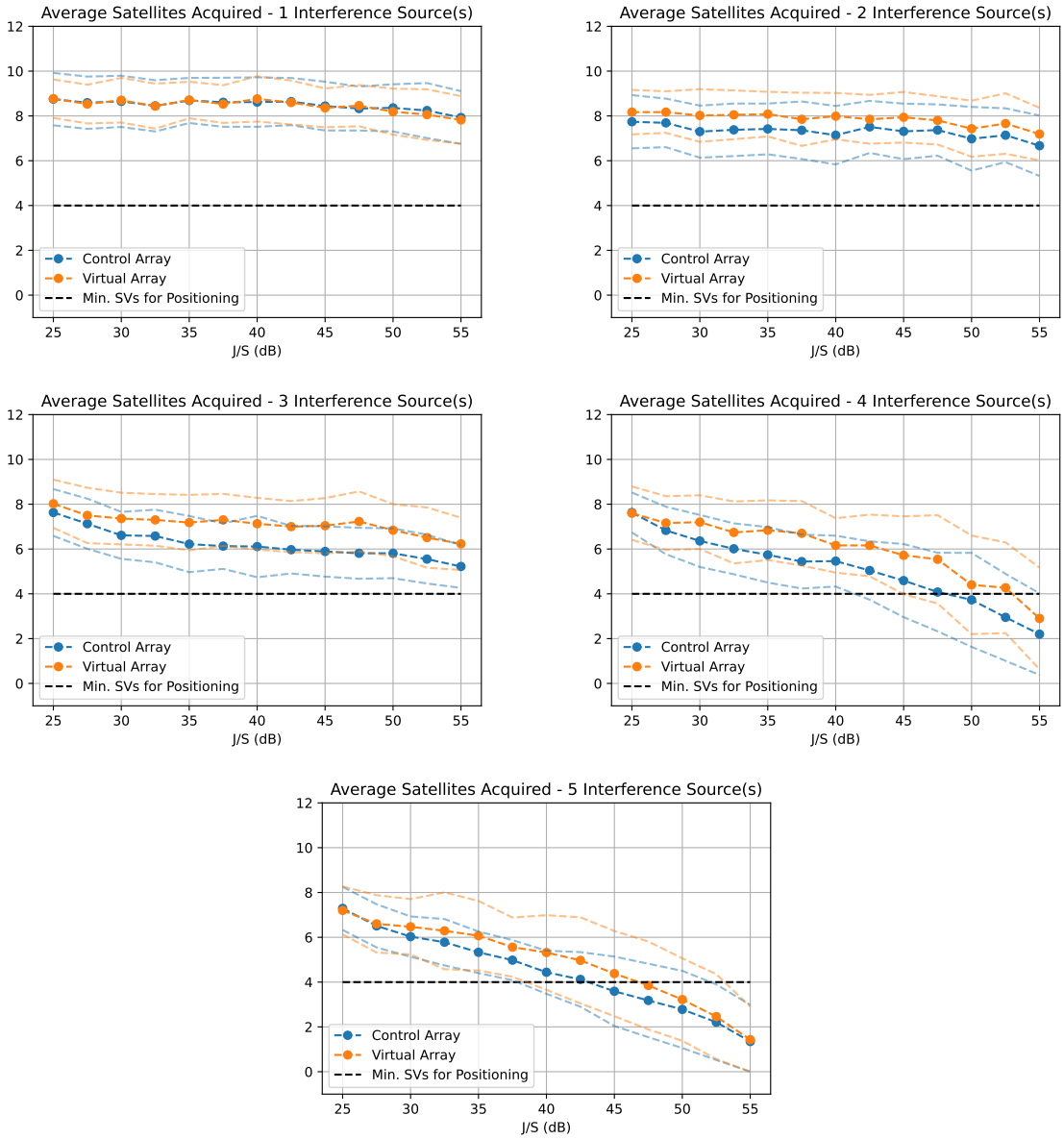

Figure A.9: LMS-in-the-loop Interference Performance ($\pm 1\sigma$)

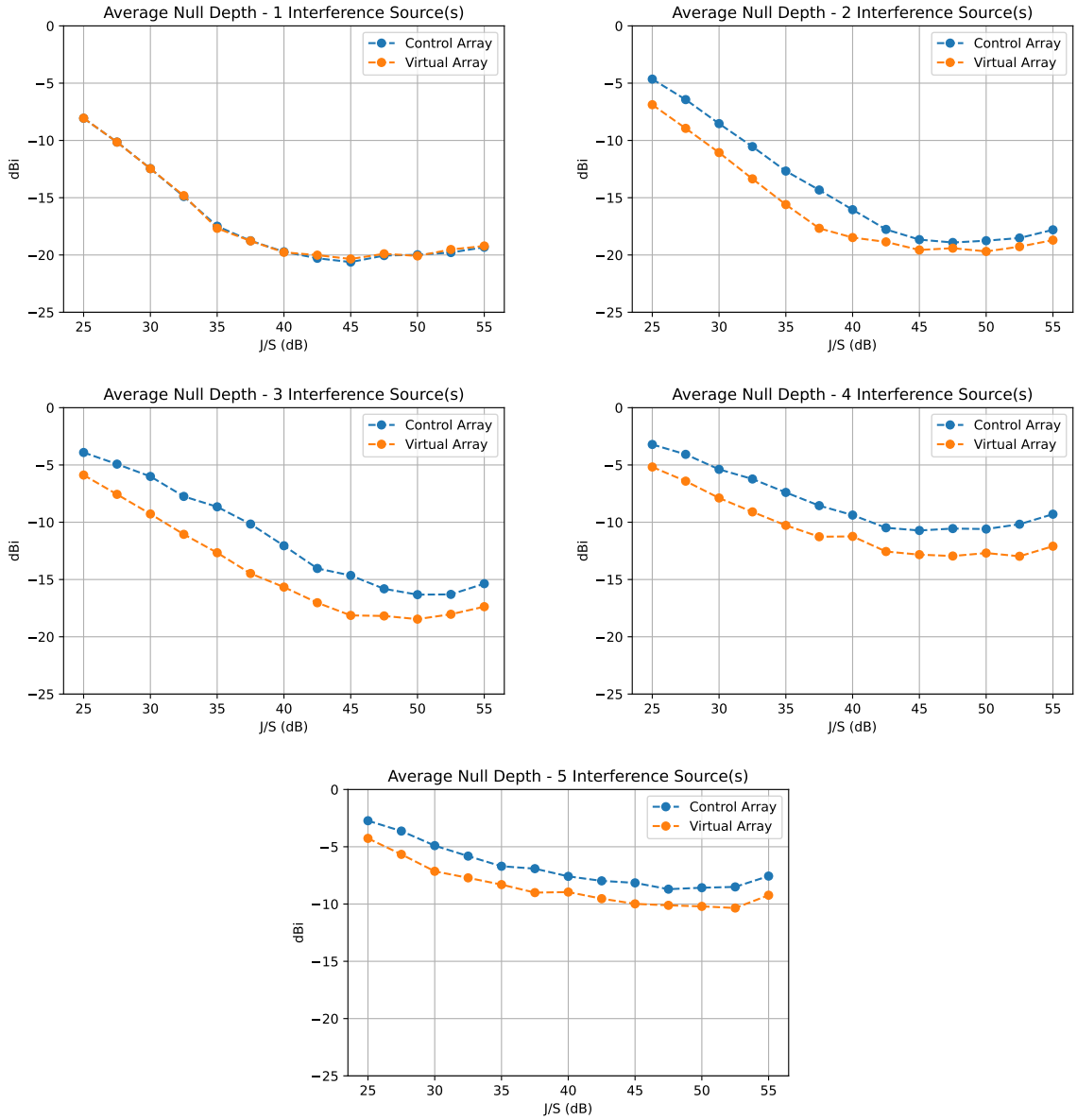

Figure A.10: LMS-in-the-loop Null Depth