DEVELOPMENT OF A COGNITIVE ARRAY SYSTEM

by

Raymond Joseph Weber

A thesis submitted in partial fulfillment
of the requirements for the degree

of

Master of Science

in

Electrical Engineering

MONTANA STATE UNIVERSITY
Bozeman, Montana

April 2010

© Copyright

by

Raymond Joseph Weber

2010

All Rights Reserved

APPROVAL

of a thesis submitted by

Raymond Joseph Weber

This thesis has been read by each member of the thesis committee and has been found to be satisfactory regarding content, English usage, format, citations, bibliographic style, and consistency, and is ready for submission to the Division of Graduate Education.

Dr. Richard S. Wolff

Approved for the Department of Electrical and Computer Engineering

Dr. Robert C. Maher

Approved for the Division of Graduate Education

Dr. Carl A. Fox

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfullment of the requirements for a master's degree at Montana State University, I agree that the Library shall make it available to borrowers under rules of the Library.

If I have indicated my intention to copyright this thesis by including a copyright notice page, copying is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for permission for extended quotation from or reproduction of this thesis in whole or in parts may be granted only by the copyright holder.

Raymond Joseph Weber

April 2010

ACKNOWLEDGEMENTS

I would like to thank my advisor, Dr. Yikun Huang for giving me the opportunity and to explore many topics of array design and signal processing encompassed within this project. I would also like to thank my committee chair Dr. Richard Wolff, and committee members Andy Olson and Dr. Brock LaMeres for all their guidance and knowledge.

I would also like to give special thanks to Ahmed Khallaayoun, Will Tidd, Aaron Traxinger and Monther Abusultan for all their assistance and contributions to this project.

This project was supported by the Montana Board of Research and Commercialization grant MBRCT # 09-09 and was co-sponsored by Advanced Acoustic Concepts in Bozeman Montana.

TABLE OF CONTENTS

1. INTRODUCTION 1

Adaptive Array Systems 2
Direction of Arrival Concepts 5
Beamforming Concepts 6
Cognitive Radio Systems 6

2. COGNITIVE ARRAY SYSTEM 8

3. DIRECTION OF ARRIVAL ESTIMATION ALGORITHM STUDY 12

Algorithm Review 12
Bartlett Algorithm 14
MVDR Algorithm 15
MUSIC Algorithm 16
Simulation Setup 18
Simulation Results 21
No Calibration Errors 21
Element Phase Calibration Errors 21
Element Amplitude Calibration Errors 25
Effect of Differences in Signal Powers 25
Resolution and Accuracy Simulations 25

4. FREQUENCY SENSING AND SPECTRAL HOLE FINDING 31

5. JOINT FREQUENCY AND DIRECTION OF ARRIVAL ALGORITHM 33

Algorithm - Derivation 33
Algorithm - Procedure 39
Simulation Study 41
Frequency Estimation Simulation Study 43
Direction of Arrival Estimation Simulation Study 44

6. BEAMFORMING METHODS 48

Algorithms 48
Shift and Sum 48
MVDR 49
Simulations 49

TABLE OF CONTENTS – CONTINUED

7. TRACKING NON-STATIONARY PARAMETERS 53

Kalman Filter 53
Simulations 53

8. RF LOCALIZATION STUDY 56

Localization Algorithms 57
DOA Estimation Algorithm 57
TOA Estimation Algorithm 58
TDOA Estimation Algorithm 58
Simulations 59

9. SYSTEM CALIBRATION METHODS 63

Beamformer Calibration 63
Method 63
Results 64
Reciever Calibration 65
Method 65
Results 67

10. LABORATORY TEST RESULTS 70

Direction of Arrival Estimation Tests 71
Joint Direction and Frequency of Arrival Estimation Test 72
Beamforming Test 74

11. FUTURE WORK 75

12. CONCLUSIONS 76

REFERENCES CITED 77

APPENDICES 82

APPENDIX A: MATLAB Code 83
APPENDIX B: Additional Calibration Spectra Datasets 90

LIST OF TABLES

Table Page

5.1 Paired Frequency and DOA Estimation Simulation Parameters 42

5.2 Paired Frequency and DOA Estimation Results for 20dB SNR 42

5.3 Paired Frequency and DOA Estimation Results for 40dB SNR 42

10.1 Lab Results for Joint Frequency and Angle Estimation Algorithm vs. Conventional MUSIC 73

LIST OF FIGURES

Figure Page

1.1 Open and Closed Loop Beamforming System Diagrams 3

1.2 Antenna beams switching between two targets 6

2.1 Cognitive Array System Diagram 9

2.2 System Hardware 11

3.1 Comparison of Theoretical Spectra for Bartlett, MVDR and MUSIC with 20dB SNR and signals at 180 and 270 degrees 14

3.2 Collected Pseudo Power Spectra showing properties 19

3.3 DOA Estimation: Two Signal Histograms and Power Plots 22

3.4 DOA Histograms: Two Signals with Uniform Phase Errors 23

3.5 DOA Power Plots: Two Signals with Uniform Phase Errors 24

3.6 DOA Histograms: Two Signals with Uniform Amplitude Errors 26

3.7 DOA Power Plots: Two Signals with Uniform Amplitude Errors 27

3.8 DOA Estimation Histogram and Power Plots with two signals and a 10× stronger fixed signal power 28

3.9 Uniform Phase Error vs Resolution for the 8 element UCA with MVDR and MUSIC 29

3.10 Number of Elements vs Resolution for the 8 element UCA with MVDR and MUSIC 30

3.11 Signal To Noise vs Resolution for the 8 element UCA with MVDR and MUSIC 30

4.1 Binary Tree/Energy Detector Spectral Hole Finding Method 32

5.1 Joint Frequency/DOA Estimation Algorithm: Frequency and Signal Detection Pairings 43

5.2 Joint Frequency/DOA Estimation Algorithm: Frequency Estimation Accuracy 44

5.3 Joint Frequency/DOA Estimation Algorithm: Two Signals Overlaid 1D Pseudo-Power Spectrums 44

5.4 Joint Frequency/DOA Estimation Algorithm: 2D Pseudo-Power Spectrum 45

LIST OF FIGURES – CONTINUED

Figure Page

5.5 Joint Frequency/DOA Estimation Algorithm: Number of Signals Detectable vs SNR 46

5.6 Joint Frequency/DOA Estimation Algorithm: Direction of Angle Estimation Accuracy vs Numerous Parameters 46

5.7 Joint Frequency/DOA Estimation Algorithm: Comparison of Mean Accuracy with Standard Deviation of DOA Estimates with Two Sources vs Azimuthal Seperation with MUSIC 47

5.8 Joint Frequency/DOA Estimation Algorithm: Effect of Mutual Coupling on Accuracy vs MUSIC 47

6.1 Comparison of Single Element Antenna vs Shift ans Sum and MVDR Beamforming methods for signals with 50° seperation 50

6.2 Shift and Sum vs MVDR Beam Patterns and Output Signals with a 60° square wave interference and a desired 75° sinusoid 51

6.3 Shift and Sum vs MVDR Beam Patterns and Output Signals with 60° square wave interference with 10x more powerful and a desired 75° sinusoid 51

6.4 LMS Beamformer signal converging to a sinusoidal wave 52

7.1 DOA Estimation Kalman Tracking Simulation Results 54

7.2 Tracking Two Targets with a Kalman Filter 55

8.1 2D DOA Hough Space with the 2D Hough Space Plots projected on the walls 59

8.2 3D DOA Hough Space 60

8.3 2D TDOA Hough Space 60

8.4 3D TDOA Hough Space 61

8.5 Examples of 3d Solids Generated By TDOA Simulations 62

9.1 Comparison of a Fixed Beam Weights to Computed Calibrated Weights . 64

9.2 Comparison of a MATLAB simulated beam to a calibrated beam using the generalized calibration matrix vs. an uncalibrated beam in hardware 65

9.3 Calibration Method Flow Chart 68

9.4 MVDR and MUSIC Spectra with Different Calibration Methods 69

LIST OF FIGURES – CONTINUED

Figure Page

10.1 Chamber Setup for Direction Of Arrival Estimation Tests 70
10.2 Direction Of Arrival Estimation Lab Test 71
10.3 Direction Of Arrival Estimation Two Signal Lab Test 71
10.4 Labratory DOA Estimation Algorithm Pseudo Power Spectra 72
10.5 Labratory DOA Estimation Algorithm Histograms 73
10.6 Labratory Two Source DOA Estimation Algorithm Pseudo Power Spectrums 73
10.7 Beamforming Algorithms Lab Test 74

ABSTRACT

This thesis proposes a design for a cognitive array system for next generation wireless communication systems, combining the techniques of cognitive radios and adaptive array systems. This novel array system allows for the possibility of greater spectral usage and reuse, and improved communication ranges.

In this thesis, numerous algorithms were studied to map an RF environment in both spatial and spectral domains that would be useful in this system. Towards this goal, direction of arrival estimation, frequency sensing and spectral hole finding algorithms were studied, in addition to a joint frequency and direction of arrival estimation algorithm. Beamforming was also studied as a means of improving signal quality and increasing range. Once the direction to a target was found, localization and tracking were studied to further refine the target's position and change in position over time. After the algorithms were studied in simulation to determine their properties, hardware calibration was performed followed by laboratory tests of the methods with a uniform circular array testbed at Montana State University to verify the expected performances.

1

INTRODUCTION

As communications technologies evolve and develop with the assistance of faster computers and digital signal processing methods, many new techniques have been developed to improve the quality, speed and range that can be achieved with wireless communications. Among the topics of current interest for research are Adaptive Array Systems (AASs) and Cognitive Radios (CRs). These technologies are often seen as competing technologies of next generation systems, but as they largely improve different limitations in the systems and as such they can be combined to further improve performance beyond that which either system could achieve on its own.

AASs work through combining signals from multiple antennas to determine the angle to users as well as creating spatial signal filters [1]. This allows for signals to be located spatially and beams electronically steered and focused towards desired targets improving the Signal to Interference Noise Ratio (SINR) as well as reducing the waisted power in un-interesting directions. These systems can also reduce the probability of signal interception and effect of jamming signals. CRs in contrast work by modifying radio parameters such as power, modulation and frequency range to avoid other users and interference sources that are found by monitoring its operating environment [2]. Combining these two systems yeilds a radio system that can fully map the radio users in its local area along with their frequency bands and other relevant information and pick a frequency and beam-pattern that avoids interference to provide the best QoS for a given situation.

The research performed during this project has resulted in several publications, across a wide variety of topics covered in this thesis [3–13].

2

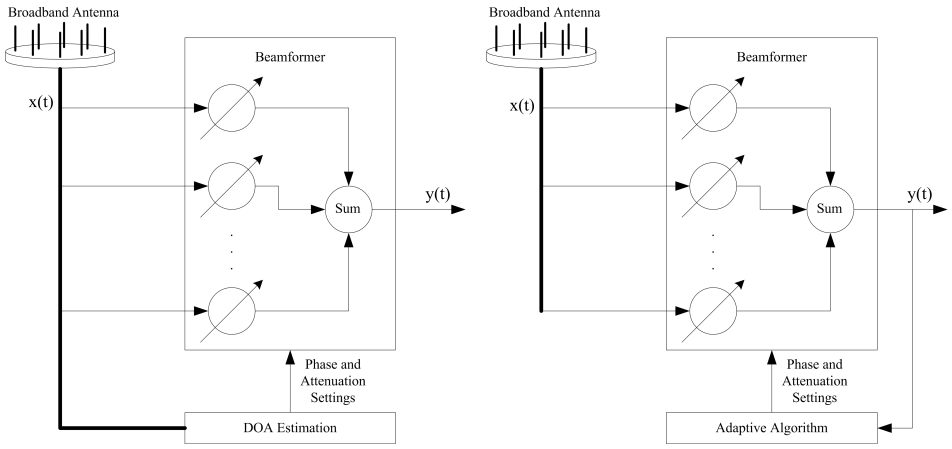
Adaptive Array Systems

Adaptive Array Systems (AASs) for fixed and mobile wireless communications have been the subject of much interest in recent decades. Unlike conventional omni-directional antennas, AASs can form one or multiple beams to system users though beamforming. This can be used to suppress interference, reduce wasted power, and thus greatly improve system performance.

Beamforming is a spatial filtering technique that uses weights (amplitude and phase shifts) that are introduced on signals collected from multiple closely spaced antenna elements. Once the computed phase shifts and amplitudes are changed on the individual elements of the antenna array, the elements waveforms are summed into a single waveform. This technique forms the basis of the "shift and sum" beamformer, which all other beamformer architectures are based on.

These AASs can be separated into two general classes depending on how the weights are determined. In closed adaptive loop systems, the output from the system is monitored and the weights are adjusted to make the output closer to a known ideal statistical measure or desired signal. This result is a set of weights that converge upon the ideal system weights over time and numerous iterations generally lead to good performance. By contrast, the open loop adaptive beamforming system looks at the signals coming from the multiple antenna elements and computes the weights directly in a single iteration based on a measure such as an estimated direction of arrival that requires no knowledge of the signal in question [14]. The open loop beamforming algorithms were the focus of this thesis.

The array geometry has a great impact on the overall performance and abilities of the system. The most common and simple of these geometries in the Uniform Linear Array (ULA). In this configuration, array elements are placed generally at less



(a) Open Loop Beamformer (b) Closed Loop Beamformer

Figure 1.1: Open and Closed Loop Beamforming System Diagrams

then or equal to $\frac{1}{2}\lambda$ intervals for a desired operating frequency. This structure greatly simplifies the math such that the difference in signals collected between adjacent array elements increases consistently as a signal impinges on the array. This is expressed through use of the spatial signature of the array and is probably the single most used equation in array studies. This equation expresses the phase and magnitude relationships between the array elements as a function of azimuthal and elevation angles, as well as frequency.

For the linear array the spatial signature is easily expressed as:

$$\mathbf{a}(\phi) = \left[1, e^{\frac{j2\pi d}{\lambda} sin(\phi)}, e^{\frac{j2\pi 2d}{\lambda} sin(\phi)} \cdots, e^{\frac{j2\pi(m-1)d}{\lambda} sin(\phi)}\right]^T \tag{1.1}$$

where d is the inter-element spacing λ is the wavelength of the signal received and ϕ is the azimuthal angle. On the linear array, their is no elevation measurements possible and as such no dependence on it in the spatial signature.

With the simplicity of the linear array comes many drawbacks. Among these is that the direction of arrival estimates are ambiguous across the both the axis of the

array for azimuthal angles ($\phi = 360 - \phi$) and nothing can be determined about the elevation without a planar array. In addition the beams change shape depending on the azimuthal angle and the array has blind spots at 0° and 180°. In practice the operation of the array to generally limited to about $110 - 120°$ of a 180° hemisphere.

To combat these limitations, Uniform Circular Arrays (UCA) can be used. These arrays have 360° coverage with negligible differences between beams at any given angle. In addition the added 2nd dimension of elements eliminates the azimuthal ambiguities of linear arrays and leaves only the elevation ambiguities above and below the plane of the array. These arrays also have a better centered frequency response for transmit gain when designed with an inter-element spacing of 0.375λ, which in turn results in higher mutual coupling between antenna elements then would be experienced for the linear array. For receive beamforming and DOA estimation, however, this limitation does not need to be followed, and larger array apertures can be used. The more complex shape also results in a more complex spatial signature, and limits the equations that the the array can use as some algorithms depend on the properties of the linear array.

The spatial signature for circular arrays takes the form:

$$\mathbf{a}(\phi,\theta) = \left[1, e^{j\beta\rho\sin(\theta)\cos\left(\phi-\frac{2\pi}{M}\right)}, e^{j\beta\rho\sin(\theta)\cos\left(\phi-\frac{2\pi 2}{M}\right)} \ldots, e^{j\beta\rho\sin(\theta)\cos\left(\phi-\frac{2\pi(m-1)}{M}\right)}\right]^T \quad (1.2)$$

where M is the number of elements, m is the element number under consideration, ϕ is the azimuthal angle, θ is the elevation angle and $\beta\rho$ is the electrical size of the array. For one dimensional situations, the elevation is fixed to 90°, removing it from the equation. For simplicity the azimuthal angle only will be used through most of this thesis.

The electrical size is found where ρ is the array radius and β is the wavenumber defined as:

$$\beta = \frac{2\pi}{\lambda} \tag{1.3}$$

This research is focused on circular array because of its 360° azimuthal coverage was used, and and also because of access to pre-existing UCA hardware developed at Montana State University [15, 16].

Direction of Arrival Concepts

The use of multiple antennas in an array allows for the use of Direction Of Arrival (DOA) estimation algorithms to be run on the signals collected from the antenna elements. This measure can then be used for any method desiring to know the angle to the source, such as in determining beam weights for an open loop beamforming system. As an additional advantage of this technique, DOA estimation algorithms are independent of the signals' properties and allow for for implementation of blind beamforming.

Algorithms implemented for these techniques can be based on many methods with differing properties. For spectral based directional of arrival methods, a spectrum is computed and the largest value found as the estimated direction of arrival for a signal. These can use Fourier based methods, the matrix structure of the autocorrelation, or be based on the signal subspace. It is also possible to implement parametric methods which directly return a direction estimate, but the approach if often at the cost of accuracy, complexity or the number of signals that can be determined [17].

Beamforming Concepts

The ability to perform beamforming is one of the major advantages of using AASs. These techniques allow for an array of omnidirectional antennas to make itself directional and electronically modify its transmission and receive beam patterns to improve signal quality.

In the example in Figure 1.2, two focused beams can be generated and selected to allow for nodes to communicate in pre-defined time slots.

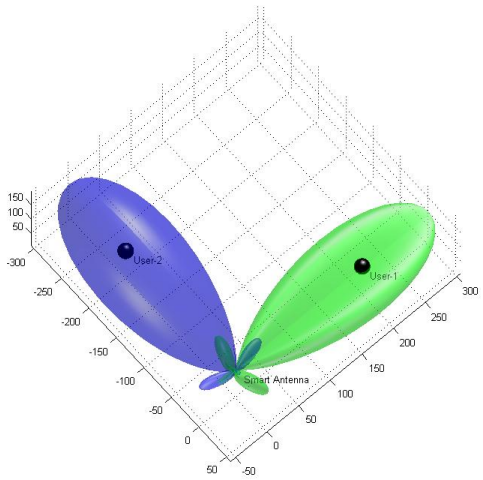


Figure 1.2: Antenna beams switching between two targets

The use of such a system reducing the possibility of interfering with other users, as well as reduce the RF power required to communicate with the remote nodes.

Cognitive Radio Systems

The concept of Cognitive Radios (CRs) was first introduced by Joseph Mitola III in [18]. He proposed that a CR should be an advanced form of a software defined

radio utilizing machine learning techniques to allow for improved communications and Quality of Service for users. This work has been expanded by others in the last several years, including Bruce Fette [2] and Hüseyin Arslan [19]. This has further refined the definition of a cognitive radio and proposed the desired operational methods of a cognitive radio, but left adaptive antenna techniques and spatial filtering as desired future developments for the system.

Cognitive radios in contrast to adaptive arrays work by monitoring the frequency usage in the radio's environment. By analyzing the radio spectrum, the radio system works to modify its operating frequency, power, modulation and other radio based parameters to avoid other users in its environment. These features allow cognitive radios to more efficiently utilize the radio spectrum available and in many cases occupy licensed spectra on a secondary basis, and leave if primary users are detected. This makes cognitive radios of much research interest as unlicensed radio spectrum has become congested, and licensed spectra is poorly utilized.

8

COGNITIVE ARRAY SYSTEM

The Cognitive Array (CA) system proposed and studied in this thesis combines AAS technology with the added ability to determine users and interferer's frequency usage through limited cognitive radio concepts. To achieve this, the signals are down-converted to baseband by the receiver board and digitized with a DAQ. Once the signals are digitized, algorithms are run to locate the signals both spatially and spectrally. This information can then be added to a database such that a signal source can be tracked and it historical tendencies and presence can be noted. From this database it is then possible to determine the best parameters to communicate with a desired node and configure the beamformer and radio appropriately (Figure 2.1).

The result of this system, is that the system is able to map the users within its range in frequency and space and track them as they move or change frequencies. This allows for our communications to then be setup, taking advantage of the adaptive antenna, to not only avoid other users in space, but in frequency and predict possible problems that are going to occur in the future as users move. This would allow for a final implemented system to negotiate new communication parameters to use before an interferer entered the adaptive antenna's beam and corrupts the user's signal and eliminate the problem of trying the re-locate users after they have been lost. This also fits within the desires of cognitive radio systems to be able to exploit holes that exist in the spectrum for a geographic environment as a secondary user can then use the spectrum until such a time that a primary user appears.

This project primarily involved developing the algorithms to fill each of these system blocks and study their abilities, such that they can be later fully integrated into a hardware based system.

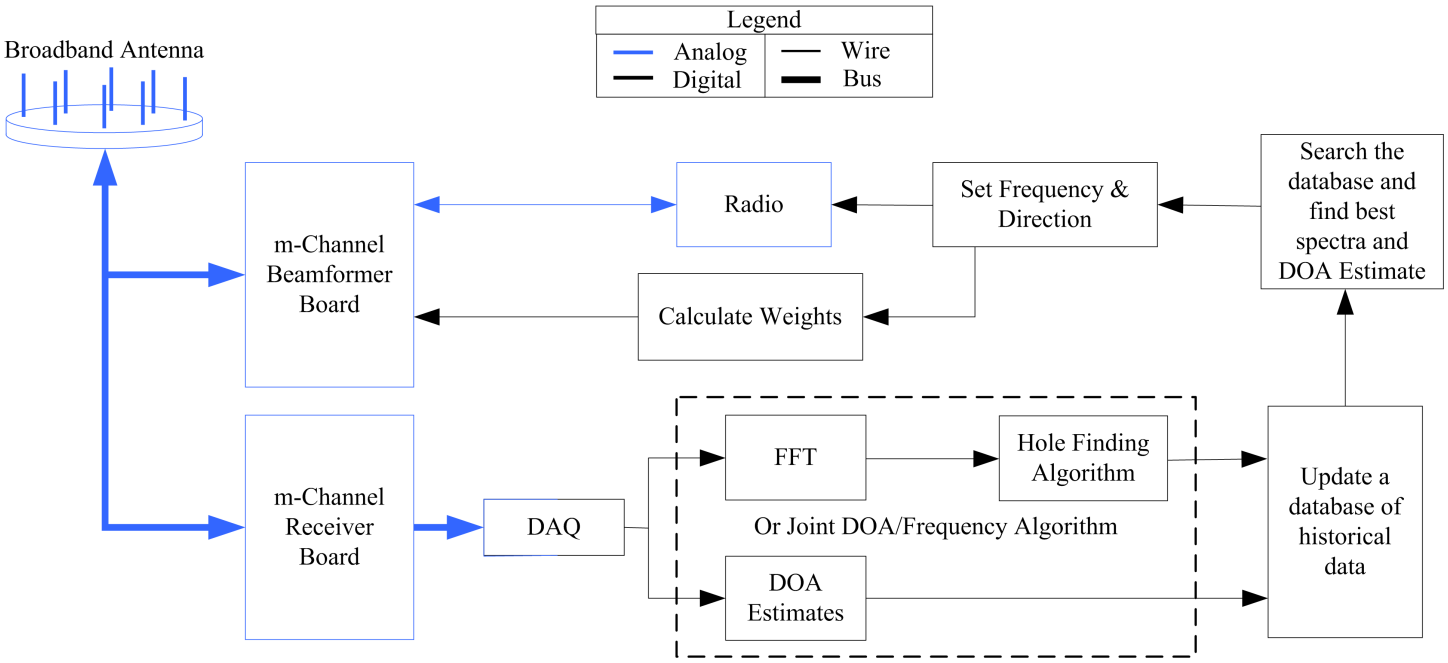


Figure 2.1: Cognitive Array System Diagram

The hardware this research focused on was an 8 element uniform circular array (UCA) of $\frac{1}{4}\lambda$ monopoles with a 0.375λ inter-element spacing with a operating frequency centered at 5.8GHz. These monopoles were on a ground plane with a quarter wavelength skirt simulating an infinite ground plane, resulting in a nearly dipole like response with a slightly lifted elevation pattern. This hardware was developed at Montana State University and as such was available for testing of the subsystem designs during the course of this research [15, 20, 21].

The beamforming board (Figure 2.2) is a hybrid analog digital system with 6 bit digitally controlled phase shifters and attenuators combined with power combiners and splitters to beamform the signals. This hardware allows for approximately 50° beams to be formed approximately every 4° with cophasel excitation or similar performance with windowed beamformers [20].

The second loop of the system comprises of an 8 channel receiver board used to downsample 5.8GHz signals to baseband and filter the signals such that it can be collected by an 8 channel, 2.5MSPS, NI-6133 DAQ using LabVIEW interfaces on a PC testbed and computations performed on the collected signals using LabVIEW or MATLAB [15].

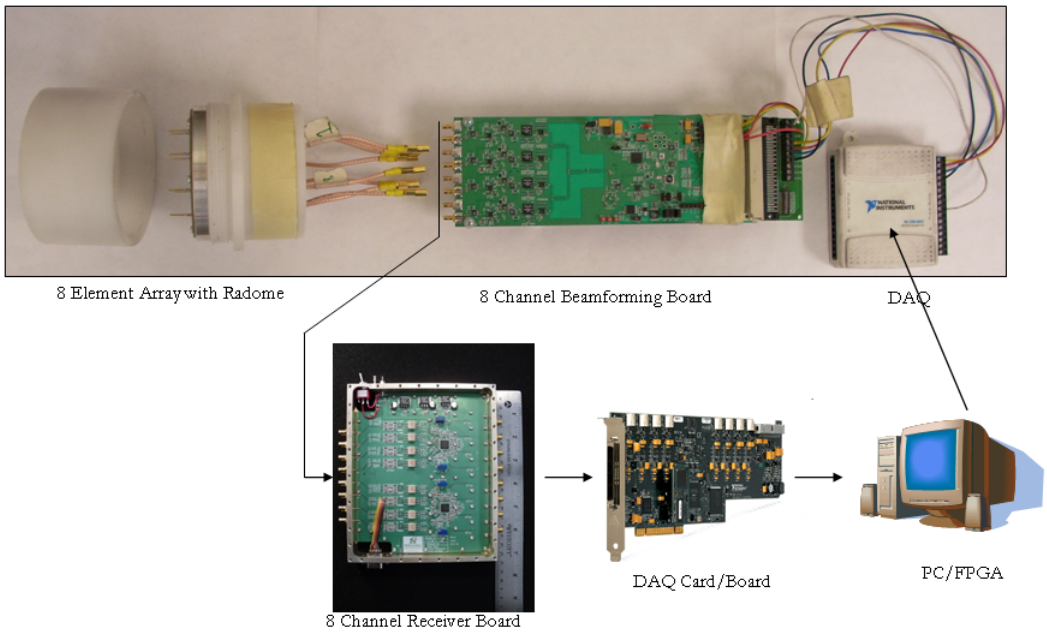


Figure 2.2: System Hardware

DIRECTION OF ARRIVAL ESTIMATION ALGORITHM STUDY

Many algorithms have been developed over the last several decades to perform direction of arrival (DOA) estimation using antenna arrays [17]. From among these, three primary algorithms were chosen to focus research on at MSU: Bartlett, Minimum Variance Distortionless Response (MVDR) and MUltiple SIgnal Classification (MUSIC) [22–24]. These comprise examples from the three primary classes of algorithms and provide good estimates as to how the other algorithms of each class would respond. In addition, the algorithms chosen had to work on arbitrary array geometries or support a circular array which eliminated the popular Estimation of Signal Parameters through Rotational Invariance Techniques (ESPRIT) algorithm from study [25]. In this chapter, these algorithms were studied to determine performance limitations for the algorithms when exposed to simulated element calibration errors of phase and amplitude. This influenced algorithm choices by exposing the parameters that most influenced accuracy and resolution for the different algorithms as well to compare their inherent abilities.

Algorithm Review

Signals from K sources impinge on an array. The received signal at the array has the form:

$$\begin{aligned} \mathbf{X}(t) &= \sum_{k=1}^{K} \mathbf{a}(\phi_k) s_k(t) + \mathbf{n}(t) \\ &= \mathbf{AS}(t) + \mathbf{n}(t) \end{aligned} \tag{3.1}$$

where $\mathbf{a}(\phi_k)$ is the spatial signature defined in Equation 1.2 with $\theta = \frac{\pi}{2}$ and $\mathbf{A}$ is:

$$\mathbf{A} = \left[\mathbf{a}(\phi_1)\vdots\mathbf{a}(\phi_2)\vdots\cdots\vdots\mathbf{a}(\phi_K)\right] \tag{3.2}$$

and $\mathbf{S}(t)$ is the matrix of the signals collected at each array element at a given time:

$$\mathbf{S}(t) = \begin{bmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_K(t) \end{bmatrix} \tag{3.3}$$

and $\mathbf{n}(t)$ is a $M \times T$ matrix of the noise on each element.

and all the algorithms studied used the autocorrelation function defined as:

$$\mathbf{R}_{xx} = E\{\mathbf{X}\mathbf{X}^H\} + E\{\mathbf{n}\mathbf{n}^H\} \tag{3.4}$$

and is estimated on a finite data set as where T is the number of samples collected with:

$$\mathbf{R}_{xx} = \frac{\mathbf{X}\mathbf{X}^H}{T} \tag{3.5}$$

The Root Mean Square Error (RMSE) of the detected angles is defined as:

$$RMSE = \sqrt{\sum_{n=1}^{N} \frac{(\hat{\phi_n} - \phi_n)^2}{N-1}} \tag{3.6}$$

All the algorithms studied here generate a spectra, the peak of which is the estimated DOA estimate, however the shape and properties of the spectra vary between algorithms as shown in Figure 3.1.

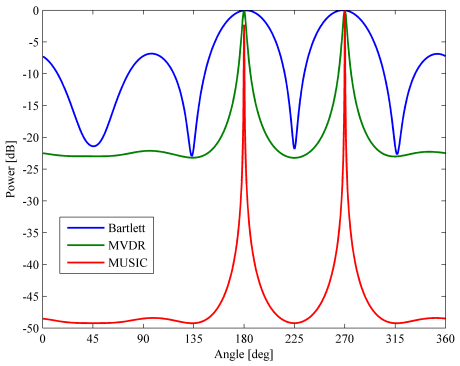


Figure 3.1: Comparison of Theoretical Spectra for Bartlett, MVDR and MUSIC with 20dB SNR and signals at 180 and 270 degrees

Bartlett Algorithm

This method beamforms the signal in all directions and computes the power in a cophasel beam pointed in each direction and is also referred to as spectral analysis. The beam whose angle has the greatest power can then be estimated as the direction of arrival for the signal.

$$\phi = \arg\max_{\phi} \frac{\mathbf{a}^H(\phi)\mathbf{R}_{xx}\mathbf{a}(\phi)}{\mathbf{a}^H(\phi)\mathbf{a}(\phi)} \tag{3.7}$$

and is generally used in a normalized form ($\mathbf{a}(\phi)^H\mathbf{a}(\phi)$ is generally normalized to 1):

$$\phi = \arg\max_{\phi}(\mathbf{a}^H(\phi)\mathbf{R}_{xx}\mathbf{a}(\phi)) \tag{3.8}$$

This method suffers from low resolution and inability to distinguish separated signals of different powers as a result of the high sidelobes which are caused by the algorithm being a Fourier transform in space with a rectangular window. This method works well for a single signal, or if there are two widely separated signals with very similar powers and can be very robust [22].

MVDR Algorithm

Minimum Variance Distortionless Response (MVDR), is a super resolution matrix method that allows for higher resolution DOA then Bartlett. This method is also frequently called Capon's Minimum Variance after its developer J. Capon. This method computes ideal beamweights from the invariance matrices and then, like Bartlett, beamforms in all directions to find a power spectra. However unlike Bartlett, this method results in narrow beams and can distinguish signals that are much closer. In addition the sidelobes for this method are much closer to 0 than Bartlett, resulting in the ability to detect signals of vastly different powers. Bartlett's sidelobes for our 8 element UCA are at about -8dB depending on the array geometry, at which point signals from different sources can become lost. By comparison MVDR's sidelobes are lower resulting in a flatter power spectra and depend more on SNR then on array geometry.

The MVDR power spectra can be found through the equation:

$$\mathbf{P}(\phi) = \frac{1}{\mathbf{a}^H(\phi)\mathbf{R}_{xx}^{-1}\mathbf{a}(\phi)} \tag{3.9}$$

The location of the peak of the power spectrum is then the direction of arrival estimate:

$$\phi = \arg\max_{\phi}(\mathbf{P}(\phi)) \tag{3.10}$$

MVDR is a relatively simple high resolution algorithm and is able to resolve signals that are much closer and have different signal powers [23].

MUSIC Algorithm

MUliple Signal Classification (MUSIC) is the highest resolution algorithm studied and its performance approaches the Cramer Rho Bound (CRB) for the theoretical limit to the performance achievable by an unbiased estimation algorithm [26]. This is achieved through the use of matrix subspace techniques which force the noise and signals into orthogonal subspaces from which the orthogonal matrix can be computed. This method generates a pseudo-power spectra with a theoretical noise floor lower and flatter floor than MVDR, which is also dependent on the SNR.

The autocorrelation can mathematically be shown to be equal to:

$$\begin{aligned} \mathbf{R}_{xx} &= E\{\mathbf{S}\mathbf{S}^H\} \\ &= E\{\mathbf{X}\mathbf{X}^H\} + E\{\mathbf{n}\mathbf{n}^H\} \\ &= \mathbf{A}\mathbf{P}\mathbf{A}^H + \sigma^2\mathbf{I} \end{aligned} \tag{3.11}$$

where $\mathbf{P}$ is a power matrix σ^2 is the noise power and $\mathbf{I}$ is an identity matrix. By use of eigenvalue decomposition, which is performed on the autocorrelation term to compute the eigenvectors ($\mathbf{U}$) and eigenvalues (diagonals of $\mathbf{\Lambda}$):

$$\mathbf{R}_{xx}\mathbf{U} = \mathbf{U}\mathbf{\Lambda} \tag{3.12}$$

This decomposition, when the alternate form of the autocorrelation is inserted and rearranged, shows the relationship between the eigen components and the input signal. This is then further extended to show that the noise eigencomponents ($\mathbf{U}_n$ and $\mathbf{\Lambda}_n$) and signals eigencomponents ($\mathbf{U}_s$ and $\mathbf{\Lambda}_s$) components are linearly independent and

can be used to separate noise and signal subspaces:

$$\begin{aligned}(\mathbf{APA}^H + \sigma^2\mathbf{I})\mathbf{U} &= \mathbf{U}\boldsymbol{\Lambda} \\ \mathbf{APA}^H + \sigma^2\mathbf{I} &= \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^H \\ &= \mathbf{U}_s\boldsymbol{\Lambda}_s\mathbf{U}_s^H + \mathbf{U}_n\boldsymbol{\Lambda}_n\mathbf{U}_n^H\end{aligned} \tag{3.13}$$

The signal powers decompose into the eigenvalues, while the array steering vectors are in the eigenvectors and completely orthogonal ($\mathbf{UU}^H = \mathbf{I}$) and form a complete space.

From this, a noise space can be defined as all the eigenvectors not belonging to signals from the above equation where $\hat{\mathbf{U}}$ is an estimate of $\mathbf{U}$ with limited samples:

$$\hat{\mathbf{U}}_n^H\mathbf{a}(\phi) = 0, \phi\epsilon\{\phi_1, \phi_2, \cdots, \phi_K\} \tag{3.14}$$

and the normalized pseudo-power spectra can be computed as:

$$\mathbf{P}(\phi) = \frac{\mathbf{a}^H(\phi)\mathbf{a}(\phi)}{\mathbf{a}^H(\phi)\hat{\mathbf{U}}_n\hat{\mathbf{U}}_n^H\mathbf{a}(\phi)} \tag{3.15}$$

and in the normalized form written as:

$$\mathbf{P}(\phi) = \frac{1}{\mathbf{a}^H(\phi)\hat{\mathbf{U}}_n\hat{\mathbf{U}}_n^H\mathbf{a}(\phi)} \tag{3.16}$$

The direction of arrival estimates are the peaks of this spectra:

$$\phi = \arg\max_{\phi}(\mathbf{P}(\phi)) \tag{3.17}$$

The result is an extremely high resolution and accurate estimate of the DOA of the signals. The main shortfall of this algorithm is the higher computational complexity from the eigenvalue computation and searches needed.

The speed of these algorithms was improved somewhat through the use of a binary tree search rather then a linear search. To do this, the complete spectrum was scanned in wide steps to determine the approximate locations of the peaks and this was used as a basis for a linear tree peak search to higher accuracy. This resulted in the algorithm being able to do much higher accuracy calculations in the same time by concentrating the peak search in areas where the peak is already known to be from the course search

Simulation Setup

Numerous parameters are of interest when studying DOA estimation algorithms. The parameters focused upon here were resolution, accuracy and analysis of the (pseudo)power spectra results.

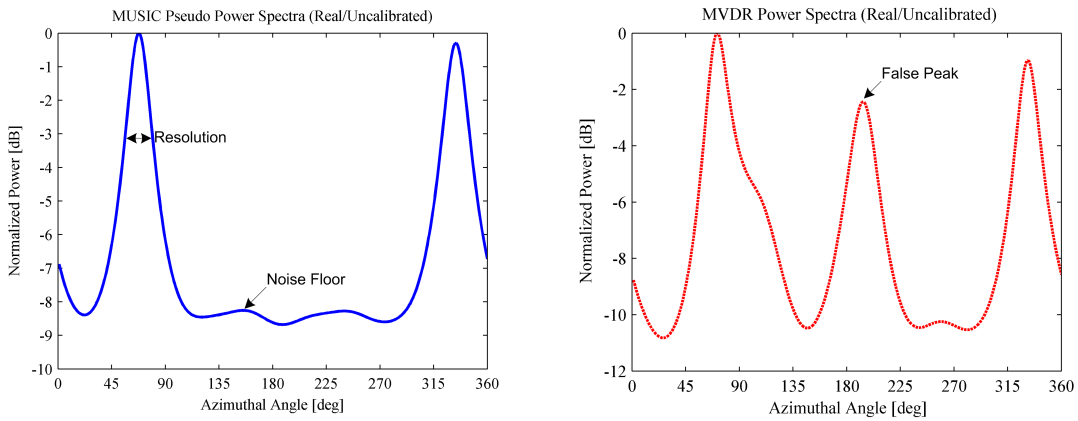
Resolution, when defined for direction of arrival estimation studies, is the minimal angle separation when two signals appear as two signals to the algorithm. When angles under this point are used the algorithm and frequently results in a single angle estimate for two sources that appears somewhere in the middle. The resolution can be estimated from the power spectra plots as the 3dB point, as spacing signals closer than this point will result in a single peak.

Accuracy, by comparison, is the closeness of an estimate to the actual location. We will express this as a root mean square error (RMSE) of the Monte Carlo simulation results.

We look at the pseudo power spectra to locate potential sources of additional error for the direction of arrival estimation. In general, the the greater the difference

between the peaks and the floor narrows the peak and is directly tied to improved resolution. In addition, by viewing these plots we can spot false estimates that in practical situations would be indistinguishable from real signals.

To exemplify these parameters on real pseudo power spectra, Figure 3.2 shows anechoic chamber collected pseudo power spectra for two sources with their distinguishing parameters and imperfections marked. These are due to real hardware imperfections and imperfect calibration (current injection phase corrections only, see chapter 9 for calibration method details).



(a) (b)

Figure 3.2: Collected Pseudo Power Spectra showing properties

To test the performance of these algorithms, the algorithms were implemented in MATLAB and the signals were synthesized at baseband as though they had been collected by an 8 channel receiver board. Monte Carlo simulations were then performed to determine both the accuracy when white Gaussian noise was introduced, as well as when phase and magnitude errors were introduced on an element by element, persistent (for the simulation run) basis. These errors were introduced by modifying

the array signal creation for a single signal from:

$$\mathbf{X}(t) = \mathbf{a}(\phi)\mathbf{s}(t) + \mathbf{n}(t) \tag{3.18}$$

to:

$$\mathbf{X}(t) = diag([AtErr_1, \cdots, AtErr_M]) \times diag([e^{jPhErr_1}, \cdots, e^{jPhErr_M}]) \times \mathbf{a}(\phi)\mathbf{s}(t) + \mathbf{n}(t) \tag{3.19}$$

where the phase and attenuation errors were 8×1 uniform distributed random matrices of the desired magnitudes with zero mean of for $PhErr$ and 1 (eg: 100% of the signals theoretical amplitude is present) for $AtErr$. These amplitude and phase error matrices were element-wise multiplied to the ideal $\mathbf{a}(\phi)$ to introduce the calibration error. The distribution of errors was uniform and distributed around the ideal so, for example, a 30° phase error simulation resulted in element phases being uniformly distributed between $-15°$ and 15° of the theoretical value. This distribution is worse then would be likely in realistic hardware, where an arctangent distribution would be more expected, but a uniform distribution is simpler and gives a lower bound for a worst case error distribution.

The two signals were created at 5.800GHz and 5.801GHz with a 5.8GHz down-sampler and a 2.5MHz sampling frequency. Also, unless otherwise noted, the SNR was fixed at 20dB of additive white Gaussian noise and the array size was set to match the available hardware (8 element UCA with $\beta\rho = 3.05$). For all simulations, two signals were assumed to be present which results in false estimate of a sidelobe when the signals resolve [3].

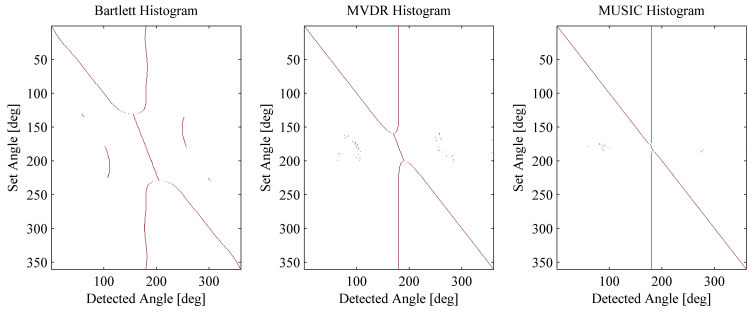
Simulation Results

No Calibration Errors

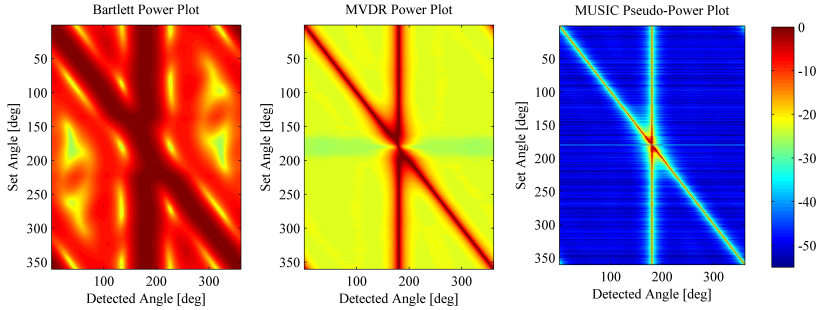
For these simulations, we looked at the output of the system with one signal fixed at 180° and a second signal moving from 0° to 360°. We then plotted the resulting pseudo-power spectra and histogram results as rows and the y axis denotes the ideal angle set for the moving signal. This allows us to visualize the effects on the algorithm of two signals with all possible angular separations in a single plot. Figure 3.3 verified the algorithms worked as expected when their were no calibration errors introduced and detected both signals until angular separation became close enough that signals resolved as clearly visible in Figure 3.3(a) to an angle between the actual angles of arrival. This occurred with Bartlett at 50°, MVDR at 21° and MUSIC at about 6°. Figure 3.3(b) shows the power plots, indicating that Bartlett clearly has high sidelobe levels that could be confused with additional signals, as well as the width of the peaks and depth of the noise floor which make determining the peak location more clear in the more recently developed high resolution techniques.

Element Phase Calibration Errors

For these simulations $PhErr$ in Equation 3.19 was set as a uniformly distributed error, computed separately for every run of the Monte Carlo study while leaving $AtErr$ fixed as 1. Figures 3.4 and 3.5 show the algorithm was susceptible to phase errors and that such errors caused a decrease in accuracy over the entire range of separation of signals. In addition, the power plots show that the peaks quickly become less distinct, but no additional sidelobes or other ambiguities appear.

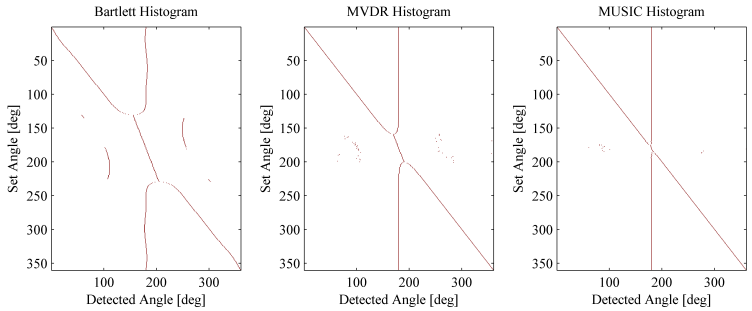


(a) Histograms

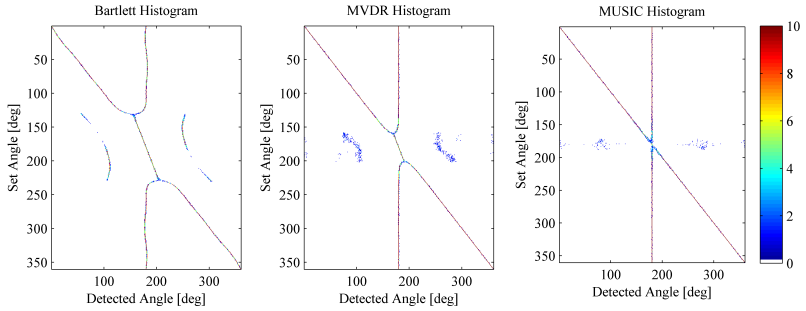


(b) Power Plots

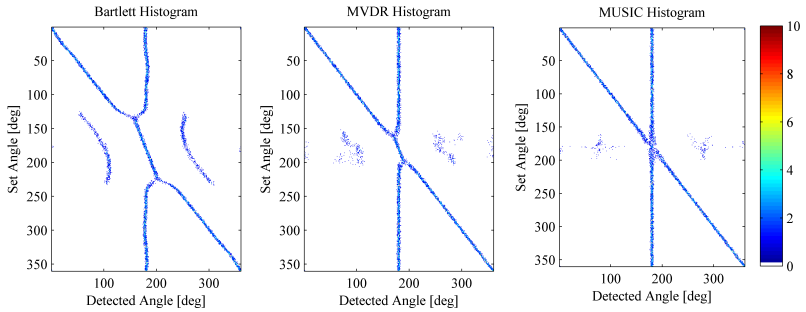
Figure 3.3: DOA Estimation: Two Signal Histograms and Power Plots



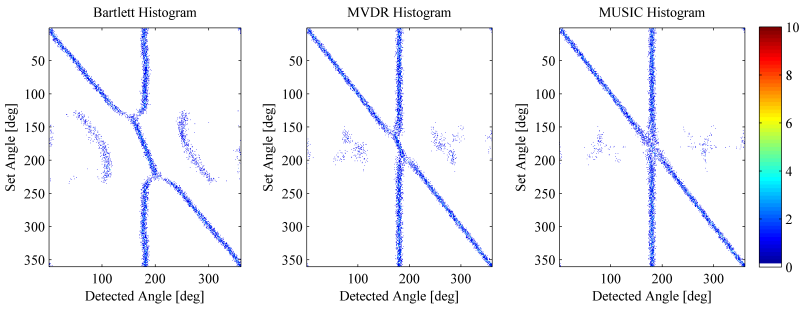
(a) Perfect Calibration



(b) 5° Uniform Phase Error

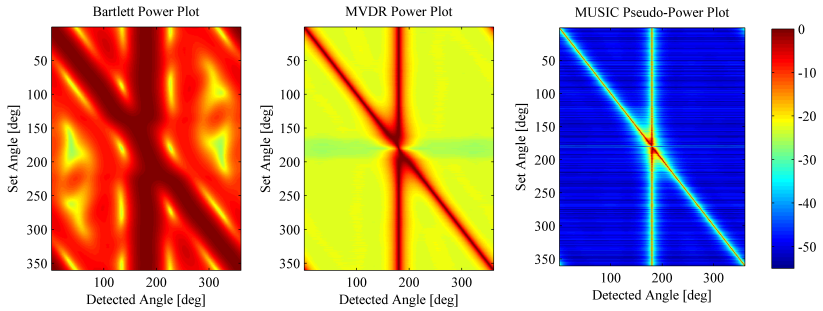


(c) 30° Uniform Phase Error

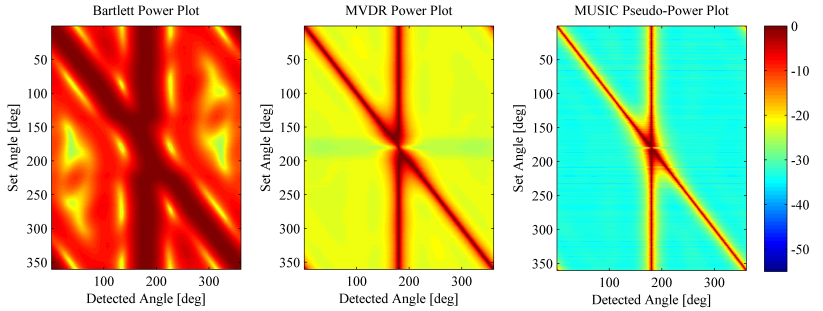


(d) 60° Uniform Phase Error

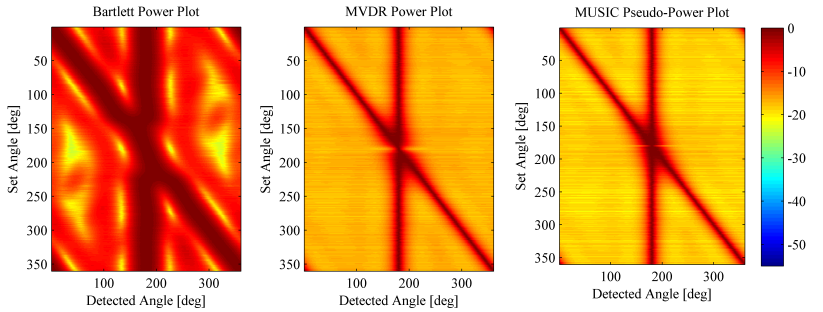
Figure 3.4: DOA Histograms: Two Signals with Uniform Phase Errors



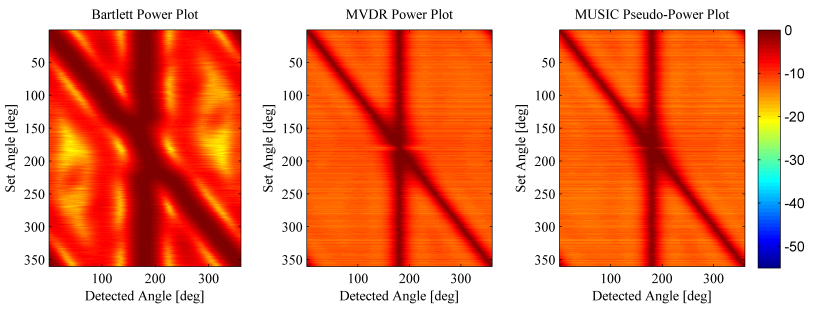
(a) Perfect Calibration



(b) 5° Uniform Phase Error



(c) 30° Uniform Phase Error



(d) 60° Uniform Phase Error

Figure 3.5: DOA Power Plots: Two Signals with Uniform Phase Errors

Element Amplitude Calibration Errors

In addition to testing the effect of element phase errors on the signals, amplitude errors were also examined but were shown to have less effect by changing the $AtErr$ parameter in Equation 3.19 while leaving $PhErr$ at 0°. The simulations results in Figures 3.6 and 3.7 show that amplitude errors have a similar effect to phase errors on the depth of the peaks in the power spectra, but have much less impact on the accuracy, only having a visible effect where the signals are already close to resolving.

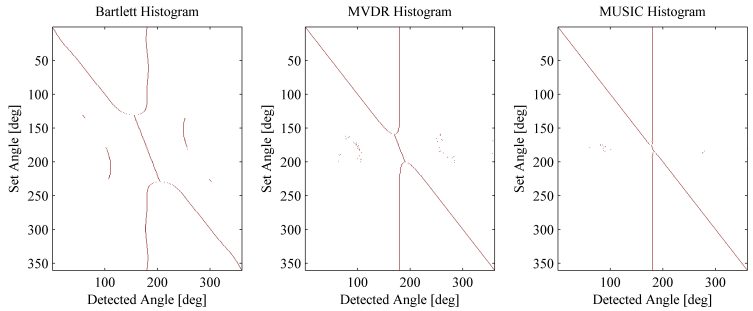
Effect of Differences in Signal Powers

A final simulation using this method was performed to examine the effect of different signal strengths (Figure 3.8). For this simulation all parameters were maintained and perfect calibration assumed, but the fixed signal was made 10× more powerful (100× voltage). This test showed a major weakness of Bartlett where it was incapable of detecting the weaker of the signals, where MVDR and MUSIC both were able to detect it. In addition it showed an interesting result where as MVDR and MUSIC will resolve to the stronger of the signals instead of to the mean angle as was observed in the earlier equal power simulations.

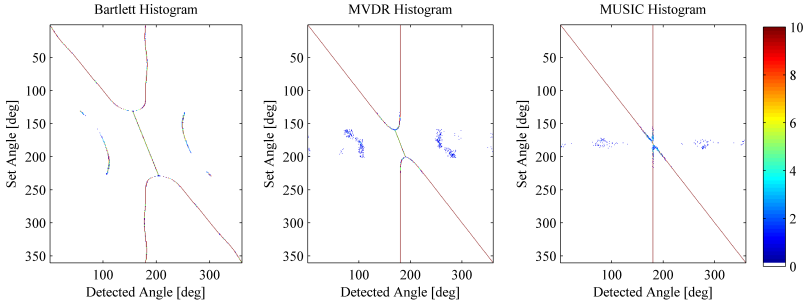
Resolution and Accuracy Simulations

Simulations were performed to more accurately quantify the accuracy and resolution of the algorithms for set situations to determine what factors caused the greatest differences, or if under some situations the more complex algorithms ever performed worse than the simpler methods.

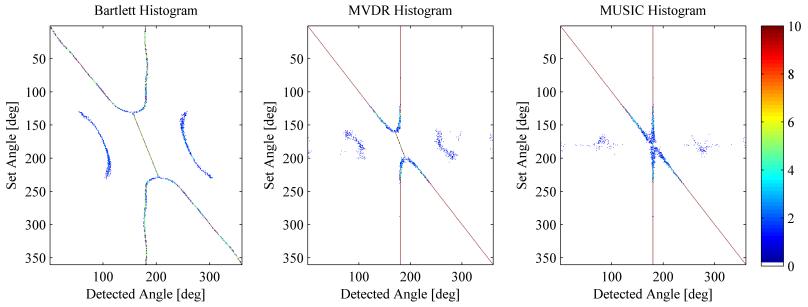
The first simulation (Figure 3.9) based on those in the previous section shows the change in array resolution as a function of the amount of systemic uniform phase



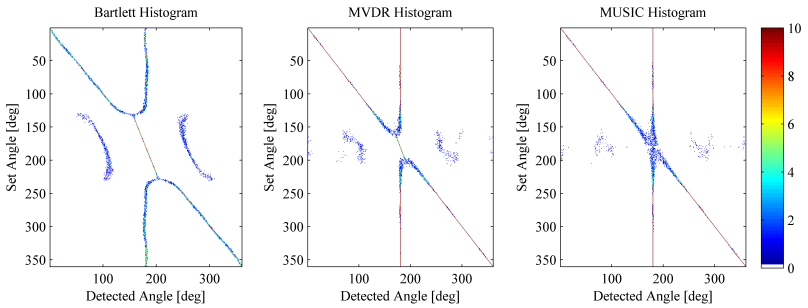
(a) Perfect Calibration



(b) 5% Uniform Amplitude Errors

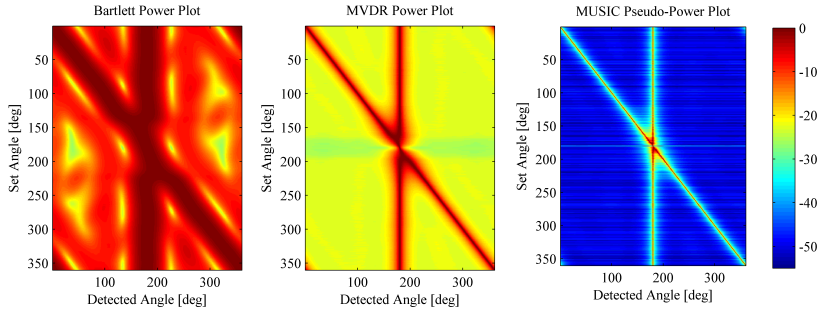


(c) 25% Uniform Amplitude Errors

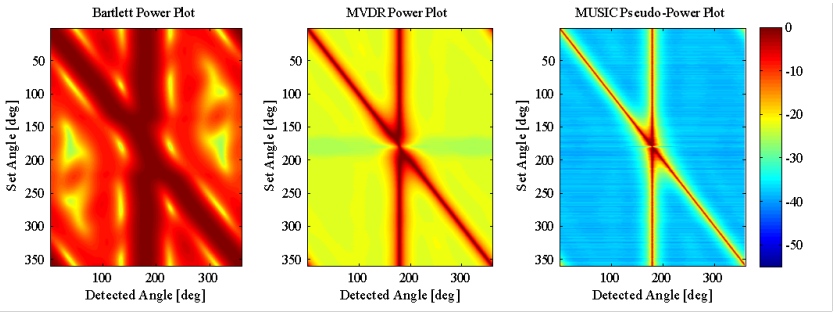


(d) 50% Uniform Amplitude Errors

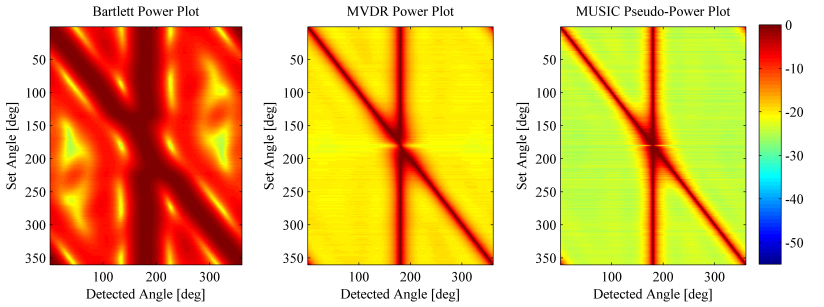
Figure 3.6: DOA Histograms: Two Signals with Uniform Amplitude Errors



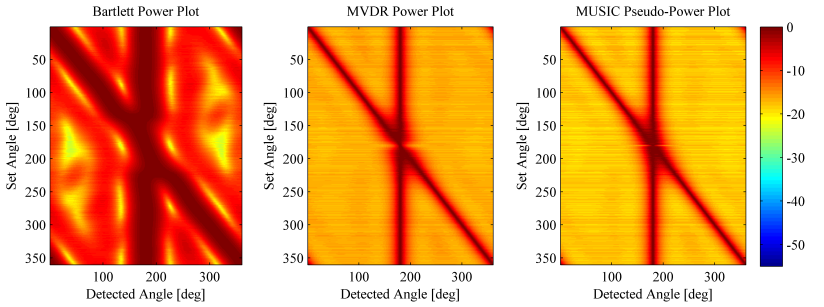
(a) Perfect Calibration



(b) 5% Uniform Amplitude Errors

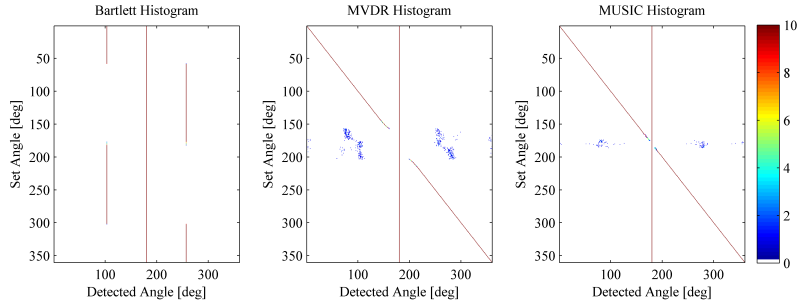


(c) 25% Uniform Amplitude Errors

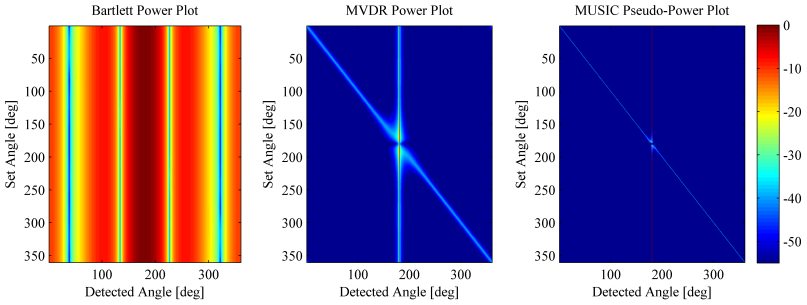


(d) 50% Uniform Amplitude Errors

Figure 3.7: DOA Power Plots: Two Signals with Uniform Amplitude Errors



(a) DOA Estimation Histogram Plots



(b) DOA Estimation Power Plots

Figure 3.8: DOA Estimation Histogram and Power Plots with two signals and a 10× stronger fixed signal power

error that the array is subjected to. This shows that while MUSIC is more sensitive to phase errors, it always maintains better resolution than MVDR.

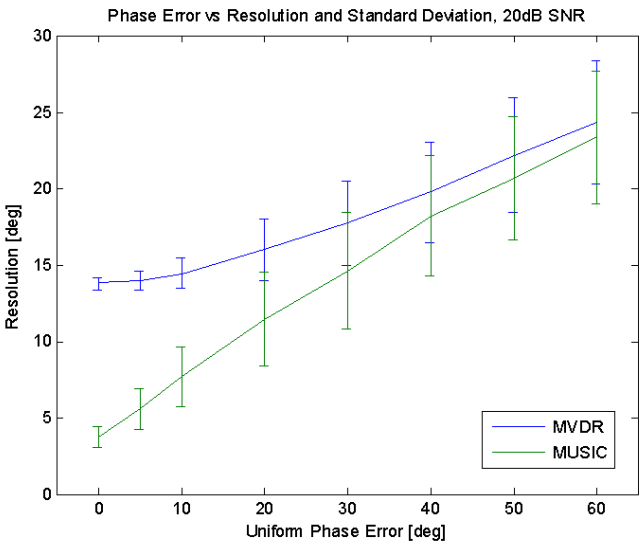


Figure 3.9: Uniform Phase Error vs Resolution for the 8 element UCA with MVDR and MUSIC

The next simulation (Figure 3.10) shows the effect of increasing the number of array elements while maintaining the same interelement spacing. The results show MVDR is an element number biased estimation algorithm.

Another simulation compares the resolution vs. Signal to Noise Ratio (SNR). In this case, the algorithms perform largely the same and MUSIC maintains its performance advantage for all SNRs (Figure 3.11). Bartlett's resolution does not depend on SNR for the values shown in this simulation and remains a constant at a level several degrees higher then MVDR at 0dB SNR.

In conclusion, for all simulations run, MUSIC had better accuracy and resolution than MVDR, despite its increased sensitivity to element errors. In adverse conditions the two methods performed nearly identically, while with clean signals MUSIC had significantly better results.

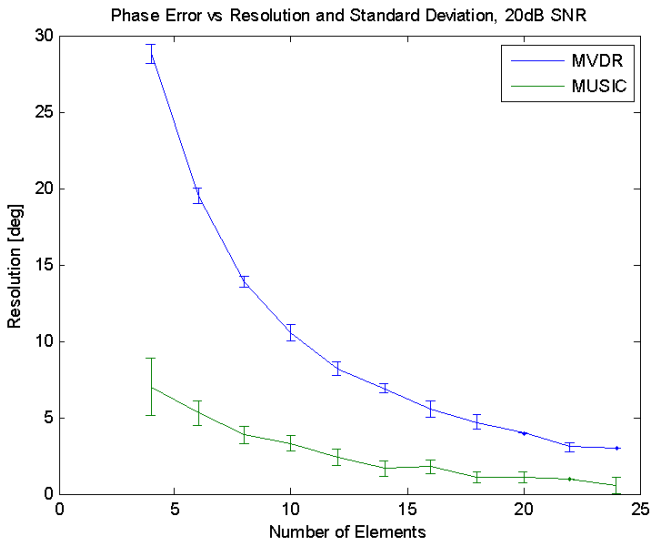


Figure 3.10: Number of Elements vs Resolution for the 8 element UCA with MVDR and MUSIC

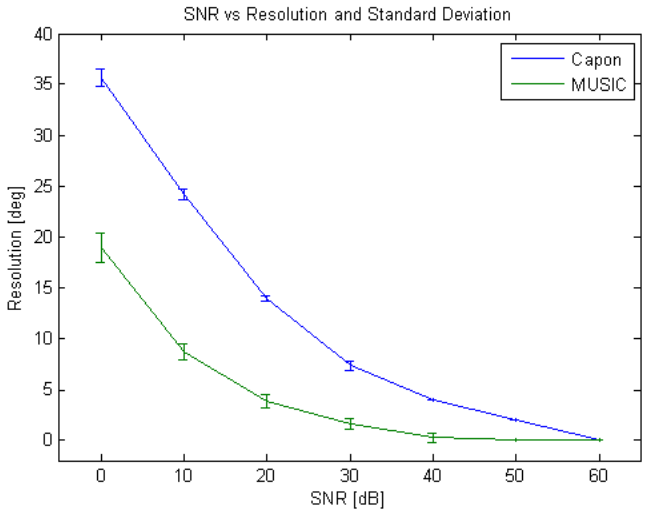


Figure 3.11: Signal To Noise vs Resolution for the 8 element UCA with MVDR and MUSIC

FREQUENCY SENSING AND SPECTRAL HOLE FINDING

With the cognitive radio functionality, it is desirable to know the location of spectral holes that are available for use by the system. There are numerous approaches that can be taken to accomplish this and it is an area of intensive research, but the simple energy detector implemented with a fast Fourier transform (FFT) was chosen for this system due to the general nature of the algorithm to be able to detect any signal power, as well as the availability of the FFT algorithm on hardware based devices such as FPGA platforms. Knowledge of the occupied and free spectrum allow for our system to avoid interference signals, as well as to avoid slow hopping jammers, to find usable spectrum and maintain communication with its users.

The simplest implementation studied for this problem was a single shot FFT that was searched with a binary tree algorithm until a spectral hole of sufficient width was discovered. This was then assumed to be a hole and could be used. Subsequent runs of this would determine if interference sources appear in that hole. We also examined a method where several FFT runs would be time weighted and the search performed to reduce the probability of occupying an already-used portion of the spectrum.

This method has the advantage of simplicity and readily available hardware for quick implementation, but also several limiting factors. First among the liming factors are the limitations of the FFT itself in properly detecting communication signals. While simple and not reliant on the signals of interest having particular statistical parameters, the FFT has a higher rate of missed signals than methods like matched filtering or cyclostationary detectors. In the implantation sense, it is better to beamform the signals and then do the FFT, as this would give a spatial/spectral map that could be used, but this also results in problems as the beamforming is frequency dependent and this would cause minor angle errors, as well as greatly increase the

storage and computational time, resulting in a much slower system. As a result in the efforts of maintaining a practical system using this approach, it was decided that spectral re-use was secondary to speed and the un-beamformed FFT was the most practical. The need for this method can also be eliminated through perfect knowledge of each source's exact frequency and paired DOA estimates as shown in the next chapter.

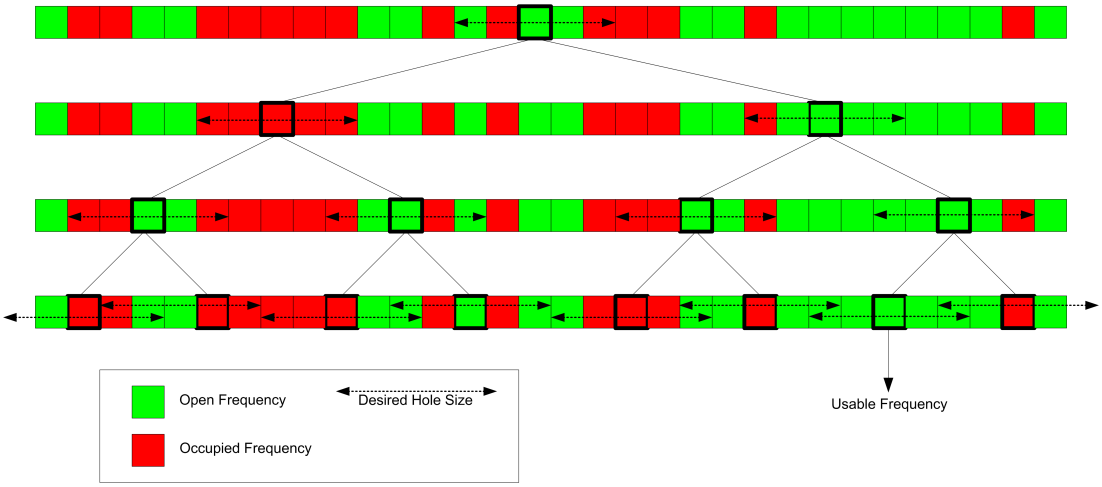


Figure 4.1: Binary Tree/Energy Detector Spectral Hole Finding Method

JOINT FREQUENCY AND DIRECTION OF ARRIVAL ALGORITHM

To better pair spectral usage and directional information, an algorithm was developed based upon the work of Michael Zoltowski [27, 28]. In addition, a temporal version of the algorithm referenced was analyzed by Hui et al. in [29] and was also referenced heavily. The developed algorithm utilized the same frequency estimation process as the cited papers, except instead of using a separate but paired cumulant based DOA, a paired MUSIC-like direction of arrival estimation step directly using a byproduct of the frequency estimation step was utilized. This code as developed is documented in Appendix A.

Algorithm - Derivation

To determine why the algorithm works, numerous works were referenced which have similar methods, but do encompass the entirety of the algorithm and its verification. Primarily referenced for this were works [24, 27–32].

K signals, $s_k(t)$'s, are collected in an array as where $K < M$:

$$\begin{aligned} \mathbf{X}(t) &= \sum_{k=1}^{K} \mathbf{a}_k \mathbf{s}_k(t) + \mathbf{n}(t) \\ &= \mathbf{AS}(\mathbf{t}) + \mathbf{n}(t) \end{aligned} \tag{5.1}$$

where the $(M \times k)$ array manifold is comprised of the steering vectors:

$$\mathbf{A} = \begin{bmatrix} \mathbf{a}_1 & \vdots & \mathbf{a}_2 & \vdots & \cdots & \vdots & \mathbf{a}_K \end{bmatrix} \tag{5.2}$$

where

$$\mathbf{a}(\phi_k) = \left[1, e^{j\beta\rho\cos\left(\phi_k - \frac{2\pi}{M}\right)}, e^{j\beta\rho\cos\left(\phi_k - \frac{2\pi 2}{M}\right)} \cdots, e^{j\beta\rho\cos\left(\phi_k - \frac{2\pi(m-1)}{M}\right)}\right]^T \tag{5.3}$$

and the $(K \times 1)$ signal vector at time t is:

$$\mathbf{S}(t) = \begin{bmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_K(t) \end{bmatrix} \tag{5.4}$$

and the $(M \times 1)$ vector of spatially white Gaussian noise at time t:

$$\mathbf{n}(t) = \begin{bmatrix} n_1(t) \\ n_2(t) \\ \vdots \\ n_M(t) \end{bmatrix} \tag{5.5}$$

A second set of data shifted in time is defined as:

$$\begin{aligned} \mathbf{Y}(t) &= \sum_{k=1}^{K} \mathbf{a}(\phi)_k \mathbf{s}_k(t-\tau) + \mathbf{n}(t-\tau) \\ &= \mathbf{A}\mathbf{S}(t-\tau) + \mathbf{n}(t-\tau) \end{aligned} \tag{5.6}$$

This can also be written as the signal Equation 5.1 with a phase shift:

$$\begin{aligned} \mathbf{Y}(t) &= \sum_{k=1}^{K} \mathbf{a}(\phi)_k e^{(j2\pi f_k \tau)} \mathbf{s}_k(t) + \mathbf{n}(t-\tau) \\ &= \mathbf{A}\mathbf{\Phi}\mathbf{S}(t) + \mathbf{n}(t-\tau) \end{aligned} \tag{5.7}$$

and $\mathbf{\Phi}$ is a $K \times K$ matrix of the time offsets in terms of phase shifts based on the signals frequency:

$$\mathbf{\Phi} = \begin{bmatrix} e^{j2\pi f_1 \tau} & 0 & \cdots & 0 \\ 0 & e^{j2\pi f_2 \tau} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & e^{j2\pi f_K \tau} \end{bmatrix} \tag{5.8}$$

The invariance matrices are then computed where T is the number of samples used for the estimate:

$$\begin{aligned} \mathbf{R}_{xx} &= E\{\mathbf{X}(t)\mathbf{X}(t)^H\} \\ &= E\{\mathbf{A}\mathbf{S}(t)\mathbf{S}(t)^H\mathbf{A}^H\} + E\{\mathbf{n}(t)\mathbf{n}(t)^H\} \\ &= \mathbf{A}\mathbf{P}\mathbf{A}^H + \sigma^2\mathbf{I} \\ &= \mathbf{A}(\mathbf{P} + \sigma\mathbf{I})\mathbf{A}^H \end{aligned} \tag{5.9}$$

$$\begin{aligned} \mathbf{R}_{xy} &= E\{\mathbf{X}(t)\mathbf{Y}(t)^H\} \\ &= E\{\mathbf{A}\mathbf{S}(t)\mathbf{S}(t-\tau)^H\mathbf{A}^H\} + E\{\mathbf{n}(t)\mathbf{n}(t-\tau)^H\} \\ &= E\{\mathbf{A}\mathbf{S}(t)\mathbf{\Phi}^H\mathbf{S}(t)^H\mathbf{A}^H\} + E\{\mathbf{n}(t)\mathbf{n}(t-\tau)^H\} \\ &= E\{\mathbf{A}\mathbf{S}(t)\mathbf{S}(t)^H\mathbf{\Phi}^H\mathbf{A}^H\} + E\{\mathbf{n}(t)\mathbf{n}(t-\tau)^H\} \\ &= \mathbf{A}\mathbf{P}\mathbf{\Phi}^H\mathbf{A}^H \end{aligned} \tag{5.10}$$

where $\mathbf{\Phi}$ can be on either side of $\mathbf{P}$ since $\mathbf{P}$,$\mathbf{S}$, and $\mathbf{\Phi}$ are diagonal square matrices their multiplication is communicative. Since the signals are independent and non-correlated a full rank K Hermitian matrix P, given by:

$$\mathbf{P} = E\{\mathbf{S}(t)\mathbf{S}^H(t)\} \tag{5.11}$$

and estimated with finite data as:

$$\begin{aligned}\hat{\mathbf{P}} &= \frac{1}{T}\begin{bmatrix} \mathbf{s}_1(t)\mathbf{s}_1^H(t) & \mathbf{s}_1(t)\mathbf{s}_2^H(t) & \cdots & \mathbf{s}_1(t)\mathbf{s}_K^H(t) \\ \mathbf{s}_2(t)\mathbf{s}_1^H(t) & \mathbf{s}_2(t)\mathbf{s}_2^H(t) & \cdots & \mathbf{s}_2(t)\mathbf{s}_K^H(t) \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{s}_K(t)\mathbf{s}_1^H(t) & \mathbf{s}_K(t)\mathbf{s}_2^H(t) & \cdots & \mathbf{s}_K(t)\mathbf{s}_K^H(t) \end{bmatrix} \\ &= \begin{bmatrix} p_1 & 0 & \cdots & 0 \\ 0 & p_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & p_K \end{bmatrix}\end{aligned} \tag{5.12}$$

the noise power σ^2 is given by a diagonal matrix since noise is only correlated with itself. This decomposition forms a rank M Hermitian matrix with finite data:

$$\begin{aligned}E\{\mathbf{n}(t)\mathbf{n}^H(t)\} &= \frac{1}{T}\begin{bmatrix} \mathbf{n}_1(t)\mathbf{n}_1^H(t) & \cdots & \mathbf{n}_1(t)\mathbf{n}_M^H(t) \\ \vdots & \ddots & \vdots \\ \mathbf{n}_M(t)\mathbf{n}_1^H(t) & \cdots & \mathbf{n}_M(t)\mathbf{n}_M^H(t) \end{bmatrix} \\ &= \begin{bmatrix} \sigma^2 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \sigma^2 \end{bmatrix} \\ &= \sigma^2\mathbf{I}\end{aligned} \tag{5.13}$$

but is not present in the cross correlation since noise is only correlated if the time and signals match, which never occurs we then have::

$$\begin{aligned} E\{\mathbf{n}(t)\mathbf{n}^H(t-\tau)\} &= \begin{bmatrix} E\{\mathbf{n}_1(t)\mathbf{n}_1^H(t-\tau)\} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & E\{\mathbf{n}_M(t)\mathbf{n}_M^H(t-\tau)\} \end{bmatrix} \\ &= \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \\ &= \mathbf{O} \end{aligned} \tag{5.14}$$

From above, the invariance matrices are defined as:

$$\mathbf{R}_{xx} = \mathbf{A}(\mathbf{P} + \sigma^{\mathbf{2}}\mathbf{I})\mathbf{A}^H \tag{5.15}$$

$$\mathbf{R}_{xy} = \mathbf{A}\mathbf{P}\mathbf{\Phi}^H\mathbf{A}^H \tag{5.16}$$

and a general EVD is performed:

$$\mathbf{R}_{xy}\mathbf{V} = \mathbf{R}_{xx}\mathbf{V}\mathbf{D} \tag{5.17}$$

Where $\mathbf{D}$ is a diagonal matrix of eigenvalues:

$$\mathbf{D} = \begin{bmatrix} \lambda_1 & 0 & 0 & \cdots & 0 \\ 0 & \lambda_2 & 0 & \cdots & 0 \\ 0 & 0 & \lambda_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \lambda_M \end{bmatrix} \tag{5.18}$$

and $\mathbf{V}$ contains the eigenvectors in the columns:

$$\mathbf{V} = \begin{bmatrix} \mathbf{v_1} & \vdots & \mathbf{v_2} & \vdots & \cdots & \vdots & \mathbf{v_M} \end{bmatrix} \tag{5.19}$$

By substituting Equations 5.15 and 5.16 into Equation 5.17 we get:

$$\mathbf{A}\mathbf{P}\mathbf{\Phi}^H\mathbf{A}^H\mathbf{V} = \mathbf{A}(\mathbf{P} + \sigma^2\mathbf{I})\mathbf{A}^H\mathbf{V}\mathbf{D} \tag{5.20}$$

and simplified to:

$$\mathbf{\Phi}^H(\mathbf{P} + \sigma^2\mathbf{I})^{-1}\mathbf{P} = \mathbf{D} \tag{5.21}$$

where $(P + \sigma^2 I)^{-1}P$ is a selection matrix on the diagonal when σ^2 is small, and pulls signal diagonals to 0 when increased or the signal is not present in P)
and since λ are then equal to:

$$\lambda_k = e^{j2\pi f_k \tau} \tag{5.22}$$

Frequencies can be extracted from the eigenvalues simply by:

$$f_k = \frac{angle(\lambda_k)}{2\pi\tau} \tag{5.23}$$

while the remainder of the eigenvalues are from noise and are discarded, but can be identified by their property:

$$|\lambda_n| \approx 0 \tag{5.24}$$

where as actual signals are:

$$|\lambda_k| \approx 1 \tag{5.25}$$

Algorithm - Procedure

In practice however this algorithm is implemented with the following set of procedures:

A real signal is collected from the array of antenna elements and immediately made analytic after digitization with a Hilbert transform:

$$\mathbf{X}(t) = \mathscr{H}\{\mathbf{A}^T(\phi)\mathbf{S}(t)\} \tag{5.26}$$

A second signal has a time shift such that it starts one sample time later:

$$\mathbf{Y}(t) = \mathscr{H}\{\mathbf{A}^T(\phi)\mathbf{S}(t-\tau)\} \tag{5.27}$$

The autocorrelation for $\mathbf{X}(t)$ is found:

$$\mathbf{R}_{xx} = E\{\mathbf{X}(t)\mathbf{X}(t)^H\} \tag{5.28}$$

as well as the cross correlation between $\mathbf{X}(t)$ and $\mathbf{Y}(t)$:

$$\mathbf{R}_{xy} = E\{\mathbf{X}(t)\mathbf{Y}(t)^H\} \tag{5.29}$$

and a general eigenvalue decomposition is performed:

$$\mathbf{R}_{xy}\mathbf{V} = \mathbf{R}_{xx}\mathbf{V}\mathbf{D} \tag{5.30}$$

A boolean indicator of the signals presence in each eigenvalue eigenvector pair can be determined by:

$$|diag(\mathbf{D}_M)| \tag{5.31}$$

and the frequencies are then found with:

$$\mathbf{f}_M = angle\left(\frac{\mathbf{D}_M}{2\pi\tau}\right) \tag{5.32}$$

A MUSIC-like pseudo-power spectrum is computed with:

$$\mathbf{P}_M(\phi) = \frac{1}{|\mathbf{a}_M^H(\phi)\mathbf{\Pi}_{MN}^{\perp}\mathbf{a}_M(\phi)|} \tag{5.33}$$

where:

$$\mathbf{\Pi}_{MN}^{\perp} = [\mathbf{V}_1 \cdots \mathbf{V}_{m-1}, \mathbf{V}_{m+1} \cdots \mathbf{V}_M][\mathbf{V}_1 \cdots \mathbf{V}_{m-1}, \mathbf{V}_{m+1} \cdots \mathbf{V}_M]^H \tag{5.34}$$

The direction of arrival angles can then be found by:

$$\phi_M = \arg\max_{\phi}(\mathbf{P}_M(\phi)) \tag{5.35}$$

Alternatively, the invariance matrices can be computed without use of the Hilbert transform in frequency by using the positive coefficients of the FFT:

$$\mathbf{X}(f) = \mathscr{F}_+\{\mathbf{S}(t) + \mathbf{n}(t)\} \tag{5.36}$$

$$\mathbf{Y}(f) = \mathscr{F}_{+}\{\mathbf{S}(t-\tau) + \mathbf{n}(t-\tau)\} \tag{5.37}$$

and the invariance matrices are then:

$$\mathbf{R}_{xx} = E\{\mathbf{X}(f)\mathbf{X}(f)^{H}\} \tag{5.38}$$

$$\mathbf{R}_{xy} = E\{\mathbf{X}(f)\mathbf{Y}(f)^{H}\} \tag{5.39}$$

and keeping the remainder of the method identical. This has the advantage of eliminating the inverse FFT present in the Hilbert transform from the computation time, as well as permitting for additional filtering to occur in the frequency domain to reduce interference signals from interfering with the parameter estimation. While the end results from this method are very similar, they currently are not as accurate and do not distinguish signals as accurately as performing the algorithm in time.

Another point of interest discovered during the development of the algorithm was that the eigenvector matrix returned from the general eigenvalue decomposition was not orthogonal as is the case with MUSIC. This did not appear to cause any major problems, but may be the responsible for the the increased sensitivity of the algorithm to mutual coupling errors.

Simulation Study

The first simulation study examined the performance and operation of the algorithm on a setup with multiple signals at different frequencies and locations. This study also verified that the pairings were accurate and correct. The signal locations and frequencies are noted in Table 5.1. The results for 20dB and 40dB SNRs are then given in Tables 5.2 and 5.3 showing that the accuracy of the results is reasonable and

improves with SNR to yield highly accurate results. A visualization of the pairings is then shown in 5.1 where each column corresponds to a signal where the first plot is a boolean indicator of the signals presence, the second column in then the frequency followed by azimuth and elevation in the next panels. The ordering of the signals is completely arbitrary. Also of interest is that the elevation angles become less accurate as the signals approach 90°. While this occurs for all DOA estimation algorithms due to the symmetry of the estimates across the plane of the array, is first visible in this simulation [8].

Table 5.1: Paired Frequency and DOA Estimation Simulation Parameters

	Frequency [GHz]	Azimuthal Angle [°]	Elevation Angle [°]
Signal 1	5.1	15	60
Signal 2	5.4	45	30
Signal 3	5.7	278	90
Signal 4	5.8	130	90

Table 5.2: Paired Frequency and DOA Estimation Results for 20dB SNR

	Frequency Estimate		Azimuthal Estimate		Elevation Estimate	
	Avg [GHz]	Std [KHz]	Avg [°]	Std [°]	Avg [°]	Std [°]
Signal 1	5.1	263.39	14.90	0.11	60.30	0.25
Signal 2	5.4	232.48	45.33	0.21	29.94	0.14
Signal 3	5.7	175.81	278.0	0.09	89.71	0.78
Signal 4	5.8	186.77	130.0	0.11	88.25	1.71

Table 5.3: Paired Frequency and DOA Estimation Results for 40dB SNR

	Frequency Estimate		Azimuthal Estimate		Elevation Estimate	
	Avg [GHz]	Std [KHz]	Avg [°]	Std [°]	Avg [°]	Std [°]
Signal 1	5.1	6.57	15.00	0.01	60.00	0.01
Signal 2	5.4	6.96	45.00	0.02	30.00	0.02
Signal 3	5.7	5.72	278.00	0.01	89.10	0.39
Signal 4	5.8	5.40	130.00	0.01	89.10	0.76

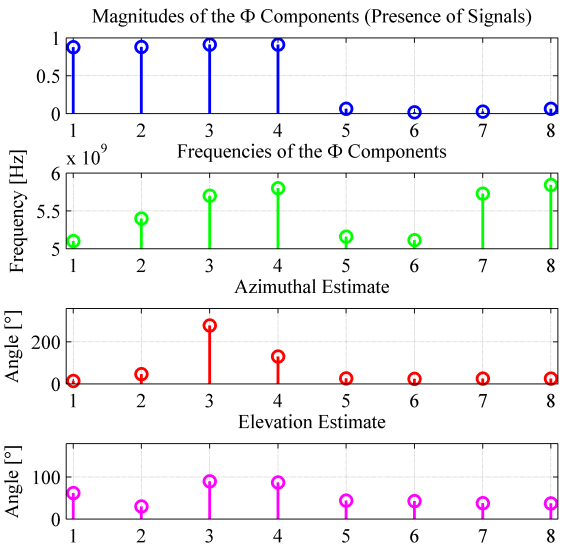
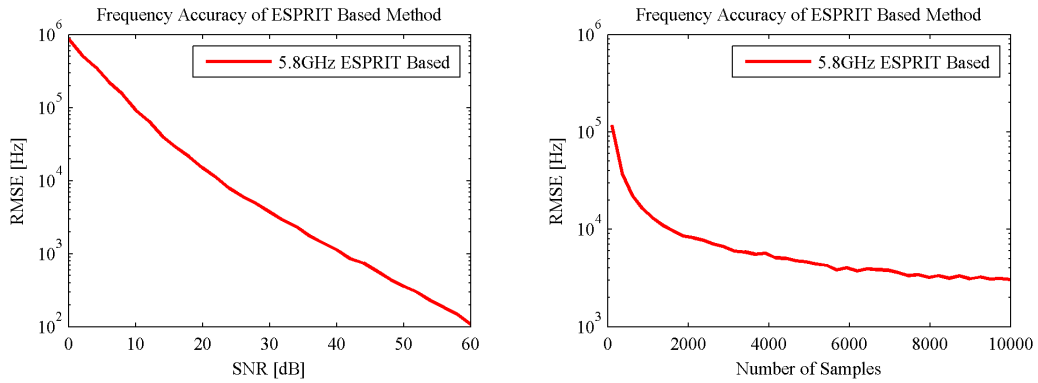


Figure 5.1: Joint Frequency/DOA Estimation Algorithm: Frequency and Signal Detection Pairings

Frequency Estimation Simulation Study

The plots in Figure 5.2 demonstrate the accuracy of the frequency estimation step. For this situation with 1000 samples and a 2GHz sample rate, a simple FFT would have a binwidth of 2MHz, while the frequency estimates start at 1MHz standard deviation at 0dB SNR and improve from there. In addition, like the FFT, adding samples causes a linear improvement in the accuracy.

Just like high resolution DOA estimation algorithms, the frequency estimates resolve as the frequencies approach each other. This however is further complicated, as the point at which the signals resolve is based both on the frequency separation and angular separation, such that signals can be detected either spatially very close or angularly very close, but not both.



(a) Frequency Estimation Accuracy vs SNR (b) Frequency Estimation Accuracy vs Number of Samples

Figure 5.2: Joint Frequency/DOA Estimation Algorithm: Frequency Estimation Accuracy

Direction of Arrival Estimation Simulation Study

The DOA estimation was performed in a very similar method to MUSIC for MVDR methods (Figure 5.3). A psuedo power spectrum was generated and the peaks of this were then found, except there was a pseudo-power spectrum generated for each signal instead of one for the entire problem.

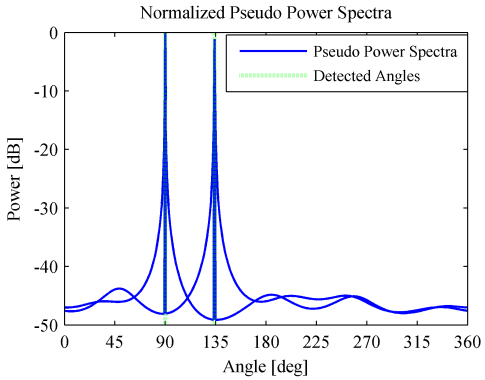


Figure 5.3: Joint Frequency/DOA Estimation Algorithm: Two Signals Overlaid 1D Pseudo-Power Spectrums

As in the earlier methods, this 1D pseudo power spectrum can be expand to two dimensional problems and allow for determination of both azimuth and elevation through location of a peak on a 2d pseudo power spectrum, also 1 per signal detected Figure 5.4.

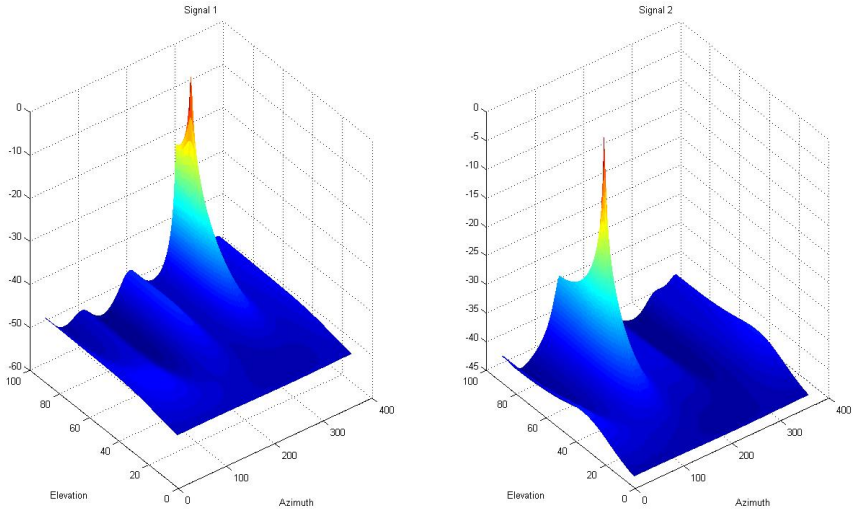


Figure 5.4: Joint Frequency/DOA Estimation Algorithm: 2D Pseudo-Power Spectrum

One shortcoming of this method is in the reduction of the number of signals that can be detected compared to conventional DOA estimation methods. For practical SNR's of 20dB, this is reduced to about 5, which is verified in [27]. Signals beyond the number capable of being detected also act as additional noise causing a further reduction in the number of signals detectable (Figure 5.5).

The result of the simulations shown in Figure 5.6 is that for just about every parameter tested, MUSIC performed slightly better, but by a small enough amount as to make the equations functionally equivalent.

Despite performing slightly worse than MUSIC for single source situations, this algorithm does outperform MUSIC when two signals are closely separated as shown

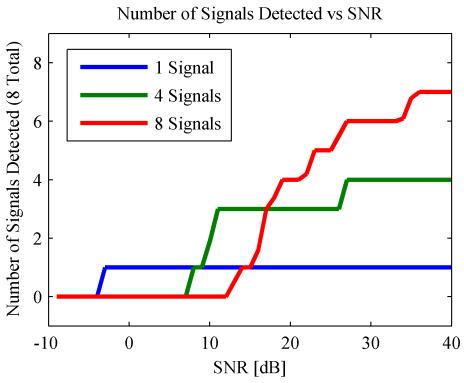
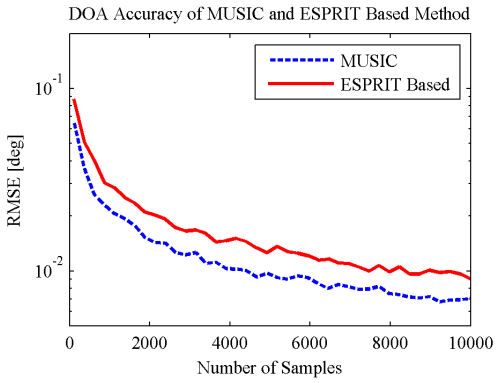
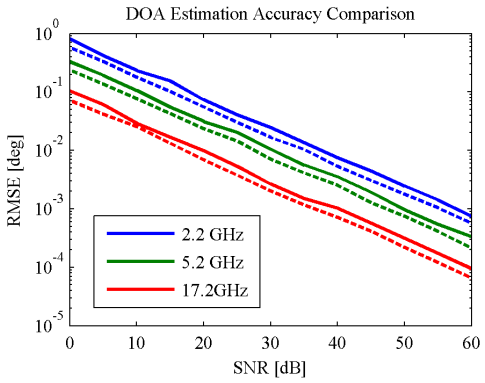


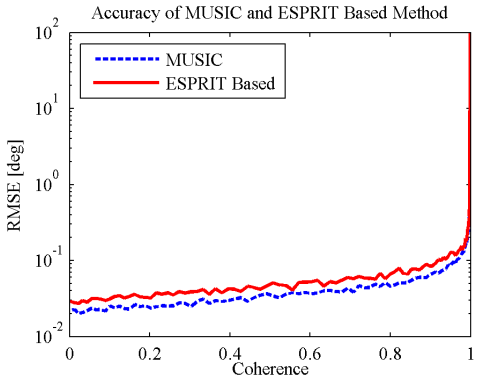
Figure 5.5: Joint Frequency/DOA Estimation Algorithm: Number of Signals Detectable vs SNR



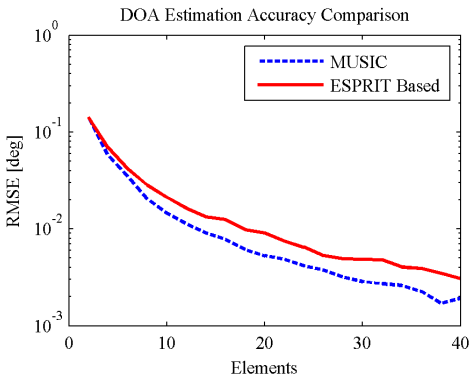
(a) DOA Estimation Accuracy vs Number of Samples



(b) DOA Estimation Accuracy vs SNR



(c) DOA Estimation Accuracy vs Coherence



(d) DOA Estimation Accuracy vs Number of Elements

Figure 5.6: Joint Frequency/DOA Estimation Algorithm: Direction of Angle Estimation Accuracy vs Numerous Parameters

in Figure 5.7. This joint algorithm does not start to return biased results nearly as quickly as the signals approach, which results in slightly improved accuracy.

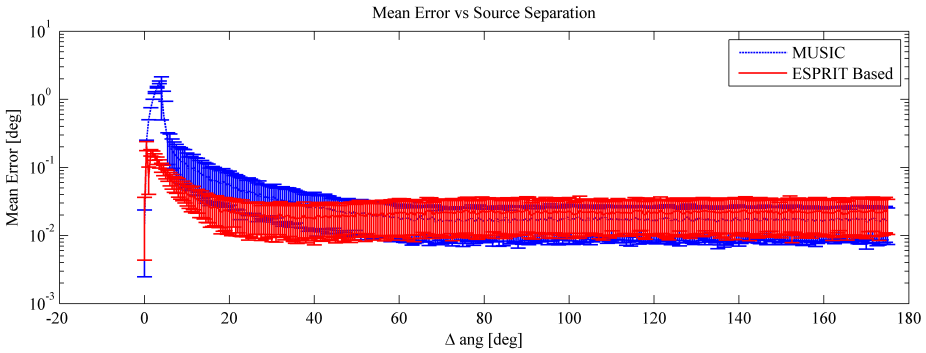


Figure 5.7: Joint Frequency/DOA Estimation Algorithm: Comparison of Mean Accuracy with Standard Deviation of DOA Estimates with Two Sources vs Azimuthal Seperation with MUSIC

The primary failing of this algorithm in it current design, however, is in its sensitivity to mutual coupling errors. These errors quickly limit to accuracy and are a primary factor with the accuracy of the algorithm with our current generation of hardware.

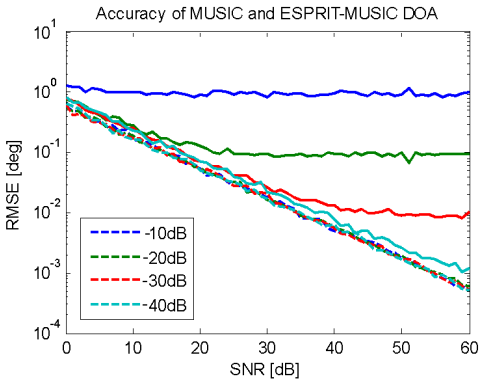


Figure 5.8: Joint Frequency/DOA Estimation Algorithm: Effect of Mutual Coupling on Accuracy vs MUSIC

BEAMFORMING METHODS

As with DOA estimation (and in many cases directly related to) beamforming algorithms have been studied to great length in the last several decades with multiple classes of algorithms and engineering trade-offs between them to consider [1]. For our Cognitive Array system we primarily focus on two algorithms: Cophasel primarily for transmit and analog beamforming and MVDR for high resolution digital receive-only beamforming.

Algorithms

Shift and Sum

Shift and sum patterns are simple and easy to generate and use as they are simply aligning a received signals phases from each element and summing all the channels. These patterns work for both transmit and receive paths and work by attempting to have the signal from each array arrive (at/from) the destination at the same time through phase shifts, while other directions generally destructively interfere resulting in less power. The weights are also trivial to compute as they are the transpose of the steering vector:

$$\mathbf{w} = \mathbf{a}^H(\phi) \tag{6.1}$$

For transmission this system also has higher efficiency, as the amplitude of the signals on all 8 elements is desired to be equal, unlike those generated from window beamformers. As a side effect of this, these weights look like a rectangular aperture which results in higher sidelobe levels.

Shift and sum beamforming for this circular array system was studied thoroughly and implemented in [16].

MVDR

The Minimum Variance Distortionless Response (MVDR) algorithm creates filter weights that act as a high resolution spatial filter and can only be used in receive mode. This algorithm returns cleanly separated signals, but signals appear to have more noise then those from Cophasel and lacks array gain. Despite this and its greater computational complexity, this algorithm is much more desirable for receive beamforming due to its extremely narrow spatial filter.

The weights for MVDR are generated in a very similar fashion to the power spectra for MVDR DOA estimation with the equation:

$$\mathbf{w} = \frac{\mathbf{R}_{xx}^{-1}\mathbf{a}(\phi)}{\mathbf{a}^{H}(\phi)\mathbf{R}_{xx}^{-1}\mathbf{a}(\phi)} \tag{6.2}$$

Simulations

The simulation study shown in Figure 6.1, illustrates an array where there are two signals 50° apart, with one a square wave and the other sinusoidal. This combination of waveforms was chosen as these waveforms, when combined, are easily visible. When only a single element is viewed, the signals are mixed and as a result the signal would likely be corrupted and no information could be retrieved from it. Cophasel beamforming is able to almost completely separate the signals, but as the signals are on the edge of the beam, some of the square wave leaks in and as the signals come closer, their quality will reduce. MVDR is the most capable and is able to almost completely separate the square wave from the sinusoid.

Figure 6.2 compares a cophasel and MVDR Beamformer, but also shows the beams resulting from each. This shows that while the weights generated from the MVDR

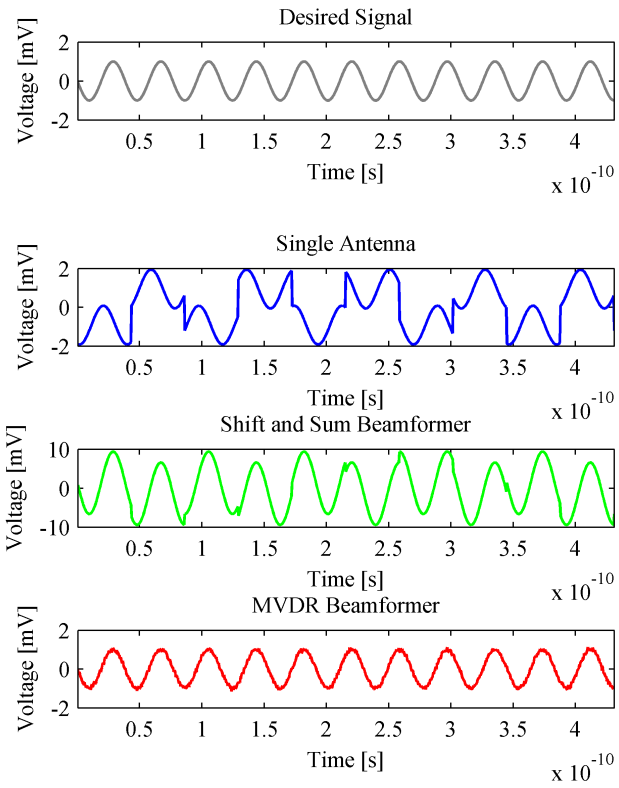


Figure 6.1: Comparison of Single Element Antenna vs Shift ans Sum and MVDR Beamforming methods for signals with 50° seperation

algorithm work very effectively for receive beamforming, the pattern that results in transmit mode is completely ineffective.

Further, as with the DOA estimation algorithm, shift and sum beamforming is incapable of picking out weak signals from strong interference, while MVDR still has good performance (Figure 6.3). The power in the spectrum is not apparent in the received spectrum due to the linear scaling on the plot and far weaker signal strength.

While adaptive beamformers were not applied in the current portion of this project, their use still bears mention. These work through recursively converging upon optimum weights over time rather then directly computing them in a single

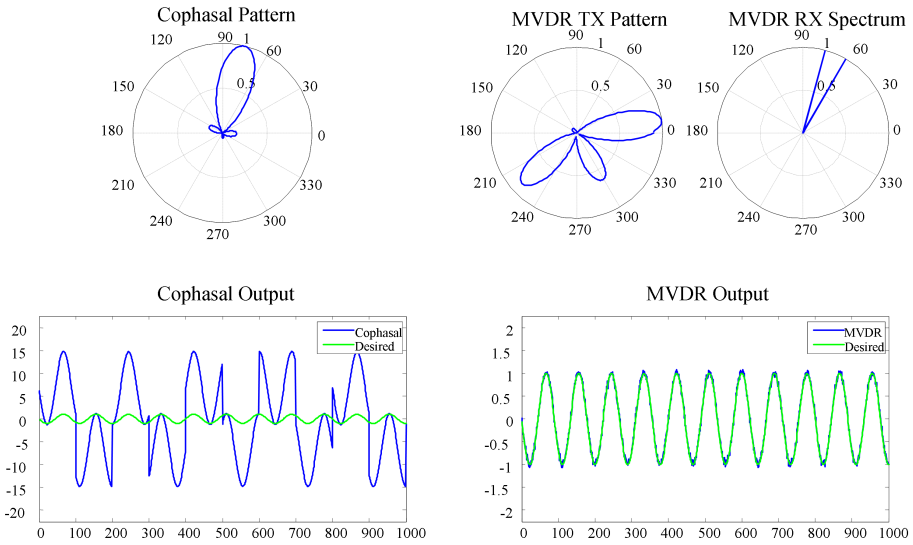


Figure 6.2: Shift and Sum vs MVDR Beam Patterns and Output Signals with a 60° square wave interference and a desired 75° sinusoid

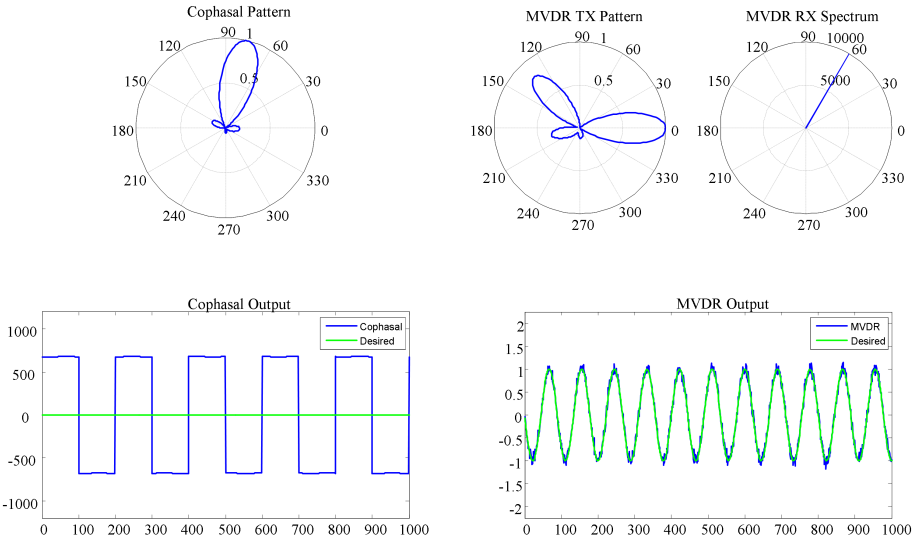


Figure 6.3: Shift and Sum vs MVDR Beam Patterns and Output Signals with 60° square wave interference with 10x more powerful and a desired 75° sinusoid

shot. This allows for these beamformers to often give very similar if not better performance then MVDR, but they do need to have prior knowledge of the statistical properties of the signal they desire. Figure 6.4 shows the output signal as a result of the changing weights as the method converges upon the correct weights to separate sinusoids separated by 55° [33].

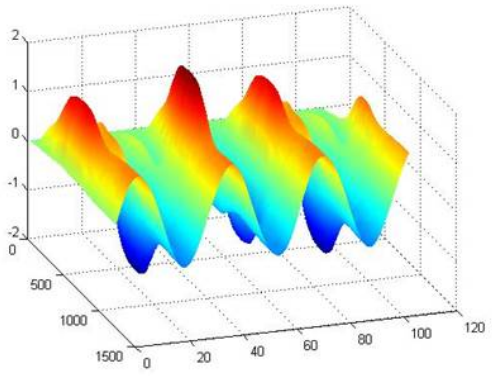


Figure 6.4: LMS Beamformer signal converging to a sinusoidal wave

TRACKING NON-STATIONARY PARAMETERS

The ability to track data sources would have advantages for a cognitive array system. The use of a tracking algorithm allows for estimation of the future location of a moving target. This permits the system to search a smaller portion of the pseudo power spectrum for DOA estimation, increasing the speed of the system, as well as track a source and the signal frequencies and signal parameters that were observed in the past. This information can then be used to predict if in the future data sources will conflict and frequencies should be changed.

Kalman filtering is the simplest and one of the oldest algorithms for tracking. This algorithm is a special case of a recursive least squares filter that predicts the velocity and acceleration of a source in the presence of noise. While many algorithms have been developed for use in systems such as this, the Kalman filter is used in this thesis due to its simplicity and popularity [34, 35].

Kalman Filter

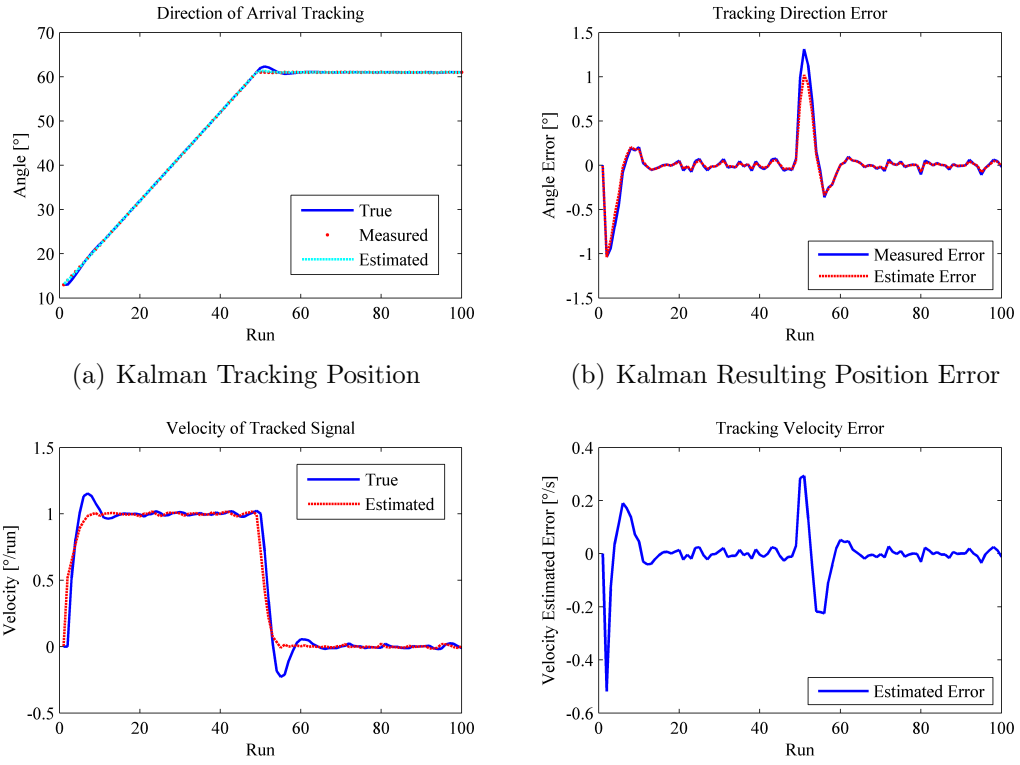
The Kalman filter implemented was based very heavily upon the implementation and code documented in [35]. This code was developed for vehicle navigation applications and was modified primarily in the inputs such that it tracked angular position instead of x and y coordinate positions.

Simulations

This technique was applied to the results of the joint frequency and DOA estimation algorithm for a scenario where a signal source is simulated moving at a constant

angular velocity from 10° to 60° and then stops. The results of this simulation are given in Figure 7.1 and show that the tracking works as expected and quickly converges upon the necessary velocities to track the target.

This simulation shows the result of the Kalman filter and its ability to track a target and estimate its velocity allowing for estimates to be made as to the target's future location. In addition this study shows the errors that occur with the Kalman filter when a target's velocity suddenly stops resulting in velocity estimate errors, and and the estimate overshoots the targets actual location.



(a) Kalman Tracking Position

(b) Kalman Resulting Position Error

(c) Kalman Tracking Velocity

(d) Kalman Tracking of Velocity Error

Figure 7.1: DOA Estimation Kalman Tracking Simulation Results

The filter was also simulated for tracking two separate targets by keeping separate tracks of each target and assuming that the point closest to the estimated next track

for a target was a continuation of the same target. These tracks then kept their data separate and run the filtering algorithm individually on each track. For the simulation in Figure 7.2, two signals were tracked with MUSIC DOA estimates as they moved linearly around the array and stopped. The slight overshoot of the tracked estimates is visible at the stop, as the Kalman filter temporarily estimates the target is still moving and only later converges back to its actual estimated location.

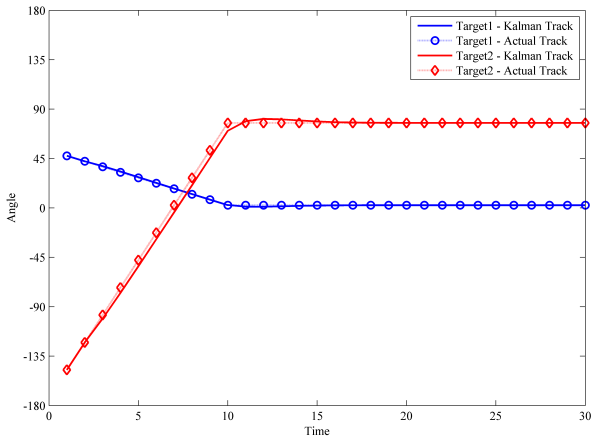


Figure 7.2: Tracking Two Targets with a Kalman Filter

RF LOCALIZATION STUDY

Numerous methods of RF localization were studied as an extension of direction of arrival estimation to provide even more accurate information in the form of the RF map to the system. This not only allows for the system to determine the angle to the sources, but also their position in 2 or 3 dimensional space. Depending on the method chosen, this could require multiple antenna arrays, or time synchronized networks which would limit the sources placed to cooperative targets.

Numerous algorithms were studied for use with a beamforming system and their advantages and disadvantages evaluated.

Direction of arrival (DOA) estimation, which is sometimes also referred to as angle of arrival (AOA) estimation in localization, is simply a triangulation method using multiple arrays to triangulate upon a target. This method relies upon a known error deviation to allow for small amounts of error in measurement to not affect results when more arrays are used. This algorithm can be used for both cooperative and non-cooperative network localization, but suffers from increased error over some of the time based methods due to small angular errors becoming large position errors when ranges are long. In addition, DOA is problematic in non line of sight (NLOS) environments.

Time of arrival (TOA) is a basic radar algorithm where by the system can determine exactly when a signal was sent and when it was received, allowing for trivial computation of distance to the target. This algorithm however, requires exact network time synchronization or to be used in active radar where the same node sends and receives the signal after is bounces off a target. This algorithm can be adapted to NLOS environments and can be highly accurate if fast A/D's are used as the accuracy is largely limited by the propagation time between sample times.

Time difference of arrival (TDOA) is very similar to time of arrival, except that instead of relying on the target to provide a synchronized time signal, the time is measured as a difference from when the signal reaches one array, to the time it reaches antenna at a known reference location. This allows for the advantages of the accuracy of time of arrival to be used when the signal sources to be located are not part of a cooperative network, or synchronized.

The last common technique which was not implemented is frequency difference of arrival (FDOA). This technique compares the frequency of the signal received to a reference or other arrays to determine the Doppler shift and as a result, the speed of a target. It is generally less accurate and requires highly accurate frequency estimates and non-stationary targets.

There are also many methods by which these algorithms can be combined to take advantage of their respective strengths. Most commonly studied are fusion based methods where methods are weighted based on their accuracy in the implemented system and directly added to create a new Hough Space. Such a paper was used to implement the three algorithms studied below, which studied multiple localization algorithms and their fusion [36].

Localization Algorithms

DOA Estimation Algorithm

Determine the probability function:

$$p(x, y|\theta_i) = \frac{\exp(-\frac{(\xi-\theta_i)^2}{\sigma_{\theta_i}^2})}{\sqrt{2\pi}\sigma_{\theta_i}} \tag{8.1}$$

where xi is the angle to a location on the meshgrid from the receivers location, θ_i is the estimated DOA to an emitter and θ_{θ_i} is the standard deviation of the DOA estimate.

Then compute the Hough Space (Probability signal is located at a given point on a mesh) by:

$$A_{DOA}(x, y) = \frac{1}{M} \sum_{m=1}^{M} p(x, y|\theta_m) \tag{8.2}$$

The peak of Hough Space is then the location estimate.

TOA Estimation Algorithm

For the TOA estimation algorithm, the received signal is correlated with a known signal from the same time as the signal was sent.

$$R_{tt} = E\{x_{ideal}^H(t)x_{rx}(t)\} \tag{8.3}$$

The peak of the diagonal is then the time of propagation for the signal:

$$t_{est} = \arg\max_t(diag(R_{tt})) \tag{8.4}$$

and from this, the distance to the target can be backed out.

TDOA Estimation Algorithm

The distance between a point and the origin is found by:

$$R_i = \sqrt{(x_{r_i} - x)^2 + (y_{r_i} - y)^2} \tag{8.5}$$

Determine the distance between array i and the reference array:

$$\begin{aligned} c\tau_{i,1} &= R_i - R_1 \\ &= \sqrt{(x_{r_i} - x)^2 + (y_{r_i} - y)^2} - \sqrt{(x_{r_1} - x)^2 + (y_{r_1} - y)^2} \end{aligned} \tag{8.6}$$

Determine the probability function where σ_r is the standard deviation of the errors in distance estimates:

$$p(x, y|\tau_{i,1}) = \frac{\exp(-\frac{(R_{i,1} - c\tau_{i,1})^2}{2\sigma_r^2})}{\sqrt{2\pi}\sigma_r} \tag{8.7}$$

Find the probability Hough Space:

$$A_{TDOA}(x, y) = \frac{1}{L}\sum_{l=1}^{L} p(x, y, |\tau_{l,1}) \tag{8.8}$$

and the location of the target is then the peak present in the Hough Space.

Simulations

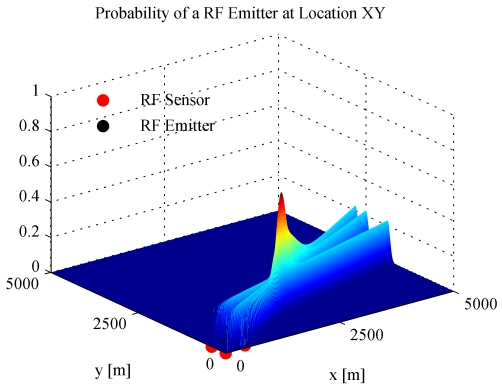


Figure 8.1: 2D DOA Hough Space with the 2D Hough Space Plots projected on the walls

Three dimensional DOA localization was performed by first locating the x and y coordinates and then using that information determine the z coordinate. This greatly reduced the workload as then only 2 planes needed to have probabilities computed and searched, instead of for all points in a 3d cube.

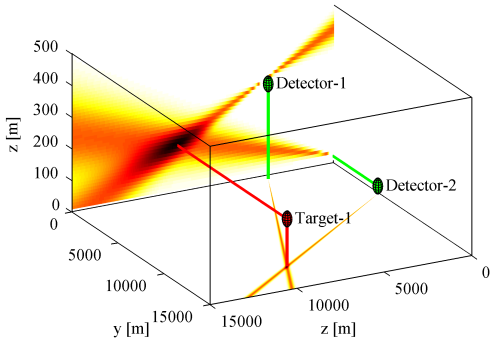


Figure 8.2: 3D DOA Hough Space

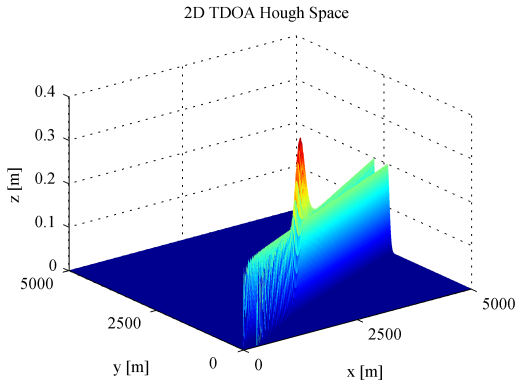


Figure 8.3: 2D TDOA Hough Space

As an expansion of the 2d Hough Spaces, 3d surfaces were generated by adding an elevation dimension and throwing away data below a threshold value. Pixels were replaced by semitransparent cubes and the figure was made more opaque in higher probability regions making them more visible.

This resulted in plots such as that in Figure 8.4 where the black location marks the estimated emitter position.

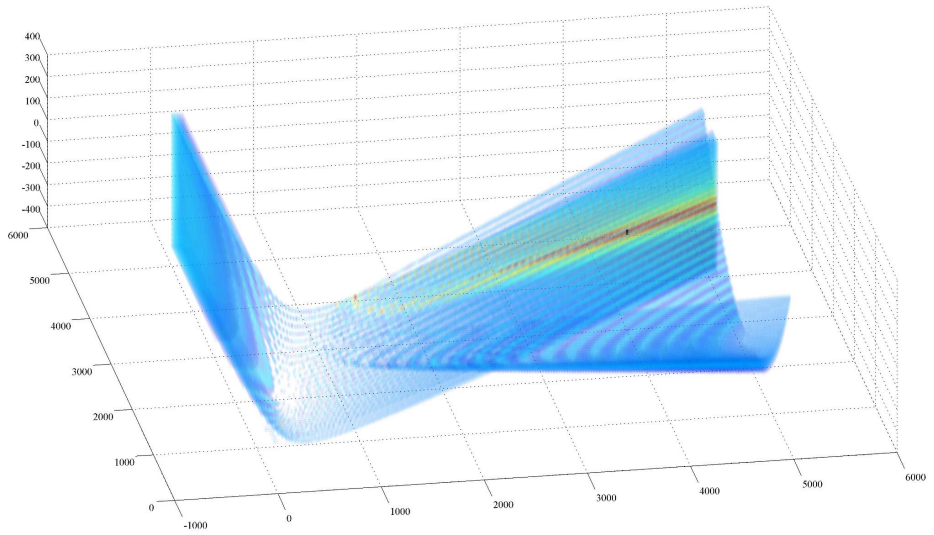
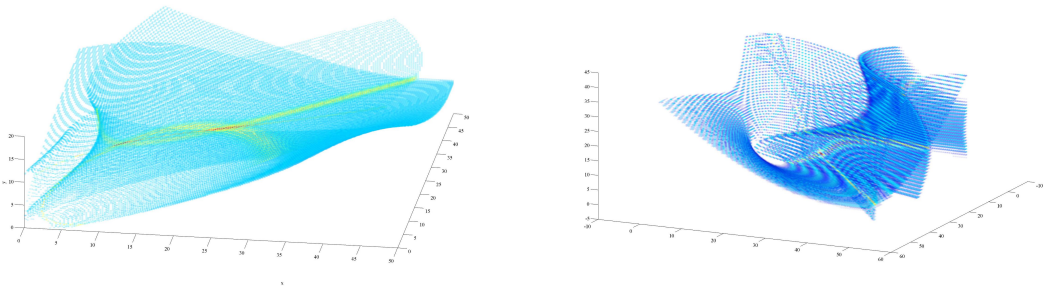
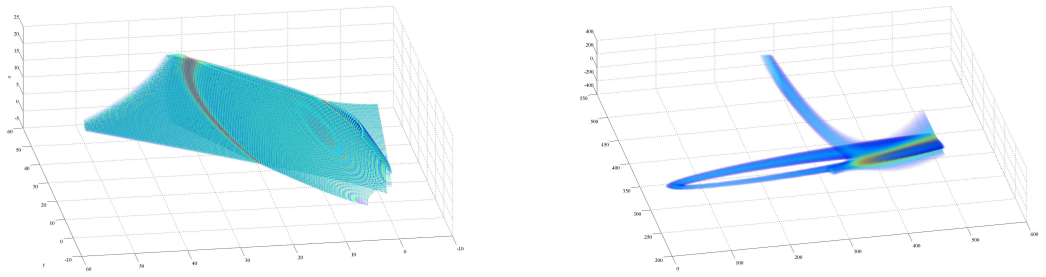


Figure 8.4: 3D TDOA Hough Space

The TDOA Hough spaces can take many different shapes depending on the array placement in comparison to that of the nodes. As examples Figure 8.5 shows the resulting spaces generated during the testing of the algorithm.



(a) (b)



(c) (d)

Figure 8.5: Examples of 3d Solids Generated By TDOA Simulations

SYSTEM CALIBRATION METHODS

Beamformer Calibration

To accurately test the methods on the hardware testbed, an accurate calibration of the system was necessary. This calibration built upon the work of earlier studies to create a generalized calibration matrix that could mitigate the effects of mutual coupling as well as permit for the formation of beams without requiring each to be individually calibrated [37]. In addition, this allowed for formation of more complex beam patterns, such as those with multiple main lobes for multi-directional communications. [38, 39].

Method

The existing automatic calibration system was used to determine usable weights for 16 beams for our 8 element array (9+ are required).

The ideal weights for those 16 known directions were placed in a matrix:

$$\mathbf{A} = [\mathbf{a}(\phi_1)\vdots\mathbf{a}(\phi_2)\vdots\cdots\vdots\mathbf{a}(\phi_M)] \tag{9.1}$$

The measured weights were then placed in a matrix with angle positions matching the ideal weights matrix:

$$\mathbf{A}_m = [\mathbf{a}_m(\phi_1)\vdots\mathbf{a}_m(\phi_2)\vdots\cdots\vdots\mathbf{a}_m(\phi_M)] \tag{9.2}$$

From this point a linear regression was performed on the pair to compute a distortion matrix:

$$\mathbf{C} = \mathbf{A}_m\mathbf{A}^H(\mathbf{A}\mathbf{A}^H)^{-1} \tag{9.3}$$

This distortion matrix could then be used to modify ideal weight settings to those usable by the equipment:

$$\mathbf{A}_{cal} = \mathbf{C}\mathbf{A}_{ideal} \tag{9.4}$$

Results

To verify that the algorithm was working, the calibrated phase weights were generated to match those the angles used for the fixed beam inputs. These tracked well and indicated a good match between the calibration matrix and the measured good weights. Attenuators had much less impact on cophasel weights and as such their comparisons are not shown.

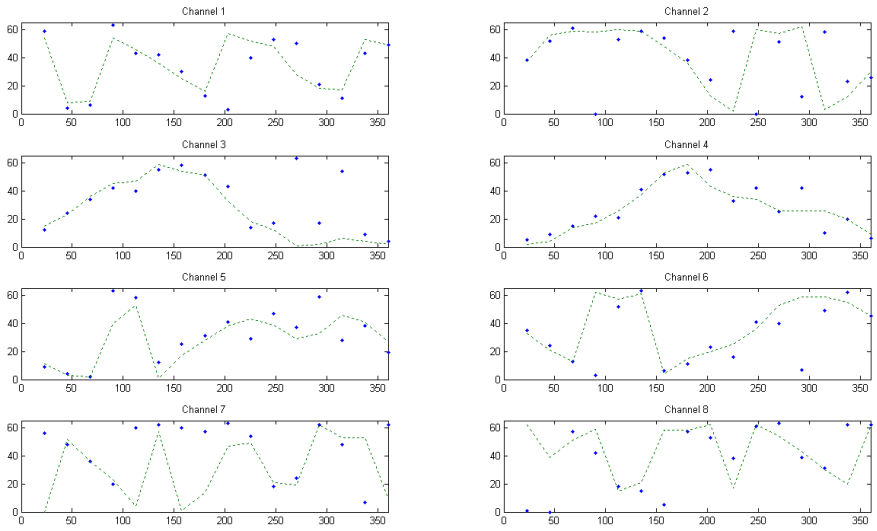


Figure 9.1: Comparison of a Fixed Beam Weights to Computed Calibrated Weights

The performance of the method was then checked by comparing simulated patterns collected in the anechoic chamber using the ideal and calibrated weights.

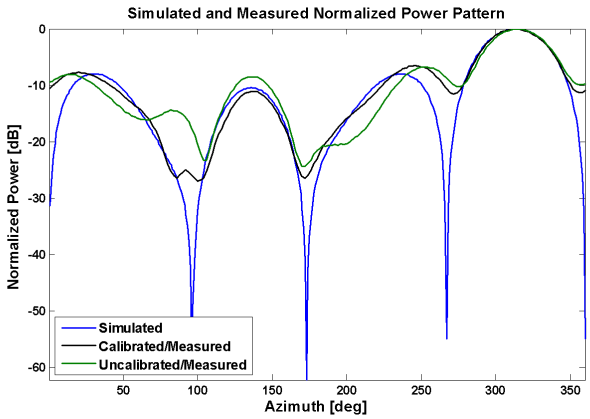


Figure 9.2: Comparison of a MATLAB simulated beam to a calibrated beam using the generalized calibration matrix vs. an uncalibrated beam in hardware

Despite providing good results for beamfoming applications, the combination of the hardware limitations and the effects of calibration errors still resulted in non-ideal null-steering as these patterns are very susceptible to any errors.

Reciever Calibration

A similar method was also applied to calibrate the receiver board [39]. This method collected digitized, single signal narrowband signals from multiple directions and computed a coupling matrix that worked at arbitrary angles to allow for accurate calibration of the array and correct for array mutual coupling. The application of this method greatly improved the resolution of the array over its uncalibrated or center element current injection phase corrections.

Method

Collect K signals where M is greater then the number of array elements. Each signal has the form where $\mathbf{C}$ is the coupling matrix, $\mathbf{A}$ is a matrix of steering vectors

that correspond to the signal matrix **S**:

$$\mathbf{X}(t) = \mathbf{CAS} \tag{9.5}$$

and the autocorrelation is estimated from this as:

$$\hat{\mathbf{R}}_{xx} = \frac{1}{T}\mathbf{X}\mathbf{X}^H \tag{9.6}$$

and a eigenvalue decomposition is performed to compute the eigenvalues and eigenvectors:

$$\hat{\mathbf{R}}_{xx}\mathbf{V} = \mathbf{VD} \tag{9.7}$$

The eigenvector corresponding to the largest eigenvector is then extracted as the estimate of the spatial signature. This is then normalized against each element as:

$$\mathbf{A}_{m\{k\}} = \left[\frac{\mathbf{v}_{max}}{\mathbf{v}(1)_{max}}, \frac{\mathbf{v}_{max}}{\mathbf{v}(2)_{max}}, \cdots, \frac{\mathbf{v}_{max}}{\mathbf{v}(M)_{max}}\right] \tag{9.8}$$

where $\mathbf{v}_max$ is the maximum eigenvector and $\mathbf{v}(i)_max$ is the ith element of the vector, which after K signals have been run results in a $\mathbf{M} \times \mathbf{KM}$ matrix $\mathbf{A}_m$

$$\mathbf{A}_m = \left[\mathbf{A}_{m\{1\}}, \mathbf{A}_{m\{2\}}, \cdots, \mathbf{A}_{m\{K\}}\right] \tag{9.9}$$

The "ideal" steering vector matrix is then determined. For the initial run a current injection calibration is performed and for subsequent runs the calibration matrix from the previous iteration is used on the data set and a MUSIC DOA estimate is found.

This is then used to generate a theoretical ideal array vector $\mathbf{a}_\theta$.

$$\mathbf{A}_{i\{k\}} = \left[\frac{\mathbf{a}_\phi}{\mathbf{a}_\phi(1)}, \frac{\mathbf{a}_\phi}{\mathbf{a}_\phi(2)}, \cdots, \frac{\mathbf{a}_\phi}{\mathbf{a}_\phi(M)}, \right] \tag{9.10}$$

where $\mathbf{a}$ is the steering vector and $\mathbf{a}(i)$ is the ith element of the vector, which after K signals have been run results in a $\mathbf{M} \times \mathbf{KM}$ matrix and are strung together as earlier to create an paired matrix to $\mathbf{A}_{measured}$ of size $\mathbf{M} \times \mathbf{KM}$

$$\mathbf{A}_i = \left[\mathbf{A}_{i\{1\}}, \mathbf{A}_{i\{2\}}, \cdots, \mathbf{A}_{i\{K\}} \right] \tag{9.11}$$

The $\mathbf{C}$ matrix is then estimated as:

$$\mathbf{C}_{est} = \mathbf{A}_m \mathbf{A}_i^H (\mathbf{A}_i \mathbf{A}_i^H)^{-1} \tag{9.12}$$

This is then process iterated until the $\mathbf{C}$ matrix converges. To apply the $\mathbf{C_{est}}$ matrix to the data signal from the array, the signals are modified by:

$$\mathbf{X}(t)_{calibrated} = \mathbf{C}_{est}^{-1} \mathbf{C} \mathbf{A} \mathbf{X}(t) \tag{9.13}$$

Results

The method was iterated for 50 runs to converge the calibration matrix to an accurate estimate of the actual matrix using 13 datasets collected from different directions. The result of this matrix worked equally well on a larger set of 34 datasets including additional angles. The result was greatly improved resolution as well as reduced false peaks, especially in MVDR, where in simpler calibration routines they verged on detection as additional signal directions.

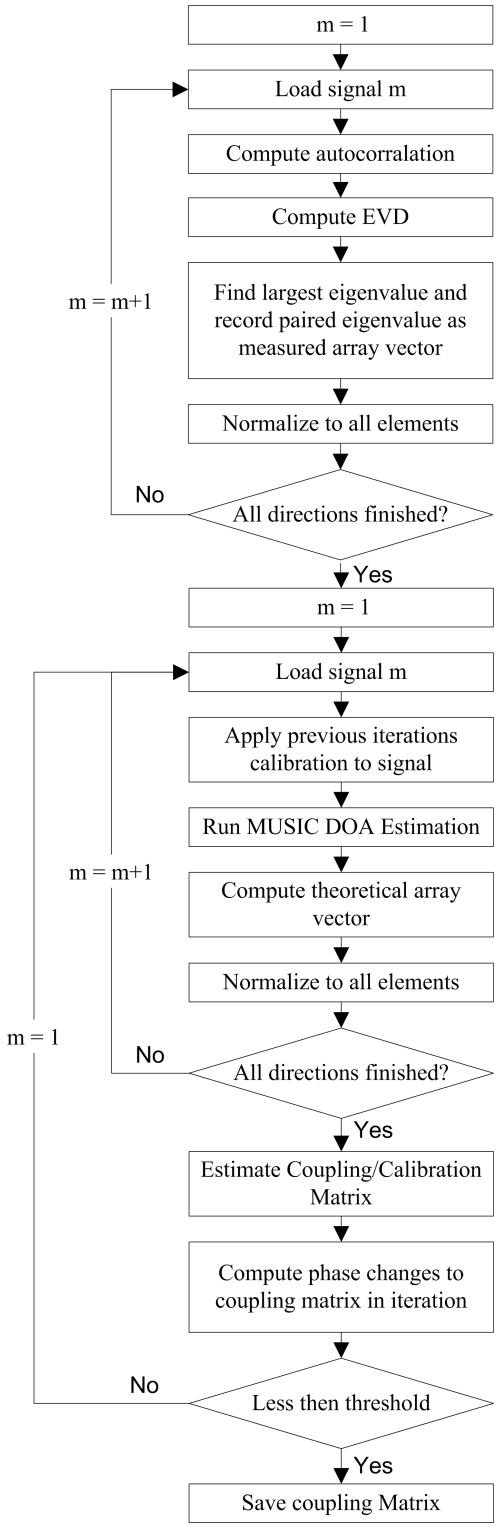


Figure 9.3: Calibration Method Flow Chart

When the hardware was tested, running with no calibration generally produced results, but resolution and accuracy were extremely poor. A calibration was used where a signal was transmitted from a center element in the array and phase differences were recorded for between the elements using an oscilloscope and corrected for on the digitized signals to take path length and part tolerance differences into account. This improved the system greatly, but resolution was still far from ideal.

To combat these remaining problems, the mutual coupling correction was applied, at which point the array system was able to be operated much closer to the theoretical limits and the effect of coupling between array elements was removed. This greatly improved the performance of the DOA estimation routines studied. The effect of this calibration method on DOA estimation results was studied in greater detail by Ahmed Khallaayoun in [15]. Additional examples for MVDR and MUSIC are given in the Appendix to show generality of the calibration across the angles of the array.

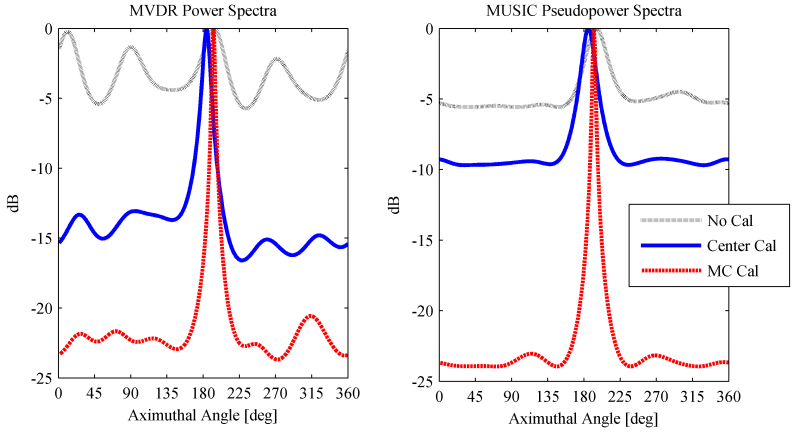


Figure 9.4: MVDR and MUSIC Spectra with Different Calibration Methods

LABORATORY TEST RESULTS

Once the algorithms had been developed and tested in MATLAB and the hardware calibration was performed, we were able to test the algorithms using hardware in the anechoic chamber and verify they performed as expected when run on real hardware. The implementation of these algorithms was done in LabVIEW with the same MATLAB code that was developed in the earlier chapters to allow for rapid implementation of the algorithms and the verification of results. For DOA estimation tests the chamber was setup with the diagram shown in Figure 10.1. A signal is sent into the chamber from a source to a horn antenna and collected by the array. The signal was then downsampled by a 8 channel receiver board and digitized by an NI-6133 DAQ from which LabVIEW and MATLAB scripts could be run on collected signals. This allowed for rapid testing of algorithms as the same code as was used during simulation could be copied and run on real signals [16].

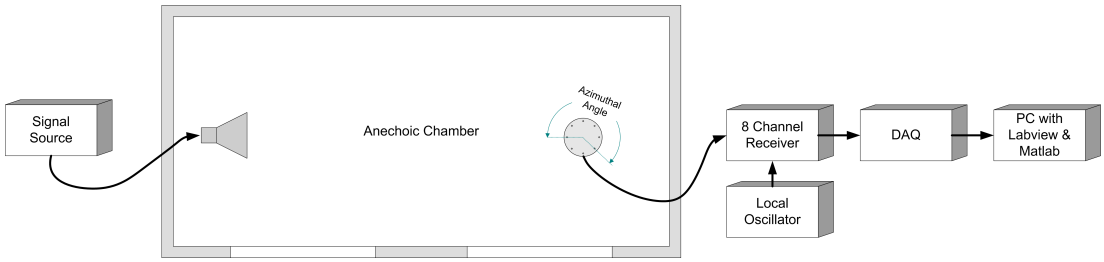


Figure 10.1: Chamber Setup for Direction Of Arrival Estimation Tests

This methods was also used to test the joint frequency and DOA estimation algorithm and compare it to MUSIC with a signal sinusoid at about 5.805GHz and a local oscillator frequency at 5.8GHz. The sample rate was 2.5MHz.

Direction of Arrival Estimation Tests

The DOA estimation algorithms were run on data collected in the chamber for the different calibration methods implemented and with the mutual coupling calibration returned accurate results and expected spectrum shapes.

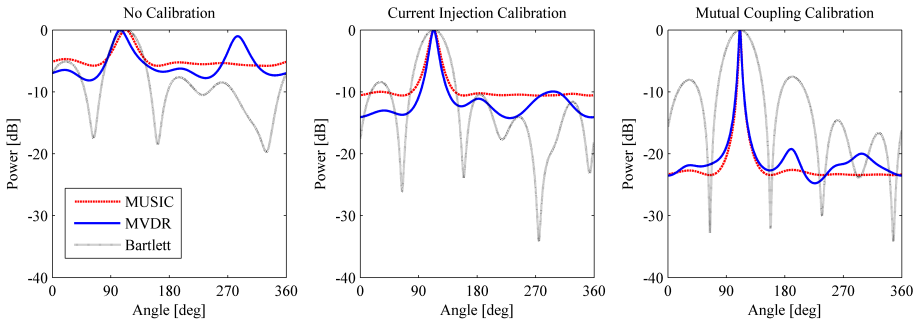


Figure 10.2: Direction Of Arrival Estimation Lab Test

The algorithms were also tested on a two signal data set and worked as expected, but the calibration in this case was necessary for both signals to detect accurately for any algorithm.

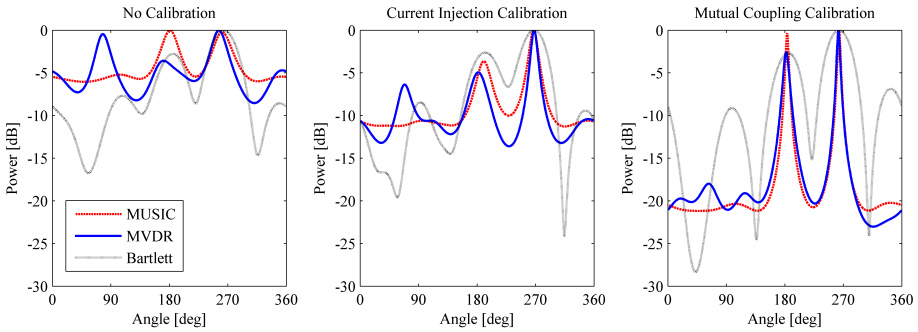
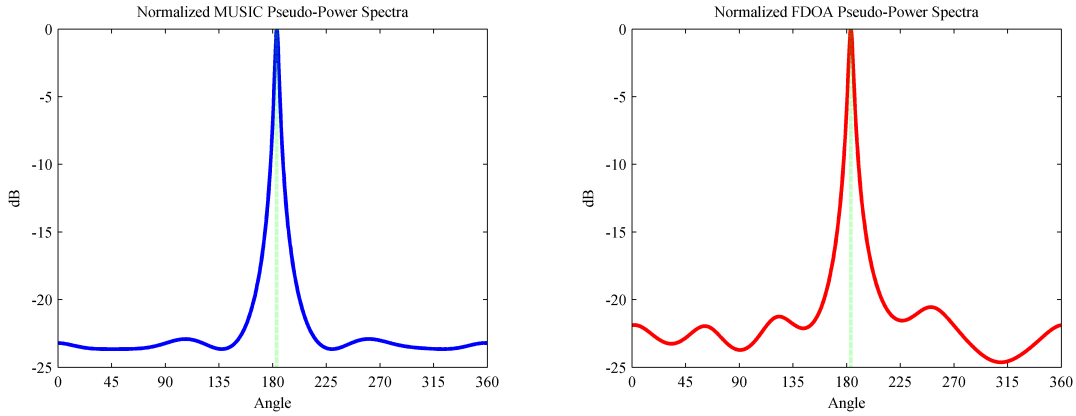


Figure 10.3: Direction Of Arrival Estimation Two Signal Lab Test

Joint Direction and Frequency of Arrival Estimation Test

The joint frequency and DOA estimation algorithm was also tested and compared to the results given by MUSIC. This algorithm generated pseudo-power spectra very similar to those of MUSIC, but had deeper noise floors with slightly more ripple in them. These ripples are likely the result of the increased sensitivity to mutual couplings effect on the spectra.

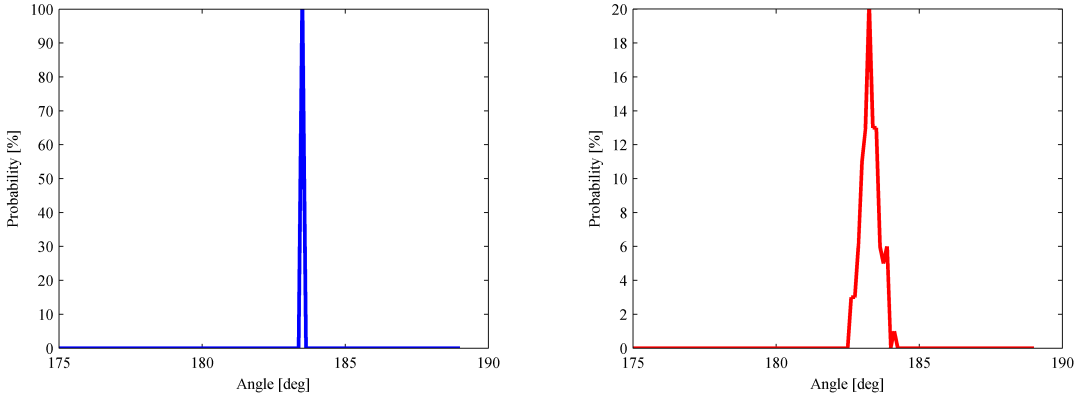


(a) Pseudo Power Plot for of MUSIC (b) Pseudo Power Plot for the Joint Algorithm

Figure 10.4: Labratory DOA Estimation Algorithm Pseudo Power Spectra

However, due to the increased sensitivity of the algorithm to amplitude calibration errors, as well as mutual coupling, the accuracy of the joint algorithm was far less that that of the results using MUSIC. In a 100 run test with the array fixed in a single location, MUSIC had a 0.0039° standard deviation between runs, while the joint algorithm had a standard deviation closer to 0.3071°.

The joint algorithm and MUSIC were also tested with a two signal case. In this situation only the peak of each power spectra for the joint algorithm is important so the large false peak is irrelevant.

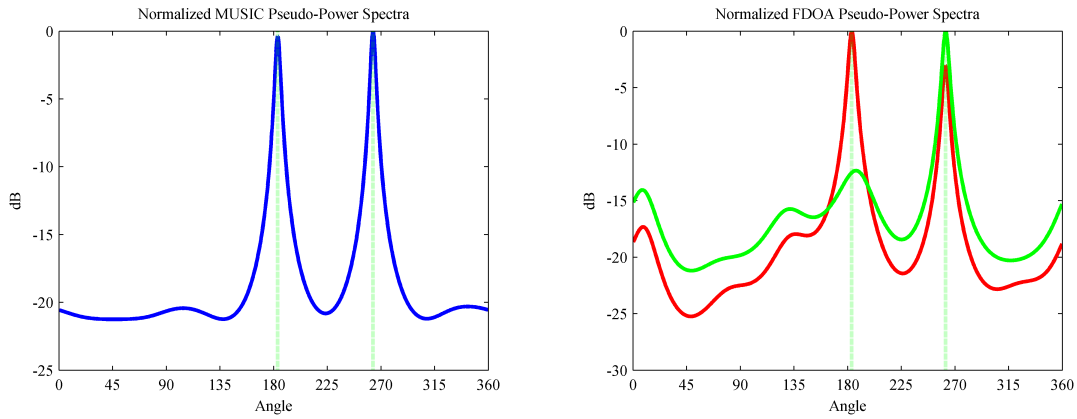


(a) Histogram of MUSIC Results (b) Histogram of the Joint Algorithms Results

Figure 10.5: Labratory DOA Estimation Algorithm Histograms

Table 10.1: Lab Results for Joint Frequency and Angle Estimation Algorithm vs. Conventional MUSIC

	Frequency Estimate		Azimuthal Estimate	
	Avg [GHz]	Std [Hz]	Avg [°]	Std [°]
MUSIC	N/A	N/A	183.4715	0.0039
Joint Algorithm	5.8002633	13.2860	183.2973	0.3071



(a) Pseudo Power Plot for of MUSIC (b) Pseudo Power Plot for the Joint Algorithm

Figure 10.6: Labratory Two Source DOA Estimation Algorithm Pseudo Power Spectrums

Beamforming Test

High resolution beamforming algorithms were then also tested against the data collected in the chamber for two sinusoids with about 90° angular separation and 200KHz frequency separation. For this test, a single antenna element was compared to MVDR as well as the maximum likelihood method (ML) method using ESPRIT byproducts proposed in [25]. While ML and MVDR should be nearly identical, in this test they were different due to the angle errors and rounding in the MVDR DOA estimate.

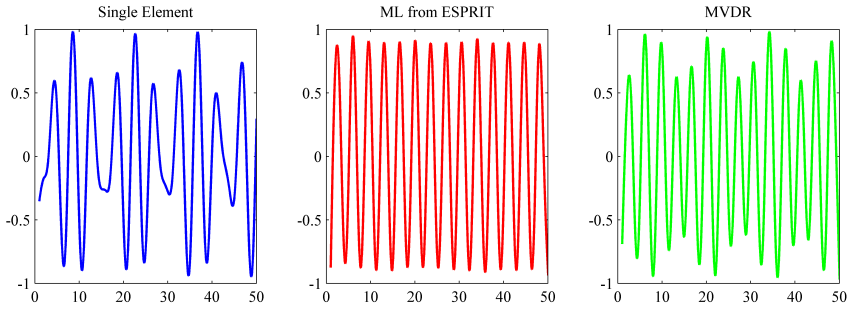


Figure 10.7: Beamforming Algorithms Lab Test

FUTURE WORK

In this work the numerous algorithms were studied separate components that could fill a role in a complete system, instead of linked components. In the future, the algorithms should be linked with a cognitive controlling algorithm to use and combine the information generated by the methods and use it to control a system. There are additional algorithms that are also desired that could be studied and implemented in the future to further improve the quality of the system, primarily in an algorithm that could be used to identify and classify received signals. Additional work is also needed to be able to fully implement the jamming suppression capabilities of the system combine the spatial and spectral systems to avoid jammers in both domains. Finally, a complete system would need a user interface such that a complete system could be controlled and used without extensive knowledge required as to the operation of the algorithms and just present the relevant data to a user.

CONCLUSIONS

The result of this work has shown that combining the power of adaptive antenna systems and cognitive radios can complement each other to greatly improve the quality of radio communications in many situations. In addition, the additional knowledge that is found by these systems can further improve other aspects of networking through exact node placement knowledge of users and interferes allows for more precise knowledge to be used by routing protocols, as well as it allows for the system to predict ahead of time if an interferer is going to causer problems in the future and preemptively take appropriate measures to insure communications are not affected. While the algorithms to do this are non-trivial, this research has shown them to be feasible and implementable, and that the algorithms to provide such information for the cognitive array system are highly accurate and realizable. Smarter array based systems have enough advantages over their single antenna based counterparts that many systems, both in research and industry could benefit from their use.

REFERENCES CITED

[1] Berry D. Van Venn, Kevin M. Buckley. Beamforming: A versatile approach to spatial filtering. *IEEE ASSP Magazine*, pages 4–24, April 1988.

[2] Bruce A. Fette, editor. *Cognitive Radio Technologies*. Academic Press, edition second, 2009.

[3] Raymond J. Weber, Yikun Huang. Analysis for capon and music doa estimation algorithms. *Antennas and Propagation Society International Symposium*, pages 1–4, Charleston, SC, June 2009.

[4] Raymond J. Weber, Yikun Huang. An automatic calibration system for smart antenna arrays. *Antennas and Propagation Society International Symposium*, pages 1–4, Charleston, SC, June 2009.

[5] Raymond J. Weber, Yikun Huang, Grant B. Brandal. A frequency and 2d direction estimation algorithm. *National Radio Science Meeting*, University of Colorado at Boulder, January 2010.

[6] Raymond J. Weber, Yikun Huang. A local positioning system for wireless networks. *National Radio Science Meeting*, University of Colorado at Boulder, January 2010.

[7] William G. Tidd, Raymond J. Weber, Yikun Huang. Cognitive array - a new approach. *National Radio Science Meeting*, University of Colorado at Boulder, January 2010.

[8] Raymond J. Weber, Yikun Huang. A wideband circular array for frequency and 2d angle estimation. *2010 IEEE Aerospace Conference*, Big Sky, MT, March 2010.

[9] Raymond J. Weber, Yikun Huang. 3d rf emitter location estimation. *2010 IEEE Aerospace Conference*, Big Sky, MT, March 2010.

[10] Raymond J. Weber, Yikun Huang. Cognitive beamforming antenna. *2010 IEEE Aerospace Conference*, Big Sky, MT, March 2010.

[11] William G. Tidd, Raymond J. Weber, Yikun Huang. A cognitive beamforming array. *Abstract Submitted to the 2010 IEEE International Symposium on Phased Array Systems and Technology*, October 2010.

[12] Ahmed Khallaayoun, Raymond J. Weber, Yikun Huang. Novel iterative calibration method for doa estimation. *Abstract Submitted to MILCOM 2010*, November 2010.

[13] William G. Tidd, Raymond J. Weber, Yikun Huang. A rf source localization and tracking system. *Abstract Submitted to MILCOM 2010*, November 2010.

[14] Robert A. Monzingo, Thomas W. Miller. *Introduction to Adaptive Arrays*. SciTech, 2004.

[15] Ahmed Khallaayoun. *High Resolution Direction of Arrival Estimation Analysis and Implementation in a Smart Antenna System*. Doctoral Dissertation, Montana State University, 2010.

[16] Michael David Panique. Design and evaluation of test bed software for a smart antenna system supporting wireless communications in rural areas. Master's Thesis, Montana State University, 2008.

[17] Constantine A. Balanis. *Introduction to Smart Antennas*. Morgan & Claypool, 2007.

[18] Joseph Mitola III. *Cognitive Radio: An Integrated Agent Architecture for Software Defined Radio*. Doctoral Dissertation, Royal Institute Of Technology (KTH), 2000.

[19] Hüseyin Arslan, editor. *Cognitive Radio, Software Defined Radio and Adaptive Wireless Systems*. Springer, 2007.

[20] Yikun Huang. Design of a dynamic beamforming system for wimax radio systems. *2008 IEEE Aerospace Conference*, 2008.

[21] Ahmed Khallaayoun, Andy Olson, Michael D. Panique, Yikun Huang. An adaptive smart antenna testbed. *2009 IEEE Mobile WiMAX Symposium*, 2009.

[22] M.S. Bartlett. Periodogram analysis and continuous spectra. *Biometrika*, 37:1–16, June 1950.

[23] J. Capon. High-resolution frequency-wavenumber spectrum analysis. *Proceedings of the IEEE*, 57(8):1408–1418, August 1969.

[24] Ralph O. Schmidt. Multiple emitter location and signal parameter estimation. *IEEE Transactions on Antennas and Propagation*, AP-34(3):276–280, March 1986.

[25] Richard Roy, Thomas Kailath. Esprit - estimation of signal parameters via rotational invariance techniques. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(7):984–995, July 1989.

[26] Petre Stoica, Arye Nehorai. Music, maximum likelyhood, and cramer-rao bound. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 37(5):720–741, May 1989.

[27] Michael D. Zoltowski, Cherian P. Mathews. Real-time frequency and 2-d angle estimation with sub-nyquist spatio-temporal sampling. *ICASSP 1993: Proceedings of the Acoustics, Speech, and Signal Processing*, pages 117–120, Washington, DC, USA, 1993. IEEE Computer Society.

[28] Michael D. Zoltowski, Cherian P. Mathews. Real time frequency and 2-d angle estimation with sub-nyquest spatio-temporal sampling. *IEEE Transactions on Signal Processing*, 42(10):2781–2794, October 1994.

[29] Chen Hui, Wang Yongliang, Wu Zhiwen. Frequency and 2-d angle estimation based on uniform circular array. *IEEE International Symposium on Phased Array Systems and Technology*, pages 547–552, 2003.

[30] Jia Weimin, Yao Minli, Song Jianshe. Joint frequency, two dimensional arrival angles estimations via marked signal subspaces. *ICSP 2006 Proceedings*, 2006.

[31] A. Lee Swindlehurst, Thomas Kailath. A performance analysis of subspace-based methods in the presence of model errors, part i: The music algorithm. *IEEE Transactions on Signal Processing*, 40(7):1758–1774, July 1992.

[32] A. Lee Swindlehurst, Thomas Kailath. A performance analysis of subspace-based methods in the presence of model errors, part ii: Multidimensional algorithms. *IEEE Transactions on Signal Processing*, 41(9):2882–2890, September 1993.

[33] Raymond J. Weber. Adaptive beamforming algorithms. Final Project for EE577: Advanced DSP, May 2009.

[34] H. W. Sorensen. Least-squares estimation from gauss to kalman. *IEEE Spectrum*, 7:63–68, July 1970.

[35] Dan Simon. Kalman filtering. *Embedded Systems Programming*, pages 72–79, June 2001.

[36] Alex Mikhalev, Richard Ormondroyd. Fusion of sensor data for source localization using the hough transform. *2006 9th International Conference on Information Fusion*, pages 1–6, 2006.

[37] Dwayne A. Folden. Automated antenna test and characterization facility. Master's Thesis, Montana State University, 2007.

[38] C.M.S. See. Sensor array calibration in the presence of mutual coupling and unknown sensor gains and phases. *Electronics Letters*, 30(5):373–374, March 1994.

[39] Aweke N. Lemma, Ed F. Deprettere, A. J van der Veen. Experimental analysis of antenna coupling for high-resolution doa estimation algorithms. *2nd IEEE Workshop on Signal Processing Advances in Wireless Communications*, pages 362–365, May 1999.

APPENDICES

APPENDIX A

MATLAB CODE

Creating Array Signals MATLAB Code

```
1  %
2  % Function to sythizise a signals as if it passed over a circular
3  %     array and was downsampled by a reciver board.
4  %
5  % Parameters:
6  %   Azimuth          - Angle in Degrees
7  %   Frequency        - Frequency of the signal
8  %   Power            - Power of the signal to create
9  %   MixerFequency    - Downsample Frequency
10 %
11 % Returns: Matrix with the signals collected at each element
12 %
13 function [x_t] = CreateSignal(Azimuth, Frequency, Power,
       MixerFrequency)
14
15 f1 = Frequency;
16 P1 = Power;
17 azimuth1 = Azimuth;
18 elevation1 = 90;
19 M = 8;                    %Array Elements
20 Noise_P = .01;            %Noise Power
21 Samp = 1000;              %Number of Samples
22 theta1 = elevation1*pi/180;
23 Win_per = 1/2e9;          %Sample Period
24 t = (0:Win_per:(Samp*Win_per - Win_per));
25 %% Compute BetaRho's for the signal frequencies
26 c = 2.99e8;
27 rho = .025;
28 beta1 = 2*pi*f1/c;
29 betaRho1 = beta1*rho;
30
31 %% Spacial Sig for array for both signal frequencies
32 a_t1(1,:) = exp(j*betaRho1*sin(theta1)*cos(azimuth1*pi/180 - (2*(
       pi/M)*(1:8) )));
33
34 %% Array received signals and noise (Create Signals)
35 S_t1 = (P1)*(exp(j*2*pi*(f1-MixerFrequency)*t));
36 NoiSe = wgn(M,Samp,Noise_P,'linear','complex');
37 x_t = a_t1.' * S_t1 +  NoiSe ;
```

Joint Frequency and DOA Algorithm MATLAB Code

```
%
% Function to run a Joint Frequency and DOA Estimation Algorithm
%    and print the results
%
% Parameters:
%   x_t                 -Signal collected from array
%
function [Frequency, Angle] = FDOA(x_t)
tau = 1/2e9;          %Sample Period
Samp = 1000;              %Number of Samples
flo = 5e9;                %Mixer Frequency
M=8;                      %Number of Array Elements
x=[];y=[];                %Variable for power plot data
c = 2.99e8;               %Speed of Light
rho = .025;               %Array Size

%% Time Delayed Versions of Array Input
X = x_t(:,1:(Samp-1));
Y = x_t(:,2:(Samp));

%% Invariance Matrixies
Rxx = X*X'/(Samp-1);
Rxy = X*Y'/(Samp-1);

%% Eigenvector Decomposition
[V, D] = eig(Rxy,Rxx);
Phi = diag(D);
f = abs(angle(Phi)/(2*pi*tau))+flo;

%% DOA Estimation Step
threshold = .99;

for sig = 1:8          %%Scan all eigenvectors
    resMat = sum(abs(Phi) > threshold);
    if abs(Phi(sig))>threshold
        betaRho = 2*pi*f(sig)/c * rho;

        if sig == 1
            eigM =  V(:,2:8);
        elseif sig == 8
            eigM =  V(:,1:7);
        else
```

```
43              eigM =  [V(:,1:(sig-1)),V(:,(sig+1):8)];
44          end
45
46          maxLevels = 52;      %Depth of tree
47          bestVal = 0;         %Value of highest peak found
48          bestAng = 180;       %Angle of highest peak found
49
50          for currentLevel = 0:maxLevels
51              if currentLevel < 10    %Search all angles to start
52                  k = 0: 360/2^(currentLevel+1) : 360;
53                  k = k( 2:2:length(k) ) ;
54              else                    %Search near best location
55                  kStep = 360/2^(currentLevel+1);
56                  k = (bestAng-2*kStep):kStep:(bestAng+2*kStep);
57              end
58
59              for currentNode = 1:length(k)   %%Do MUSIC for a
                    single point and compare to previous best
60                  a_t = exp( 1i*betaRho*cos(k(currentNode)*pi
                        /180-(2*(pi/M)*(1:M)')) );
61                  P_Music = 1./(a_t'*eigM*ctranspose(eigM)*a_t);
62                  P_Music_spectrum = diag(P_Music);
63                  val = max(real(P_Music_spectrum));
64                  if bestVal < valid      %%if new is better,
                        replace value
65                      bestVal = val;
66                      bestAng = k(currentNode);
67                  end
68                  x = [x,k(currentNode)]; %%(for power plot data)
69                  y = [y,val];
70              end
71          end
72      bestAngFinal(sig) = bestAng;
73      end
74  end
75
76  %% Print Results
77  disp(' ');disp(' ');
78  disp(['Number of Signals: ', num2str(resMat)]);disp(' ');
79  disp('Signal 1:')
80  disp(['    Frequency: ' num2str(f(1), '%.0f' ) ' Hz'])
81  disp(['    Angle: ' num2str(bestAngFinal(1),'%.4f'),' deg']);
```

Receiver Calibration MATLAB Code

```
%
% Function Calibrate the Receiver Board with collected data
%
% Imput(In Code): Collected Datasets
% Returns: Mutual Coupling Calibration Matrix
%
% Note: A_m computation can be moved outside the loop to
%       improve speed (doesn't change with iterations)
%
function [C]= RxCalibration()
C = eye(8);                         % Initial Calibration Matrix
A_m = []; A_i = [];

for diffAng = 1:1:200                % Loop Calibration Routines
    for loop = 1:13                 % Loop for to go through signals
        if loop == 1;  x_t = load('dataset1.lvm');  end
        if loop == 2;  x_t = load('dataset3.lvm');  end
        if loop == 3;  x_t = load('dataset7.lvm');  end
        if loop == 4;  x_t = load('dataset9.lvm');  end
        if loop == 5;  x_t = load('dataset12.lvm'); end
        if loop == 6;  x_t = load('dataset16.lvm'); end
        if loop == 7;  x_t = load('dataset18.lvm'); end
        if loop == 8;  x_t = load('dataset21.lvm'); end
        if loop == 9;  x_t = load('dataset24.lvm'); end
        if loop == 10; x_t = load('dataset27.lvm'); end
        if loop == 11; x_t = load('dataset29.lvm'); end
        if loop == 12; x_t = load('dataset30.lvm'); end
        if loop == 13; x_t = load('dataset33.lvm'); end

        x_t = hilbert(x_t).';

        % Compute Autocorralation
        Rxx = 1/size(x_t,2) * x_t * x_t';

        % Compute SVD
        [V,D] = eig(Rxx);

        % Sort by eigenvalue magnitudes
        [D,ind] = sort(diag(D), 'descend');
        V = V(:,ind);

        % Record 'measured' steering vector
```

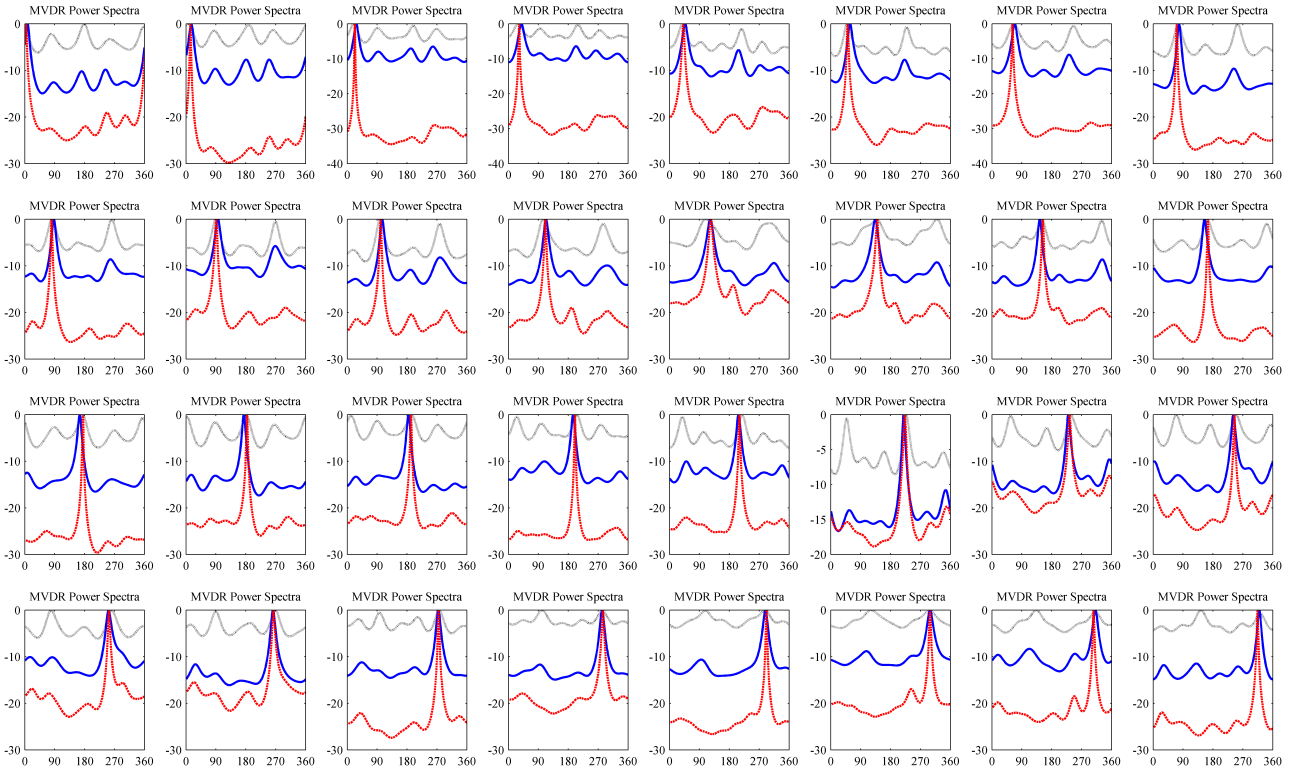
```
43          A_m = [A_m, (V(:,1)*ones(1,8))./(V(:,1)*ones(1,8))'];
44
45          % Find DOA and Compute 'ideal' steering vector
46          x_t = inv(C)*x_t;
47          for tt = 1:8
48              x_t(tt,:) = (x_t(tt,:)-mean(x_t(tt,:)))/(std(x_t(tt
                ,:)));
49          end
50
51          % Compute Autocorralation
52          Rxx = 1/size(x_t,2) * x_t * x_t';
53
54          % Compute SVD
55          [V,D] = eig(Rxx);
56
57          % Sort by eigenvalue magnitudes
58          [D,ind] = sort(diag(D), 'descend');
59          V = V(:,ind);
60
61          eigM =  V(:,2:8);
62          betaRho = 3.05;%3.1416;
63          M=8;
64
65          % Arbitrary Accurate MUSIC DOA
66          maxLevels = 22;      %Depth of tree     52
67          bestVal = 0;         %Value of highest peak found
68          bestAng = 180;       %Angle of highest peak found
69          for currentLevel = 0:maxLevels
70              if currentLevel < 10    %Search all angles to start
71                  k = 0: 360/2^(currentLevel+1) : 360;
72                  k = k( 2:2:length(k) ) ;
73              else                    %Search near best location
74                  kStep = 360/2^(currentLevel+1);
75                  k = (bestAng-2*kStep):kStep:(bestAng+2*kStep);
76              end
77              for currentNode = 1:length(k)
78                  a_t = exp( -1i*betaRho*cos(k(currentNode)*pi
                        /180-(2*(pi/M)*(1:M)')) );
79                  P_Music = 1./(a_t'*eigM*eigM'*a_t);
80                  P_Music_spectrum = diag(P_Music);
81                  val = max(real(P_Music_spectrum));
82                  if bestVal < val
83                      bestVal = val;
84                      bestAng = k(currentNode);
85                  end
```

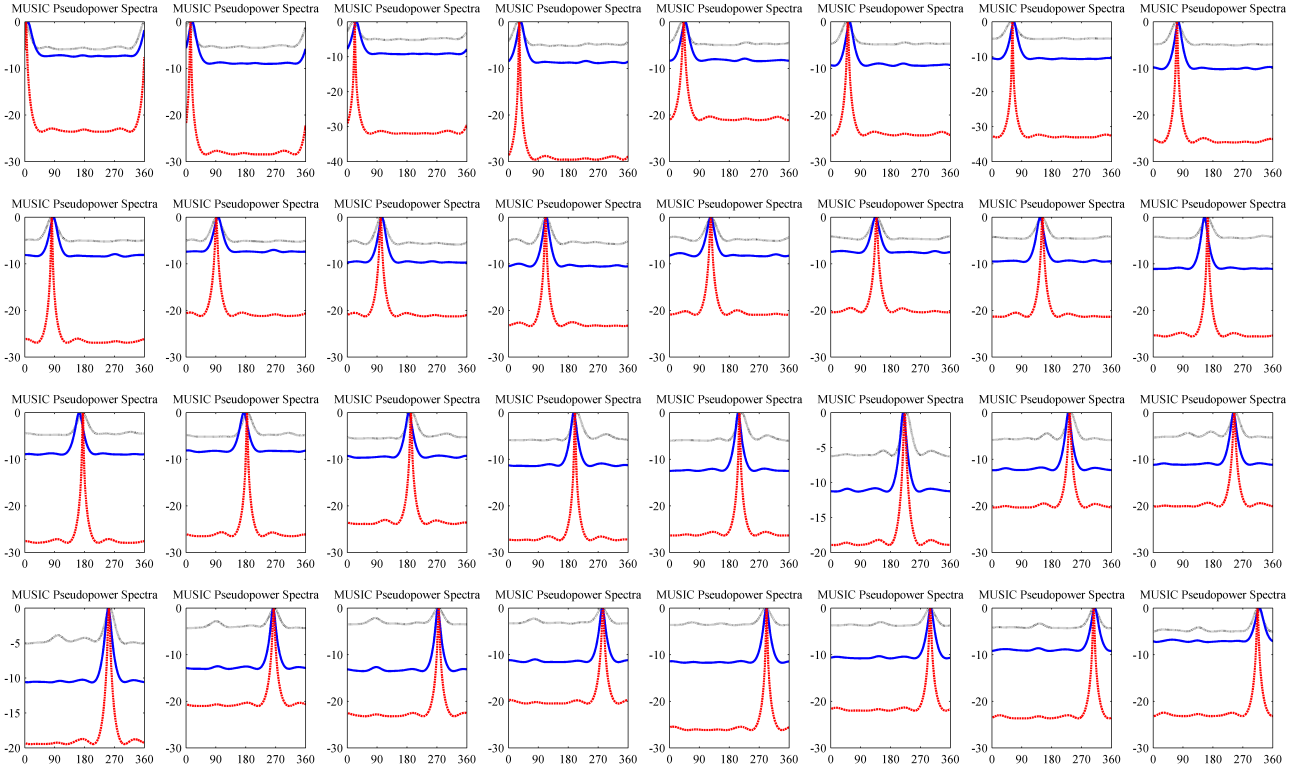
```
86                  end
87              end
88
89              % Compute, normalize and store steering vector
90              A_t = exp( -1i*betaRho*cos(bestAng*pi/180-(2*(pi/M)*(1:M)
                    ')) );
91              A_i = [A_i, (A_t*ones(1,8))./(A_t*ones(1,8))'];
92          end
93
94          % Compute Distortion Matix
95          Cold = C;
96          C = A_m*A_i'*inv(A_i*A_i');
97
98          dC(diffAng) = sum(sum(abs(angle(C)-angle(Cold))));
99
100         figure(1);
101         subplot(1,2,1);
102         imagesc(20*log10(abs(C)),[-20,0]);axis square;colorbar;title(
                'Magnitude');
103         subplot(1,2,2);
104         imagesc(angle(C)*180/pi);colorbar;axis square;title('Phase')
105
106         figure(2);semilogy(dC);
107     end
```

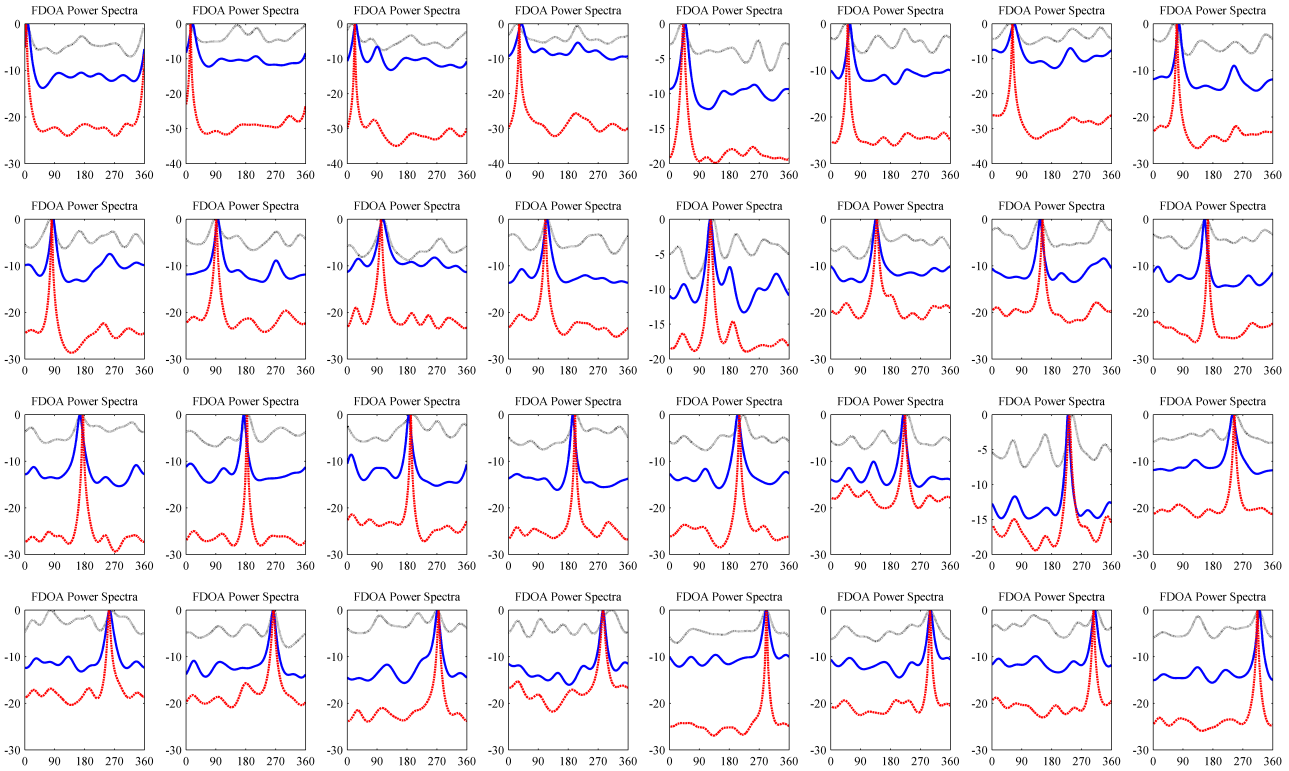
APPENDIX B

ADDITIONAL CALIBRATION SPECTRA DATASETS

Single source signals were collected in the anechoic chamber for 32 directions and processed in MATLAB. For the calibration method, 13 sources were used spread over the 360° range to estimate the calibration matrix. The calibration matrix was then applied to all the data sets, and for different methods to compare the improvements of compensation for mutual coupling compared to current injection (channel phase correction), or no calibration. In all cases the mutual coupling compensation outperformed the other calibration options explored.


MVDR Power Spectra
0 90 180 270 360


MUSIC Pseudopower Spectra
0 90 180 270 360


FDOA Power Spectra