UNIVERSITY OF OKLAHOMA

GRADUATE COLLEGE

SIMULATION AND ADAPTIVE APERTURE ALLOCATION FOR ALL-DIGITAL PHASED-ARRAY RADAR

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

In partial fulfillment of the requirements for the

Degree of

MASTER OF SCIENCE

By

CORBAN ANDREW RECKNAGEL

Norman, Oklahoma

2019

Downloaded from SHAREOK. Persistent Handle URL: https://hdl.handle.net/11244/321113

SIMULATION AND ADAPTIVE APERTURE ALLOCATION FOR ALL-DIGITAL PHASED-ARRAY RADAR

A THESIS APPROVED FOR THE SCHOOL OF ELECTRICAL AND COMPUTER ENGINEERING

BY

Dr. Nathan Goodman, Chair

Dr. Mark Yeary

Dr. Caleb Fulton

© Copyright by CORBAN ANDREW RECKNAGEL 2019

All Rights Reserved.

Contents

1 Introduction 1

1.1 Organization 3

2 Fundamentals 4

2.1 All-Digital Phased-Array Architectures 4

2.2 Phased-Array Spatial Processing 5

2.3 MIMO Operational Tradeoffs 9

3 MIMO & Phased-Array Considerations 11

3.1 Waveform Introduction 11

3.2 Waveform Modification 12

3.3 Beampattern Synthesis 23

3.3.1 Cross-Correlation Prediction 33

4 Simulation 38

4.1 Simulation Overview 38

4.2 Radar Signal Processing 39

4.3 Noise & Interference Model 40

4.4 Element Pattern 41

4.5 Simulation Validation 42

4.5.1 Image Maps 42

4.5.2 Target Detection 51

5 Energy Allocation Method 63

5.1 Parameter Space 63

5.2 Power Derivation 68

5.3 Return Detection and Profit Function 72

6 Subarray Placement 77

6.1 Exceptions 81

7 Allocation & Packing Evaluation 85

7.1 Operation Modes for Comparison 85

7.2 Performance Metrics 86

7.3 Results 88

8 Conclusions & Future Work 105

8.1 Conclusions 105

8.2 Future Work & New Potentials 106

9 Bibliography 109

List of Figures

Figure 2.1: Spatial sampling and wavenumber visualization 7

Figure 3.1. Result of matched filter on signal before interpolation. Blue is the auto-correlation and yellow is the cross-correlation 13

Figure 3.2. Result of matched filter on signal after interpolation by labeled methods. Auto-correlation are shown in blue and sample cross-correlations are shown in yellow. 14

Figure 3.3. Result of matched filter on different signals after interpolation by labeled method. 15

Figure 3.4. Bandwidth of signal after interpolation by labeled method. 16

Figure 3.5. Amplitude of signal after interpolation by labeled method. The real values are yellow, and blue represents the magnitude of the IQ waveform. 17

Figure 3.6 Chebyshev generation and conditioning flowchart 18

Figure 3.7 Qualitative comparison of waveform in time-domain before (top) and after (bottom) modification. The real values are yellow, and blue represents the magnitude of the waveform. 20

Figure 3.8 Qualitative comparison of waveform in frequency-domain before (top) and after (bottom) modification. 21

Figure 3.9 Qualitative comparison of waveform matched filter response before (top) and after (bottom) modification. 22

Figure 3.10: Normalized array factors for rectangular, traditional phased-array radars with dimensions described in legend 24

Figure 3.11: Element distribution for simulated sub-arrays 25

Figure 3.12: Array pattern for one sub-array aimed at broadside in elevation dimension 25
Figure 3.13: Beampatterns of four sub-arrays aimed in different directions in elevation dimension 26
Figure 3.14: Coherent combination of four beampatterns of Figure 3.13 in elevation dimension 26
Figure 3.15: Beampatterns of four sub-arrays aimed in different directions in elevation dimension and their coherent combination 27
Figure 3.16. Flowchart for array factor generation 28
Figure 3.17. Beampatterns and combined patterns for phased-MIMO radar with non-zero cross-correlation and different aiming vectors 28
Figure 3.18 Cross-Correlation Floor 29
Figure 3.19. Comparisons of peak and sidelobe levels of matched filter output in Elevation. 31
Figure 3.20. Matched filter results with one waveform much larger than all other waveforms. 32
Figure 3.21. Matched filter with cross-correlation prediction and simulated with error measurement. 36
Figure 4.1. Element pattern in polar coordinates with amplitude as fraction of total power and angles in degrees. 41
Figure 4.2. Range-Doppler map of traditional phased-array radar without noise (left) and with noise (right). 43

Figure 4.3. Range slice along maximum Doppler and spatial frequencies without noise (left) and with noise (right)..... 44
Figure 4.4. Zoomed Range-Doppler map of traditional phased-array radar without and with noise..... 44
Figure 4.5. Zoomed Range-Doppler map of phased-MIMO array with two Chebyshev waveforms without noise (left) and with noise (right)..... 45
Figure 4.6. Range slice along maximum Doppler and spatial frequencies of Phased-MIMO configuration with two Chebyshev waveforms without noise (left) and with noise (right)..... 45
Figure 4.7. Zoomed Range-Doppler map of phased-MIMO array with four Chebyshev waveforms without noise (left) and with noise (right)..... 46
Figure 4.8. Range slice along maximum Doppler and spatial frequencies of Phased-MIMO configuration with four Chebyshev waveforms without and with noise..... 47
Figure 4.9. Spatial frequency map of a target without noise, 10 GHz center frequency, and four Chebyshev waveforms..... 48
Figure 4.10. Spatial frequency map of a target without noise, 10 GHz center frequency, one Chebyshev waveforms, and increased resolution through zero-padding..... 49
Figure 4.11. Spatial frequency map of a target with noise, 10 GHz center frequency, one Chebyshev waveforms, and increased resolution through zero-padding..... 50
Figure 4.12. Spatial frequency map of a target with noise, 10 GHz center frequency, four Chebyshev waveforms, and increased resolution through zero-padding..... 51
Figure 4.13. CA-CFAR 4D cube used for detection..... 53

Figure 4.14. Histogram with probability normalization for the voltage resulting at each matched filter with the number of transmitted waveforms listed for each subplot. 54
Figure 4.15. Histogram with probability normalization for the voltage resulting from the combination of all matched filters with the number of transmitted waveforms listed for each subplot. 57
Figure 4.16. Histogram with probability normalization for the noise resulting from the combination of all matched filters with the number of transmitted waveforms listed for each subplot. 58
Figure 4.17. PDFs for signal and noise with threshold shown in orange. 60
Figure 4.18. Receiver operation curve derived with CA-CFAR method for random trials of 9 targets spawned within a broadside beam over 90 runs. The number of simultaneous transmit waveforms is included in the legend. 61
Figure 5.1. Two example cases of MAU Reduction 64
Figure 5.2. Potential sub-array divisions for one example MAU-reduced array from Figure 5.1. 65
Figure 6.1. Array Allocation flowchart overviewing the process of array size reduction and packing. 79
Figure 6.2. Array Allocation flowchart with two examples of MAU Reduction and Array Packing. The element-level configuration is also shown for clarity. 80
Figure 6.3. Array Allocation flowchart with an example of a potential loner exception. 82

Figure 7.1. Histogram with probability normalization for the target update time with a 4x4 array simulated without noise. 89
Figure 7.2. Histogram with probability normalization for the difference between the calculated RCS and its true value with a 4x4 array simulated without noise. 91
Figure 7.3. Histogram with probability normalization for the difference between the predicted return and its actual value with a 4x4 array simulated without noise. 93
Figure 7.4. Histogram with probability normalization for the average target return when updated with a 4x4 array simulated without noise. 95
Figure 7.5. Histogram with probability normalization for the update time since the last update for each target with a 4x4 array simulated without noise. 97
Figure 7.6. Histogram with probability normalization for the difference between the calculated RCS and its true value with a 4x4 array simulated with thermal noise. 98
Figure 7.7. Histogram with probability normalization for the difference between the predicted return and its actual value with a 4x4 array simulated with thermal noise. 100
Figure 7.8. Histogram with probability normalization for the average target return when updated with a 4x4 array simulated with thermal noise. 102

ABSTRACT

Increased flexibility afforded by all-digital radar architectures and operational concepts such as MIMO radar can be leveraged with new waveforms and aperture allocation methods to improve radar performance in diverse situations. A phased-array/MIMO continuum of operation is possible through all-digital architectures and further increases the degrees of freedom for adaptive-transmit radar. This thesis describes modeling this flexibility provided by all-digital radar and initiates potential strategies for waveform designs and aperture segmentation for improved performance for tracking and detection in target-dense environments.

These initial strategies leverage information from previous target measurements to predict the return strength for future configurations. By the careful definition of possible configurations and beam-steering directions, the performance predictions can be condensed into a manageable profit metric. The profit metric implemented in this thesis favors configurations that are expected to produce some minimally required signal to noise ration (SNR) while still providing meaningful target insight. By utilizing sub-sets of the normally required resources, other resources are freed for additional tasks, improving efficiency. The proposed allocation method requires high SNR for effective operation, which may be difficult to achieve in real systems. However, the goal of the allocation method is to provide initial strategies for allocation and parameter condensation that may mature into methods without this high SNR requirement. More sophisticated methods using similar strategies with additional consideration for environmental noise and interference may develop as MIMO radar matures.

1 INTRODUCTION

The promulgation of FPGA technology, improvements in radio-frequency (RF) integrated circuits, and advances in analog-to-digital converters (ADCs) have enabled rapid advances in all-digital radar system architectures, which possess waveform synthesis and signal digitization capability at every element of an antenna array [1]. Because of the fine control enabled by such an architecture, aperture reconfigurability is a key opportunity for performance improvement. Combined with software-defined radar concepts [2], emerging levels of radar adaptability are unprecedented, but still require new algorithms and waveforms to achieve their full potential. This thesis describes an all-digital simulation methodology, sample waveforms used for different transmit sub-arrays, and an approach for determining transmit sub-array size and sub-array placement into the overall array.

Phased-array radar systems comprise an array of radiating elements that allow electronic steering through phase and amplitude weightings applied to each element. Phased-array radar systems demonstrate better resolution and directionality as well as improved flexibility than their single-element counterparts [3]. Multiple-input, multiple-output (MIMO) radar expands on phased-array radar ideas by transmitting two or more unique waveforms and combining their results via signal processing. MIMO technology is well researched in the field of communication, and its radar applications have since become a topic of interest. MIMO radar can offer better slow-moving target detection [4], increased beamwidth shape flexibility [5], and simultaneous scanning of multiple directions at the cost of signal-to-noise ratio (SNR) [6], complexity of computation and waveform generation [4], and/or increased dwell

time [4] when compared to phased-array radar. Though MIMO operation is not intrinsically better than coherent phased-array operation in many scenarios, the implementation of an adaptive all-digital radar aperture allows for a phased-MIMO continuum [14]-[16] in which fully MIMO, fully traditional, or hybrid configurations can be selected to optimize desired characteristics for the radar's situational objectives. This thesis discusses some of these tradeoffs, describes the development of an all-digital radar simulation, and presents an aperture "packing" approach to optimizing transmit sub-array configurations.

1.1 Organization

The remaining sections of this thesis are structured as follows. Section 2 explains fundamental physics and mathematics essential to the operation of phased-array and MIMO radars. It uses these fundamentals to explain benefits and compromises of each mode and the value of their combinational use. Section 3 describes assumptions and limitations of MIMO waveform design and introduces a waveform used in this study. Additionally, it evaluates the interaction of these waveforms when transmitted in different directions simultaneously. Section 4 overviews a simulation environment and signal processing methods that model a digital array radar with arbitrary control of transmit sub-arrays and their waveforms. This section includes validations used to verify the design of the simulation and its performance. Section 5 introduces a novel aperture allocation method that leverages MIMO waveforms used in a hybrid configuration that demonstrates characteristics of phased-array and MIMO methods. Section 6 explains a companion method used for the spatial placement of the sub-arrays generated by the allocation method from Section 5. Section 7 presents figures of merit and comparison modes created to evaluate the methods introduced in Sections 5 and 6. Section 8 concludes the thesis and calls attention to future topics of research.

2 FUNDAMENTALS

Although many fundamentals of radar signal processing and engineering explain the operation of MIMO and phased-array radars, some concepts require a greater depth of understanding for the functionalities and concepts developed in this thesis. This section is intended to build some of the requisite knowledge for understanding the more advanced physics and the consequential limitations and benefits discussed later.

2.1 All-Digital Phased-Array Architectures

All-digital radar architectures allow the digital transmission and reception of radar signals. Digital systems are often more flexible than their analog counterparts, and this reconfigurability allows increased freedom in the transmitted signal, signal processing methods, and order of processing. Whereas traditional systems may only transmit a limited set of waveforms, an all-digital architecture allows the transmission of different waveforms restricted only by the ability to digitally construct the signals and the ability of the digital to analog converter or DAC units to produce them.

However, the increased flexibility of all-digital architectures comes at the cost of the generation, internal transmission, and processing of large quantities of data at transmission and reception. FPGA technology is commonly added to the processing chain to reduce system latency and total load handled by traditional computational units while still allowing reconfigurable performance and adaptive processing methods. For example, within the University of Oklahoma's Horus system, FPGA units perform some fundamental beamforming operations and data routing.

For the purposes of this study, it is assumed that each element of the hypothetical all-digital array is digitized and input to the processing units as a raw datacube. This means we have full flexibility in the order of operations for processing. In addition, we do not attempt to model the latency of the radar's components, but a limited set of imperfections are modeled, such as errors in the fast-time domain due to the digitization of the analog returns through interpolation of the return signal, a complex envelope with rise and fall times, and cross-correlation effects due to the transmission of multiple waveforms.

2.2 Phased-Array Spatial Processing

Two of the primary benefits of phased-array technology are the ability to electronically steer the transmit beam through phase-weighting and the ability to estimate the angle of arrival through beamforming. These functionalities are possible because of the spatial diversity of the array and are performed by applying phase-weightings at transmit and by measuring the received phase-shifts caused by the propagation direction of target returns. These operations are built upon the concept of spatial frequencies and are closely related to the Fourier transform and Fourier transform pairs. The wavenumber of a propagating waveform, in cycles per distance, can be determined from its carrier frequency (F_c) and the speed of light (c) following

$$\mathrm{k} = \frac{2\pi}{\lambda} = \frac{2\pi F_c}{c}. \tag{2.1}$$

When samples of an array are used to measure spatial components of the wavenumber, the result is the projection of the propagating wave onto the dimensions of the array. For example, if an array is made of a line of elements on the y-axis, then

only the y-component of the spatial frequency is directly measured by the array. In three dimensions the wavenumber vector can be described as

$$\bar{K} = K_x\hat{x} + K_y\hat{y} + K_z\hat{z} \tag{2.2}$$

with magnitude K. Because the carrier frequency is known, the magnitude of K is known from (2.1), and it is possible to determine the components of the direction of arrival, DOA, of the waveform as a unit vector as

$$\bar{u}_{\mathrm{k}} = \frac{K_x\hat{x} + K_y\hat{y} + K_z\hat{z}}{K}. \tag{2.3}$$

The magnitude of the combination of unmeasured dimension(s) is implied by the magnitude of the overall wavenumber as

$$K_u = \sqrt{K^2 - \sum_{i \in meas} K_i^2}\,, \tag{2.4}$$

where K_u denotes the magnitude of the wavenumber distributed in some unknown fashion in the dimensions perpendicular to the measured magnitudes. When two-dimensional arrays are used, all three dimensions can be determined, but only one dimension is directly measured with linear arrays. As in any frequency measurement, aliasing can occur if the signal is not sampled at the Nyquist rate. For spatial frequencies, this sampling interval is one-half wavelength in each measured dimension [19]. An unmodulated waveform propagating across a 2-D array with half-wavelength spacing is pictured in Figure 2.1

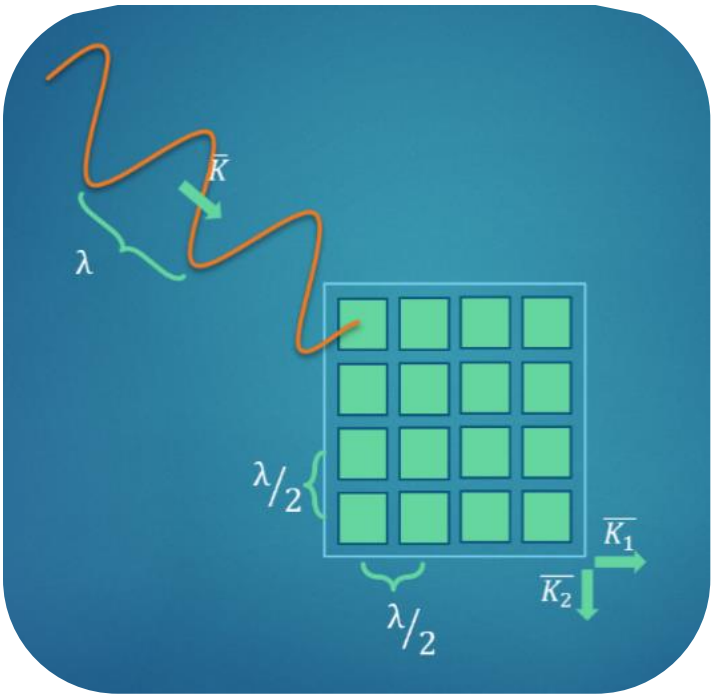

Figure 2.1: Spatial sampling and wavenumber visualization

Though many methods exist for direction-of-arrival estimation, non-digital arrays often use filter banks for each spatial frequency, which can be digitally implemented through FFT operations along each array dimension. A key benefit of the all-digital architecture is the ability to perform digital beamforming at transmit and receive after the data has already been digitized and stored. An FFT operation determines coarse measurements for a few receive beams, and more precise methods such as maximum likelihood or Capon beamforming can be applied after detection. The implementation of other processing methods and combinations of methods within a simulation environment could be valuable for future research.

The mathematics of transmit beamforming is analogous to receive beamforming. First, a spatial frequency vector $\overline{K}_t$ corresponding to a desired transmit direction $\bar{u}_{K_t}$ and wavenumber of the carrier signal K is chosen following

$$\overline{K}_t = K\bar{u}_{K_t}. \tag{2.5}$$

The transmit phase weightings to produce the desired transmit direction can be derived from the frequency shifting property of the Fourier Transform that states a shift in the frequency domain is equivalent to multiplication by a complex exponential in the time-domain as

$$e^{j\omega_0 n}x[n] \overset{\mathcal{F}}{\Leftrightarrow} X\left(e^{j(\omega-\omega_0)}\right), \tag{2.6}$$

where ω_0 is the frequency shift, n is the sampling index, x[n] is the time-domain signal, and $X\left(e^{j(\omega-\omega_0)}\right)$ is the Fourier Transform. With one-dimensional spatial sampling, we replace the time index n with the transmit sub-array position r_t, the frequency index ω with the spatial frequency or wavenumber K, and the frequency shift ω_0 with the wavenumber of interest K_t. It is important to note we are representing K_t as a scalar in this example because it is being measured in only one dimension. With these assumptions, (2.6) becomes

$$\mathrm{e}^{\mathrm{j}K_t r_t}\mathrm{x}[r] \overset{\mathcal{F}}{\Leftrightarrow} \mathrm{X}\left(\mathrm{e}^{\mathrm{j}(\mathrm{K}-K_t)}\right). \tag{2.7}$$

When the dimensionality is expanded by adding more elements and dimensions, such as the Cartesian coordinate system, the frequency shift contains multiple dimensions. The equivalent Fourier Transform relationship with multiple dimensions is denoted as the scalar product of the two vectors as

$$\mathrm{e}^{\mathrm{j}\overline{K}_t\cdot\bar{r}}\mathrm{x}[r] \overset{\mathcal{F}}{\Leftrightarrow} \mathrm{X}\left(\mathrm{e}^{\mathrm{j}(\overline{K}-\overline{K}_t)}\right). \tag{2.8}$$

In phased-array radars with isotropic element patterns and simultaneous transmission without phase shifts, the energy is focused in the $\overline{K}_t = 0$ or broadside

direction. To change the steer angle from broadside, we can calculate the following weighting

$$w_t = e^{-j\bar{K}_t \cdot \bar{r}_t}. \tag{2.9}$$

This complex weighting will steer the array in the direction $\bar{K}_t$. It is important to note the negative sign produces the complex conjugate of the desired angle, so the spatial frequencies sum coherently in the frequency domain. As in the receive case, the transmit weightings can only steer the beam in array dimensions that have multiple elements. Those dimensions with only one element have transmit patterns matching the element pattern.

2.3 MIMO Operational Tradeoffs

MIMO radar is sometimes lauded as the natural successor to single input single output (SISO) phased-array radar because of its increased flexibility and greater waveform diversity. Although these characteristics allow new operational modes and can overcome some traditional radar limitations, they introduce new imperfections and restrictions. One of the most troublesome characteristics of coherent MIMO operation is the reduction in SNR proportional to the number of transmit sub-arrays. Although some authors assume the number of pulses within each CPI can be increased proportionally to the number of transmit waveforms to offset this SNR reduction, this extension increases the computational complexity, reduces the number of CPI that can be adaptively controlled over the same duration which may result in reduced coverage rates, and may be impractical or impossible for other reasons. It is also important to note the increase in CPI length and improved angular resolution from MIMO operation can also improve slow-moving target detection [4].

There are many MIMO operation techniques, as the only requirement is the transmission and reception of separable waveforms. Separable waveforms can be formed in the time-domain or frequency-domain, but we will focus on waveforms separated through waveform diversity [7]. Waveforms separated through waveform diversity are designed to have low cross-correlation and high auto-correlation [7]-[9], so that they can be separated by comparing the return data with the transmitted waveforms through matched filters. MIMO operation has recently been shown to improve radar operation in passive radar [10], [11], synthetic aperture radar (SAR) [12], and allow target velocity information from individual pulses [13].

When simulations and calculations of MIMO performance are made, it is often assumed that the waveforms are perfectly orthogonal, meaning they have zero cross-correlation, but in practice, real systems can only achieve 'very-low' cross-correlation as no two waveforms can be orthogonal over all possible relative delays and Doppler shifts. This non-zero cross-correlation introduces a new source of interference that can also degrade performance of MIMO radars when compared to ideal MIMO operation and SISO operation. The emergence of all-digital architectures allows adaptive negotiation of the compromises of MIMO and phased-array modes through operation within a phased-MIMO continuum [14]-[16] in which fully MIMO, fully traditional, or hybrid configurations can be selected to optimize desired characteristics for the radar's situational objectives. Operation within this continuum will be explored in later sections of this thesis.

3 MIMO & PHASED-ARRAY CONSIDERATIONS

3.1 Waveform Introduction

To simulate potential MIMO and multi-beam configurations of a digital array, we have implemented waveforms based on Chebyshev maps traditionally used for cryptology [9]. Chaotic sequences have been gaining interest for their apparent randomness and favorable cross-correlation properties [9], [17]. The chaotic properties mean they are deterministic sequences that vary significantly with minor differences in input condition and have been shown to have low cross-correlation [9],[17]. The combination of these two properties allows for the generation of an arbitrary number of sequences via selection of different input parameters, with all waveforms having low cross-correlation properties.

Chebyshev polynomials φ can be recursively defined for order m and sequence x as

$$\begin{aligned} \varphi_0(x) &= 1, \\ \varphi_1(x) &= x, \\ \varphi_m(x) &= 2x\varphi_{m-1}(x) - \varphi_{m-2}(x); \quad m \geq 2. \end{aligned} \tag{3.1}$$

These polynomials can also be defined using trigonometric functions as

$$\varphi_m(x) = \cos\left(m\cos^{-1}(x)\right), \quad m = 0,1,2... \tag{3.2}$$

The sequence(s) are determined by substituting the value (x_n) at the preceding index (n) into the polynomial function $\left(\varphi_m(x_n)\right)$ to calculate the value (x_{n+1}) at the next index $(n+1)$ as

$$x_{n+1} = \varphi_m(x_n)\,, x \in [-1, 1], \tag{3.3}$$

where the initial value can be any value between -1 and 1 as

$$x_0 \in [-1, 1]. \tag{3.4}$$

Because these sequences are seemingly random but deterministic where slight changes in initial value result in different output sequences and have been shown to exhibit low cross-correlation [9], it is possible to create an arbitrary number of unique sequences that define useful MIMO waveforms limited only by the digitized resolution on the interval $[-1,1]$ from which x_0 is selected. For our purposes, the value of x_0 is selected from a uniform distribution within this range of $[-1,1]$. For the benefits described above, Chebyshev waveforms are chosen as the primary waveform for our simulation.

3.2 Waveform Modification

The Chebyshev chaotic sequences exhibit some ideal properties for MIMO operation such as low cross-correlation and the ability to generate an arbitrary amount. However, there is also potential to increase their transmitted energy and provide bandwidth control [7]. MIMO waveforms based on modified versions of the Chebyshev maps have been simulated to evaluate their peak-to-average power ratios (PAPR), adjust their desired bandwidths, and study their cross-correlation properties. In-phase (I) and quadrature (Q) channels are created by generating two different Chebyshev sequences. To maintain normalized power per waveform, the waveform formed from the I/Q sequences is scaled by its magnitude to create a complex-valued unit vector. The resulting bandwidth of the synthesized sequence depends on various parameters, making it difficult to directly generate sequences with the desired time

duration and bandwidth. Once the sequence is generated, however, the sampling rate can be adjusted such that the bandwidth matches the desired bandwidth, and the waveform can then be resampled [18] to match the desired digital-to-analog converter (DAC) rate.

Interpolation performed during resampling can change cross-correlation properties, however, so it was important to determine which interpolation method should be used. We evaluated linear, FFT-based, cubic, and spline interpolation methods for their ability to retain cross-correlation and PAPR during the resampling process. Each method causes distortions that manifest in compromises in the spectral versus time-domain characteristics, waveform power, and cross- and auto-correlation peak differences. These metrics were evaluated for each interpolation method, and the results are shown in Figure 3.1 through Figure 3.5.

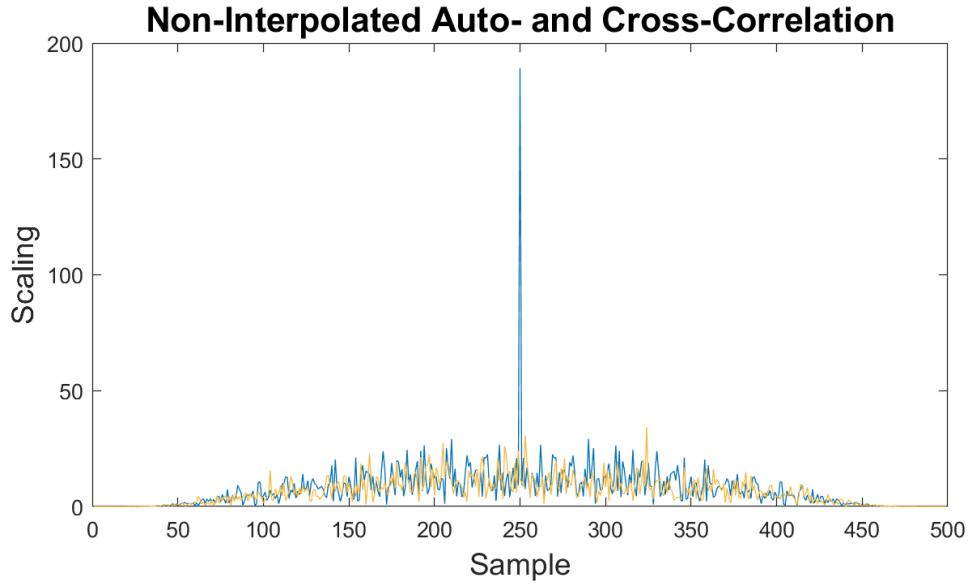

Figure 3.1. Result of matched filter on signal before interpolation. Blue is the auto-correlation and yellow is the cross-correlation

Figure 3.1 shows the un-interpolated signal's auto-correlation in blue and a sample cross-correlation in yellow. It shows a baseline for the performance of the unmodified

Chebyshev sequence. The waveform was created with a maximum amplitude of one, and the scaling of Figure 3.1 approximates the energy of the signal measured in samples. Because the maximum value per sample is one, the maximum magnitude of the matched filter output is the number of samples. The tapering applied to the waveform reduces the maximum to less than the number of samples. The tapering of this example reduces the energy of the initial waveform to approximately 87% of the number of samples. Figure 3.2 shows the auto-correlations after each method of interpolation. The auto-correlation peak is maximized by Cubic and Linear interpolation, meaning the PAPR should be better for these interpolation methods.

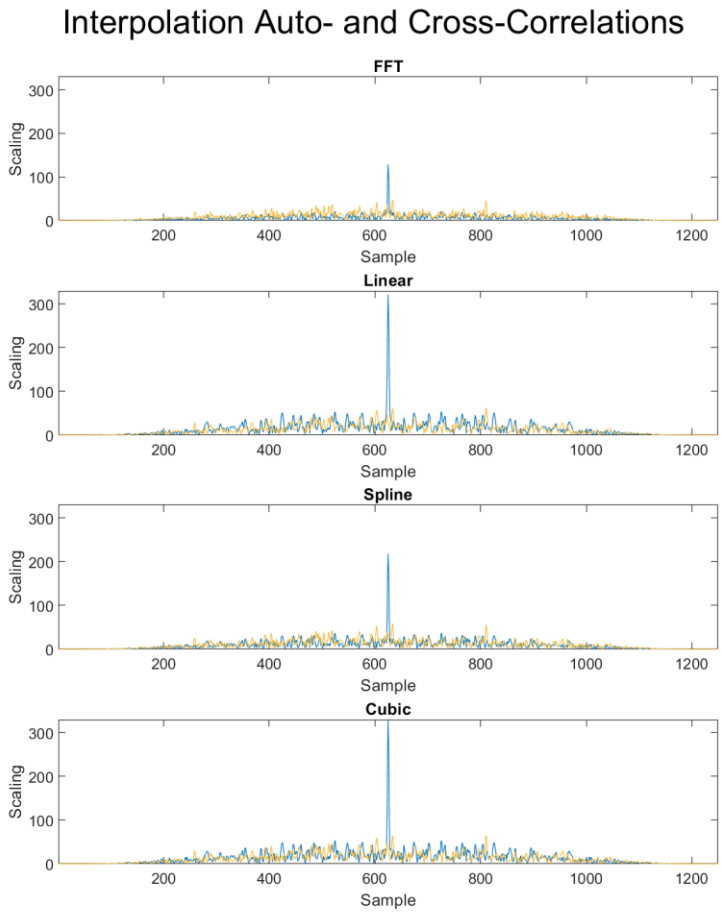

Figure 3.2. Result of matched filter on signal after interpolation by labeled methods. Auto-correlation are shown in blue and sample cross-correlations are shown in yellow.

Because the cross-correlation is low relative to the peak in Figure 3.2, Figure 3.3 was created for better indication of the structure of the cross-correlation after different interpolation methods. The cross-correlation magnitude and shape do not vary significantly between the interpolation methods in the included example.

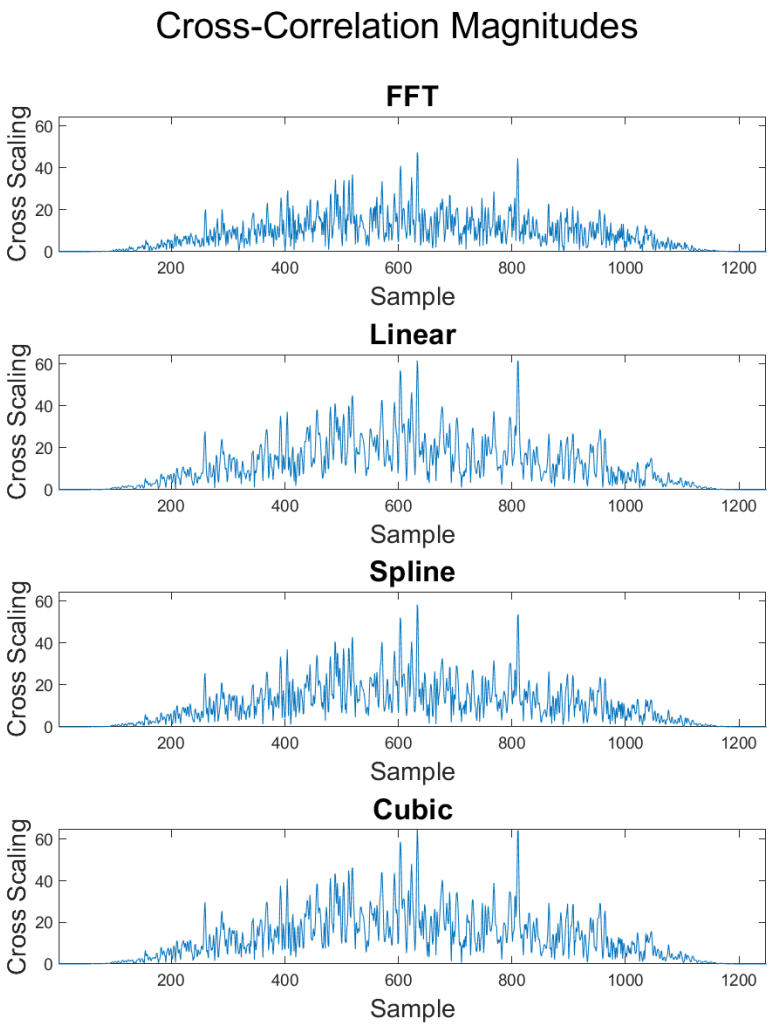

Figure 3.3. Result of matched filter on different signals after interpolation by labeled method.

After interpolation, the waveform is scaled so the maximum value of any sample is one. For this reason, the maximum value of the interpolated result is again the number of samples after interpolation. The initial sequence includes 250 samples,

and the final sequence includes 500 samples. This explains the increase in the maximum energy of some of the interpolated results as there are more total samples. The reduction in energy of the matched filter responses of the other methods indicates that their interpolation reduces the energy per sample of the resulting waveform.

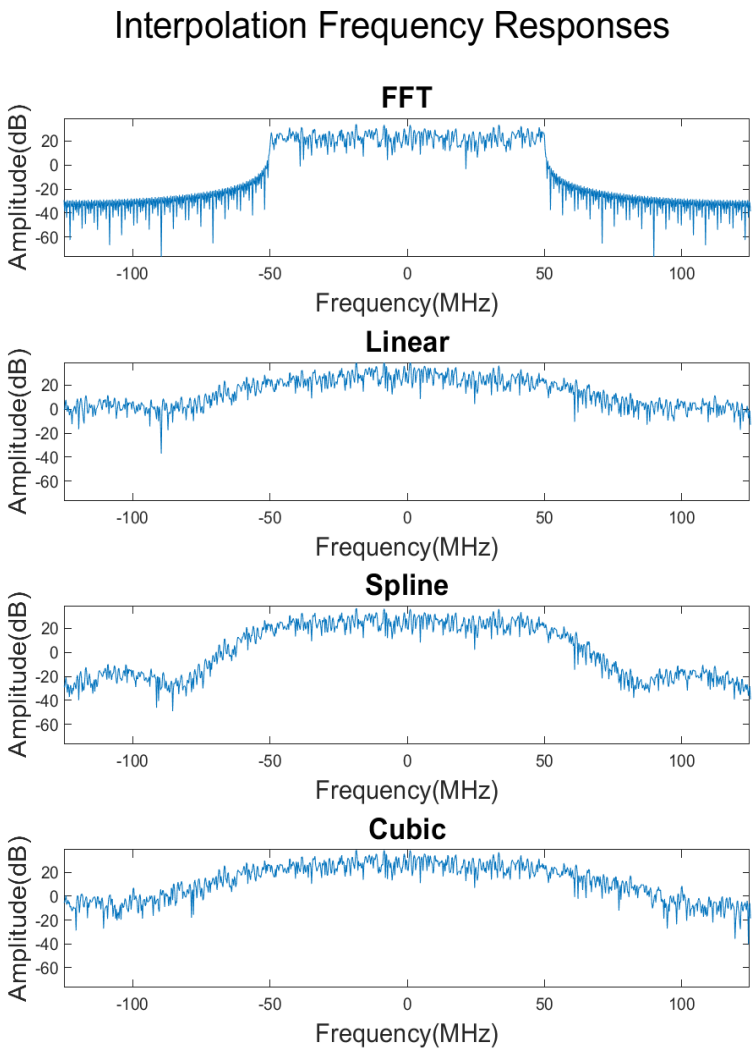

Figure 3.4. Bandwidth of signal after interpolation by labeled method.

Figure 3.4 shows the waveform's frequency spectrum after each interpolation method where the FFT-based method results in the best frequency containment and spline interpolation possesses the next best containment.

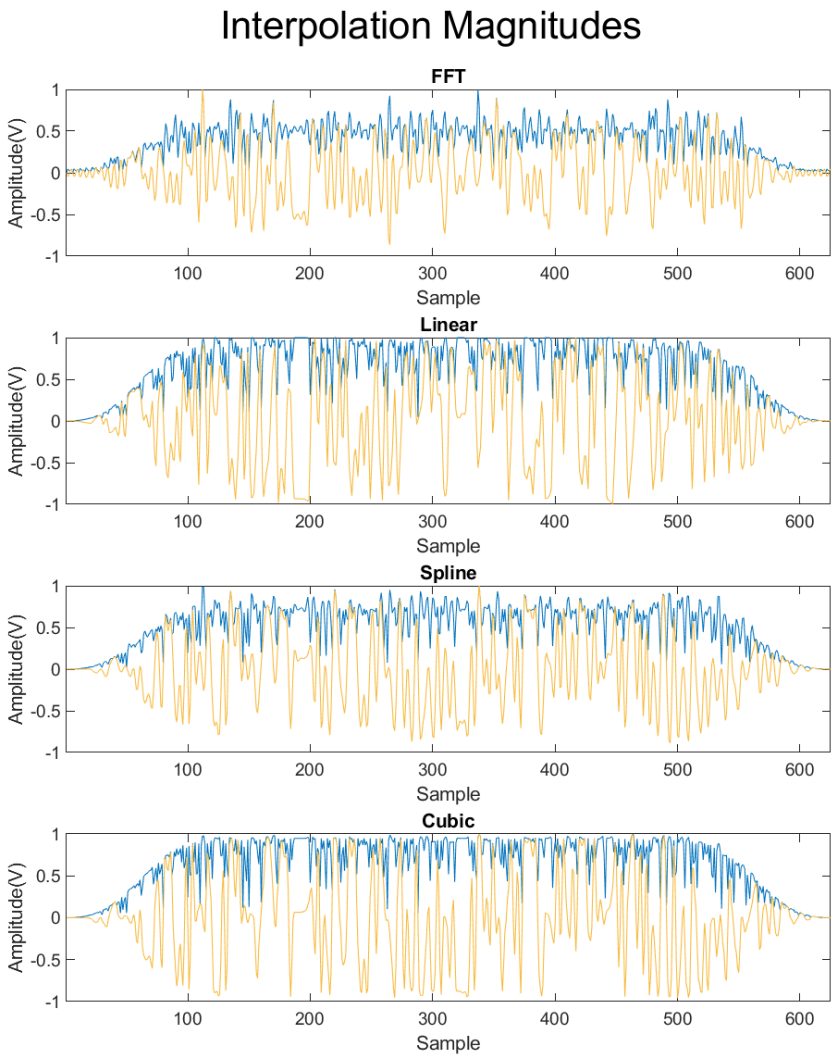

Figure 3.5. Amplitude of signal after interpolation by labeled method. The real values are yellow, and blue represents the magnitude of the IQ waveform.

Figure 3.5 shows the time-domain amplitude of the interpolated signals after their peak amplitude is scaled to one. The waveforms have smooth tapers at the beginning and end to model finite rise and fall rates. From the figure, the linear and cubic methods clearly exhibit the best time-domain characteristics. The amplitude of the FFT and spine methods are lower on average and contain less energy.

Table 1. Comparison of Interpolation Methods RMS Power

	RMS Power	*Percent RMS/Ideal*	*Auto-Cross Diff.*	*Matched Gain*
FFT	0.470	54.07	103.4	138.3
Linear	0.706	81.14	253.7	311.5
Spline	0.571	65.61	158.5	203.6
Cubic	0.723	83.06	268.3	326.4
Pre-Interp.	0.870	100.0	163.4	189.3

Table 1 shows the RMS power, the RMS power divided by its ideal value, the difference between the maximum value of the auto- and cross-correlations, and the maximum value of the auto-correlation for the waveforms shown in Figure 3.5 as well as the pre-interpolation sequence.

While FFT-based interpolation is the best at preserving frequency characteristics, linear and cubic interpolation are better at maintaining the time-domain characteristics, and the spline and cubic methods produce better compromises between time- and frequency-domain performance. Because the average power and difference between the auto-correlation and cross-correlation are paramount for MIMO systems, cubic interpolation was chosen for implementation in the main simulation because of its superior performance in these categories and acceptable preservation of the frequency spectra.



Figure 3.6 Chebyshev generation and conditioning flowchart

The flowchart of Figure 3.6 overviews the process used to generate the waveforms used in this research analysis. These waveforms result from an I/Q combination of two Chebyshev sequences seeded with different initial values, with unit scaling, and bandwidth/sample rate obtained through cubic interpolation. Furthermore, these waveforms have been shown to have low cross-correlation, even after interpolation, and that an arbitrary number of waveforms can be generated. These waveforms are used in cryptology because they are difficult to decipher without knowledge of the initial values used for their generation. The complexity of decoding the sequences is useful for radar waveforms because they will appear as noise when the generation seed is unknown, and the generation seed can change on each CPI to further obstruct prediction. This apparent unpredictability reduces the information extractable by non-friendly interceptors, while still allowing use within mesh radar systems if the initial values are communicated after generation or if some schedule of initial values is chosen prior to transmission.

Figure 3.7 includes a comparison of sample instantiations of Chebyshev waveforms before and after the modification process. The real values of the sequences are shown in yellow, and the magnitudes are shown in blue. Because the unmodified waveform has only real values, the entire sequence is represented by the yellow plot. The magnitude of the modified waveform is closer to the shape of the ideal complex envelope with tapering. The closer the waveform amplitude is to this ideal envelope, the more energy it will transfer. Because the unmodified Chebyshev waveforms appear as a random distribution of numbers between negative and positive

one and the unmodified sequence makes no efforts to operate in saturation, the total energy of the unmodified sequence has lower average amplitude.

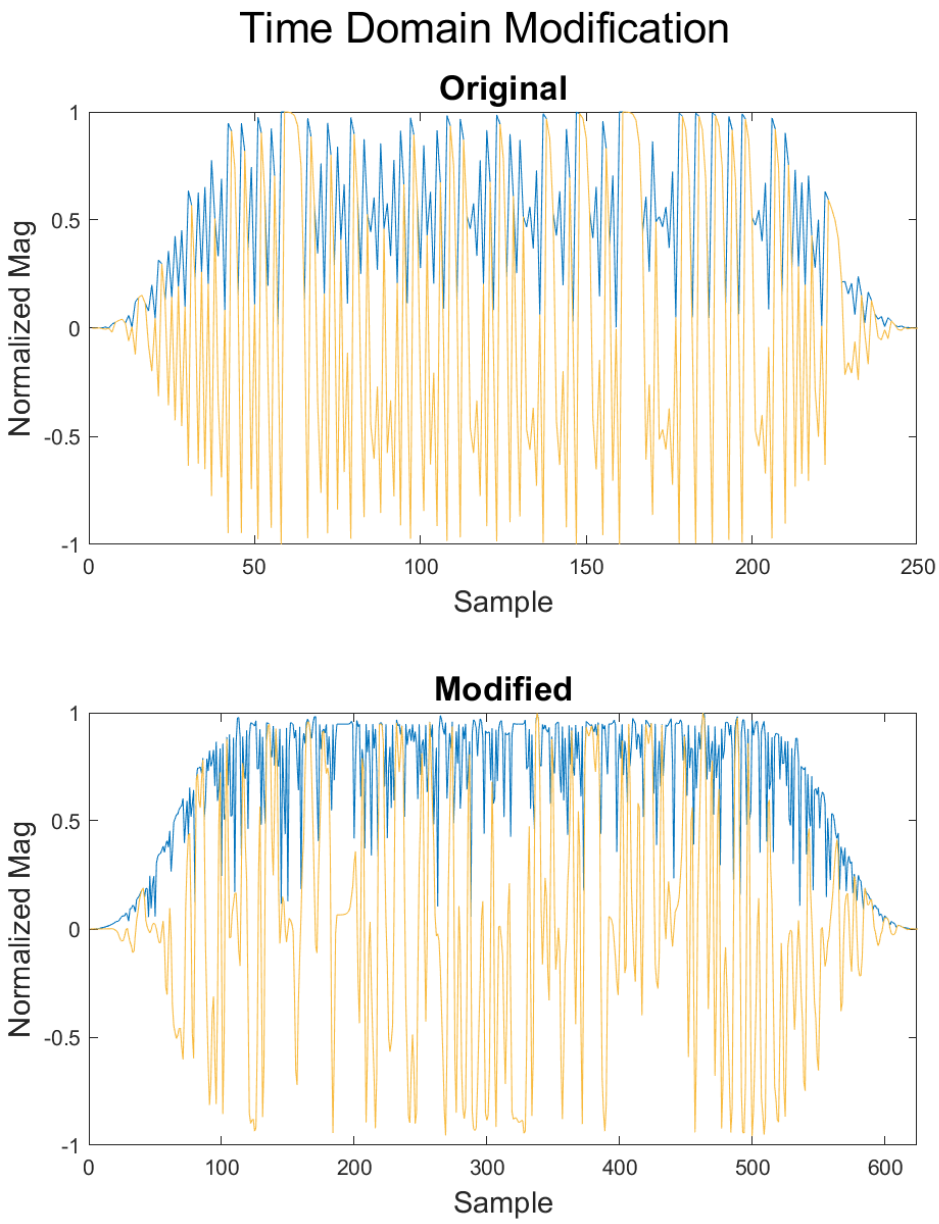

Figure 3.7 Qualitative comparison of waveform in time-domain before (top) and after (bottom) modification. The real values are yellow, and blue represents the magnitude of the waveform.

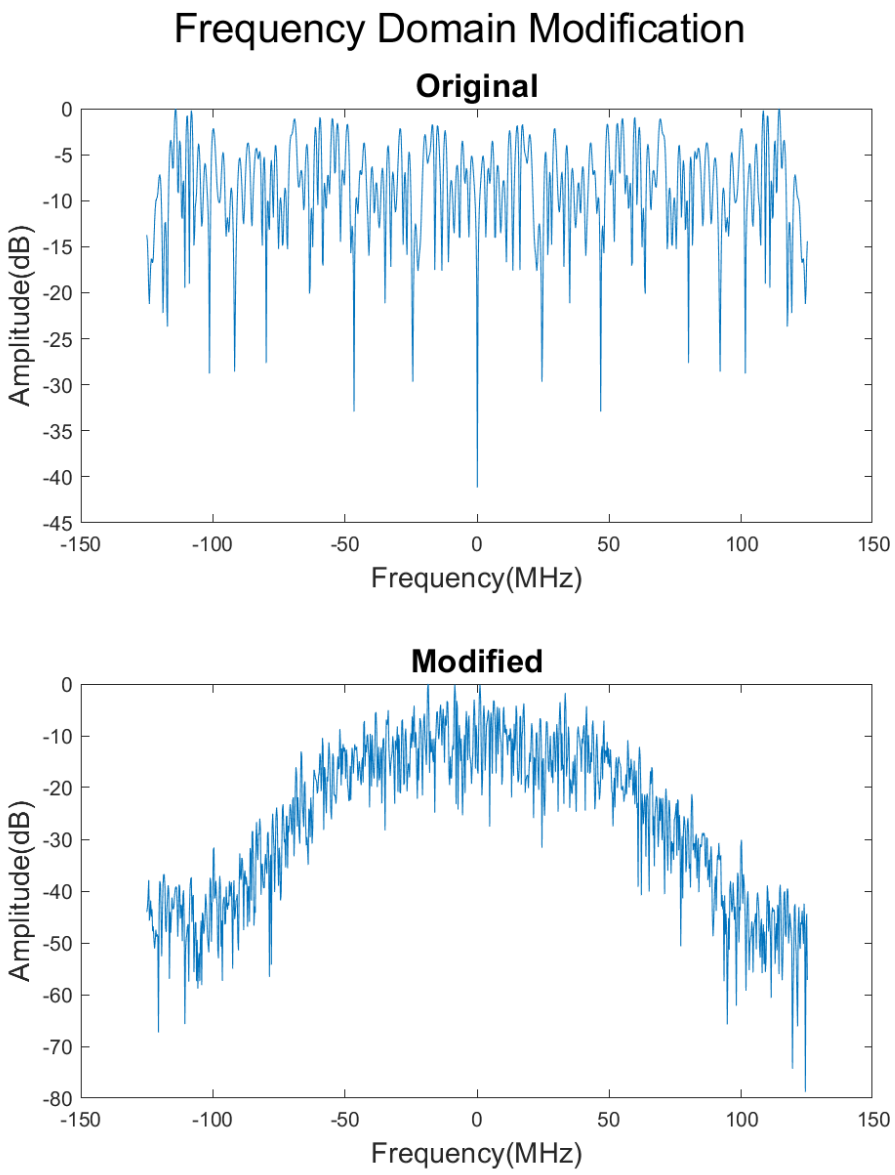

Figure 3.8 Qualitative comparison of waveform in frequency-domain before (top) and after (bottom) modification.

Figure 3.8 includes a comparison of the frequency components of the waveforms shown in Figure 3.7. The unmodified waveform has frequency limited only by the sample rate and exhibits a flat spread of its energy. The modified waveform was designed for a 100Mhz bandwidth and has less energy outside of its bandwith. The shape of the frequency spectrum of the modified waveform is comparable to the cubic

example included in Figure 3.4. The application of bandpass filters after waveform modification was not considered in this study. However, bandpass filters may provide better spectral containment while possibly decreasing the auto- and cross-correlation properties of the generated waveforms.

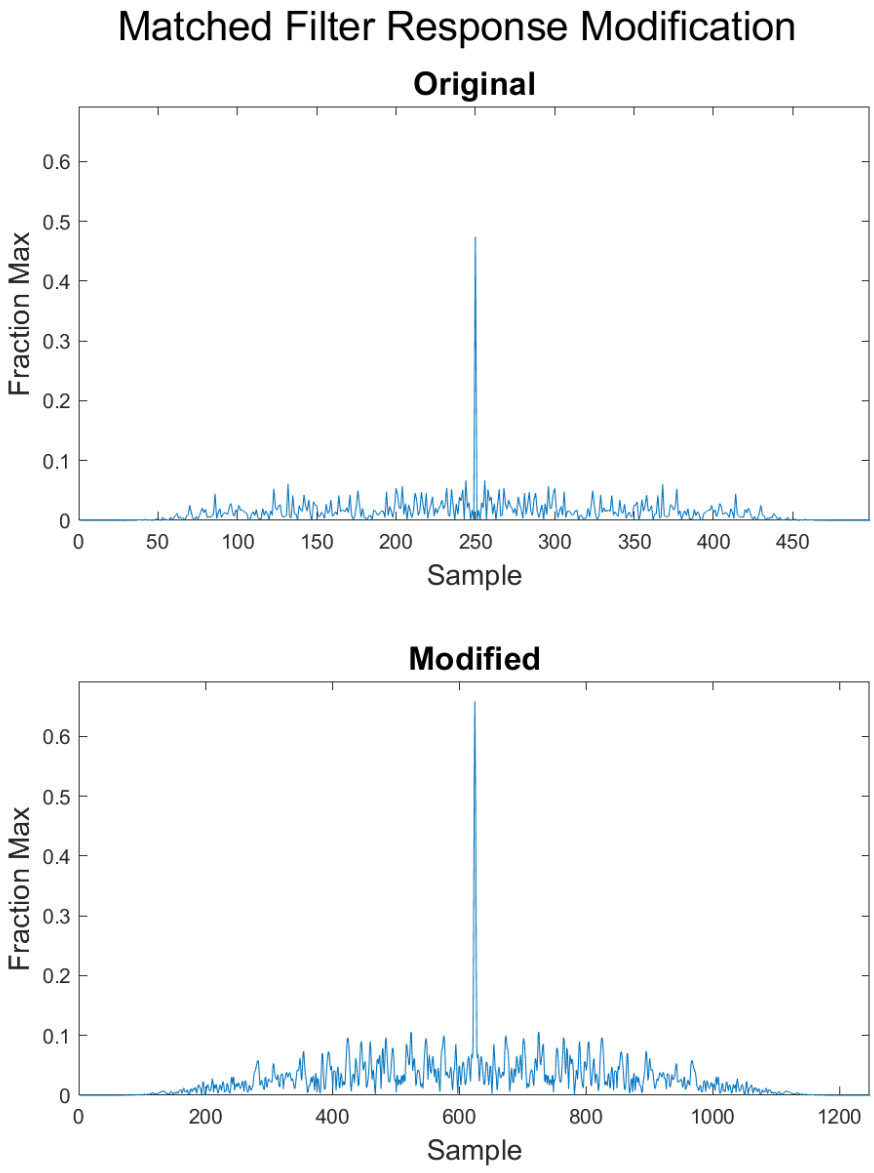

Figure 3.9 Qualitative comparison of waveform matched filter response before (top) and after (bottom) modification.

Figure 3.9 compares the response of the matched filters for sample instantiations of the modified and unmodified waveforms. The magnitude is normalized for the maximum value of the waveforms influenced by the complex envelope and number of samples. This normalization is produced by dividing the matched filter responses by the energy of the ideal complex envelopes resulting from the specific taper shape and sequence length. The peak of the matched response for the modified waveform is closer to the ideal fraction of 1 than its unmodified counterpart. The IQ components of the modified waveform bring the amplitude closer to the complex envelope, capturing more of the available energy as in the discussion of Figure 3.7.

3.3 Beampattern Synthesis

Now that we have established the transmit waveform, we can discuss its nonideal function within MIMO operation, especially when the MIMO beams are formed in multiple directions. Furthermore, because this functionality utilizes phased-array beamforming within the phased-MIMO continuum, it is important to understand the beampattern for traditional phased-array radars and how these concepts transfer with multiple beams.

Transmit beamsteering can be performed by selecting a direction for transmission and determining the appropriate element weightings as described in Section (2.2). If we assume the direction of the qth targets from the radar results in a wavenumber vector of $\overline{K}_q$, then the gain of each target due to the transmitting array is

$$G_l(\overline{K}_q) = \sum_{i=1}^{N_l} e^{-j\overline{K}_{aim}\cdot r_i} e^{j\overline{K}_q\cdot r_i} = \sum_{i=1}^{N_l} e^{j(\overline{K}_q-\overline{K}_{aim})\cdot r_i}, \tag{3.5}$$

where l is the waveform index, i is the element index within each waveform, and N_l is the number of transmit elements for each subarray. The transmit gain G_l is the response of each target q to each transmit beam l, and the maximum array gain of N_l can be observed when $\overline{K}_q = \overline{K}_{aim}$. To evaluate the beampattern for a transmit array, a vector of spatial frequencies $\overline{K}_q$ can be created and evaluated against a single steering direction $\overline{K}_{aim}$.

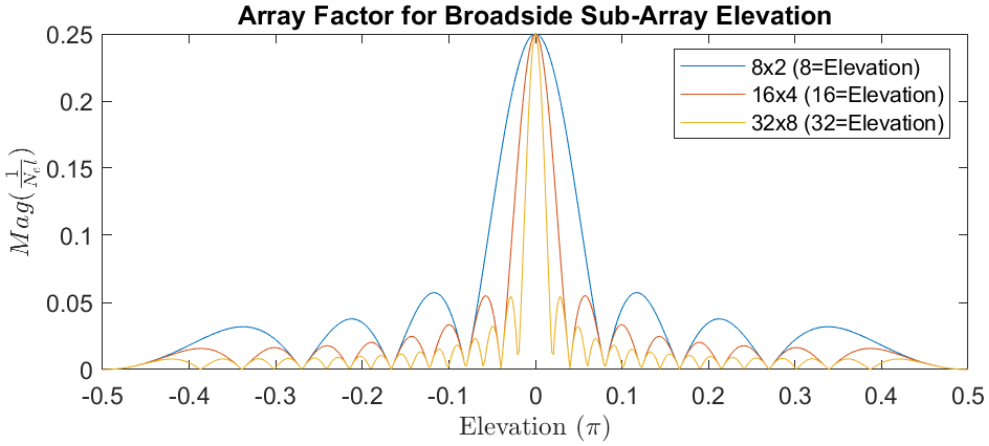

Figure 3.10: Normalized array factors for rectangular, traditional phased-array radars with dimensions described in legend

Figure 3.10 shows the normalized beampattern for a series of full-aperture phased-array radars with rectangular element distribution defined in the legend with x-axis measured in π radians. Figure 3.10 was generated by steering K_{aim} at broadside, and then evaluating a vector of targets generated at equally spaced elevation angles and zero azimuth. The magnitude is normalized to allow shape comparison independent from the changing transmit element count.

All-digital architectures with individual control of the transmit elements allow for the transmission of multiple waveforms with different aiming directions. This flexibility in waveform generation allows the full-aperture to be divided into a set of

smaller phased-arrays with reduced power. A sample set of the elements of such sub-arrays are displayed in Figure 3.11 where each sub-array is shown in a different color.

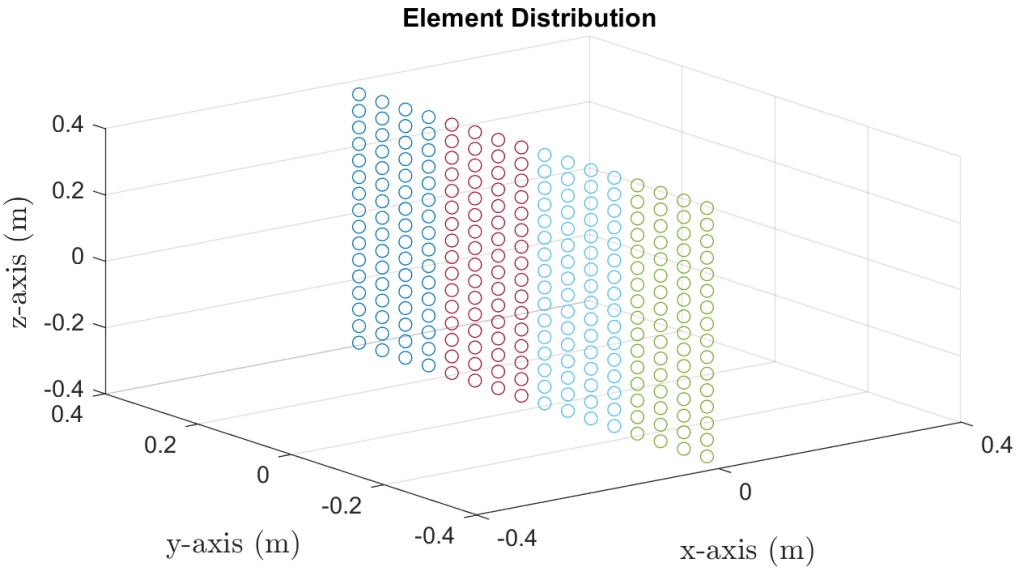

Figure 3.11: Element distribution for simulated sub-arrays

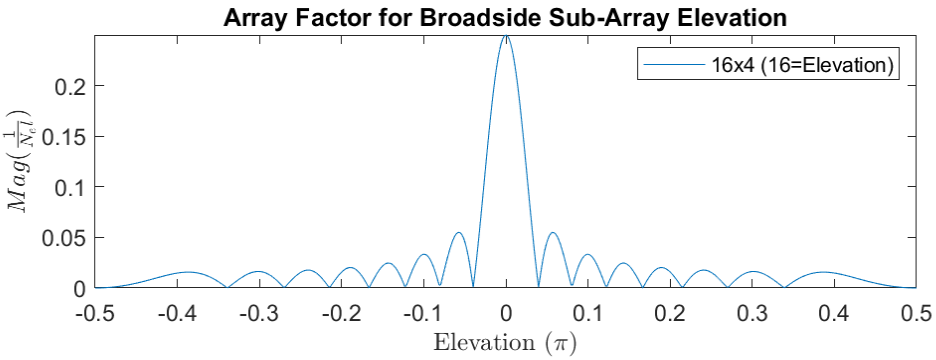

Figure 3.12: Array pattern for one sub-array aimed at broadside in elevation dimension

The broadside beampattern in the elevation dimension for one of these sub-arrays is shown in Figure 3.12. As we mentioned before, the all-digital architecture allows us to choose different aim directions for each sub-array. If we aim the beams of these sub-arrays in different directions, we will produce a different beam-pattern for each transmit waveform and steering direction. To better understand this, we can calculate each steered beam-pattern and overlap them as in Figure 3.13.

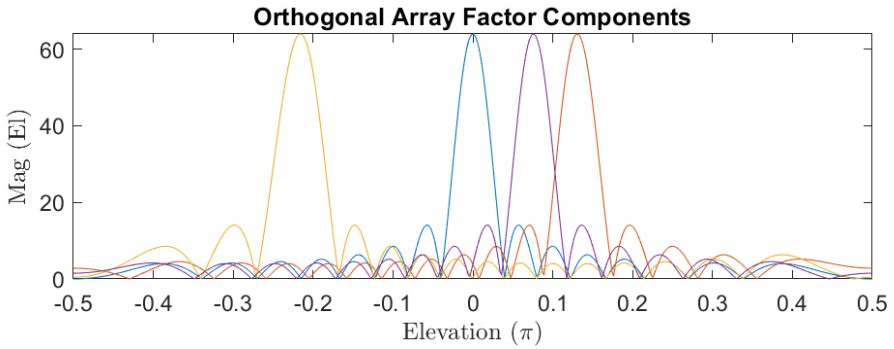

Figure 3.13: Beampatterns of four sub-arrays aimed in different directions in elevation dimension

Displaying the array patterns for each sub-array in this fashion assumes each waveform is orthogonal and produces no cross-correlation in the other beampatterns. However, as we have previously discussed, it is impossible for multiple waveforms to be perfectly orthogonal, especially when Doppler shifts and other imperfections are added, which will be considered later within this section. If we continue with the orthogonal assumption and coherently add the beampatterns of Figure 3.13 to form an effective beampattern for the full transmit array, we produce Figure 3.14.

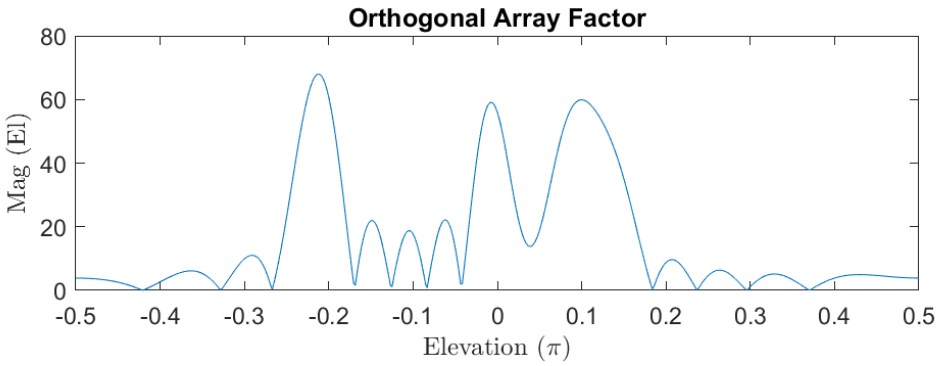

Figure 3.14: Coherent combination of four beampatterns of Figure 3.13 in elevation dimension

When the beampatterns are combined, constructive and destructive interference is introduced. In this example, the mainlobes are generally increased in the combination, but their amplitudes will decrease in other cases. Additionally, there is strong constructive interference at .15 π radians that could cause targets or interference

sources in this direction to be amplified in an undesirable way. To better compare the amplitudes before and after addition, Figure 3.15 was created.

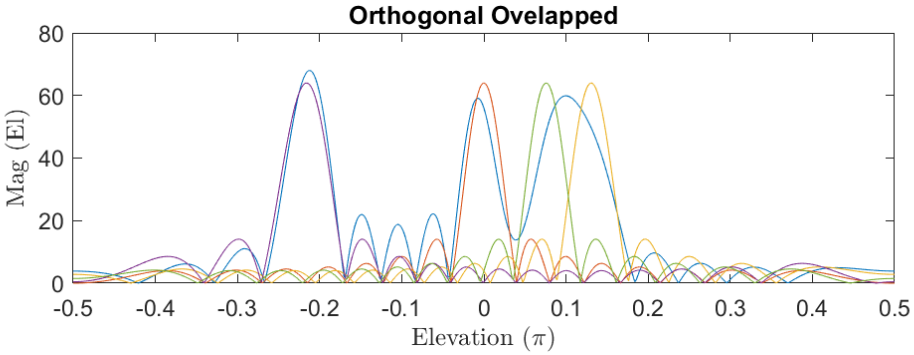

Figure 3.15: Beampatterns of four sub-arrays aimed in different directions in elevation dimension and their coherent combination

The orthogonal assumption is useful for comparison and theoretical observations, but realizable beampatterns will differ from those observed in Figure 3.13 through Figure 3.15. To better model achievable results, a simple cross-correlation model was employed where Chebyshev waveforms are generated for each transmit sub-array and superimposed to model the received return resulting from all sub-arrays. Matched filtering is applied separately for each transmitted waveform to separate the energy resulting from each transmit subarray. The maximum value of the output of each matched filter was then chosen as the array gain for each waveform. This process was repeated for each evaluation angle individually. The array factors determined for each transmit angle and influenced by the steering angle were the only gain applied to each waveform. That is, the effects of signal propagation loss, element pattern responses, and target-specific parameters were omitted in order to isolate the effects of cross-correlation based on transmit amplitude controlled by array factor. This process of array factor generation is shown in Figure 3.16.

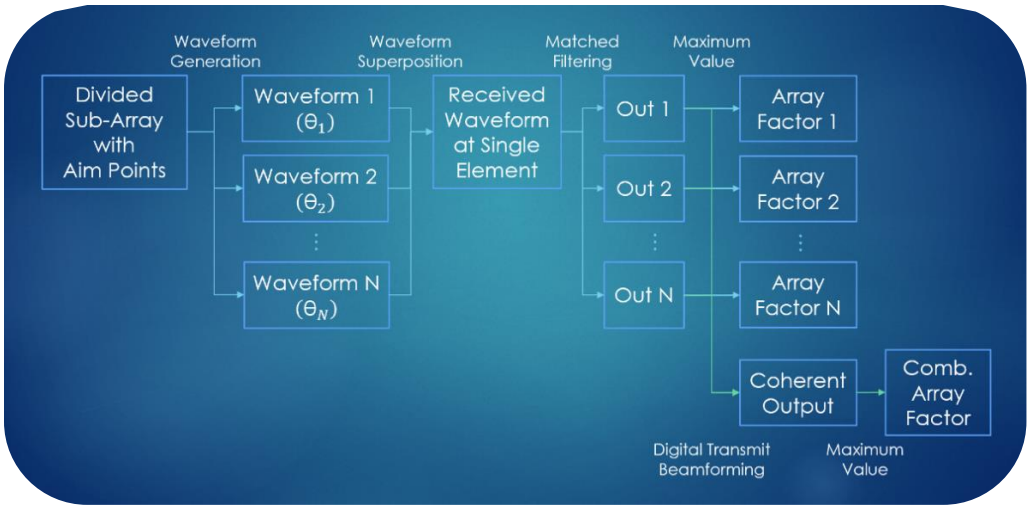

Figure 3.16. Flowchart for array factor generation

It is important to note that Figure 3.16 includes the generation of the effective beampattern in the bottom branch. This is the apparent beampattern for the full array with non-orthogonal, Chebyshev waveforms. Figure 3.17 shows the results with the aiming directions of Figure 3.13 through Figure 3.15 and non-zero cross-correlation.

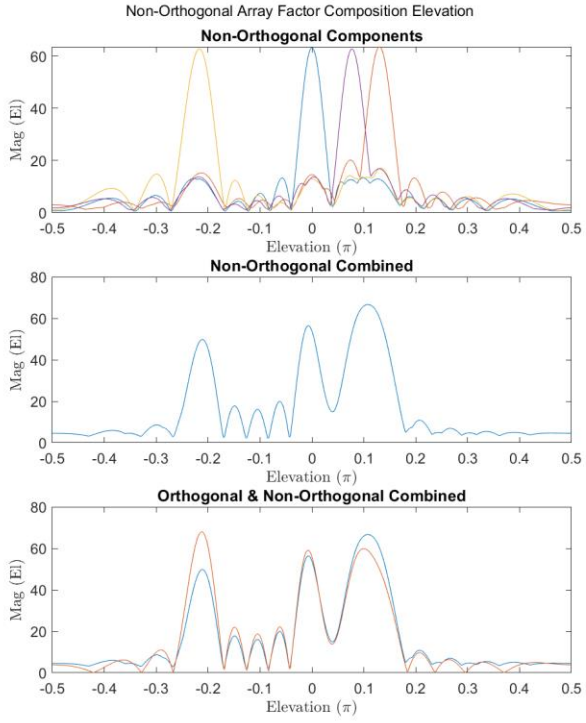

Figure 3.17. Beampatterns and combined patterns for phased-MIMO radar with non-zero cross-correlation and different aiming vectors

The main change caused by the cross-correlation in the separate beampatterns is a cross-correlation floor that raises the minimum value of each beampattern. The cross-correlation is strongest in the directions corresponding to the mainlobes in other waveforms. The cross-correlation covers the sidelobes in these directions and could overpower the mainlobes with the right combination of waveform cross-correlation properties and subarray size. The differences introduced by the lack of orthogonality become more noticeable in the full array beampattern, as can be observed in the second and third subplots. To better visualize the cross-correlation, Figure 3.18 was created with an isolated cross-correlation floor. This cross-correlation floor was generated by predicting the cross-correlation shown in Figure 3.17. Though the cross-correlation floor is introduced conceptually here, the mathematics, and method for its prediction, will be discussed later.

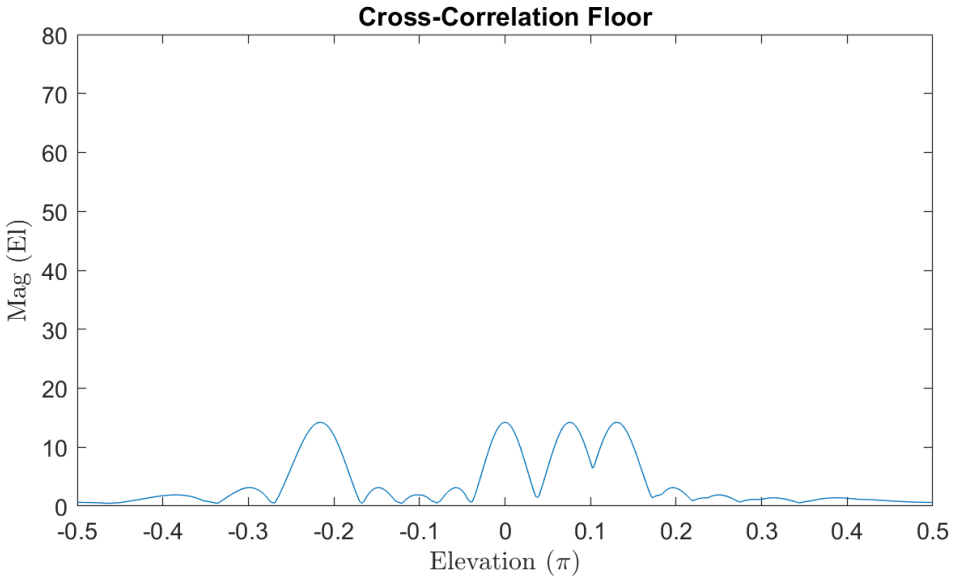

Figure 3.18 Cross-Correlation Floor

The cross-correlation is comparable to the minimum values shown in the first subplot of Figure 3.17. This indicates we have some ability to predict the cross-correlation.

Different waveforms will reduce or increase the magnitude of the cross-correlation floor based on their cross-correlation properties.

Because the maximum value of the matched filter was used instead of a more practical detection method, it is important to measure the bias introduced by the simplification. Three measurements were made to evaluate the error introduced by these assumptions. The first measurement is the difference between the maximum value of each matched filter output and its average value and is displayed in the first subplot of Figure 3.19. We expect this difference to closely match the shape of the 'Non-Orthogonal Components' plot, which would indicate the maximum value is an acceptable simplification. The second measurement is the average value of the matched filter response, which we refer to as the mean sidelobe level, and is included in the second subplot. We expect this to remain lower than the peak value by a large margin to indicate the peak value is not due to interference. The final measurement is the mean sidelobe fraction of the mainlobe included in the third subplot. We expect this ratio to remain low when the response is from the waveform of interest because the mainlobe should be much larger than the sidelobes. This value should be higher where the sidelobes are close in amplitude to the mainlobe, which should correspond to high cross-correlation.

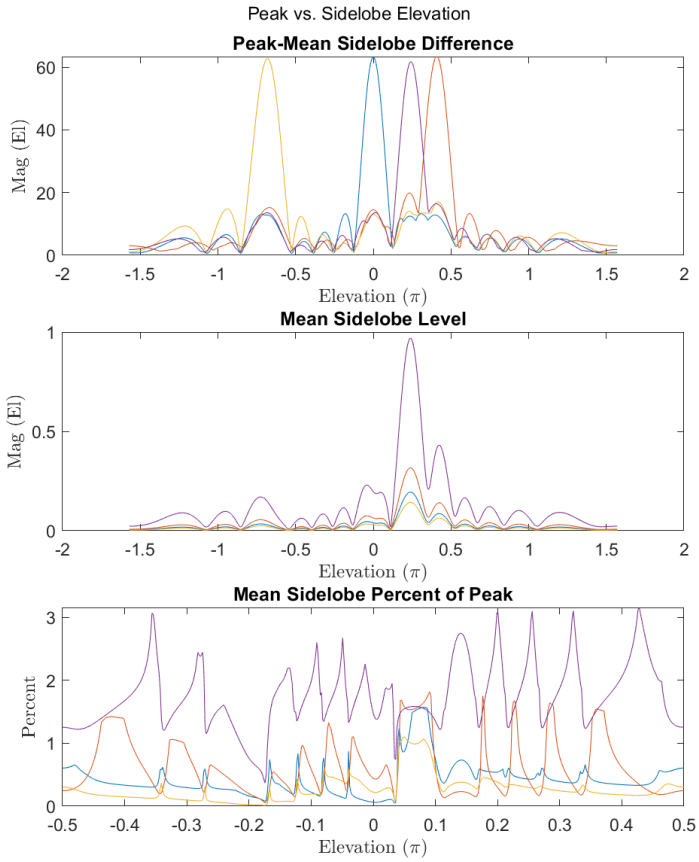

Figure 3.19. Comparisons of peak and sidelobe levels of matched filter output in Elevation.

The first subplot in Figure 3.19 maps the difference and closely matches the output after the maximum is chosen, showing that the peak value is adequately greater than the mean sidelobe value. The second subplot shows that the mean sidelobe level is less than one, which is under 2% of the peak. The third subplot shows that the mean sidelobe level scales with the peak value except where other waveforms are strongest. These all indicate that the maximum simplification is acceptable.

To this point, all sub-arrays have been the same size and shape, but they may vary in real all-digital systems. If one sub-array is much larger than the others, the maximum gain of its beam is proportionally larger and will produce higher cross-

correlation in other matched filters. If the size disparity is large enough, the return from the smaller sub-arrays can be covered by the cross-correlation of stronger arrays. Figure 3.20 shows a case in which one waveform is much stronger than the others, and its return dominates the outputs of other matched filters. This figure was generated with the same sub-array dimensions shown in Figure 3.11 but with stronger element power applied through scalar multiplication of each waveform by 1, 4, 9, and 16 times the original energy. The orthogonal beampatterns, beampatterns with cross-correlation, and their differences are shown in Figure 3.20.

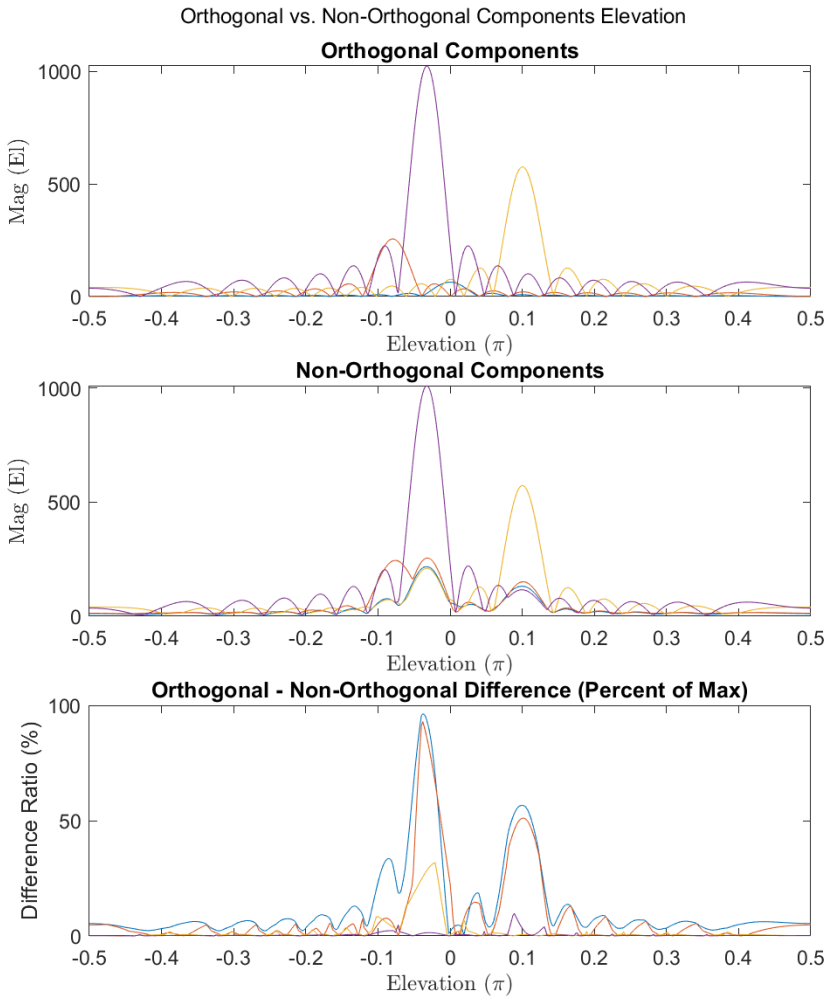

Figure 3.20. Matched filter results with one waveform much larger than all other waveforms.

The blue beampattern corresponds to the unity scaled array and is difficult to distinguish from the signals with stronger scaling. The purple waveform received the largest scale factor and is the easiest to observe. In the second subplot, the cross-correlation from the purple waveform overpowers all other returns in the direction of its mainlobe. A similar, but reduced effect is observed in the cross-correlation produced by the yellow waveform. Although this situation is problematic, there is still hope because we can predict the situations in which one sub-array dominates the others or when a new sub-array will not have enough energy to overcome the cross-correlation of the pre-existing sub-arrays.

3.3.1 Cross-Correlation Prediction

If the transmitted signal for waveform m before amplitude scaling is $(x_m[n])$, and its transmission configuration produces a gain of $(G_m[\overline{K}_\theta])$ in the direction of interest $\overline{K}_\theta$, then the transmitted waveform $x_{T,m}$ is

$$x_{T,m} = G_m[\overline{K}_\theta]x_m[n], \tag{3.6}$$

and its matched filter is the waveform's time-reversed conjugate transmit scaling as

$$h_m = x_m[-n]^*. \tag{3.7}$$

The auto-correlation is the convolution of the waveform with its own matched filter and can be expressed as

$$A_{m,m}[n, \overline{K}_\theta] = G_m[\overline{K}_\theta]x_m[n] * h_m, \tag{3.8}$$

and the cross-correlation for this matched filter resulting from waveforms $l \neq m$ is

$$A_{m,l}[n, \overline{K}_\theta] = G_l[\overline{K}_\theta]x_l[n] * h_m. \tag{3.9}$$

The total cross-correlation energy of this matched filter is the maximum cross-correlation from all other waveforms that are not *m* represented as $\bar{m}$ is

$$A_{m,\bar{m}}[n,\bar{K}_\theta] = \underset{l\neq m}{max}\left(A_{m,l}[n,\bar{K}_\theta]\right) = \underset{l\neq m}{max}(G_l[\bar{K}_\theta]x_l[n] * h_m). \quad (3.10)$$

The auto-correlation of the signal at the point of interest will be indistinguishable from the cross-correlation when the cross-correlation energy is some fraction η of the maximum gain in the direction of interest expressed as

$$A_{m,m}[n,\bar{K}_\theta] > \eta A_{m,\bar{m}}[n,\bar{K}_\theta] = \eta \underset{l\neq m}{max}(G_l[\theta]x_l[n] * h_m). \quad (3.11)$$

The processing required to calculate the cross-correlation for every-time sample, especially with additional waveforms or time samples, is often impractical. If the transmitted waveforms are from the same family or class with an average cross-correlation approximated as a scalar fraction of the waveform energy $\mathrm{E_l[n]}$, such as Chebyshev waveforms, then we can represent the cross-correlation as a fraction of the peak to mean ratio or α_{PMR}. The gain of each transmitting sub-array $G_l[\bar{K}_\theta]$ can then be combined with α_{PMR} and $\mathrm{E_l[n]}$ to estimate the average cross-correlation energy received by the matched filters of the other waveforms. This can be used to modify (3.7) to form

$$A_{m,l}[n,\bar{K}_\theta] = \alpha_{PMR} G_l[\bar{K}_\theta]\mathrm{E_l[n]}. \quad (3.12)$$

The combined cross-correlation received at each matched filter is then

$$A_{m,\bar{m}}[n,\bar{K}_\theta] = \alpha_{PMR} \underset{l\neq m}{max}(G_l[\bar{K}_\theta]\mathrm{E_l[n]}). \quad (3.13)$$

The ideal auto-correlation value is the transmit waveform energy, and we can substitute this value for $A_{m,m}$ from (3.8) and form

$$A_{m,m}[n, \overline{K}_\theta] = G_m[\overline{K}_\theta]\mathrm{E}_m[\mathrm{n}], \tag{3.14}$$

and we can modify (3.10) with (3.12) forming

$$G_m[\overline{K}_\theta] > \frac{\eta\alpha_{PMR}\underset{l\neq m}{max}(G_l[\overline{K}_\theta]\mathrm{E}_\mathrm{l}[\mathrm{n}])}{\mathrm{E}_\mathrm{m}[\mathrm{n}]}. \tag{3.15}$$

We can observe that the target return from waveform m should be distinguishable from the cross-correlation when the gain of waveform m is less than some ratio of the maximum transmit energy of all other waveforms in direction $\overline{K}_\theta$. This is useful for predicting and preventing configurations in which the signal from one waveform will cover the returns of the other waveforms. It is important to note that real targets will fluctuate in radar cross-section, RCS, range, Doppler, and direction adding a target dependence to the gain, but this will be considered in later sections.

The beampattern can be predicted with the cross-correlation and the auto-correlation equations. When we assume the beampattern in an angle can be extracted from the matched filter response with a maximum function, we are assuming the beampattern $B(\overline{K}_\theta)$ without cross-correlation is

$$B(\overline{K}_\theta) = \max_{n\in N}\left(A_{m,m}[n, \overline{K}_\theta]\right). \tag{3.16}$$

With the addition of cross-correlation effects, we find the maximum value over all n corresponding to each $\overline{K}_\theta$ and create

$$B(\overline{K}_\theta) = \max_{n\in N}\left(A_{m,m}[n, \overline{K}_\theta] \cap A_{m,\overline{m}}[n, \overline{K}_\theta]\right). \tag{3.17}$$

We can then use (3.14) and (3.12) to form

$$B(\bar{K}_\theta) = \max_{n \in N} \left(G_m[\bar{K}_\theta]\mathrm{E}_m[\mathrm{n}] \cap \alpha_{PMR} \max_{l \neq m} (G_l[\bar{K}_\theta]\mathrm{E}_l[\mathrm{n}]) \right) \tag{3.18}$$

Figure 3.21 was created to visualize the prediction of the beampatterns using (3.18) against the results of the simulation.

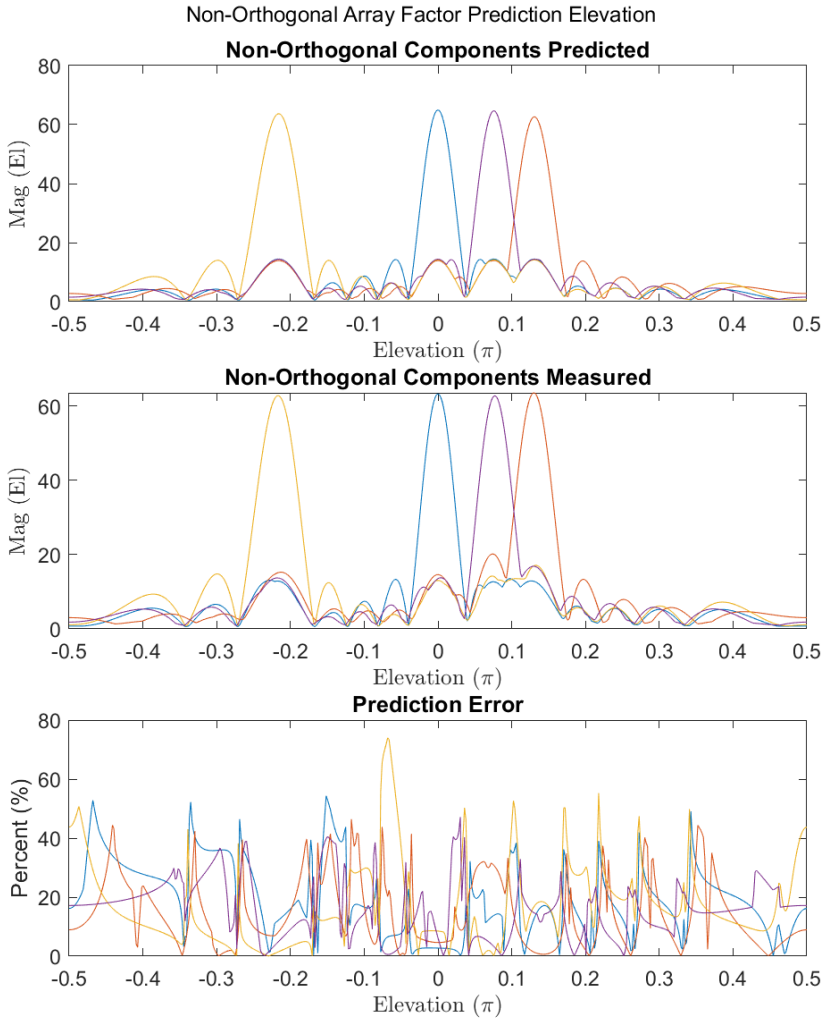

Figure 3.21. Matched filter with cross-correlation prediction and simulated with error measurement.

The first subplot includes the cross-correlation prediction, the second subplot shows the simulation results, and the third subplot shows the error between the predicted and actual values. Although the error varies with transmission angle, it tends to be close to

or under 50%. This is useful for predicting the cases in which the cross-correlation will cover the mainbeam of other returns. The cross-correlation error is lowest in the mainlobes of each beampattern, because the mainlobe gain to interference ratio is higher in these places. Although this rough prediction does not predict the cross-correlation effects accurately enough for reliable correction, it is useful for predicting situations producing unacceptable amounts of interference. The output in some directions appears to result from isolated auto-correlation, but the cross-correlation may reduce the dynamic range between the desired signal and cross-correlation floor, producing low signal to interference and noise ratios (SINR) and rendering the returns ineffective. The next section describes the simulation environment created for expanded examination of cross-correlation and other limitations of real systems.

4 SIMULATION

A simulation environment was created with MATLAB to model an all-digital phased-array system. Although the simulation can be used to model various frequency bands, aperture sizes, waveforms, and other system characteristics, we have tested it using parameters and architectures that mimic an all-digital S-band system being developed at the University of Oklahoma. Targets are treated as point scatterers with randomly generated parameters such as Cartesian coordinates, velocity, acceleration, and radar cross sections (RCS).

4.1 Simulation Overview

One of the important features of a digital array is that every element or sub-array possesses its own DAC, which means that each element or sub-array can transmit its own unique waveform. In practice, this feature may be limited by software control, communication bandwidth across the array, mutual coupling considerations, or other factors. The implemented simulation allows arbitrary, rectangular groups of elements to be organized into transmit sub-arrays, with the same waveform transmitted within each transmit sub-array and unique waveforms for each sub-array. Each sub-array may also possess its own unique steering angle. The array factor and antenna gain are calculated for each sub-array size and shape, and the transmitted signal is scaled according to a target's location within the transmit beam.

The received signal is then modeled according to its propagation delay and amplitude as specified by the radar range equation. Delays that are not an integer multiple of the ADC sampling interval are interpolated before accumulating into the

overall radar measurement. The received signal is also multiplied by a relative phase term that models the number of wavelengths traveled to and from the target. For the transmit path, this distance is calculated from the phase center of each sub-array. On receive, the signal is modeled to every element in support of digital beamforming across the entire array; therefore, the phase term is modeled as being due to a propagation path to the center of the array, plus the standard array phase factor for every element that depends on the dot product between the propagation wavevector and the element's position vector. On subsequent pulses within a coherent processing interval (CPI), Doppler effects are modeled by a rotating pulse-to-pulse phase term that depends on the target's range rate. After each target return is properly scaled, the phase shifts due to transmit sub-array location, Doppler shift, and propagation to every element (on receive) can be applied, and the contributions due to multiple targets are superimposed.

In the simulation, new Chebyshev waveforms are created for each transmit sub-array and CPI, and a copy of each waveform is saved for matched filtering. The precise energy of each transmitted waveform fluctuates with variations in the waveform amplitude caused by the interpolation used for bandwidth control, the IQ modification, and waveform tapering that models realizable rise- and fall-times. Other waveforms can be implemented as well.

4.2 Radar Signal Processing

There are many potential processing architectures on receive with varying order of processing steps, number of full-aperture beams formed, and so on. Here, we perform Doppler processing and digital beamforming via FFT-based processing and

then perform matched filtering to isolate the signals due to the individual waveforms such that the fully-formed output datacube is a 5-D structure covering range, Doppler, two spatial frequencies, and transmitted waveform. Digital transmit beamforming can then be performed by a weighted sum of the 4-D datacubes for each waveform with a phase correction according to a relative phase induced by the transmit sub-array phase centers.

4.3 Noise & Interference Model

The limits on radar performance imposed by noise and interference have produced entire fields dedicated to the mitigation of these effects. In this simulation, complex Gaussian white noise is added to the signal to afford basic insight on the noise challenges of real systems. This noise is added to the 4-D data cube of the return signal before any signal processing has been performed, and is simulated with an average power equal to

$$P_n = k_B TBF. \tag{4.1}$$

where k_B is Boltzmann's constant, *T* is an operating temperature typically assumed to be room temperature (290 K), *B* is the radar ADC sampling rate, and *F* is an assumed operating noise figure of the receiver.

Although real systems will be subject to natural noise sources as well as potential electronic warfare systems, these effects and clutter were not modeled. The radar is assumed to be an air-facing radar that limits ground clutter effects, and it is assumed to operate in the absence of other external sources of interference. The main sources of interference in the simulation are the cross-correlation of waveforms,

sidelobes of the transmit beams, sidelobes from processing, thermal noise, and interpolation imperfections.

4.4 Element Pattern

The angular performance of phased-array radar is influenced by the previously discussed array pattern as well as the thus far omitted transmission patterns of individual elements. Precise simulations of phased-array systems require careful description and estimation of the element pattern. For our purposes, a simple sine model

$$P_{el} = \begin{cases} \sin\left(\frac{\pi}{2}\left(\frac{|K_x|}{K} + 1\right)\right) & x > 0 \\ 0 & else \end{cases}, \tag{4.2}$$

is used for the element pattern with power maximized at broadside and zero when perpendicular to broadside. This scaling can be visualized in Figure 4.1.

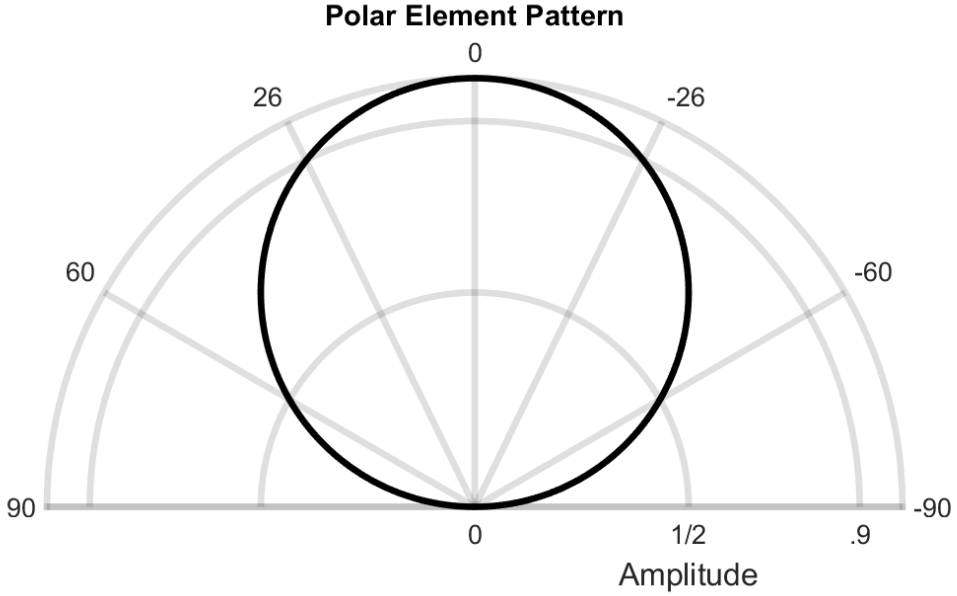

Figure 4.1. Element pattern in polar coordinates with amplitude as fraction of total power and angles in degrees.

Figure 4.1 includes a polar plot of the element pattern. The steering angle away from broadside is listed along each ray, and the fractional amplitude is included as the

radius. The power is maximized at broadside and decreases to zero as it approaches perpendicular (90 degrees) from broadside. The selection of element pattern for our purposes is not as important as the ability to closely model its gain. More realistic elements can be simulated if their element pattern is well defined mathematically or through sufficient empirical measurements.

4.5 Simulation Validation

The value of a simulation is built on its ability to accurately represent the system it models. This section presents a subset of the data and figures used to verify the integrity of the simulation and prove its fidelity. The verification was through the qualitative evaluation of the simulation results through 2-D maps of the processed datacube as well as quantitative proof through Monte-Carlo simulations of the probabilistic distributions.

4.5.1 Image Maps

the processed data is multi-dimensional, it is often visualized in 2-D slices. This section includes range-Doppler, range-angle, angle-Doppler, and angle-angle images created from these 2-D slices. Range-Doppler maps are imaging methods popular for pulse-Doppler radars and include target ranges measured against their range-rates. Figure 4.2 includes two sample range-Doppler maps with a single target spawned at a high elevation and near-zero azimuth, where zero elevation and azimuth correspond to broadside or the x-axis. The radar was simulated as an eight-by-eight phased-array with maximum element power of 2500 Watts and a single transmitted waveform. The element power was selected to allow simulation of a smaller radar

with adequate SNR at the selected target range and angles. This configuration was simulated without noise and then with noise for comparison. Range and velocity sidelobes can be observed to spread around the target along both axes. These sidelobes result from any filtering process and the differences between the signal and matched filter due to time delay and Doppler shift.

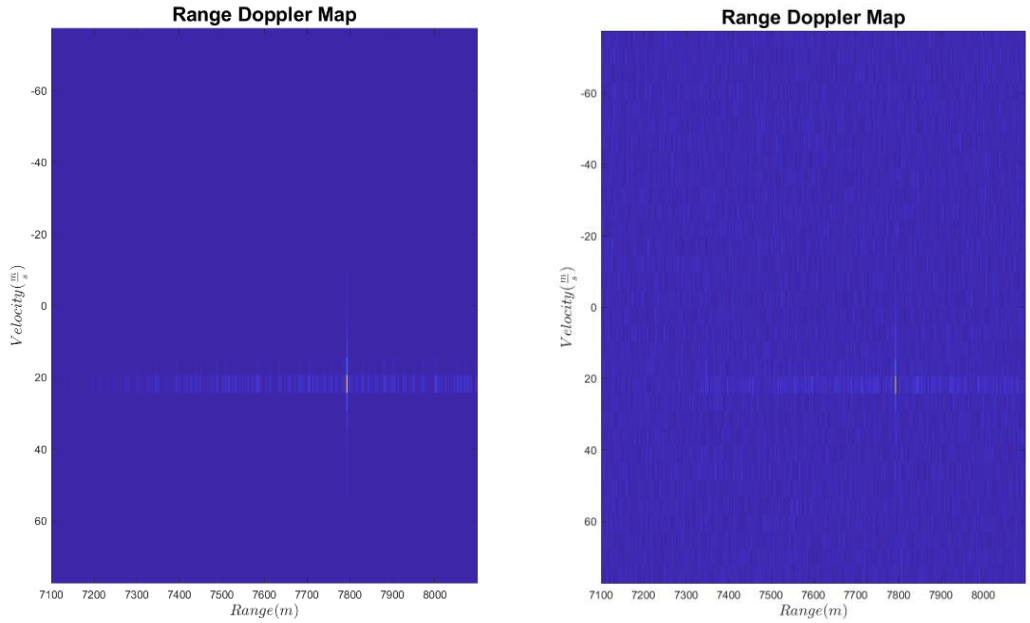

Figure 4.2. Range-Doppler map of traditional phased-array radar without noise (left) and with noise (right).

The addition of noise changes the shape of the sidelobes in both the range and velocity dimensions. To better illustrate this effect, Figure 4.3 was created by taking a slice of the range-Doppler map along the simulated ranges where the Doppler and spatial frequencies are maximized. In the cases with and without noise, the peak is at the same location, but the magnitudes change. The peak is reduced with the addition of noise but may increase or decrease in other scenarios. The sidelobes exhibit less structure due to the interference of the noise and are partially covered by it in the provided example.

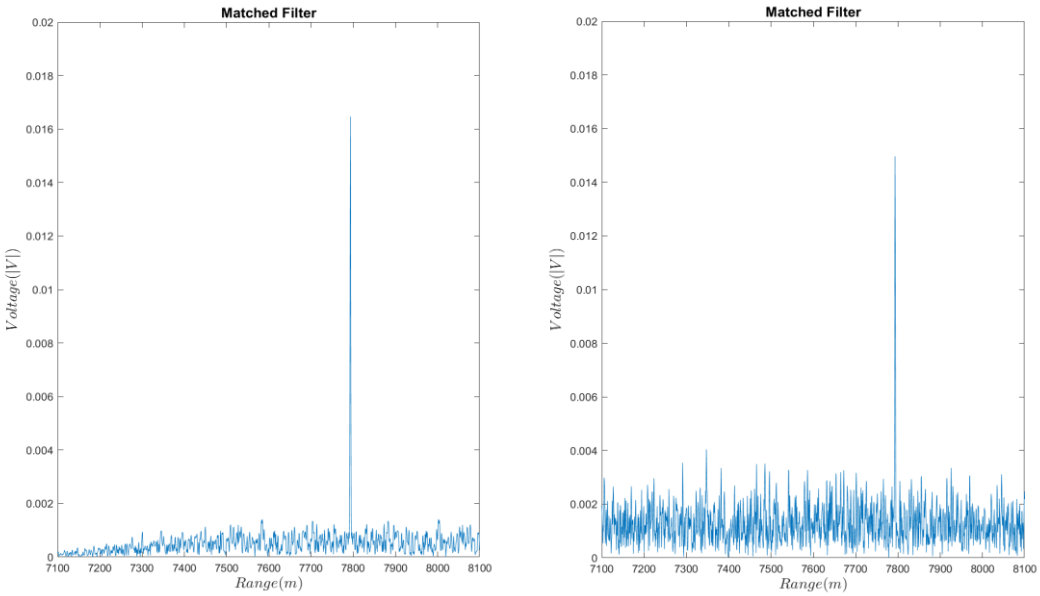

Figure 4.3. Range slice along maximum Doppler and spatial frequencies without noise (left) and with noise (right).

The set of ranges and velocities of Figure 4.2 show the structure of the returns across a wide span, but a closer view can provide a more succinct profile of the target returns. Figure 4.4 includes a zoomed-in section of Figure 4.2.

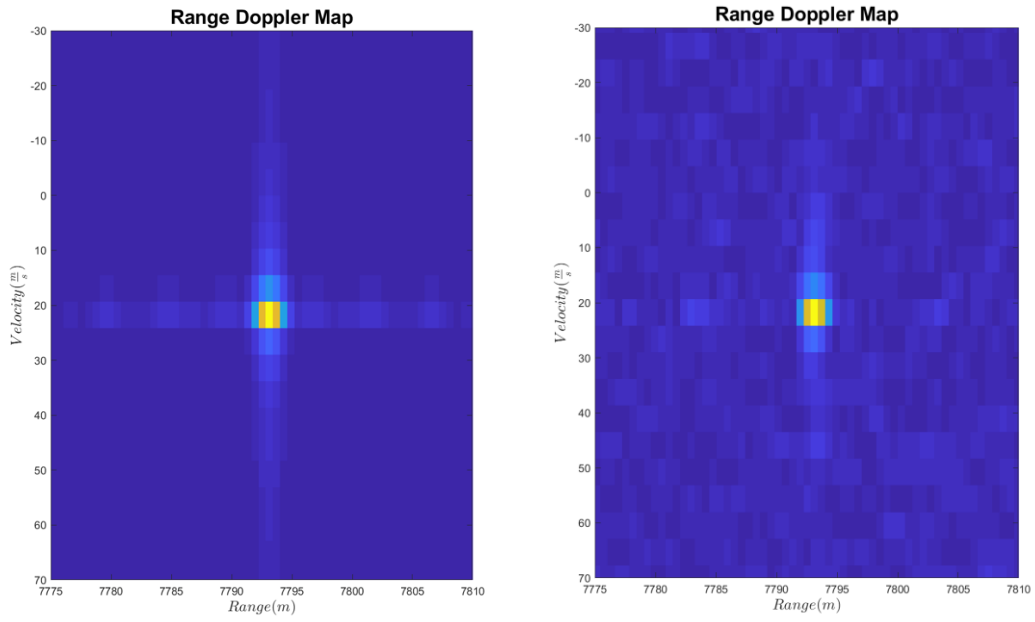

Figure 4.4. Zoomed Range-Doppler map of traditional phased-array radar without and with noise.

To demonstrate the simulation's ability to model phased-MIMO configurations, Figure 4.5 and Figure 4.6 were created with two sub-arrays transmitting different Chebyshev waveforms. Copies of the same noise used in the data of Figure 4.2 through Figure 4.4 were used to isolate random fluctuations.

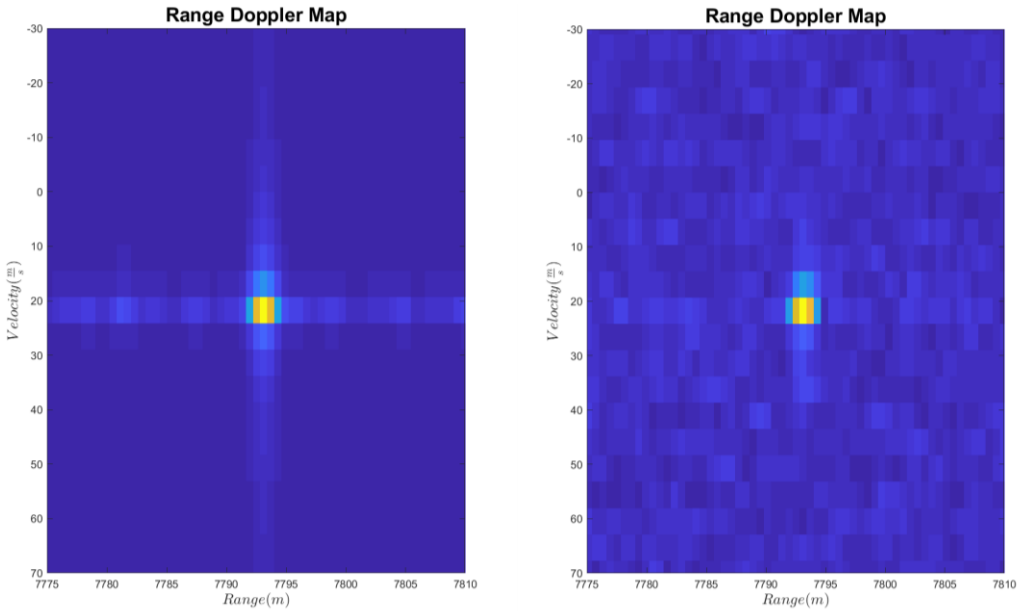

Figure 4.5. Zoomed Range-Doppler map of phased-MIMO array with two Chebyshev waveforms without noise (left) and with noise (right).

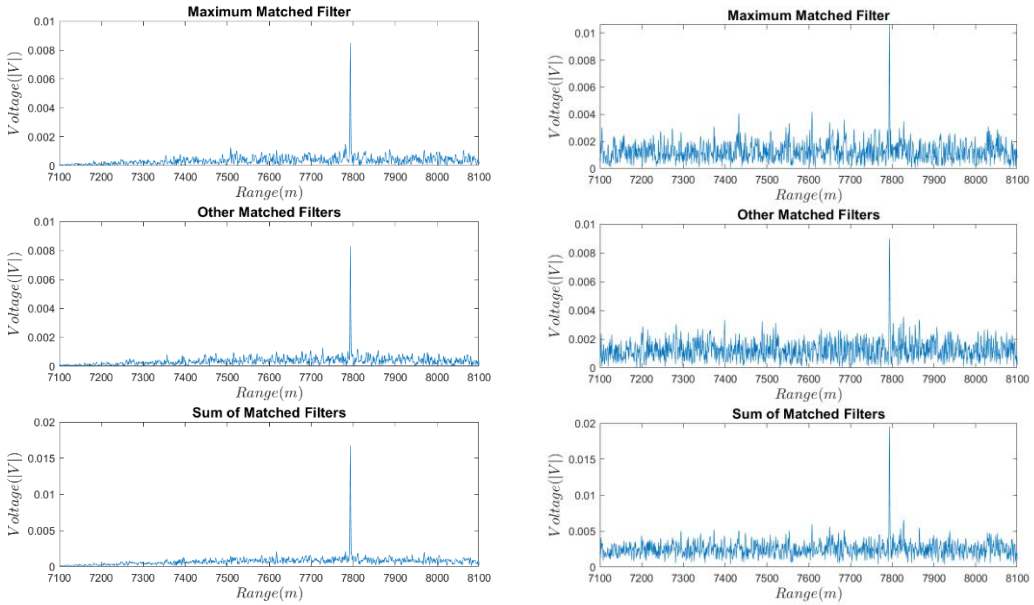

Figure 4.6. Range slice along maximum Doppler and spatial frequencies of Phased-MIMO configuration with two Chebyshev waveforms without noise (left) and with noise (right).

The range slices are displayed differently because of the multiple matched filters associated with the phased-MIMO operation. Each additional waveform increases the number of matched filters. For this reason, the figures are separated into the results of each matched filter as well as their combination. When the range-Doppler maps and range slices of the combination of matched filters for the phased-array and phased-MIMO configurations are compared, the differences are minimal except for the splitting of the energy in the matched filters. Their similarities result because only two waveforms were transmitted, and the Chebyshev waveforms exhibit low cross-correlation properties. The transmission of more waveforms is expected to cause larger changes in the range-Doppler maps and corresponding range slices. To verify this conclusion, Figure 4.7 and Figure 4.8 were created with the simultaneous transmission of four Chebyshev waveforms.

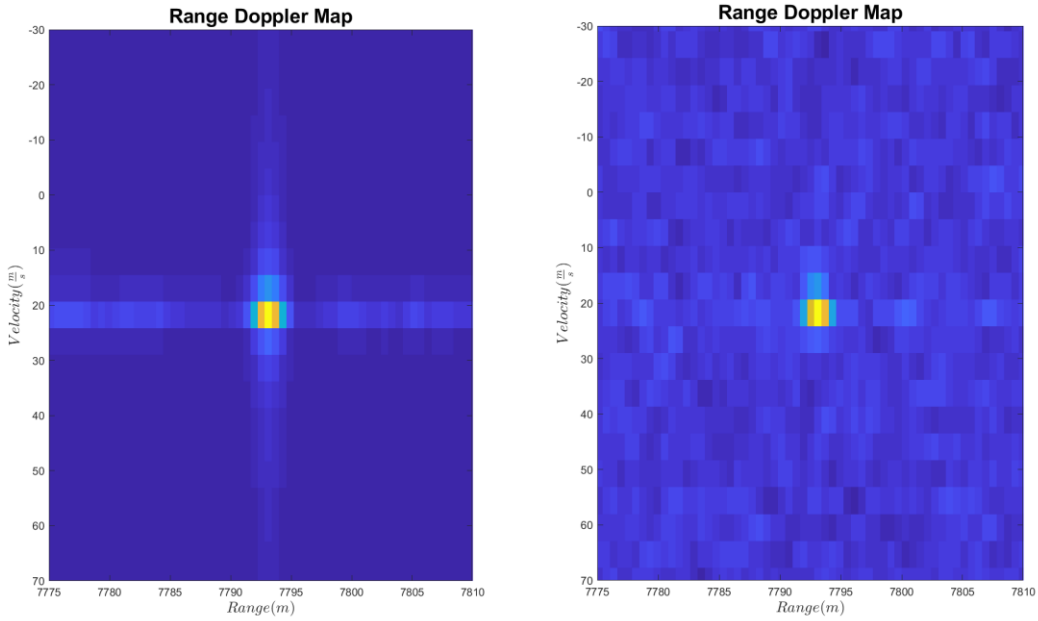

Figure 4.7. Zoomed Range-Doppler map of phased-MIMO array with four Chebyshev waveforms without noise (left) and with noise (right).

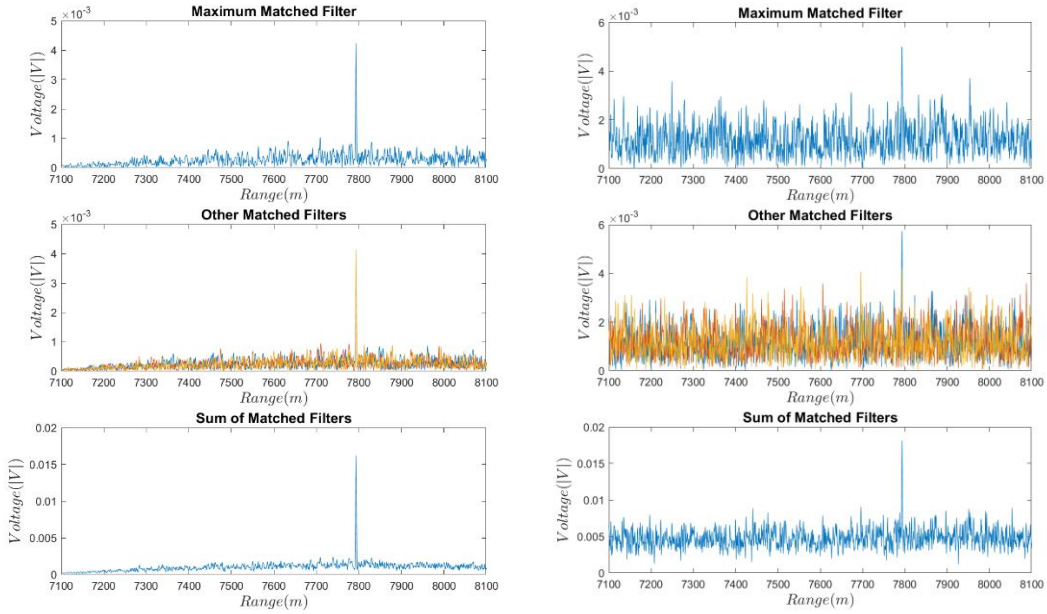

Figure 4.8. Range slice along maximum Doppler and spatial frequencies of Phased-MIMO configuration with four Chebyshev waveforms without and with noise.

From the range-Doppler maps we see increased average power in the range sidelobes as more matched filters are required. The range-slices of the noiseless simulation have a reduced peak. The simulation with added noise exhibits a higher average amplitude caused by the sidelobes and noise. This is due to the non-zero correlation of the matched filters with the noise. Slices of the datacube along the other data dimensions can be visualized as spatial frequency maps such as Figure 4.9.

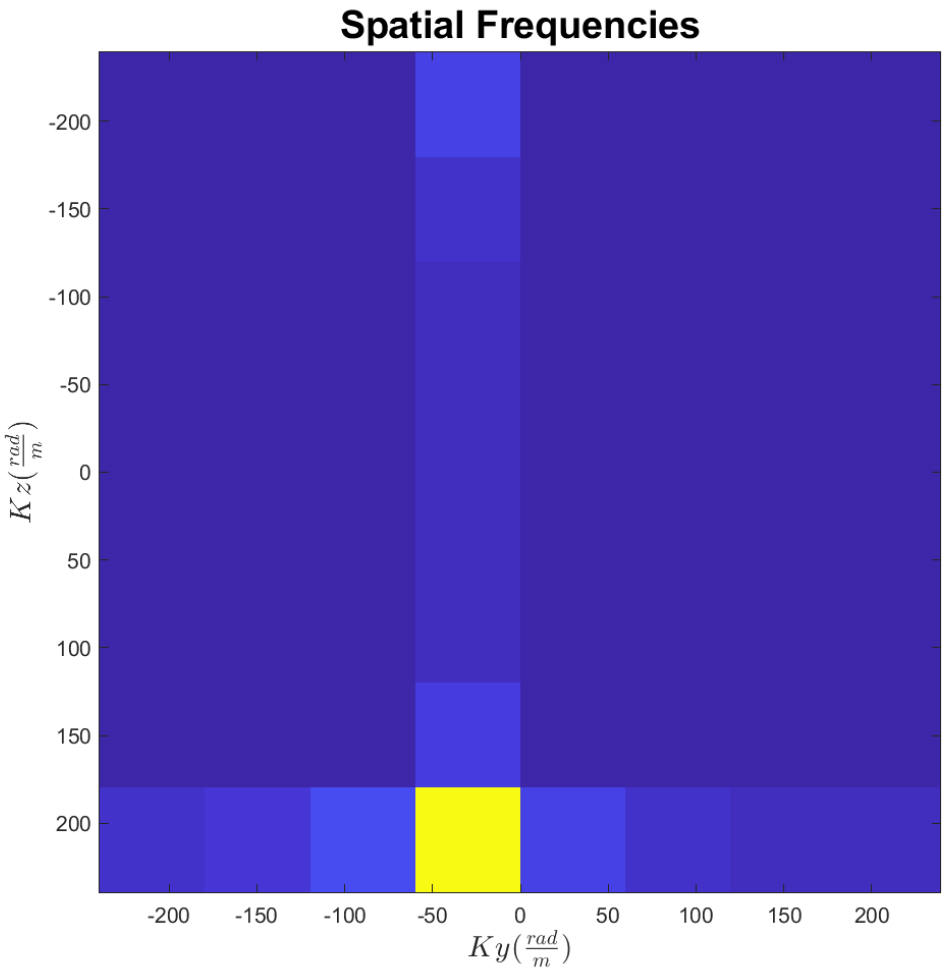

Figure 4.9. Spatial frequency map of a target without noise, 10 GHz center frequency, and four Chebyshev waveforms

The target corresponds to the yellow peak with y-component close to zero and a large z-component corresponding to the high elevation and near-zero azimuth mentioned before. The values displayed in the axes correspond to the possible wavenumbers for a carrier frequency of 10 GHz. The peak in the displayed direction is caused by the received waveform's propagation across the array due to the target's location and scattering characteristics which were discussed in Section 2.2. The raw samples of the element position are translated to the spatial frequencies through the application of an FFT. The FFT produces a spatial filter for a set of discrete

directions, and the output of the FFT is maximized at the spatial filter closest to the wavenumber of the target return shown in yellow.

Figure 4.9 allows some insight into the spatial processing of the radar, but its low resolution makes it difficult to distinguish some characteristics. Increased resolution can be achieved by zero padding the signal before the application of the FFT. This process is equivalent to increasing the number of spatial matched filters applied to the data. To further illustrate this concept, Figure 4.11 was created by zero padding the spatial samples of Figure 4.9 to four times their initial length before the application of an FFT.

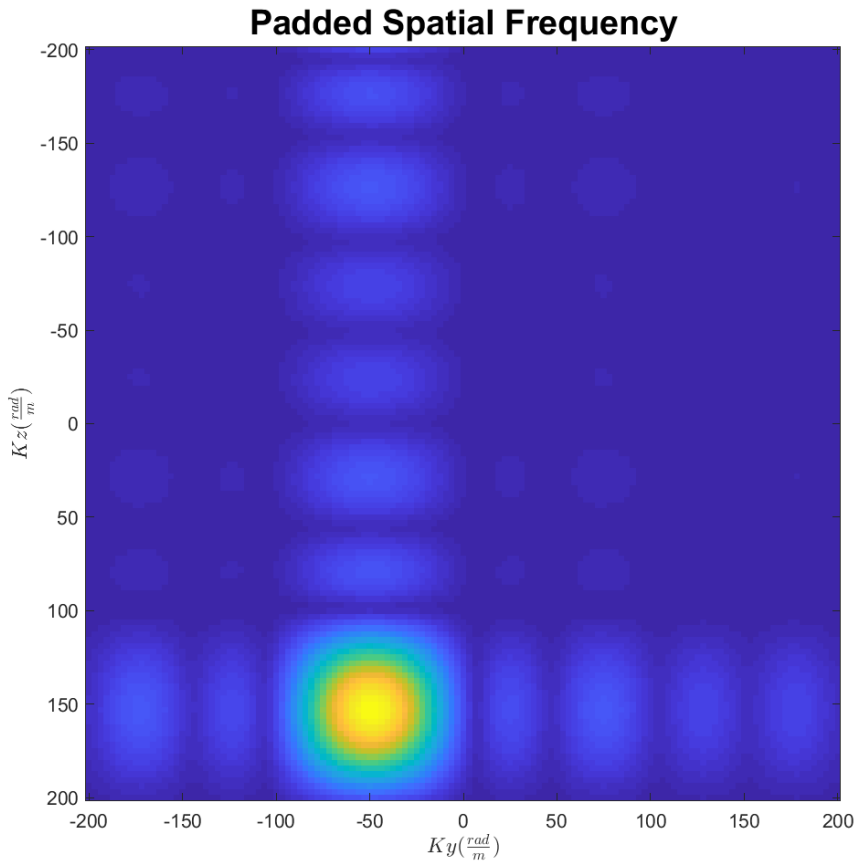

Figure 4.10. Spatial frequency map of a target without noise, 10 GHz center frequency, one Chebyshev waveforms, and increased resolution through zero-padding.

Figure 4.10 more clearly shows the sidelobe structure while also improving the estimation of the target's location. This demonstrates the gains in resolution possible by evaluating additional spatial samples or choosing a subset of spatial filters to evaluate. Figure 4.11 was created from the same data of Figure 4.10 with the same zero padding, but with the addition of noise.

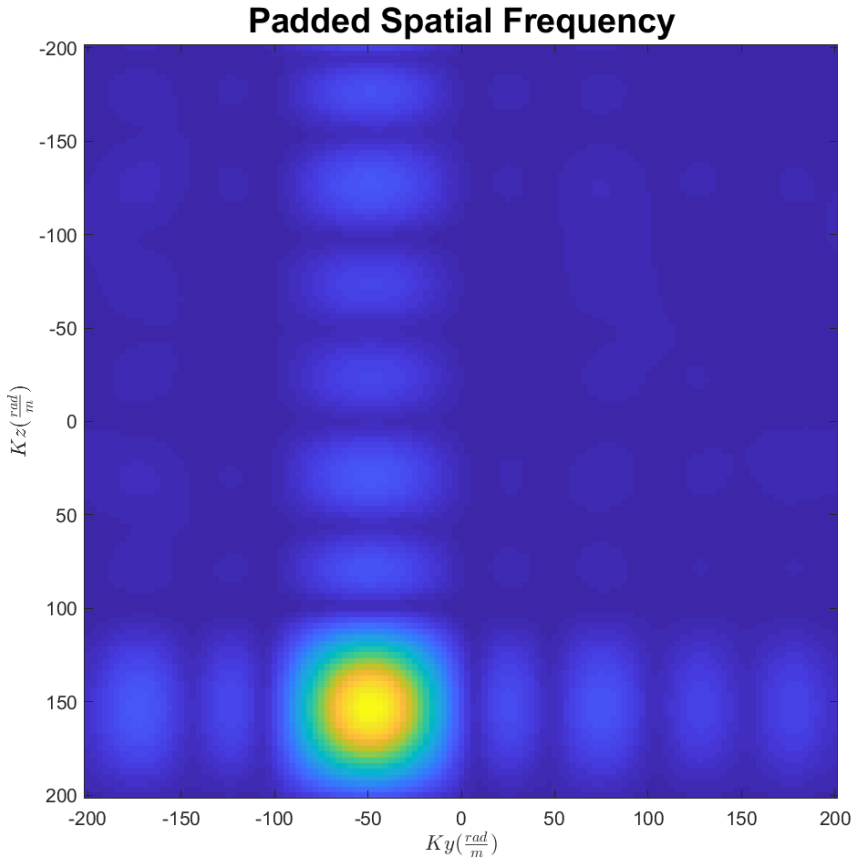

Figure 4.11. Spatial frequency map of a target with noise, 10 GHz center frequency, one Chebyshev waveforms, and increased resolution through zero-padding.

The sidelobe structure of Figure 4.11 does not change significantly from Figure 4.10. The noise can be observed outside of the sidelobes, but the main structure remains. We expect the transmission of additional waveforms to further diffuse the sidelobe structure, and Figure 4.12 was created with the same methods, noise, and the transmission of four Chebyshev waveforms to examine this trend.

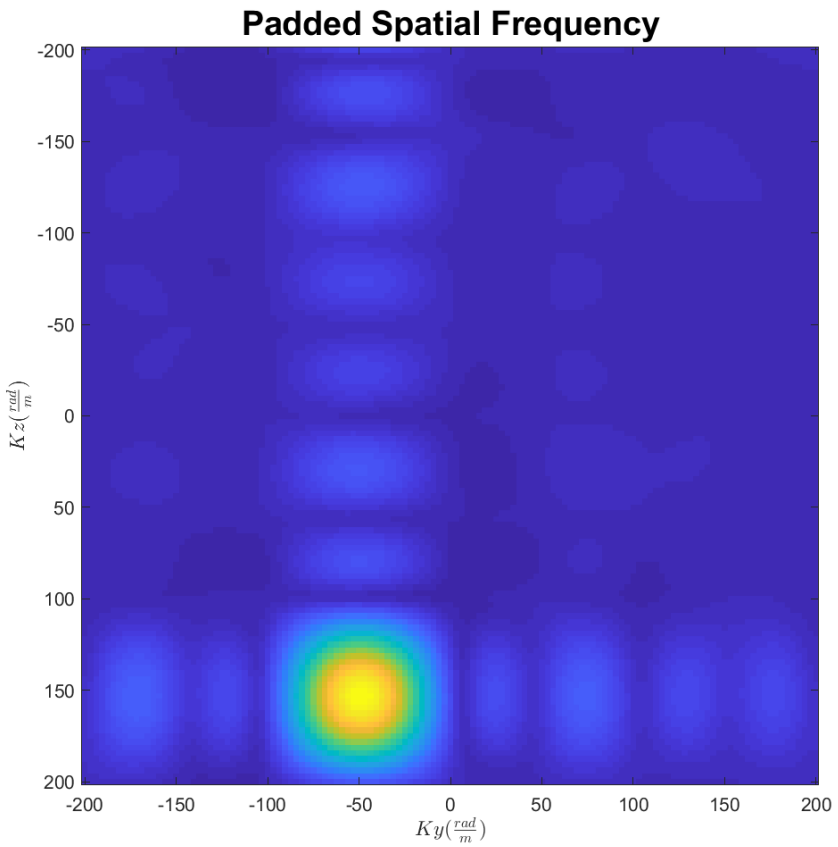

Figure 4.12. Spatial frequency map of a target with noise, 10 GHz center frequency, four Chebyshev waveforms, and increased resolution through zero-padding.

The noise of Figure 4.12 does not appear to change significantly. The main aspects of the sidelobe structure are maintained, but the amplitudes and shape display some minor distortion. The structure of the sidelobes degrades with the transmission of additional waveforms because cross-correlation between waveforms causes sidelobes to sum arbitrarily.

4.5.2 Target Detection

With the ability to simulate phased-array, MIMO, and phased-MIMO radar modes and the completion of a qualitative examination of the simulation, quantitative analysis is necessary for further assessment. Target detection was chosen because of

its ubiquity within defense systems and its simple implementation. This section describes the detection method and performance within the simulation.

4.5.2.1 Constant False Alarm Rate Detection

Although there exist many detection techniques, the cell-averaging Constant False Alarm Rate (CA-CFAR) method was chosen. CA-CFAR is a simple and effective adaptive thresholding method for target detection, although its performance varies in different scenarios [20], [21]. Ordered Statistic Constant False Alarm Rate methods have been shown to demonstrate improved performance [20] but the CA-CFAR was evaluated for its simplicity and reduced computational requirements. The CA-CFAR method was implemented with a 4D averaging filter with guard cells that protect against energy leaked to adjacent cells. A set of *N* Training cells with $\alpha/_N$ weighting, where α is a design parameter chosen to offset the small sample size, are used to determine the local average of the CA-CFAR filter. The shape of the averaging filter is visualized in Figure 4.13 with green representing training cells, blue representing guard cells, and a light-green cube in the center representing the cell under test (CUT). In this figure, the height and width of the individual rectangular prisms represent the Doppler and range dimensions (each tile in the figure is a range-Doppler map), and the digitally beamformed angles are represented by the array of tiles.

This cell-averaging window is convolved with the datacube to determine thresholds for target detection at each cell in the datacube, and values greater than this threshold are chosen as detections. For the sake of assessing performance and to

mitigate the target/track association problem, these detections are then compared to the known locations of *true* targets to distinguish true detections from false alarms.

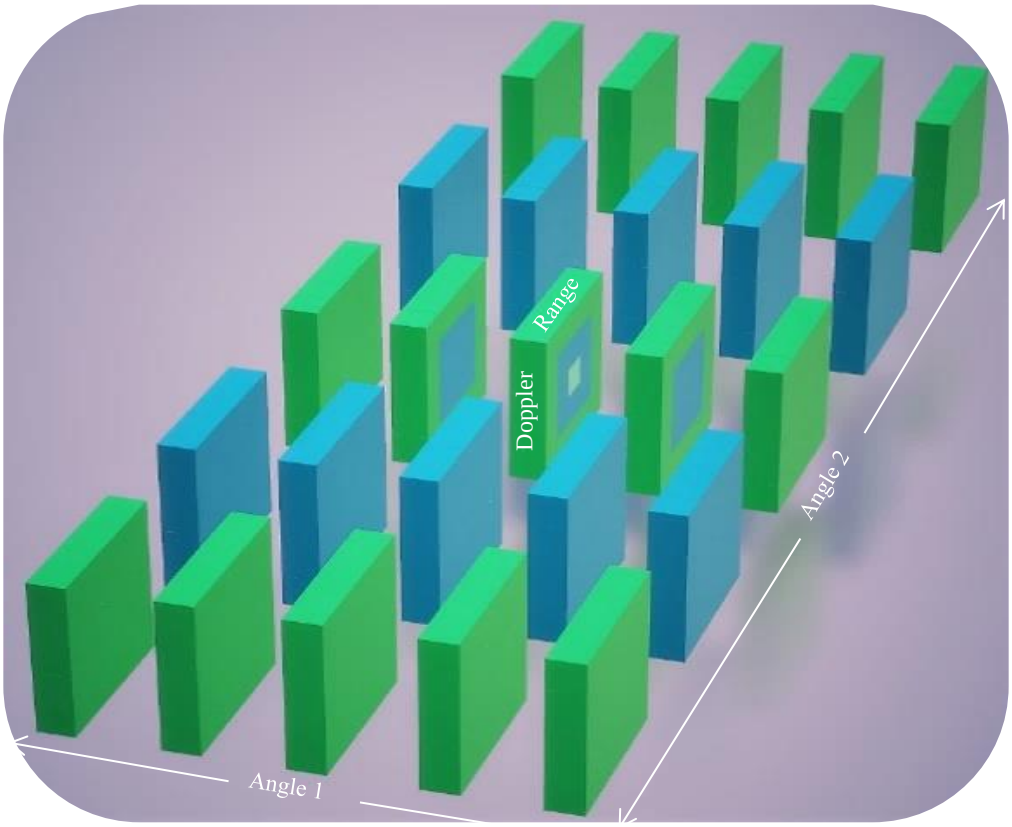

Figure 4.13. CA-CFAR 4D cube used for detection.

Because the results of the target detection are binary values, they can be illustrated as probabilistic signal measurements and rates of detection against false-alarm.

4.5.2.2 Probability Density Function

Probability density functions (PDFs) for the processed noise and signal plus noise were created through Monte-Carlo simulations to analyze the physical responses of the system. These distributions were created by running a simulation of nine targets with random RCS, spawned within a broadside beam formed from one, two, or four transmit sub-arrays. This process was repeated 70 times with new targets.

The return from the strongest target was then chosen as a sample for the PDF. The processing of the signal and noise was replicated on the noise-only data to determine its magnitude within the chosen cell. The resulting histograms follow in Figure 4.14 through Figure 4.16

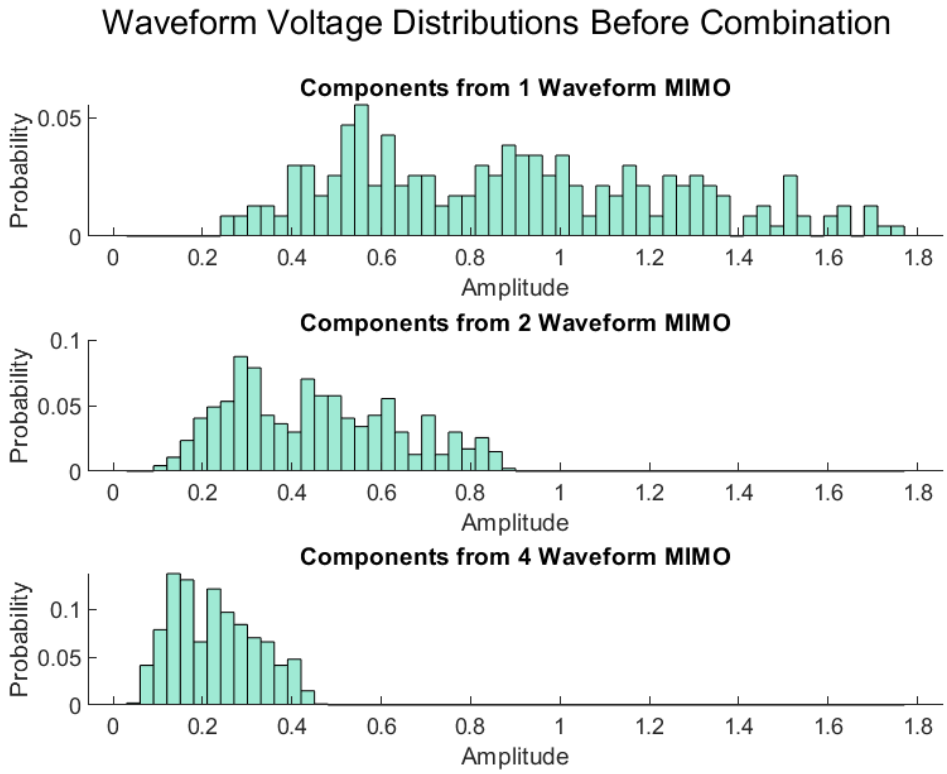

Figure 4.14. Histogram with probability normalization for the voltage resulting at each matched filter with the number of transmitted waveforms listed for each subplot.

Figure 4.14 was created by simulating the same set of targets and noise for one, two, and four unique waveforms with the signals separated at receive by matched filtering. The amplitude resulting for each matched filter was then plotted in the histogram to show the signal return resulting for each sub-array. The distributions of signal and noise are spread around central values that correspond to the single waveform case divided by the number of transmitted waveforms. This relationship of reduced voltage

per waveform is expected because the total energy is conserved. If the power is the return voltage squared or

$$P_{tot} = V_{tot}^2, \tag{4.3}$$

and the voltage for each waveform m is V_m with N_w transmit waveforms, then the total voltage is the sum of the returns for each waveform as

$$V_{tot} = \sum_{m=1}^{N_w} V_m. \tag{4.4}$$

If we assume the transmit sub-arrays for each waveform are the same size, transmit approximately the same total energy, and that the cross-correlation between the waveforms is negligible, then this sum is the transmit voltage of one waveform multiplied by the number of waveforms is approximated by $N_w V_1$. Because the voltage of each waveform is the full-aperture voltage V_0 divided by the number of waveforms, we then see

$$V_{tot} \approx N_w V_1 = N_w \left(\frac{V_0}{N_w}\right) = V_0. \tag{4.5}$$

If we combine these relationships, we can show that

$$P_{tot} = V_{tot}^2 = V_0^2, \tag{4.6}$$

is true for the defined voltages and total power. The total transmission energy is conserved when the equally sized sub-arrays exhibit a V_0/N_w voltage scaling as can be observed in Figure 4.14 and Table 2.

Table 2. Component Waveforms Central Tendencies

	Mean	*Median*	*Std*	*Range*
1 Waveform	0.8907	0.8777	0.3647	1.5002
2 Waveforms	0.4526	0.4375	0.1869	0.7638
4 Waveforms	0.2283	0.2221	0.0936	0.4201
Ratio ½	1.9680	2.0061	1.9517	1.9640
Expected ½	2.0000	2.0000	2.0000	2.0000
Ratio ¼	3.9009	3.9516	3.8976	3.5711
Expected ¼	4.0000	4.0000	4.0000	4.0000

The first three rows of Table 2 include measurements of the mean, median, standard deviation, and range for distributions based on the transmission of one, two, and four orthogonal waveforms. This table also includes calculations of the ratios for these central tendencies as well as the ratios' expected values. The transmit energy will be conserved regardless of the number of waveforms because the waveform duration, element count, and transmit power remain constant, but the waveform cross-correlation and modulation energy may produce slight variations. With the correct scaling of the sub-array returns, the coherent addition of the outputs of each matched filter should produce the same total voltage for any number of waveforms. The voltages resulting from this method of combination for one, two, and four waveforms are shown in Figure 4.15.

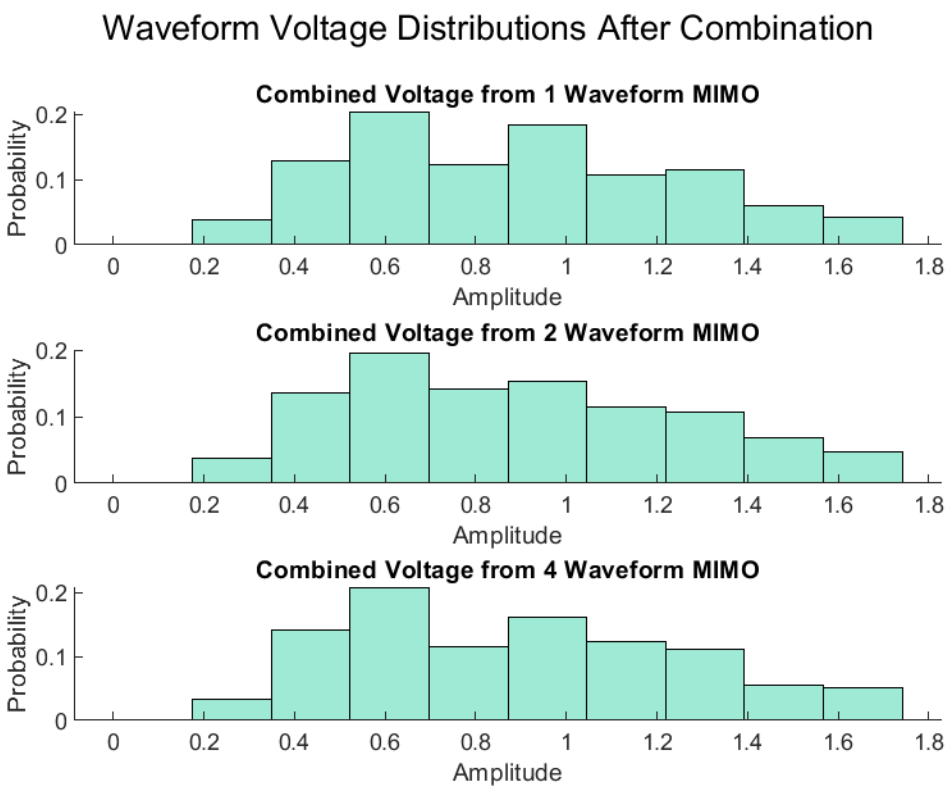

Figure 4.15. Histogram with probability normalization for the voltage resulting from the combination of all matched filters with the number of transmitted waveforms listed for each subplot.

As we expected, the distributions' centers and shapes are similar. To further demonstrate this, Table 3 was created with measurements of the distribution parameters.

Table 3. Combined Waveforms Central Tendencies

	Mean	*Median*	*Std*	*Range*
1 Waveform	0.8907	0.8777	0.3653	1.5002
2 Waveforms	0.8936	0.8635	0.3695	1.5081
4 Waveforms	0.8909	0.8726	0.3658	1.4842
Ratio ½	0.9968	1.0164	0.9887	0.9947
Expected ½	1.0000	1.0000	1.0000	1.0000
Ratio ¼	0.9998	1.0059	0.9986	1.0107
Expected ¼	1.0000	1.0000	1.0000	1.0000

The configurations have closely matched the ideal case, so an examination of noise in these scenarios is valuable.

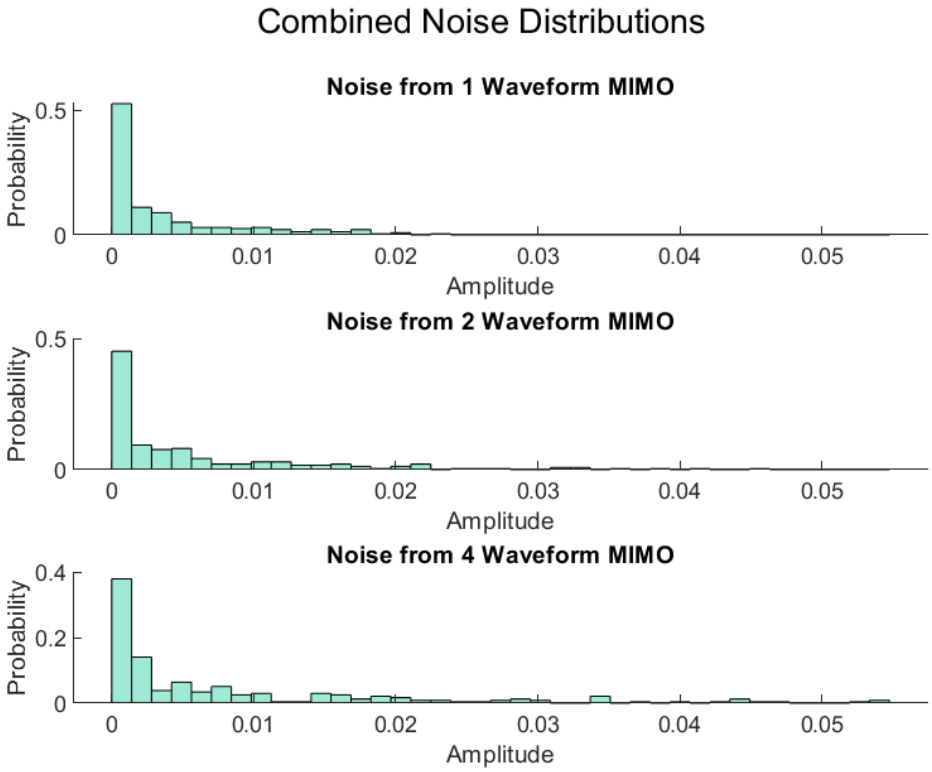

Figure 4.16. Histogram with probability normalization for the noise resulting from the combination of all matched filters with the number of transmitted waveforms listed for each subplot.

Figure 4.16 shows histograms for the noise-only data for the configurations of Figure 4.14 and Figure 4.15. With each matched filter adding cross-correlation effects and independent noise, the mean and variance of the noise power increases with the addition of waveforms. Table 4 includes measurements of the noise distribution parameters analogous to the measurements of Table 2 and Table 3 and demonstrates the trend of additional noise with each additional waveform. The expected ratio for each noise set is the ratio that would indicate the total noise energy was conserved.

We expect the actual ratios to be lower because the noise will add randomly after the application of different matched filters producing increased noise.

Table 4. Noise Distribution Parameters

	Mean	*Median*	*Std*	*Range*
1 Waveform	0.0037	0.0011	0.0051	0.0228
2 Waveforms	0.0059	0.0021	0.0084	0.0449
4 Waveforms	0.0084	0.0023	0.0118	0.0547
Ratio ½	0.6299	0.5267	0.6083	0.5079
Expected ½	1.0000	1.0000	1.0000	1.0000
Ratio ¼	0.4389	0.4845	0.4299	0.4168
Expected ¼	1.0000	1.0000	1.0000	1.0000

As predicted, the measured ratios are less than the expected ratios that would conserve the total energy. Because the addition of matched filters increases noise power and does not significantly change the signal power, the SNR will decrease as expected with the transmission of additional matched filters. The addition of waveforms has a negative impact on SNR and will be seen in scenarios with weak SNR.

4.5.2.3 Receiver Operation Characteristic Curves

Receiver Operation Characteristic (ROC) curves are a figure of merit for binary detection systems with a rich history in radar, and they weigh the probability of detection (PD) against its false alarm rate (FAR). When thresholding methods are used, the above-threshold integration of the PDFs for the noise distribution and the signal-plus-noise distribution determine the probabilities of false alarm and detection, respectively. To better visualize this method, normal distributions labeled as noise and signal plus noise are overlaid in Figure 4.17 with a threshold displayed. The probability of detection is the area under the curve of the signal-plus-noise (S+N)

above the threshold and the probability of false alarm is similarly derived from the noise-only PDF.

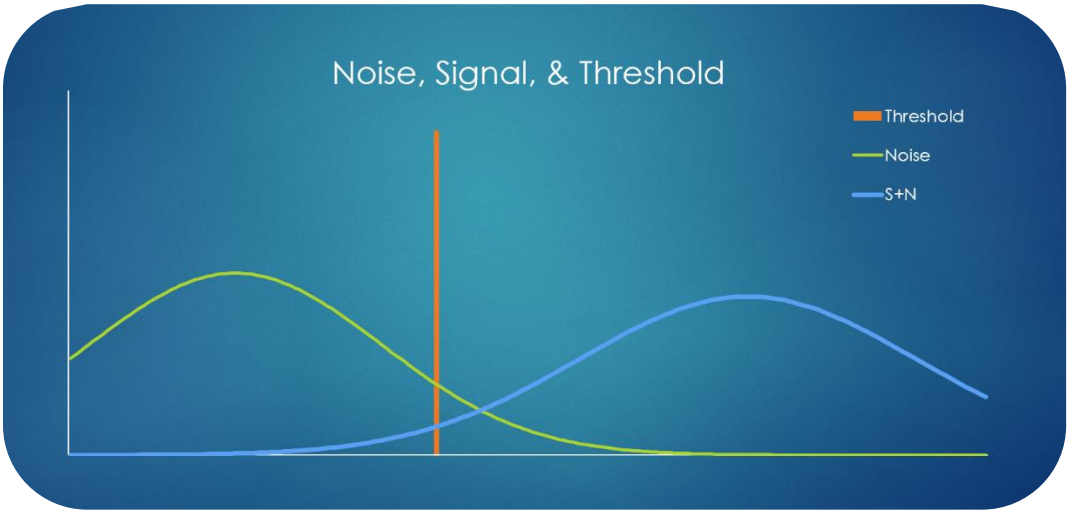

Figure 4.17. PDFs for signal and noise with threshold shown in orange.

With this understanding of detection probabilities and the PDFs from Section 4.5.2.2, we can predict trends in the ROC curves for static thresholding methods. The same thresholding concepts apply when CA-CFAR methods are used, but the threshold is adapted for each cell, resulting in increased resilience to false alarms caused by variations in noise, sidelobes due to cross-correlation, and other interferences. Within a CA-CFAR system, the PD and FAR are adjusted through the tuning parameter α. Sweeping through different α values can determine the scenario-specific ROC curves for comparison with other modes and scenarios or for the selection of an α that will produce the desired PD and FAR. ROC curves generated from 90 Monte-Carlo trials with 9 targets are shown in Figure 4.18.

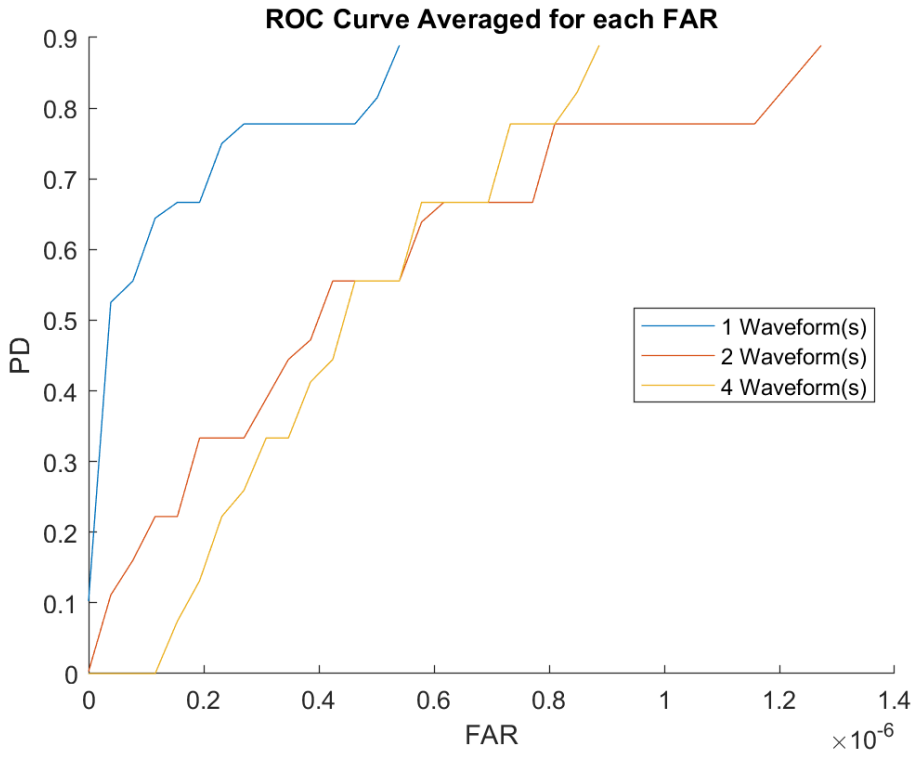

Figure 4.18. Receiver operation curve derived with CA-CFAR method for random trials of 9 targets spawned within a broadside beam over 90 runs. The number of simultaneous transmit waveforms is included in the legend.

Additional transmit waveforms increase the apparent noise power and decrease SNR, such that the ROC curves will exhibit worse detection with more simultaneous waveforms. The addition of waveforms provides the benefit of more instantaneous coverage at the cost of SNR. The reduction of SNR and ROC performance with the transmission of additional waveforms is one tradeoff of MIMO operation that indicates it will be useful to improve operational flexibility instead of replacing traditional methods. The transmission of two or four waveforms produces ROC curves in Figure 4.18 that are holistically worse than the single waveform case, but do not exactly match our predictions. This is probably due to the limitations in the number of targets that can be simultaneously defined while still providing separable returns. If we define too many simultaneous targets, the returns may overlap, so only nine targets were modeled. This

limited set of targets reduces the resolution of the PD metric and could cause this inconsistency.

Through the process of implementing and debugging the CA-CFAR method, PDFs that characterize the underlying physics and ROC curves that characterize the detection performance were created. The next sections will introduce a novel method for selecting transmit sub-arrays and steering directions to improve the utilization of the available resources of all-digital radars. This method is built within the simulation environment to allow an appraisal of the method's performance against simple comparison modes.

5 ENERGY ALLOCATION METHOD

Adaptive allocation of transmit sub-arrays exploits the versatility of all-digital radar, and this section introduces an adaptive resource allocation method that improves tracker performance by steering transmit beams in multiple directions simultaneously using low-correlation (MIMO) waveforms and processing. Theoretical comparisons of some energy allocation methods can be found in [22]. Our proposed method uses the configuration and power measurements of previous CPI to predict subsequent aperture requirements needed to update each target. A profit metric for each direction is determined based on the number of targets updated relative to the resources required. A recursive algorithm derives an efficient distribution of the radar's power by repeatedly choosing the best profit and recalculating the resulting profits for the remaining configuration options. The goal here is an initial test of adaptive transmit sub-arrays in the context of a detailed simulation of all-digital arrays.

5.1 Parameter Space

The energy allocation method will evaluate a set of options for possible configurations of the radar's resources. Because it is impossible to consider every possibility for transmitting waveforms from every combination of sub-array, we must define a subspace of the possible sub-array sizes and beam directions. This section will define the set of configurations considered, as well as methods for reducing the number of considered options.

First, we will reduce the size of the array to be more manageable. Our hypothetical all-digital radar consists of a full-aperture of N by M individually controlled elements where sub-arrays can be formed from any number and distribution of elements. To simplify this dimension, we only consider sub-arrays formed with rectangular shape and contiguous elements where it is preferred, due to computational complexity and physical limitations on array control, to group the total array of N by M elements into Minimum Array Units (MAU). These MAU are the finest granularity with which the system software/firmware will allow unique waveforms. Let the MAU have size $[N_0, M_0]$. The total available MAUs is then $N_T = {}^{N}/_{N_0}$ and $M_T = {}^{M}/_{M_0}$ for the two dimensions. That is, we take the full aperture of N by M elements and combine subsets of this aperture of N_0 by M_0 elements so that we form a new, effective aperture with dimensions of N_T by M_T total MAU. This process of MAU reduction is visualized in Figure 5.1 where the full aperture of N by M elements is on the left and two samples of MAU reduced arrays are on the right.

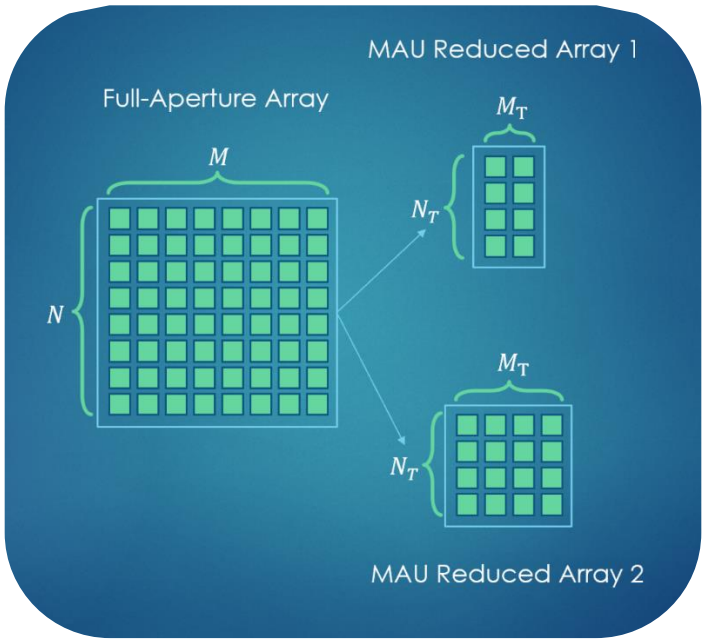

Figure 5.1. Two example cases of MAU Reduction

The reduced aperture of Figure 5.1 on top becomes a 4x2 set of MAU where each MAU is two N elements and four M elements. The reduced aperture on bottom becomes a 4x4 set of MAU where each MAU is two N elements and two M elements. The options remaining after the MAU reduction are the individually controllable array resources. In some systems, the MAU size can stay as small as a single element, but this is only practical in arrays with few total elements or when the elements are not co-located. If we limit sub-arrays to have rectangular shape, and adjacent elements, then the possible sub-arrays are rectangles with total length and width less than the number of MAU in each dimension. If we examine the possible configurations for the MAU Reduced Array 2 in Figure 5.1, we can form the configurations shown in Figure 5.2.

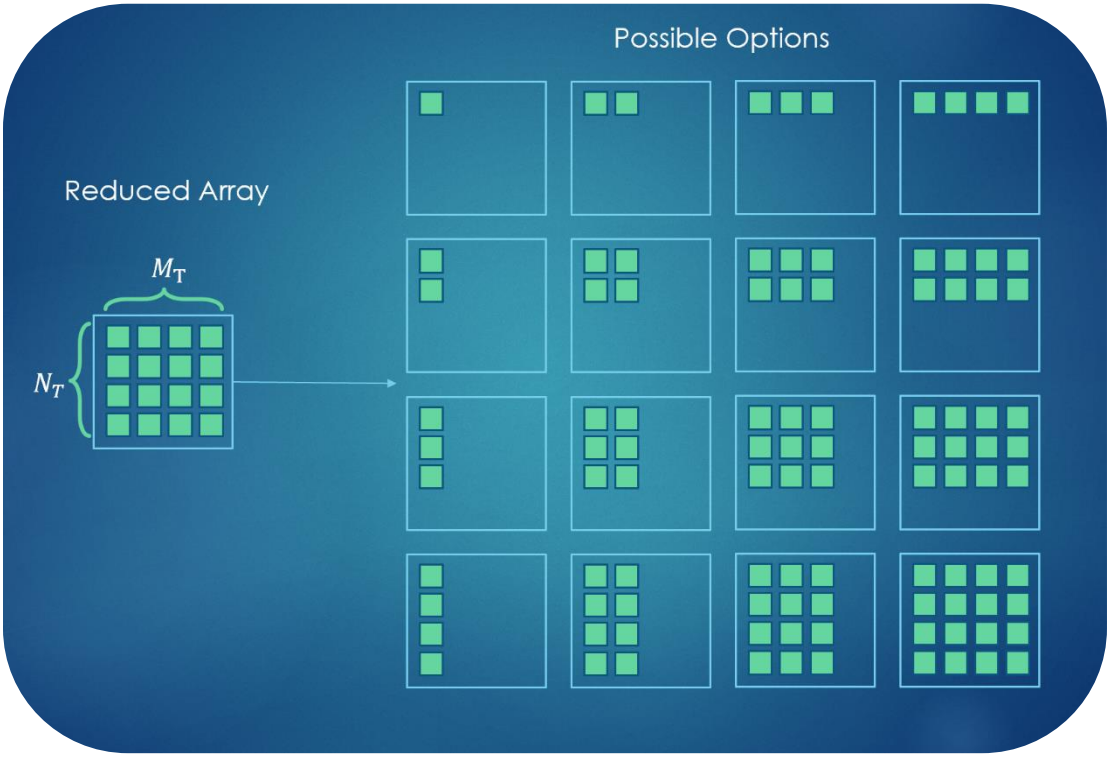

Figure 5.2. Potential sub-array divisions for one example MAU-reduced array from Figure 5.1.

The sub-array orientation matters because it effects the beam-shape. The beam narrows as sub-array size increases in each dimension, and a single MAU will have the widest beam while the full-aperture will have the narrowest beam. For our purposes, the location of the sub-arrays within the full-aperture does not matter because we assume the targets are far enough away from the radar that the slight differences in sub-array position will be negligible. Stated more simply, we will consider the unique orientations for each sub-array division of the MAU-reduced aperture that is rectangular and co-located without consideration for the sub-array location within the full aperture.

With our set of possible elements defined, we will consider the ways in which the sub-arrays can be steered. The possibilities for our steering angles are limited by less clear boundaries such as the ADC rate that discretizes the steering directions and limits the exact steering angle. Additionally, not all sub-arrays can transmit in all steering angles because more than one sample (element) is required to steer a beam in any dimension. For example, a 1xM or Nx1 array will only have steering control in the dimension with multiple elements, and the beampattern in the single element dimension will match the element pattern. Though this space of possible steering angles is bound by the radar-specific implementation, there will still be many possible solutions. To reduce its size and make computation more manageable, a subset of steering directions must be determined. Within this thesis, we establish these possible steering directions as unit vectors derived from predictions of the target position. That is, we consider steering the beam in angles near each target. However, the set of potential steering directions could be defined as a uniformly sampled grid of the

search area, a grid with more samples in areas of interest, or whatever method best fits the situational objectives of the radar.

With the sub-array configurations and a set of possible steering directions, we can then determine the expected return signal strength resulting from each target reflection. This return prediction can be calculated using an RCS estimate derived from previous measurements for each target. If we have an estimate for a target's range, angle, speed, and RCS, we can predict the SNR produced by each sub-array and aiming direction. The prediction's equation can be derived from the radar equation and is examined more thoroughly in Section 5.2. The predicted effect on each target forms the final dimension of the search space.

The combination of possible sub-array dimensions of n_{MAU} and m_{MAU}, aiming directions $\overline{K}_{\theta}$, and their effect on each target q define a set of configurations that can be used to predict signal returns, and these predictions are organized as a multi-dimensional map. These predicted signal strengths can be compared with the noise to determine the SNR for each target update, and a threshold can be applied to the SNR measurements to form a binary map indicating which targets would be updated acceptably by each set of beam location and sub-array size. Transmitting multiple waveforms introduces cross-correlation interference, and the binary map can be modified with cross-correlation predictions to mitigate the SINR effects of this additional interference. The next section will use the radar equation to develop the relationships that predict the return power and enable the SINR prediction.

5.2 Power Derivation

An estimation of the RCS value is critical to the operation of the array allocation method as it is the key to SINR estimation. Although there are high-fidelity and generalized-physics methods for modeling target RCS [30]-[32], we desire a simple estimation of its value based on past measurements. This will allow us to predict the return power for the same target when the radar operates under different conditions. We will derive this estimate from a rearrangement of the radar equation with gain estimate substitutions. This section aims to thoroughly describe the estimation technique for the veracity of the tested allocation method and simulation description. This section could be described as introductory information but is included here as it is a key step in the energy allocation method.

If q is the index of a target, $\overline{K}_q$ is the wavevector corresponding to the target, P_T is the power of the transmit waveform, $G_T(\overline{K}_q)$ and $G_R(\overline{K}_q)$ are the transmit and receive gains in the direction of the target, λ is the wavelength of the carrier frequency, $\sigma_{rcs}(q)$ is the radar cross section of each target, and $R(q)$ is the range of the qth target, then the received power of a single input, single output SISO radar is

$$P_R(q) = \frac{P_T G_T(\overline{K}_q) G_R(\overline{K}_q) \lambda^2 \sigma_{rcs}(q)^2}{(4\pi)^3 R(q)^4}. \tag{5.1}$$

When the radar transmits multiple waveforms, the power of the transmit waveform P_T, the transmit gain G_T, and the return power P_R also depend on the transmit waveform l or its transmission angle $\overline{K}_l$. When we assume no cross-correlation between waveforms, (5.1) becomes

$$P_R(q,l) = \frac{P_T(l)G_T\left(\overline{K}_q, \overline{K}_l\right)G_R\left(\overline{K}_q\right)\lambda^2\sigma_{rcs}(q)^2}{(4\pi)^3R(q)^4}. \tag{5.2}$$

If we create a time index for each CPI and assume the RCS does not change significantly between each measurement, (5.2) becomes

$$P_R(q,l,cpi) = \frac{P_T(l,cpi)G_T\left(\overline{K}_q, \overline{K}_l, cpi\right)G_R\left(\overline{K}_q, cpi\right)\lambda^2\sigma_{rcs}(q)^2}{(4\pi)^3R(q,cpi)^4}. \tag{5.3}$$

The transmit power is also dependent on the output of the matched filter which is determined by the modulation and complex envelope of the transmitted waveform. Because we generate different Chebyshev waveforms each CPI, the effective transmit power will also vary. If we represent the maximum transmit power for each element as P_{el} and the un-scaled signal with unity peak amplitude as S with time index n and N total samples, then the transmit power P_T is

$$P_T(l,cpi) = P_{el}\sum_{n=1}^{N} S(n,l,cpi)^2. \tag{5.4}$$

The transmit gain is the array response to each target determined by the spacing of the elements, the steering direction of propagation of the transmit waveform, and the target direction. If we assume the lth transmit sub-array uses N_l total elements with coordinates r_t and element signal strength in the target direction $\rho_{el}(q)$ then the transmit gain is

$$G_T\left(\overline{K}_q, \overline{K}_l\right) = \rho_{el}(\overline{K}_q)\left|\sum_{t=1}^{N_l} e^{j\left(\overline{K}_q - \overline{K}_l\right)\cdot r_t}\right|. \tag{5.5}$$

More detail on the derivation of the element pattern can be found in Section 4.4. The receive gain is similar to the transmit gain but the array size is always the full-

aperture of N_{el} elements. Receive beamforming is implemented with a 2D-FFT operation that can be written as

$$X\left(e^{j\bar{K}_\theta}\right) = \sum_{i=1}^{N_{el}} x[i]e^{-j\bar{K}_\theta \cdot \bar{r}_i}, \tag{5.6}$$

where $\bar{K}_\theta$ is evaluated for a discrete set of spatial frequencies limited by the number of spatial samples (*n* and *m*) and FFT length, and i is the element index for the full-aperture array. If the set of evaluated angles is Ƙ_{FFT}, and we assume the maximum return occurs at the evaluation angle closest to the target angle, then the receive gain can be modeled as

$$G_R\left(\bar{K}_q\right) = \rho_{el}\left(\bar{K}_q\right) \underset{\bar{K}_\theta \in \text{Ƙ}_{FFT}}{max} \left(\left| \sum_{t=1}^{N_{el}} e^{j\left(\bar{K}_q - \bar{K}_\theta\right)\cdot r_t} \right| \right). \tag{5.7}$$

If we assume the spatial filter bank covers a wide enough set of aiming directions such that there is a $\bar{K}_\theta \approx \bar{K}_q$, we can simplify (5.7) by assuming the exponent in the sum approaches zero. The sum of the complex exponentials can then be approximated by the number of full-aperture receive elements N_{el} or as

$$G_R\left(\bar{K}_q\right) \sim \rho_{el}\left(\bar{K}_q\right) N_{el}. \tag{5.8}$$

It is important to note $\bar{K}_\theta$ will never exactly match $\bar{K}_q$ because the filter evaluates a finite number of $\bar{K}_\theta$, and $\bar{K}_q$ is in a continuous space. The approximation is improved with increases in the number of receive elements or FFT bins. If $\bar{K}_q$ for a particular target is well known, its value could be used instead of the FFT operation to further reduce error. We can now substitute $G_T\left(\bar{K}_q, \bar{K}_l\right)$ and $G_R\left(\bar{K}_q\right)$ into (5.3) to form

$$P_R(q,l,cpi) = \frac{P_{el}\rho_{el}\left(\bar{K}_q\right)^2 N_{el}\lambda^2\left(\sum_{n=1}^{N} S(n,l,cpi)^2\right)\left(\sum_{t=1}^{N_l} e^{j\left(\bar{K}_q-\bar{K}_l\right)\cdot r_t}\right)\sigma_{rcs}(q)^2}{(4\pi)^3 R(q,cpi)^4}. \quad (5.9)$$

If we use the most recent measurement of P_R at $cpi-1$, then (5.9) can be rearranged to estimate the RCS as

$$\sigma_{rcs}(q)^2 = \frac{P_R(q,l,cpi-1)(4\pi)^3 R(q,cpi-1)^4}{P_{el}\rho_{el}\left(\bar{K}_q\right)^2 N_{el}\lambda^2\left(\sum_{n=1}^{N} S(n,l,cpi-1)^2\right)\left(\sum_{t=1}^{N_l} e^{j\left(\bar{K}_q-\bar{K}_l\right)\cdot r_t}\right)}. \quad (5.10)$$

With equations (5.9) and (5.10) we can estimate the RCS from past measurements and then use this estimate to predict the power of future returns.

We have previously assumed the RCS will not change between CPI, but errors in our RCS estimate will cause variations between updates. To provide some resilience to these variations, we instead use the RCS value averaged over several previous CPI. Within a real system, the RCS value will change between CPI outside of just measurement errors. A moving average smooths the RCS estimates used to predict the necessary sub-array size. Some RCS profiling method that rejects outliers may provide better performance but is not considered here. Substituting the averaged value of the RCS into (5.8) will create

$$P_R(q,l,cpi) =$$

$$\frac{P_{el}\rho_{el}\left(\bar{K}_q\right)^2 N_{el}\lambda^2\left(\sum_{n=1}^{N} S(n,l,cpi)^2\right)\left(\sum_{t=1}^{N_l} e^{j\left(\bar{K}_q-\bar{K}_l\right)\cdot r_t}\right)\left(\frac{\sum_{k=1}^{cpi-1}\sigma_{rcs}(q,k)}{cpi-1}\right)^2}{(4\pi)^3 R(q,cpi)^4}. \quad (5.11)$$

We then say that each waveform has transmit steering angle $\bar{K}_l$ and sub-array size of n_l by m_l so that the total number of elements N_l becomes $n_l m_l$, and (5.11) becomes

$$\frac{P_{el}\rho_{el}\left(\bar{K}_q\right)^2 N_{el}\lambda^2\left(\sum_{n=1}^{N} S(n,l,cpi)^2\right)\left(\sum_{t=1}^{n_l m_l} e^{j\left(\bar{K}_q-\bar{K}_l\right)\cdot r_t}\right)\left(\sum_{k=1}^{cpi-1}\sigma_{rcs}(q,k)\right)^2}{(4\pi)^3 R(q,cpi)^4 (cpi-1)^2}. \quad (5.12)$$

This equation is primarily a modification of the radar equation and will be employed in later sections to predict the return power for various radar configurations.

5.3 Return Detection and Profit Function

Using the RCS estimate, we can predict the return power for various configurations of sub-array sizes and aiming angles for each target. The primary concept used in the array allocation is that there exists some minimal energy required to achieve an adequate return for target track update. We can use the RCS estimate to predict such configurations and derive a profit metric to weigh the increase in target information against the overall resource requirements and the value of the updated target information.

Target trackers stand to benefit from these predictions. When carefully configured, trackers create good estimates of the target range and angle for future CPI. This section outlines a method designed to determine efficient transmit directions and array sizes for target tracking. For evaluation purposes, we have abstracted the configuration method from the implemented tracking method by performing pseudo-tracking that leverages the clairvoyant information of the actual position and velocity of the target to predict its next location. Target association identifies the true targets by determining the maximum value within some allowable region around the clairvoyant location. Within a real system, these clairvoyant methods for tracking and correlation will not be practical, but practical tracking methods may achieve comparable performance.

To quantify the primary goal of determining the minimal resources that achieve an adequate return for target track update, we define adequate returns as those that achieve some SINR β. The first step in building this allocation mode is the calculation of the expected signal strength of possible configurations to predict the SINR. To predict the signal strength, we must determine the possible configurations and map their effective energy transmission. Our configuration space is the set of possible rectangular arrays, a set of look angles (which can be provided by the tracker), and their predicted results for each target as previously discussed in Section 5.1. If we denote the newest sub-array and waveform combination with index m, we can use the possible configurations defined in Section 5.1 with indices $q, \overline{K}_m, n_m, m_m$ to produce a datacube of expected returns from (5.12). We then threshold this datacube to determine the configurations that achieve the desired SINR β to form a binary indication map B_{New}. The indication map is true when the expected return, noise power P_n, and interference P_I achieve the SINR β for each new sub-array as

$$B_{New}(q, \overline{K}_m, n_m, m_m, cpi) = \left(P_R(q, \overline{K}_m, n_m, m_m, cpi) \geq \beta\left(P_n + P_I(l \neq m)\right)\right). \tag{5.13}$$

Within our simulation P_I is the interference noise that is limited to the cross-correlation of the transmitted waveforms as defined by (3.13) in Section 3.3.1. B_{New} indicates which configurations will achieve the desired SINR but does not consider the cross-correlation effects induced on existing waveforms. These cross-correlation effects reduce the SINR of the previously selected sub-arrays, and this increased interference can overpower the signal when the SINR is reduced enough. We can

calculate a second binary map that predicts when potential waveform m will decrease the SINR of existing waveforms l below the threshold as

$$B_{Old}(q, \overline{K}_m, n_m, m_m, cpi) = \\ all_{l \neq m}\left(P_R(q, \overline{K}_l, n_l, m_l, cpi) \geq \beta\left(P_n + P_I(m)\right)\right). \tag{5.14}$$

B_{Old} indicates if potential sub-array m with steering directions and dimensions of $\overline{K}_m, n_m,$ and m_m will not invalidate any of the preexisting sub-arrays l. This invalidation occurs when the potential sub-array increases the interference power sufficiently to reduce the SINR below the threshold β.

We can combine the two Boolean matrices B_{New}, which indicates if potential sub-arrays will achieve an SINR above the threshold β, and B_{Old}, which indicates if potential sub-arrays will not invalidate the SINR threshold of pre-existing sub-arrays, to form a combined validation indication map B_{Valid}. This combined indication map indicates valid new configurations that will not invalidate any previous selections as

$$B_{Valid}(q, \overline{K}_m, n_m, m_m, cpi) = B_{Old} \cap B_{New}. \tag{5.15}$$

With this combined binary map that predicts if the return from target q will achieve SINR β by aiming a sub-array of size n_m by m_m in direction $\overline{K}_m$, we can determine a profit metric that prioritizes configurations deemed the most valuable. A sub-array increases in value as it updates multiple targets and targets that have not been recently updated while simultaneously using fewer resources and achieving the desired SINR without invalidating the returns of other sub-arrays. It has no value when it invalidates the returns of other sub-arrays or does not update any targets with return stronger than the SINR β.

The first step to creating this profit metric is the summation of B_{Valid} for all targets within each configuration to count the number of targets the configuration should update. We will divide by the number of sub-array elements $n_l m_l$ to penalize the use of more resources creating

$$P(\bar{K}_m, n_m, m_m, cpi) = \frac{\sum_q B_{Valid}}{n_m m_m}. \tag{5.16}$$

If the CPI is sufficiently short, we do not expect the target parameters to change significantly between updates. Because the parameters defining the profit would remain relatively constant, we would expect a set of configurations to consistently offer the most profit. That is, if there is minimal change in the input parameters, we expect the function to output the same value(s) every time. To prevent this loop of the same configuration, we can change the inputs by applying a function that biases the output. The application of a priority scaling will promote the equal coverage of all targets and prevent looping through the same configuration. This priority function $p(q, cpi)$ is a scalar weighting that increases with the time since the last update of each target, and is calculated as

$$p(q, cpi) = (p(q, cpi - 1) + 1)^2. \tag{5.17}$$

$p(q, 0)$ is the initial value of the priority function and can be set to any scalar value. In the evaluated simulations, we initialize $p(q, 0)$ to one, and the corresponding section of the priority function is reset to zero after a target is updated. Nulling the priority function eliminates the profit of updating the same target in consecutive CPI, but we will temporarily remove this restriction if no profitable options remain. The

final profit function is a combination of the original profit function (5.16) and the priority function (5.17) as

$$P(\bar{K}_m, n_m, m_m, cpi) = \frac{\sum_q B_{valid}(q,\bar{K}_m,n_m,m_m,cpi)p(q,cpi)}{n_m m_m}. \quad (5.18)$$

With this profit metric, we can then begin choosing configurations that will provide meaningful information without interfering with previously chosen sub-arrays. The configuration with the best profit is chosen first, and then the procedure is repeated to find the next best sub-array configuration, and so on. Note that the profit for each transmit sub-array accounts for the number of targets that can be updated with sufficient SINR for a particular beam location; therefore, the metric balances bigger sub-arrays for enhanced SINR versus smaller sub-arrays for better instantaneous coverage. As the sub-array configurations are determined, they must be *packed* into the available array, and this is performed with the sub-array placement algorithm described in the next section.

6 SUBARRAY PLACEMENT

The previous section explained the selection of the sub-arrays' sizes and directions, and this section describes the optimization of their fit within the available array. This optimization problem is a subset of general packing problems often referred to as knapsack problems [24]-[29]. This problem occurs when there is 1) a boundary, such as a knapsack; 2) multiple objects that can fit inside this boundary, such as tools and supplies; and 3) parameters, such as volume, weight, and utility that require consideration to allow the most valuable packing scheme. In our case, the boundary is the full aperture, the objects are sub-arrays, and the parameters are the number of MAU used and the profit metric of the selection. Because the profit selection algorithm is adaptive and recursive, the number of sub-arrays is not known prior to its completion, and the packing method must place the chosen sub-arrays as they are selected to prevent physically impossible configurations.

In this section, we will design a method for placing sub-arrays as they are chosen so that new sub-arrays can be determined until the full array is allocated. To facilitate this procedure, we create a block placement mask with information on the available resources for the array and how pre-existing sub-arrays have already been placed. This mask will be fed back into the profit function so that new sub-arrays will be compatible with the existing sub-arrays.

When the first sub-array is chosen, the full N_T by M_T array is available, and it makes intuitive sense to place the first sub-array at [1,1] index corresponding to the upper left corner of the full array. This step reduces the remaining maximum array

size, and a mask that covers subsequently invalid sub-array options can be created. This mask is initialized with total size N_T by M_T to match the MAU-reduced array with all values set to true, corresponding to the fully empty or available array. As sub-arrays are allocated and *packed*, the locations of the mask corresponding to elements of the array that are now assigned are given a value of false because they are now filled or unavailable. To align this mask with the profit array, it is replicated by the number of unique sub-array configurations (see Figure 5.2), the number of aiming directions, and each target. The replicated mask can be combined with the profit array to remove the profits of physically impossible configurations.

After a sub-array with adequate SINR has been assigned to cover a target, the dimension of the mask corresponding to the updated target is set to false to remove the profit of redundant target updates. This prevents all beams from being assigned to the same target (as it would always have the best profit) to emphasize the coverage of the most targets. The use of the mask prevents the expense of recalculating the total binary map after each sub-array selection, but still requires the recalculation of the profit metric, which is a simple summation.

The more challenging placement decisions begin after the selection of the second sub-array. To allow a more compact final configuration, the MAU mask is reset to true or empty after the selection of each sub-array, and the packing process is repeated. With this fresh mask and two or more sub-arrays to place, the first placement is not obvious. The placement method packs the sub-arrays in order of their total elements so that the smallest sub-arrays are packed last. After the first block is placed in the mask's upper-left corner, the next longest block is placed. To

determine the available locations, an array of ones with dimensions matching the sub-array is convolved with the current full-aperture array. The values of the result greater than zero are the locations where the placement sub-array can be located. This matrix is then multiplied by each element's vertical and horizontal indices. The lowest, non-zero value in the resulting matrix is chosen as the initial index for the next placement to favor dense packing around the upper left corner. If at any point in this process there are no possible placement points, the new mask is discarded, and the mask reverts to the last valid configuration before it was reset. Then a new sub-array with the lowest, non-zero profit is chosen, if such a sub-array exists. If no profitable options exist, the array is fully packed and the array configuration for the next CPI is determined.

The following figures are used to visualize the process of reducing the full aperture from element units to MAUs for calculation simplification, the resulting sub-arrays and their placement due to the energy allocation method and packing, and a visualization of the full-aperture array with the selected sub-arrays. Two cases were chosen to demonstrate differing MAU sizes, subarray selection, and packing in Figure 6.2. Figure 6.1 includes a high-level description of each step.

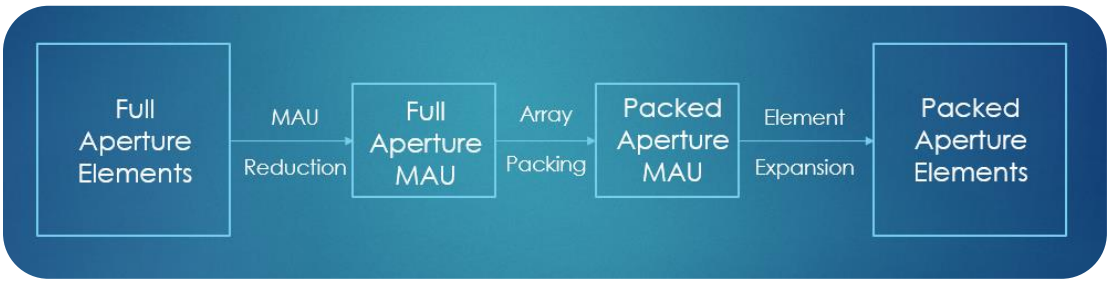

Figure 6.1. Array Allocation flowchart overviewing the process of array size reduction and packing.

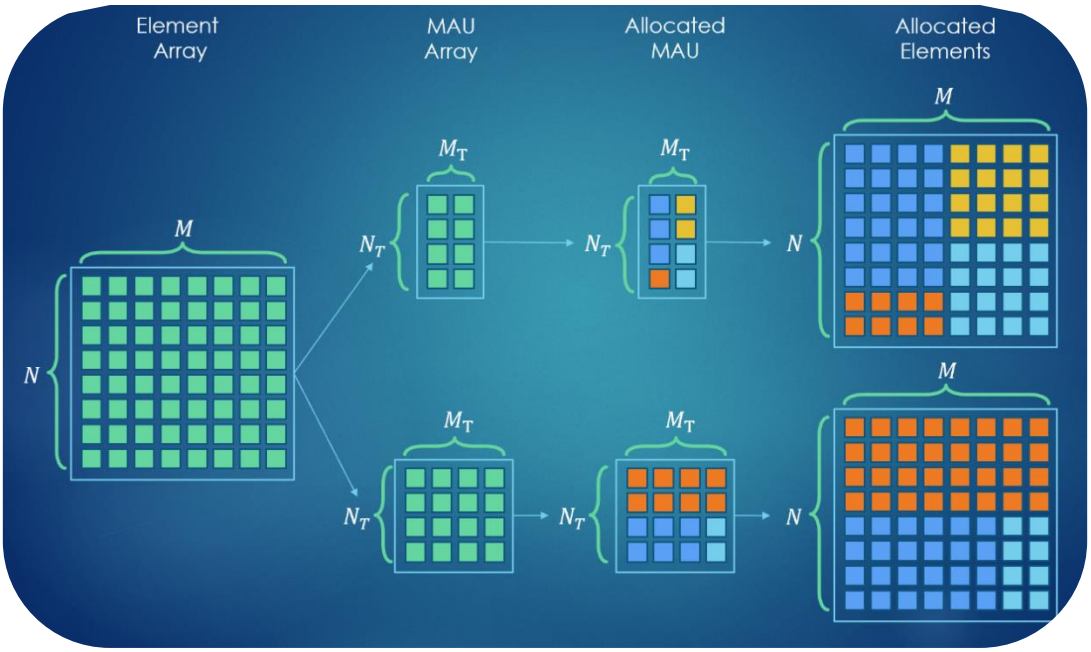

Figure 6.2. Array Allocation flowchart with two examples of MAU Reduction and Array Packing. The element-level configuration is also shown for clarity.

Figure 6.2 includes the same example arrays shown in Figure 5.1 but with the additional steps of sub-array selection and placement. In Figure 6.2, each sub-array was assigned a different color. In the top example the dark blue sub-array is a 3x1 MAU array corresponding to a 6x2 element array, the orange sub-array is a 1x1 MAU array corresponding to a 2x4 element array, the yellow and light blue sub-arrays are 1x2 MAU arrays corresponding to 4x4 element arrays. Each of the sub-arrays are assigned a different aiming direction and update different targets. In the top example, the dark blue sub-array will have the narrowest beam in the vertical dimension and the orange beam will have the widest beam in this dimension. All of these beams will exhibit similar beamshapes in the horizontal dimension.

If we summarize the sub-array placement process, we start with the available MAU and select one sub-array. This sub-array is placed in the upper left corner and a second sub-array is selected from the remaining elements. The placement process

then repeats by placing the largest remaining sub-array until all sub-arrays have been placed. At this point a new sub-array is chosen, and the process repeats until all elements are allocated. Notice the largest sub-arrays in Figure 6.2 are placed near the upper left corner because the method starts placement with the largest sub-arrays in this section of the full array. In the examples of Figure 6.2, four and three total sub-arrays were chosen, but the number of sub-arrays, number of elements in each sub-array, and sub-array orientation will vary. This array packing method is agnostic to the size of the full aperture and can select an arbitrary number of sub-arrays within the limits of the full-aperture dimensions.

6.1 Exceptions

The previous section gave a conceptual and technical overview of an array packing method. This masking method handles most cases but allows problematic quirks that permit the selection of sub-arrays with impossible dimensions. For ease of communication, the first exception is the ‘loner exception,’ where loners are any non-edge-contiguous and unassigned MAU within the array. This exception occurs when sub-arrays are placed so that an un-assigned element has assigned elements between its location and the edge of the array. If we revisit the MAU reduction shown in Figure 5.1 and Figure 5.2 with different sub-arrays we create Figure 6.3, demonstrating an example of the loner exception.

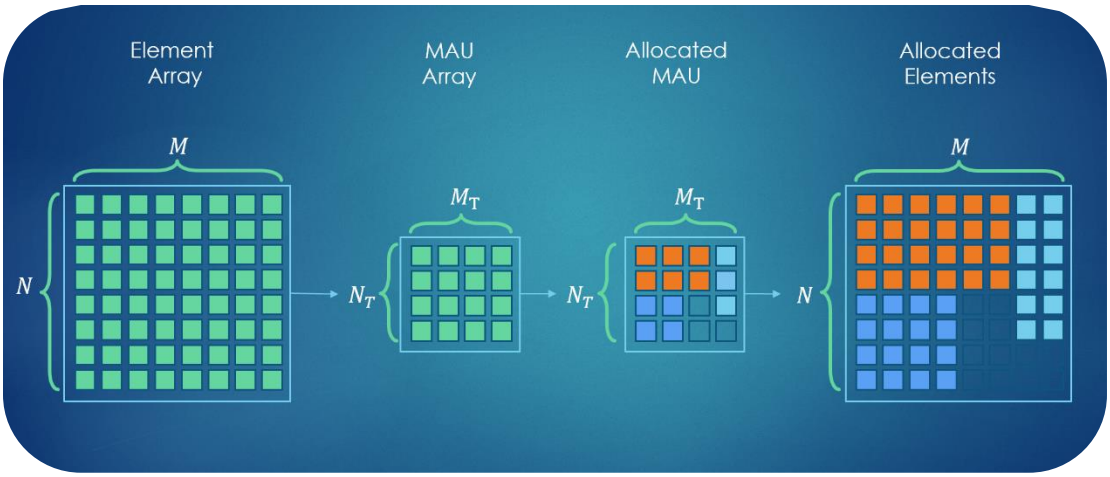

Figure 6.3. Array Allocation flowchart with an example of a potential loner exception.

If, in the example of Figure 6.3, we assume the process of sub-array selection and packing is not yet finished, then the bottom right corner still includes three MAU or 12 elements that have not been allocated. If we additionally assume the test scenario that produced this configuration still has profitable sub-arrays available, the unmodified block placement mask would allow the selection of a 2x2 MAU (4x4 elements) which would conflict with the light blue sub-array, as both sub-arrays would be assigned the same MAU. The simultaneous transmission of two waveforms from this MAU is not possible, so we must include precautions that prevent the selection of these invalid sub-arrays.

A mask that will cover this exception is created by counting the number of elements that would invalidate each sub-array using the block placement mask defined in Section 6. The mask that prevents the loner exception is created by performing a 2D convolution of the block placement mask with a matrix of ones in all locations except a zero at the top left index with total size one larger than the block placement mask or $[N_T + 1, M_T + 1]$. The fourth quadrant (or bottom right quadrant) of the result of this convolution is a mask with zeros at permissible positions. The

non-zero values of the mask count the MAU that prevent the placement of an array in each index. The potential sub-arrays rendered impossible by their loner element(s) can be covered by using the inverse of this loner exception mask in place of the block placement mask. The loner exception mask includes the information of the block placement mask with the loner exceptions removed. For its correct application, its size must be expanded to match the dimensionality of the profit function in the same way as the expansion of the block placement mask in Section 6.

As more of the array is filled, the remaining aperture available for new allocations, sub-array shape flexibility, and number of meaningful target updates decreases, resulting in a consistently reduced probability for the existence of new sub-arrays with non-zero profit. Once only zero profit options remain, the last effort for finding meaningful update candidates is the recalculation of the profit without the priority function. If zero profit options are still the only options that remain, then there is some number of un-allocated elements to handle. They could be used to expand the size of previously allocated sub-arrays, but this could cause non-rectangular arrays, reduce the profit by narrowing the beam width to exclude secondary targets within the same beam, or add more computation for minimal benefit. All un-allocated MAU could send separate waveforms steered toward broadside that could be combined after matched filtering and digitally steered to detect new targets. However, these additional waveforms would act as interference sources that increase the cross-correlation energy of the previously allocated sub-arrays and may reduce the SINR below the acceptable threshold. The chosen solution is to transmit no power from these elements to prevent added interference. Although

this reduces the total energy transmitted, this drawback is less severe than the other methods considered. There may be more beneficial methods for handling these un-allocated MAU that were not considered. Perhaps using the remaining elements for communication or combining the elements into a sparse transmit sub-array with irregular beam pattern could gain value from the remaining elements without unnecessary increases in the cross-correlation.

The described packing method attempts to place sub-arrays densely within a full-aperture without adding undue computational burden. However, slight modifications could improve performance for practical systems. Accepting the original placement without re-packing the sub-arrays could reduce the computational cost while increasing the chance of producing less-dense placement. Methods that combine the packing problem and profit calculation could improve performance. Machine learning algorithms could determine packing trends associated with full-aperture configurations or trends that result after the selection of some number of sub-arrays. The packing order could be tuned to prioritize placing the largest sub-arrays, filling each row or column, or some other parameter that would improve the placement density as influenced by full-aperture size, target distance, and target density.

7 ALLOCATION & PACKING EVALUATION

The proposition of a new programming mode does not have much meaning if it is not tested against other methods to determine its value. To verify the performance of the proposed energy allocation method, figures of merit are defined, and a series of simple control methods are implemented to create benchmarks. This section is intended to provide initial indications of the novel method's performance, and is not comprehensive.

7.1 Operation Modes for Comparison

The first allocation method created was based on a simple target tracking scheme for a full-aperture phased array which will be referred to as the 'greedy' method. In this method, a full-aperture beam is directed towards a different target each CPI. The targets are cycled through sequentially until all targets have been updated, and the process repeats for the specified number of CPI. This method is a brute-force method that should achieve the strongest updates of each target and result in the longest average time between updates for individual targets.

Another method was created that divides the full-aperture into a series of the 2x2 sub-arrays. Each sub-array is aimed at different targets, and each beam is combined coherently on receive to update all targets simultaneously. If there are more targets than sub-arrays, the targets are cycled as in the greedy method, and if there are more sub-arrays than targets, the sub-arrays are cycled so that some targets are updated over sequential CPI. This method is referred to as the 'phased-MIMO' method. The third method is based on the new allocation method described in

Sections 5 through 7, and will be referred to as the ‘energy allocation’ method. It is shortened to ‘EnAll’ within figures.

7.2 Performance Metrics

To compare the performance of the control methods and the new energy allocation methods, figures of merit must be created. As the time between target updates grows, errors can compound, and targets may be lost. Because the allocation method is meant to improve performance in the presence of multiple targets, the update time provides a measure of each method’s ability to keep targets consistently updated. Even if the average update time is minimized, the chance one or more targets will be lost increases with the maximum update time, so it is also measured.

Because the energy allocation method is designed for all-digital architectures and is adaptive, it will likely be implemented as a complementary method of multiple modes on systems with the ability to rapidly change operation mode. For this reason, a history of the estimated RCS is useful so other programming modes could utilize it and smoothly transition into the new energy allocation method. The standard error of the estimated RCS,

$$E_{RCS} = \left| \frac{Ideal - Experimental}{Ideal} \right| \times 100\%, \tag{7.1}$$

indicates each method’s ability to cooperate with the adaptive allocation mode and other RCS-based modes. Because the precision of the RCS error is important for consistent estimates of the return, the RCS variance between updates is examined. The impetus for RCS estimation is the reliable prediction of the return signal strength to minimize the signal strength above a threshold. Each method’s performance in this

category is characterized by the error between the expected and actual return signal strength and is also measured as a standard error. The return signal is more valuable when it is much stronger than the noise and interference because it is more likely to be detected reliably. The average return power is measured to examine the utility of each method's returns.

Now that we have well-defined performance metrics, we must discuss key parameters for the test configuration. The simulation produces a processed datacube with range, Doppler, and angle dimensions, and a target association step uses clairvoyant information on the target locations to make this information usable for the energy allocation method. The association step determines the coordinates of the maximum value within a tolerance region around the target's clairvoyant value of range, Doppler, and angle. This is not practical for real systems but simplifies the evaluation of the allocation mode's performance.

Clairvoyant information is also used for the RCS initialization. We assume a practical system would implement a detection mode before deploying its tracker that would output an acceptable estimation of the RCS constant. Because the RCS calculation will fluctuate between updates due to estimation errors, the mean value of the RCS estimations is used for prediction. Other central tendency methods such as median, mode, or ordered statistics in which outliers are discarded before the central measurements are determined might allow better performance but are not evaluated.

7.3 Results

To evaluate the adaptive allocation method described in Section 5, figures of merit and comparison methods were defined in Sections 7.1 and 7.2 respectively and will be evaluated in this section. Over 100 trials, five unique targets were defined with positions within an allowable set of angles and ranges. These angles and ranges were centered around broadside where the elevation angles can be any value and azimuth angles are limited to positive or negative 90°. The evaluated ranges were defined by a 700 m swath centered around a randomly selected range between 4 km and 14 km. The space for potential targets was limited by these upper and lower range limits and the possible angles.

The initial target velocities were chosen within a sphere bounded by a maximum velocity of $25\ \frac{m}{s}$, and an acceleration profile is created as a set of random numbers with maximum acceleration $1\ \frac{m}{s^2}$ and smoothing applied. A random RCS is chosen from a uniform distribution with maximum $100\ \frac{1}{m^2}$ and minimum $10\ \frac{1}{m^2}$. The new allocation method described in section 5 and comparison modes of section 7.1 were simulated with an 4x4 element array with 10 W element power. The system was evaluated over seven CPI with eight pulses each. The angular resolution is controlled by an 4x4 2D FFT without zero-padding for computational simplicity. This limited spacing reduces the validity of the assumption that the receive gain is well approximated by the number of total elements as discussed in section 5. Histograms were created for the average update time, RCS error, return power amplitude, and return power error. Tables including central tendency measurements for the update

time, RCS, power value, and power prediction error are also included. These qualities are evaluated for the proposed adaptive allocation method evaluated when the desired SINR $\beta = 12dB$ as well as the phased-MIMO and greedy methods. Additionally, thermal noise is omitted to isolate the performance of the estimations with few sources of error. Figure 7.1 includes a histogram of the target update times for each evaluation mode.

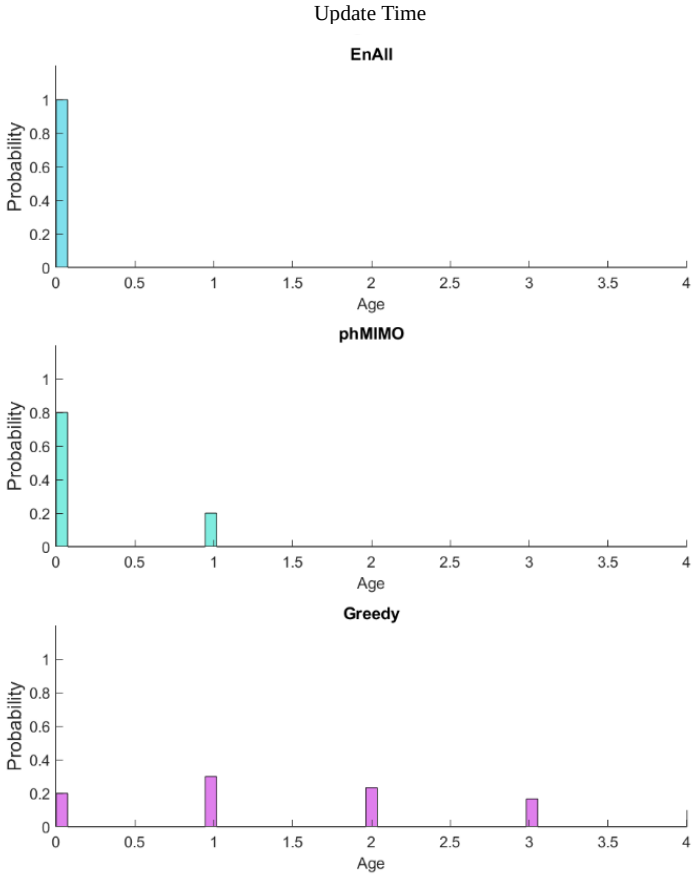

Figure 7.1. Histogram with probability normalization for the target update time with a 4x4 array simulated without noise.

Because the new energy allocation method allows simultaneous update of multiple targets within each CPI, it has the minimum update time, as we would expect. The phased-MIMO exhibits a middling update time because it is split to cover

as many targets as possible but does so without consideration for the best way to split the power. In this set of five targets and 4 MAU (2x2), it can cover all but one target simultaneously, and must cycle in the excluded target each CPI. When the energy allocation method is performing at its worst, it should still perform as well or better than the greedy methods which take a full CPI for each target update. Table 5 includes measurements of the central tendencies of the update time distribution.

Table 5. Update Time Distribution

	Update Time		
	Max	*Mean*	*Std*
EnAll	0	0	0
phMIMO	1	0.2	.4
Greedy	4	1.7	1.2

The priority function of the adaptive allocation method incentivizes the reduction of the maximum update time, but the other methods do not make efforts to reduce the maximum update time. The average update time of the greedy methods will approach the number of targets as the number of CPI increases, but it is lower in our results because the update time is set to zero at the beginning of its execution, and a limited number of CPI are simulated. We will next consider the RCS error for each evaluation method.

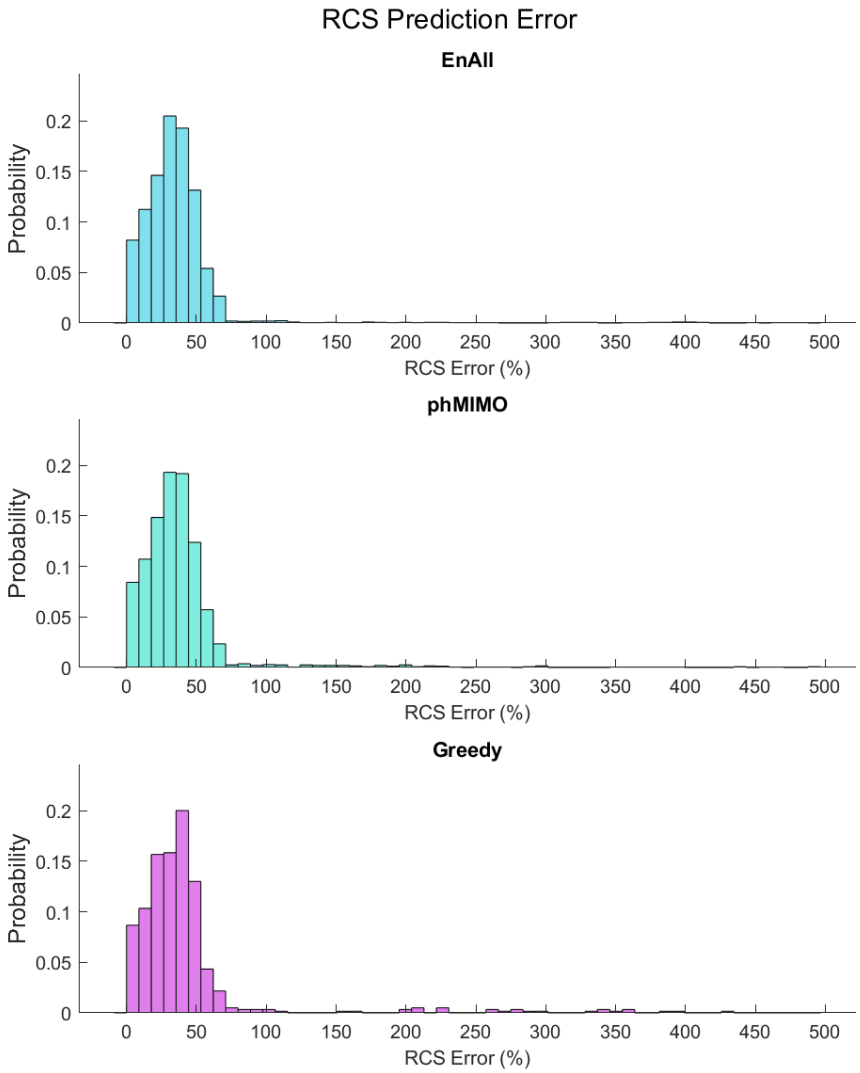

Figure 7.2. Histogram with probability normalization for the difference between the calculated RCS and its true value with a 4x4 array simulated without noise.

The RCS error is similar for all methods. This is reasonable because we assume there is no noise in the environment. We expect the estimation to vary more between methods with additional sources of interference such as noise. Table 6 includes specific measurements from this distribution including the maximum and mean errors.

Table 6. RCS Error Distribution

	RCS Error (%)	
	Max	*Mean*
EnAll	27000	135
phMIMO	22000	144
Greedy	120000	375

Some extreme outliers were not plotted in the previous figure and show up in Table 6. These few outliers inflate the maximum and mean measurements and probably result from the large set of possible ranges. Without the outliers, the mean value is below 50%. This should be acceptable as the RCS is used a prediction tool, and the errors seem to be tightly clustered. RCS measurements with higher precision will have more consistent performance and more potential for correction through calibration. The introduction of additional waveforms reduces the precision of all allocation methods, and the greedy method is expected to exhibit the lowest RCS variance because it transmits only one waveform.

We estimate the RCS for the sole purpose of predicting the return signal strength, so the return power prediction error is the most important evaluation metric. We next consider the error between our predicted and actual power measurement in which the return power of the ‘next’ CPI is compared against the predicted value based on the estimated RCS.

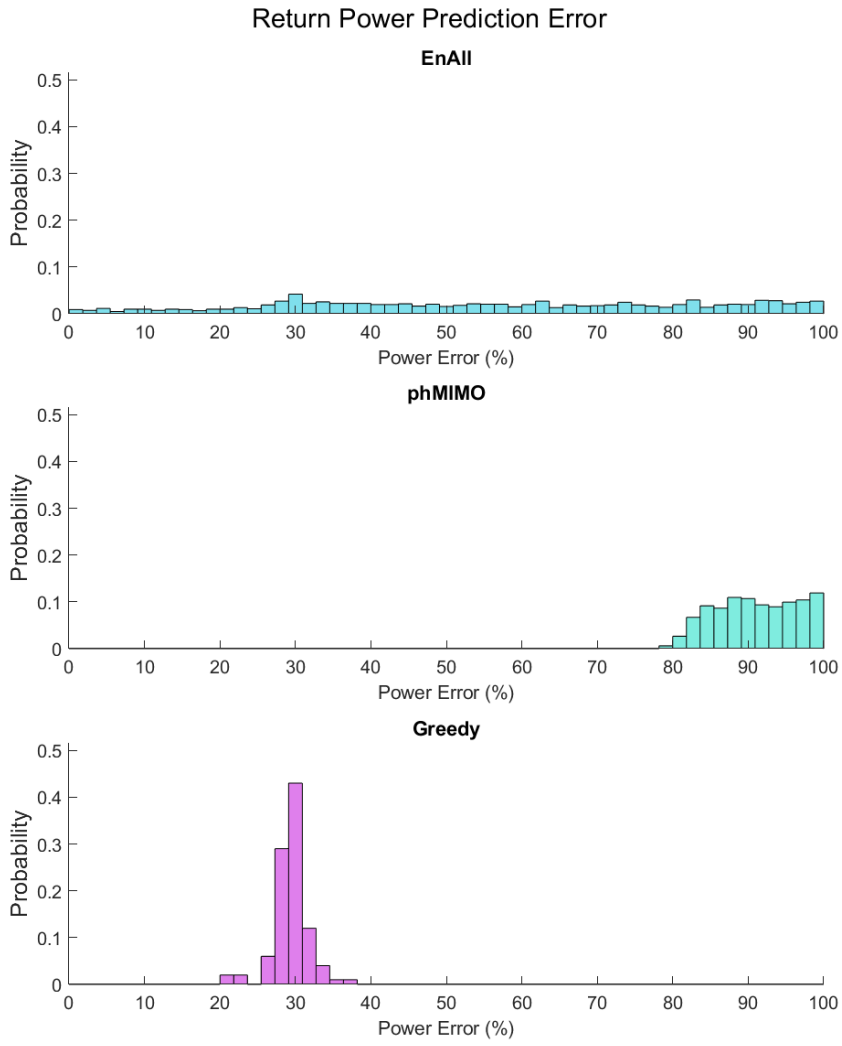

Figure 7.3. Histogram with probability normalization for the difference between the predicted return and its actual value with a 4x4 array simulated without noise.

The return power error seems to remain lower than 100% in most cases. Because the power error is measured as a ratio, an error of 100% will produce a ratio of two between the predicted and actual value. This should be acceptable for our purposes because the energy allocation method has a 3 dB factor built into its requirement on the SINR. This allows reliable detection of high-error target returns even when the return power does not match its prediction. The greedy method displays the lowest power prediction error, and this is expected because it transmits

one waveform from the full aperture and does not introduce cross-correlation effects. The energy allocation method is widely distributed but seems to remain below 100%. This means the energy allocation method should be effective in real systems. The energy allocation method's efforts to mitigate cross-correlation effects through prediction and reduced number of transmit waveforms explains its error performance compared to the phased-MIMO method. The phased-MIMO method always operates with the maximum cross-correlation (allowed by the MAU size) and does not attempt to reduce cross-correlation through intelligent waveform steering. Because the phased-MIMO method does not account for the cross-correlation, it exhibits the worst prediction error. A table including numeric measurements of the central tendencies of the histograms of Figure 7.3 follows in Table 7.

Table 7. Power Error Distribution

	Power Error (%)		
	Max	*Mean*	*Std*
EnAll	99.97	55.9	27.0
phMIMO	99.99	91.1	5.4
Greedy	36.50	29.4	2.4

Table 7 clearly indicates the trends observed in the power prediction error histograms. The greedy method has the lowest error and most precise measurements. The energy allocation method has lower error than the phased-MIMO method, but at the cost of a wider distribution indicated by its increased standard deviation. The phased-MIMO method exhibits the highest error due to its high cross-correlation without any consideration for its reduction and prevention. Next, we will consider the magnitude of these returns as it pertains to their utility.

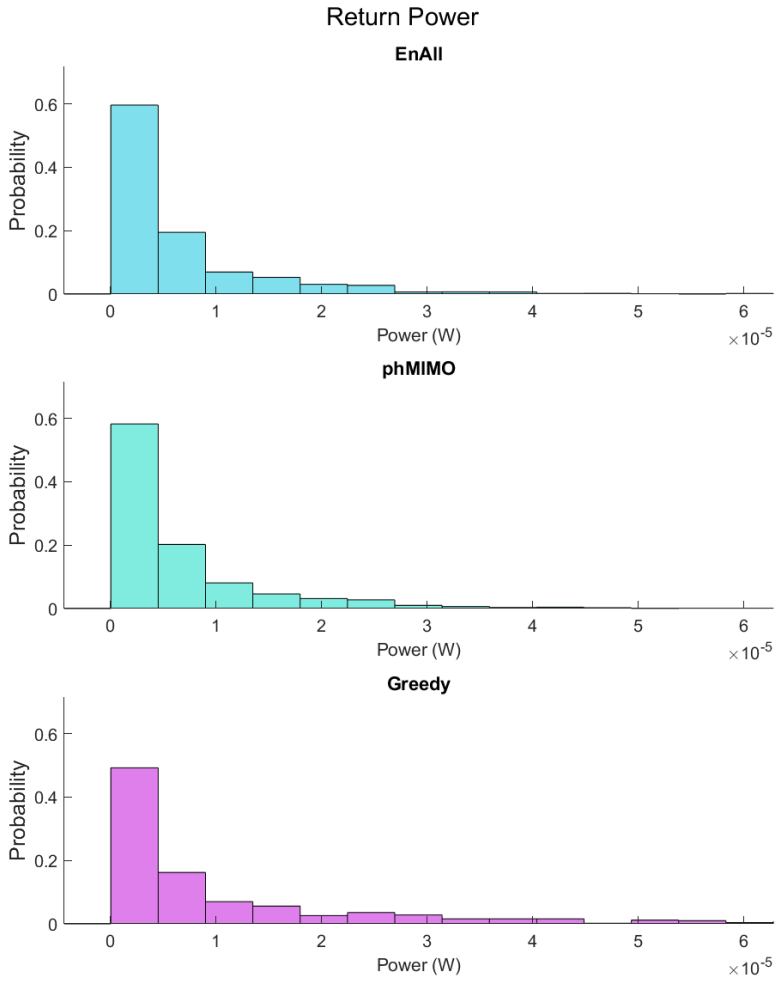

Figure 7.4. Histogram with probability normalization for the average target return when updated with a 4x4 array simulated without noise.

The magnitude of the target returns, independent of their expected value, will be correlated with the performance of these methods in real systems. In these environments, the noise will be less predictable, nefarious electronic warfare systems may be deployed, and other interferences can cover or alter the target returns. Stronger target returns make reliable detections more likely. The proposed allocation method produces weaker target returns because it aims to transmit the minimal energy resulting in useful returns. This reduces the resources wasted on unnecessary increases in SINR at the cost of increased computation and weaker average returns.

This illustrates the method's value in situations in which the RF hardware limits are stricter than those on the computational systems as well as target-dense environments and low-SNR environments. The greedy method transmits using the full array and produces stronger returns, though this is not obvious in Table 8 because the energy magnitude is spread by the variance in possible target ranges.

Table 8. Power Distribution

	Power (mW)		
	Max	*Mean*	*Std*
EnAll	.06	.01	.01
phMIMO	.07	.01	.01
Greedy	.25	.01	.03

Next, we will reevaluate the previous environment with the addition of thermal noise. Because the thermal noise will degrade the SNR and we simulate a relatively small 4x4 array, we increase the element power to 1000 W, increase the number of pulses to 16, and spawn seven targets closer to the radar so that their center range can vary from 1 km to 1.4 km. Additionally, we increase the spatial resolution by zero-padding our FFT operation to create an 8x8 array of spatial samples. Real systems will have more elements with weaker individual powers and be able to cover much further ranges, but we use this unrealistic radar to create an SNR environment more similar to those a real radar would operate within. We imagine the stronger element power models MAU with size of 10x10 Elements where each element has a maximum transmit of 10 W. This is not technically correct because the actual elements would produce much narrower beams than our model, but we are interested in examining the possible utility of such an allocation method rather than its actual performance.

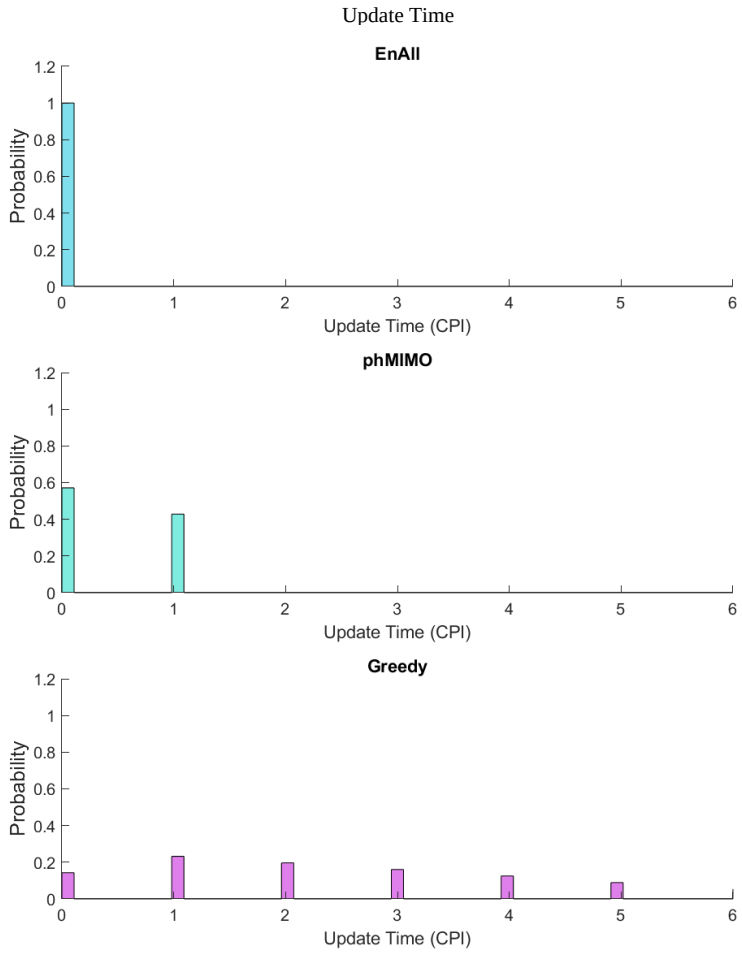

Figure 7.5. Histogram with probability normalization for the update time since the last update for each target with a 4x4 array simulated without noise.

The update time histogram shown in Figure 7.5 demonstrates similar trends to the example without noise. The greedy method has the longest time between updates, the phased-MIMO has the second longest, and the energy allocation method updates all targets simultaneously. Table 9 agrees with these update time observations.

Table 9. Update Time Distribution

	Update Time		
	Max	*Mean*	*Std*
EnAll	0	0	0
phMIMO	1	0.4	.5
Greedy	6	2.4	1.7

The target update order for the greedy and phased-MIMO methods does not depend on noise estimates and should operate in the same sequence regardless of the noise conditions. The energy allocation method will vary depending on the interference environment, but in this case, there was enough power to update each target simultaneously. In lower SNR scenarios, we expect its update time performance to degrade toward the greedy method, its worst-case equivalent. Next, we consider the RCS prediction error in Figure 7.6.

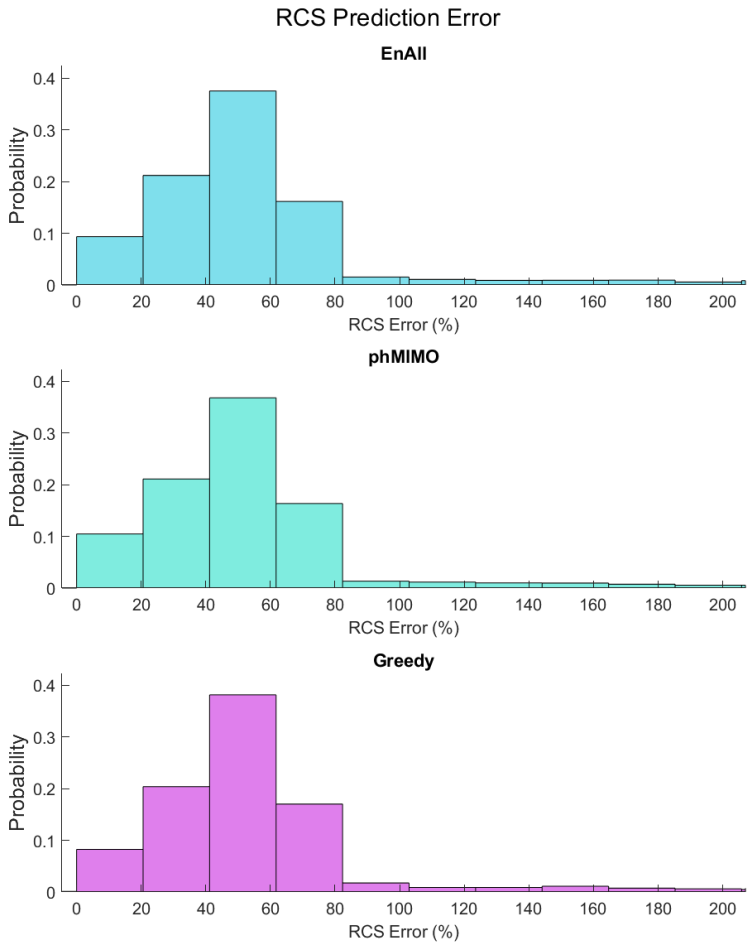

Figure 7.6. Histogram with probability normalization for the difference between the calculated RCS and its true value with a 4x4 array simulated with thermal noise.

The RCS error is similar for each distribution, as it was in the noiseless case, and this is probably because the noise is relatively weak to the signal returns. We expect the greedy method to have the lowest error because it does not have cross-correlation, but this does not appear to be the case.

Table 10. RCS Error Distribution

	RCS Error (%)	
	Max	*Mean*
EnAll	1.1E5	356
phMIMO	1.0E5	344
Greedy	3.3E5	658

Table 10 shows the mean errors are similar for the energy allocation method and phased-MIMO methods. This is probably because the SNR is low enough that the energy allocation method transmits with the same number of MAU as the phased-MIMO method in most of the trials. The greedy method exhibits worse mean and max error, but its distribution in Figure 7.6 seems similar to the other methods. This likely means that its maximum outlier of 3.3E5 percent is causing an artificial inflation of its mean. These RCS estimation errors could be problematic, but our goal with the RCS estimation is to accurately predict the return power. We will now examine the error in the return power.

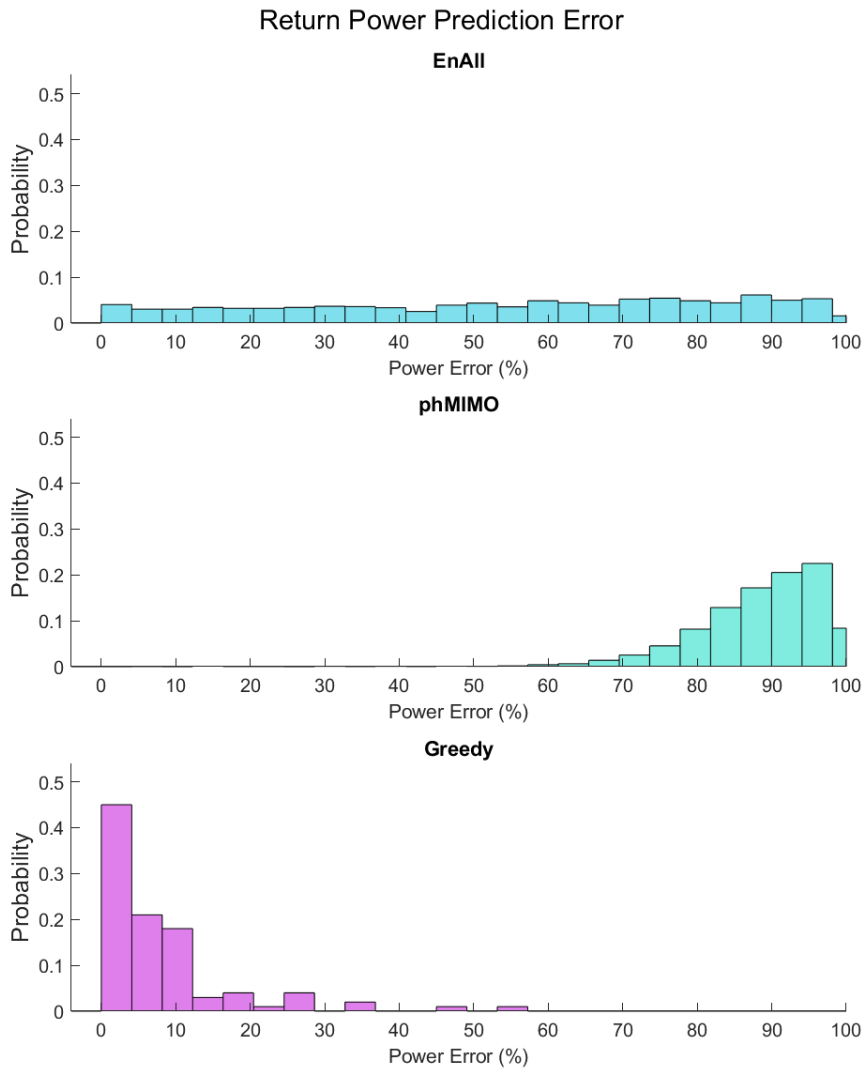

Figure 7.7. Histogram with probability normalization for the difference between the predicted return and its actual value with a 4x4 array simulated with thermal noise.

Although the RCS exhibits large errors, the return power shown in Figure 7.7 is still modeled to a reasonable degree. Although the RCS prediction error is similar for each method, the calculated RCS can be used to effectively predict the return power, and the power prediction errors more closely match our expected results than the RCS errors. The greedy method has the least error because it is not subject to as much interference. The phased-MIMO method exhibits the worst error because its

beam placement does not consider the effects of cross-correlation. The energy allocation method exhibits performance that straddles the performance of these methods because it adds cross-correlation while aiming the beams in directions that attempt to minimize the cross-correlation. Table 11 includes central tendency measurements of the previous distribution.

Table 11. Power Error Distribution

	Power Error (%)		
	Max	*Mean*	*Std*
EnAll	230	55	30
phMIMO	100	88	9.2
Greedy	53	8.1	9.6

Because the targets were moved closer, the mean error decreases. This is probably because the R^4 scaling produces a smaller impact on the digitization errors when the targets are closer. The maximum error for each method increases because the noise allows increased deviation in the return power from the expected value. The greedy performs the best, followed by the energy allocation and phased-MIMO methods, respectively. The energy allocation method has the widest distribution because it is adaptive and allows more variation in the energy configurations, and the other methods demonstrate tighter distributions because their operation is fixed. Finally, we will examine the power of the target returns in Figure 7.8.

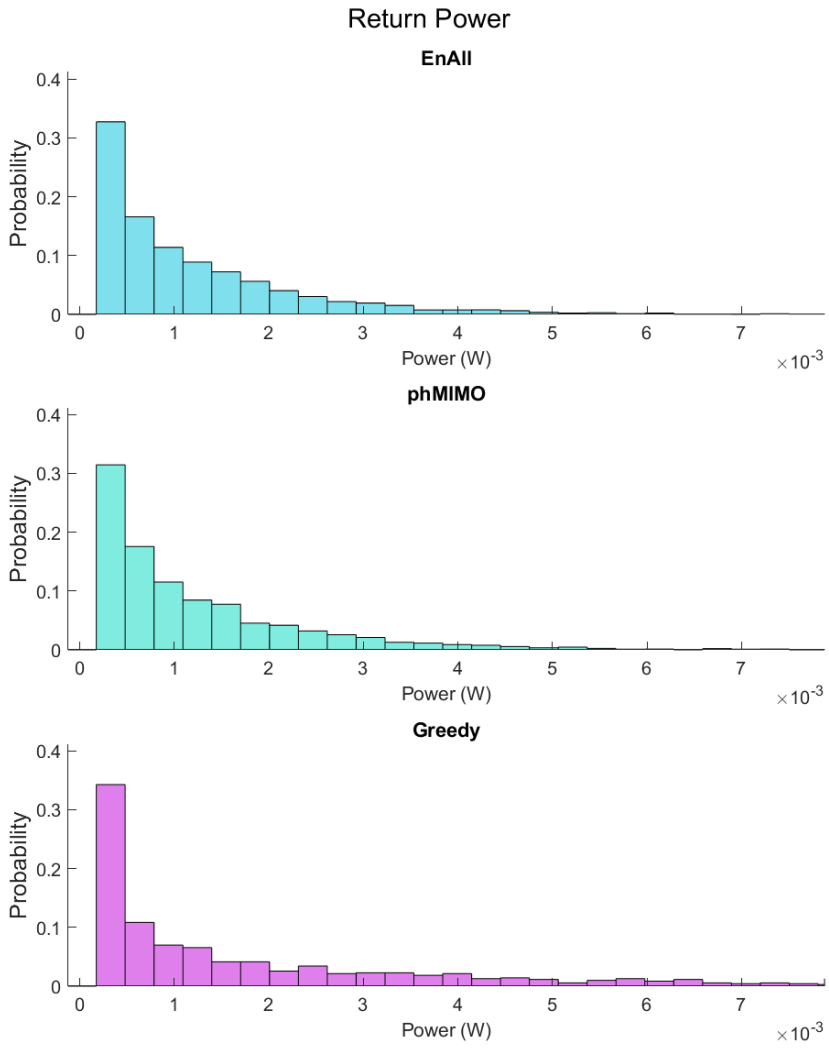

Figure 7.8. Histogram with probability normalization for the average target return when updated with a 4x4 array simulated with thermal noise.

Figure 7.8 demonstrates a similar distribution to the noiseless example of Figure 7.4 but with a more obvious increase in the returns of the greedy method. We expect the power will not vary much between methods but do expect the greedy method to have the strongest returns because it uses the full array. Table 12 further demonstrates this trend of increased power in the greedy method. The similarities in the mean values of the energy allocation and phased-MIMO methods also indicate they tend to transmit using the same sized sub-arrays.

Table 12. Power Distribution

	Power Error (%)		
	Max	*Mean*	*Std*
EnAll	.013	1.2E-3	1.2E-3
phMIMO	.011	1.2E-3	1.2E-3
Greedy	.017	2.2E-3	2.7E-3

We expect the energy allocation method to have higher average power when larger sub-arrays are more consistently required in lower SNR environments. The maximum power is stronger for the energy allocation method than the phased-MIMO method which suggests it was required to employ larger sub-arrays than the phased-MIMO method in some of the trials.

The comparison of the energy allocation method's performance in noisy and noise-less environments demonstrates its dependence on relatively clean signal returns in high SNR environments. In noisy environments, we are still able to predict the power return to an acceptable degree, but the RCS is not as close to the truth value. The increased RCS estimation error means the RCS estimate would be less useful if used in other methods that require more exact RCS estimation, and indicates the RCS estimations may not be useful for predicting new returns. This is because the RCS estimation appears to be a situational measurement that may vary significantly over time. This will always be the case because the RCS can change drastically with different viewing angles depending on the geometry and structure of the reflecting object, but the results indicate the RCS may have large variations even between CPI.

Improved system performance is possible by mitigating the energy spreading effects of range-walk (by limiting the number of pulses per dwell), increasing the spatial resolution by using larger phased-arrays and interpolating the spatial filter

outputs, and other SNR improvement methods. The spreading effects of range-walk indicate the proposed energy allocation method cannot account for MIMO SINR reductions by simply increasing dwell-time with additional pulses as has been suggested by some sources [4]. The next section will draw conclusions from this thesis and present future potentials of this technology.

8 CONCLUSIONS & FUTURE WORK

8.1 Conclusions

Within this thesis, we have expanded the functionality of a MIMO waveform based on Chebyshev Chaotic Polynomials, defined a simulation environment that models the operation of phased-array and MIMO radars, and developed an algorithm for phased-array radar operation with increased reconfigurability. Chebyshev waveforms allow the simple generation of an arbitrary number of MIMO waveforms with bandwidth control and IQ channel support. The simulation environment models waveform propagation, point scatterer targets, and radar signal processing to provide insight into the realizable performance of the waveform modification and the comparison of radar modes. These fundamentals enable the derivation of the proposed energy allocation method and indicate its potential as a useful tool for expanding the flexibility of operation and improving the overall performance of all-digital radar systems.

Verification procedures established the overarching accuracy of the simulation environment and the simulation's ability to model the compromises of MIMO waveforms that are convenient to ignore. This simulation was also used for preliminary evaluation of a new energy allocation method through statistical analysis and comparison with other radar allocation modes. These discoveries and new methods present opportunities for additional research and operational benefits outlined in the next subsection.

8.2 Future Work & New Potentials

Though this study covers some of the valuable realizations and novel methods of all-digital operation, there is always more to be done. We will outline a selection of new known-unknowns found through the efforts of simulation and allocation development, paths to improve the transmit waveforms and allocation method, and possible simulations that would expand our understanding of all-digital architectures and their benefits.

Chebyshev waveforms were employed for their convenient properties of low cross-correlation and simple creation. However, many chaotic polynomials, such as Lorentz waveforms, share similar properties and may outperform Chebyshev polynomials in specific situations. More research into these waveforms, their individual performance, and their specific performance in the allocation method may be valuable. More research on chaotic waveform conditioning to improve auto-correlation and reduce cross-correlation could further improve performance.

Beyond the adjustment of the allocation method's individual components, its overarching design can be enhanced. Decision structures that balance the benefits of combining waveforms against the leaving them separated could produce better RCS estimates. Additionally, the domain of possible solutions for the allocation method was reduced by assuming the number of pulses and the pulse repetition periods are static and that CPI cannot overlap. Processing architectures that allow these parameters to vary or increased reduction of the dimensions could improve performance. The proposed allocation method 'turns off' unused transmit elements to

prevent the addition of unnecessary interference, but alternative approaches could use these wasted elements to add value.

The described allocation method is designed to outline benefits of the all-digital architectures and allows easier description of the method's underlying principles. Redesigning it with a focus on implementation-specific optimization and simplicity could be useful. The method operates recursively, and its performance varies with target parameters, so evaluating some figure of merit for the computational complexity and efficiency would be valuable. The method calculates the configuration for each CPI, but determining a batch of CPI configurations or using the same configuration for multiple CPI could overcome potential latency issues related to its processing time. Additionally, some operations could be parallelized and implemented on field-programmable gate arrays, or FPGAs, to further reduce latency.

Other radar modes could benefit from the concepts developed in this thesis and allocation mode. The allocation mode could be adapted for use with detection algorithms by creating a set of hypothetical targets spread over evenly distributed angular divisions or angular divisions emphasizing a detection area. The RCS value of each potential target could be set to the desired minimally detectable RCS and the distance to the desired maximum range of detection. The allocation mode, as currently defined, could then be used to spread the energy coverage in this area so that potential targets above the minimum RCS, within the maximum range, and within the specified coverage angles will produce strong enough returns and provide some optimization of dwell-time for each angle.

A primary obstacle to accurate and precise RCS estimation in low-noise environments is cross-correlation. As outlined in section 3.3.1, sub-array size disparities can result in cross-correlation that overpowers the returns of smaller sub-arrays, and more research into cross-correlation reduction or its accurate prediction would be valuable. Additionally, this research would be useful for other all-digital and MIMO modes and could reduce intra-system interference of multi-function or MIMO radars.

To further evaluate the performance of the energy allocation method, more extensive simulations covering a wider range of target count, distances, and other variations could provide insight into its benefits and disadvantages. Additionally, an adaptation of the allocation method or application of its operational concepts with improved performance in noisy and clutter-rich environments would be interesting and allow its operation in more diverse situations. However, this thesis has outlined a novel energy allocation method and investigated this new and exciting technology.

9 BIBLIOGRAPHY

[1] S.H. Talisa, K.W. O'Haver, et al., "Benefits of digital array radar," *Proc. of the IEEE*, vol. 104, no. 3, Mar. 2016, pp. 530 – 543.

[2] T. Debatty, "Software defined RADAR a state of the art," in *Proc. 2010 2nd International Workshop on Cognitive Information Processing*, Elba, 2010, pp. 253-257.

[3] T. W. Jeffrey, *Phased-Array Radar Design: Application of Radar Fundamentals*. Raleigh, NC: Scitech Publishing, 2009.

[4] J. Bergin and J. R. Guerci, *MIMO Radar: Theory and Application*. Boston, MA: Artech House, 2018.

[5] J. Li and P. Stoica, "MIMO radar with colocated antennas", *IEEE Signal Process. Mag.*, vol. 24, pp. 106-114, Sep. 2007.

[6] F. Daum and J. Huang, "MIMO radar: Snake oil or good idea", *IEEE Aerosp. Electron. Syst. Mag.*, May 2009, pp. 8-12.

[7] S. D. Blunt and E. L. Mokole, "Overview of radar waveform diversity," in *IEEE Aerospace and Electronic Systems Magazine,* vol. 31, no.11, Nov. 2016, pp.2-42.

[8] G. Cui, H. Li and M. Rangaswamy, "MIMO radar waveform design with constant modulus and similarity constraints," in *IEEE Transactions on Signal Processing*, vol. 62, no. 2, 15 Jan. 2014, pp. 343-353.

[9] H. Esmaeili Najafabadi, M. Ataei and M. F. Sabahi, "Chebyshev chaotic polynomials for MIMO radar waveform generation," in *IET Radar, Sonar & Navigation*, vol. 11, no. 2, Feb. 2017, pp. 330-340.

[10] D. E. Hack, L. K. Patton, B. Himed and M. A. Saville, “Detection in passive MIMO radar networks,” in *IEEE Transactions on Signal Processing,* vol. 62, no. 11, 1 June 2014, pp. 2999-3012.

[11] I. Ziskind and M. Wax, “Maximum likelihood localization of multiple sources by alternating projection,” in *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 10, Oct. 1988, pp. 1553-1560.

[12] G. Krieger, “MIMO-SAR: Opportunities and pitfalls,” in *IEEE Transactions on Geoscience and Remote Sensing,* vol. 52, no. 5, May 2014, pp. 2628-2645.

[13] J. Xu, G. Liao, S. Zhu, L. Huang and H. C. So, “Joint range and angle estimation using MIMO radar with frequency diverse array,” in *IEEE Transactions on Signal Processing*, vol. 63, no. 13, 1 July 2015, pp. 3396-3410.

[14] A. Hassanien and S. A. Vorobyov, "Transmit/receive beamforming for MIMO radar with colocated antennas," *2009 IEEE International Conference on Acoustics, Speech and Signal Processing,* Taipei, 2009, pp. 2089-2092.

[15] A. Hassanien and S. A. Vorobyov, "Why the phased-MIMO radar outperforms the phased-array and MIMO radars," *2010 18th European Signal Processing Conference*, Aalborg, 2010, pp. 1234-1238.

[16] A. Hassanien and S. A. Vorobyov, "Phased-MIMO radar: A tradeoff between phased-array and MIMO radars," in *IEEE Trans. on Signal Proc.*, vol. 58, no. 6, June 2010, pp. 3137-3151

[17] M. S. Willsey, K. M. Cuomo and A. V. Oppenheim, "Quasi-Orthogonal wideband radar waveforms based on chaotic systems," in *IEEE Transactions on Aerospace and Electronic Systems*, vol. 47, no. 3, July 2011, pp. 1974-1984.

[18] M. A. Richards, *Fundamentals of Radar Signal Processing*. McGraw-Hill Education, 2014.

[19] C. A. Balanis, *Antenna Theory Analysis and Design*. New York: Wiley, 2016.

[20] H. Rohling, "Radar CFAR thresholding in clutter and multiple target situations," in *IEEE Transactions on Aerospace and Electronic Systems*, vol. AES-19, no. 4, July 1983, pp. 608-621.

[21] S. Watts, "The performance of cell-averaging CFAR systems in sea clutter," in *Record of the IEEE 2000 International Radar Conference* [Cat. No. 00CH37037], Alexandria, VA, 2000, pp. 398-403.

[22] D. Lucking and N. Goodman, "Resource allocation for multi-variate dynamic Gaussian estimation," in *Proc. 2018 IEEE Radar Conference*, 2018.

[23] C. Recknagel and N. Goodman, "Simulation and adaptive sub-array packing for an all-digital phased-array radar," Presented at 2019 IEEE Radar Conference, Boston, MA, 2019.

[24] S. Martello and P. Toth, Knapsack Problems: Algorithms and Computer Implementations. New York, NY: John Wiley & Sons, 1990.

[25] P. Chu and J. Beasley, "A genetic algorithm for the multidimensional knapsack problem," in *Journal of Heuristics,* vol. 4, no. 1, June 1998, pp. 63-86.

[26] E. Horowitz and S. Sahni, "Computing partitions with applications to the knapsack problem," in *Journal of the Association for Computing Machinery,* vol. 21, no. 2, Apr. 1974, pp. 277-292.

[27] P. Sinha, A. A. Zoltners, "The multiple-choice knapsack problem," in *Operations Research,* vol. 27, no. 3, June 1979, pp. 503-515.

[28] K. W. Ross and D. H. K. Tang, “The stochastic knapsack problem,” in IEEE Transactions on Communication, vol. 37, no. 7, July 1989, pp. 740-747.

[29] D. Pisinger, “Algorithms for knapsack problems,” M. S. thesis, The Pennsylvania State University, State College, PA, 1995.

[30] E. F. Knott, “Radar cross section,” in *Radar Handbook*, 2nd Ed., M. Skolnik, Ed. McGraw-Hill, 1990, pp. 11.1-11.34.

[31] D.J. Andersh, M. Hazlett, S.W. Lee, D.D. Reeves, D.P. Sullivan and Y. Chu, "Xpatch: A high frequency electromagnetic-scattering prediction code and environment for complex three-dimensional objects," in *IEEE Antennas Propagat. Mag.*, vol. 36, Feb. 1994, , pp. 65-69.

[32] R. Bhalla, H. Ling, J. Moore, S.W. Lee, D.J. Andersh and J. Hughes, "3-D scattering center representation of complex targets using shooting and bouncing ray technique: A review,” in *IEEE Antennas and Propagation Magazine*, vol. 40, no. 5, Oct. 1998, pp. 30-39.