The Pennsylvania State University

The Graduate School

Department of Electrical Engineering

SOFTWARE DEFINED RADIO BASED ANGLE OF ARRIVAL FOR FIRST-RESPONDER SCENARIOS

A Thesis in

Electrical Engineering

by

Pradyumna Vasudeo Desale

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

May 2010

The thesis of Pradyumna Vasudeo Desale was reviewed and approved* by the following:

Sven G. Bilén
Associate Professor of Engineering Design, Electrical Engineering, and Aerospace Engineering
Thesis Advisor

Julio Urbina
Assistant Professor of Electrical Engineering

W. Kenneth Jenkins
Professor of Electrical Engineering
Head of the Department of Electrical Engineering

*Signatures are on file in the Graduate School

ABSTRACT

The maturing of Software-defined-radio (SDR) technology and evolving concepts of Cognitive Radios hold great promise for improved integrated communication in a disaster or first-responder scenario. The first responder can better focus on the incident or threat by effecting radio operations ranging from routine to complex through the use of cognitive radio. A SDR-based Direction-of-Arrival System can detect available and authorized RF resources, can use knowledge of geolocation, spectrum, and network to minimize interference, automatically reconfigure, and connect to the network. This thesis explores specific examples of how a SDR with geolocation capabilities can be implemented for disaster scenarios. Specifically, we investigate angle-of-arrival (AOA) methods and propose a reduced subspace method for low complexity 2-D AOA method. Fading and interference effects are also investigated through simulations. A proof-of-concept model is evaluated for such a system on Universal Software Radio Peripheral (USRP) and GNU radio platform and a plan for implementing a planar array based angle-of-arrival scheme on the USRP platform is proposed.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I would like to thank my parents for their unceasing love for me and their faith that I could accomplish anything that I put my mind to. Their presence in my life has lead me here and now more than ever I realize how truly amazing they are.

I would like to thank Dr. Sven Bilén, Associate Professor of Engineering Design, Electrical Engineering, and Aerospace Engineering at The Pennsylvania State University, and thank Dr. Julio Urbina, Assistant Professor of Electrical Engineering at Penn State, for without their guidance and support, this thesis would not have been possible

I would specially like to thank Dr. Bilén for his availability and for his trust in letting me work independently; he has done nothing but challenge and encourage me from the day I have gotten here. I am immensely grateful for his patience to guide me and review my work. Dr. Bilén has been an exceptional mentor, teacher and a very knowledgeable guide throughout and I wish one day I could know as much as he does.

I also want to thank Chris Gardner for being the perfect accomplice for hardware implementation of various algorithms. Special thanks to Mr. Robert (Bob) Capuro, who patiently reviewed my work. His invaluable suggestions and patience have helped me immensely. I want to thank Okhtay Azarmanesh for his help at various points in time.

Last but not the least; I would like to thank all of my friends, all the professors at Penn State and Penn State Electrical Engineering Department for providing me this wonderful experience of graduate studies.

मम्मी आणि पपांसाठी

(To my parents)

Chapter 1 Introduction

In the 21st century when new facets of communication are being explored, advances in radio technology continue to revolutionize wireless communications. There is a need for secure and robust communication as increasing numbers of entities rely on wireless technology, either directly or indirectly. Ongoing efforts are focused on improving and/or optimizing radio communication systems for adaptability and awareness of environment, thereby increasing usability and reliability. In order to adapt, an awareness of transmitter or receiver location is needed and this is accomplished through a process called geolocation. The most widely used wireless geolocation methods include Angle-of-arrival (AOA), Time of Arrival (TOA), Time Differences of Arrival (TDOA), Received Signal Strength (RSS), and Received Signal Phase (RSP).

1.1 Motivation

The architectures of geolocation systems can be grouped roughly into two main categories: mobile-based architectures and network-based architectures. With the network-based system architectures, like TOA, TDOA, and RSS based methods, the mobile station can be implemented as a simple structured transceiver with small size and low power consumption. The TOA method requires knowledge of the transmit time from the receiver, thus requiring strict time synchronization between transmitter and receiver. Although the TDOA method does not require knowledge of the transmit time, it does require time synchronization among all the receivers. Due to shadow fading effects and phase delays resulting from multipath delays, RSS and RSP methods result in large range estimation errors. This is the *raison d'être* for why in this work we

pursue the problem of estimating direction-of-arrival (DOA) parameters for signals in noisy background and in the presence of heavy multipath fading. This is the type of scenario that might exist, for example, in a region with a disaster response.

The well known subspace-based methods, based on the decomposition of the observation space into signal subspace and noise subspace provide high resolution DOA estimations with good accuracy. However, the classical subspace-based methods such as the MUSIC (MUltiple SIgnal Classification) [*Pauraj et al*., 1986] and ESPRIT (Estimation of Signal Parameters via Rotational Invariance Techniques) [*Roy and Kailath*, 1989] methods involve the estimation of the covariance matrix and its eigen decomposition. As a result, the classical subspace-based methods are rather computationally intensive, especially for the case where the model orders in these matrices are large.

The Reduced-Order Correlation Kernel Estimation Technique (ROCKET) [*Witzgall and Goldstein*, 2003] and ROCK MUSIC algorithm [*Witzgall and Goldstein*, 2000] methods have been presented for high-resolution spectral estimation. These methods do not need the inverse of the covariance matrix but the ROCK MUSIC technique needs the forward and backward recursions of the multi-stage Wiener filter (MSWF) [*Joham and Zoltowski*, 2000], which increase the computational complexity of the algorithm. Moreover, the ROCKET algorithm involves complex matrix–matrix products to find the reduced-rank data matrix and the reduced-rank autoregressive (AR) weight vector. Thus the computational cost of these two algorithms is high.

In this work, we have chosen the low computational complexity MUSIC [*Huang et al.*, 2005a] and ESPRIT [*Huang et al.*, 2005a] based method for DOA estimation, that rely on the MSNWF for separation of noise and signal subspace. Unlike the ROCK MUSIC [*Witzgall and Goldstein*, 2000] technique, the chosen methods separate the signal and noise subspaces using only the forward recursions of the Multi Stage Nested Wiener Filter (MSNWF). By eliminating the need for backward recursion, the computational complexity of the algorithm is effectively

reduced by a factor of 2. Compared to the classical eigen decomposition–based methods, the proposed methods avoid the estimate of the covariance matrix and its eigen decomposition. Thus, these methods are computationally much more efficient. Basically, the proposed low complexity AOA estimation method works exactly like the classical 2-D MUSIC method for circular arrays, but finds the noise subspace more efficiently.

1.2 Contributions of Thesis

This work had developed several elements necessary for AOA determination of transmitters in a disaster or first-responder scenario, including:

- An extension of 2-D Angle-of-Arrival MUSIC algorithm based on Multi Stage Nested Weiner Filter (MSNWF) is proposed.
- 1-D Angle-of-Arrival algorithms based on MUSIC and ESPRIT algorithms were implemented in GNU Radio.
- A roadmap for developing fully fledged Angle-of-Arrival algorithms is proposed.

1.3 Organization

The remainder of the thesis is organized as follows. Chapter 2 provides the background of software-defined radio, GNU Radio, and (Universal Software Radio Peripheral) USRP that are the chosen platforms used in this work. Chapter 3 presents the problem of angle-of-arrival measurement and we discuss the methods proposed by *Huang et al.* [2005a; 2005b] for measurements of angle of arrival in a plane (1-D measurements). We also present the problem of angle-of-arrival measurement in space - characterized by azimuth and elevation (2-D AOA). A quick review of the theory of MSNWF is presented to familiarize readers with notations. Chapter

3 concludes with presentation of the proposed the algorithm for 2-D AOA. Chapter 4 and Chapter 5 discuss the results of simulations and hardware implementations, respectively, and in Chapter 6 we conclude the thesis after proposing a plan for implementing a fully fledged angle-of-arrival system based on planar antenna arrays on a software-defined-radio platform.

Chapter 2 Background

In this chapter we introduce the concept of SDR, provide background on the GNU Radio Project and the Universal Software Radio Peripheral (USRP), and introduce the problem of multipath fading. Finally, we briefly discuss related work done by the SDR community using the GNU Radio and the USRP for geolocation.

2.1 Software-Defined Radio (SDR)

The term software radio [*Reed*, 2000] was first coined by Joseph Mitola in 1993 as a "class of reprogrammable or reconfigurable radios" [*Mitola and Maguire*, 1993]. The terms software radio and software-defined radio are used interchangeably throughout this work. In general, software-defined radios implement one or many of the physical layer (baseband) signal processing functions in software. SDR is an attractive platform for testing and evaluation of signal processing algorithms because of the flexibility and reconfigurability it offers.

2.2 Cognitive Radio (CR)

Cognitive Radio (CR) was formally introduced to the radio community in 1999 by Mitola and Maguire [*Mitola and Maguire*, 1999] as an extension of an SDR. CR improved the overall performance of the radio in relation to its interaction with the spectrum by using a cognition cycle. *Mitola* [2000] states that a "*CR is a goal-driven framework in which the radio autonomously observes the radio environment, infers context, assesses alternatives, generates plans, supervises multimedia services, and learns from its mistakes.*" While other definitions have been developed by research groups across the SDR community, the two components that are most

often considered core features of the CR involve awareness of the RF environment and adaptation and/or learning algorithms to improve the performance of the radio.

2.3 Universal Software Radio Peripheral (USRP)

The USRP1 as shown in Figure 2-1 is an RF device with a simple design that allows for a wide range of SDR-related uses [*Ettus*]. The USRP is not a stand-alone SDR. USRP's architecture requires signal processing functions to be executed on another host device through a USB 2.0 link.

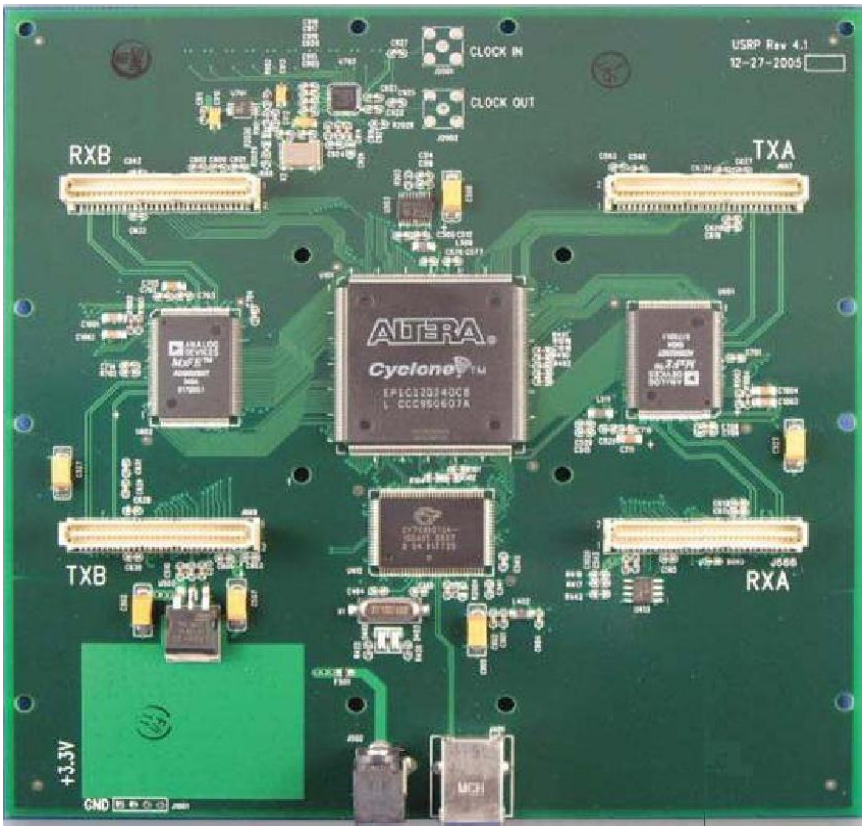


Figure 2-1 Photograph of a USRP 1 board [*Ettus*]

The host device is typically a personal computer with at least one USB 2.0 connection, but the host device can be any kind of signal processing device that can be connected via USB

2.0, e.g., any kind of signal processing system that includes components like General Purpose Processors (GPPs), Digital Signal Processors (DSPs), Field Programmable Gate Arrays (FPGAs), Application Specific Integrated Circuits (ASICs), etc.

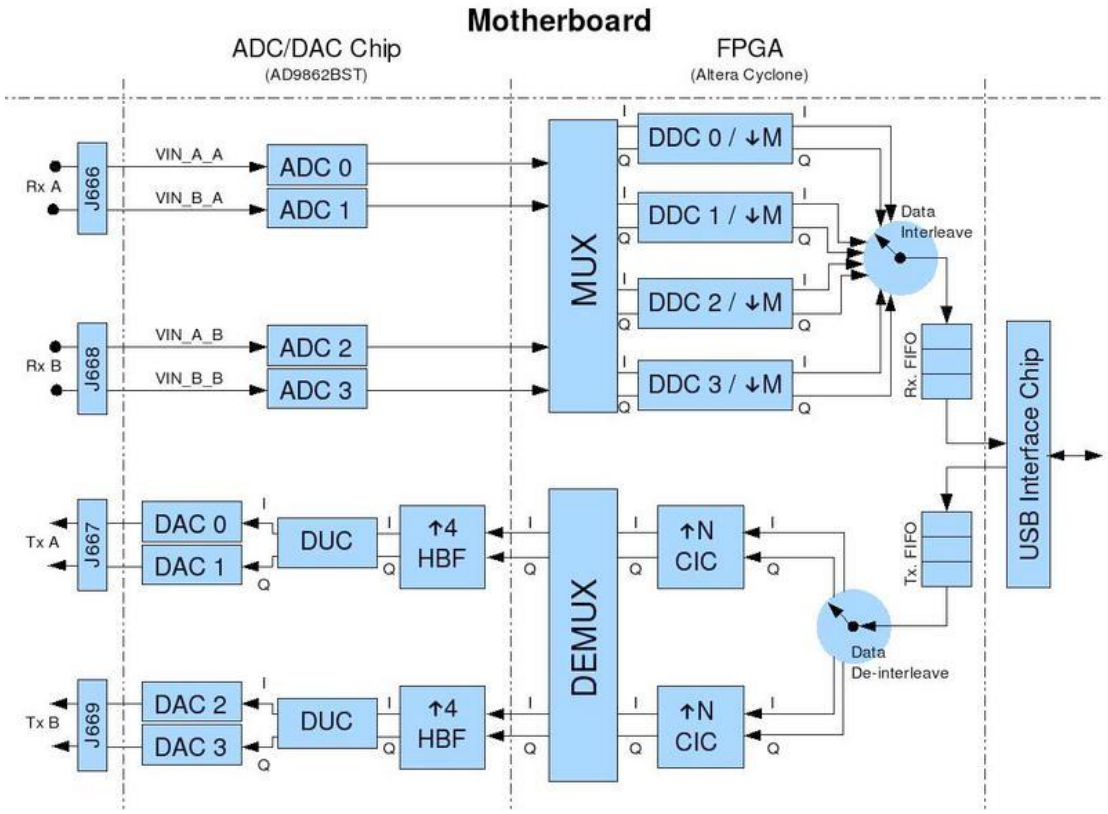


Figure 2-2 USRP motherboard block diagram

The design of USRP is completely open to the public— including the FPGA source code [*Edgewall*; *GNU Radio*]. The USRP subsystem contains many different components resident on the core motherboard, which contains two analog-to-digital (A/D) converters (Analog Devices AD9864), an FPGA (Altera Cyclone II) used for decimation, filtering, up/down conversion, and a USB 2.0 driver for connection with a host device. The motherboard provides large bandwidth capability (limited by the bandwidth of the A/D converters) accommodating many different

modular RF front-ends. Figure 2-2 depicts a block diagram of the current architecture of the USRP1 motherboard [*Patton*, 2007].

Currently, ten RF front-ends (daughterboards) are available with frequency ranges from DC to 2900 MHz [*Ettus*]. For each motherboard, multiple RF front-ends can be attached to the USRP (up to a maximum of four daughterboards-two transmit and two receive).

2.3.1 Limitations of the USRP

While the USRP is certainly a very flexible SDR platform, limitations are inherent within any radio design. The maximum throughput from the USRP to the host device is a well-known limitation of the USRP [*Scaperoth*, 2007]. Some have reported the USB 2.0 interface can support data rates around 32 MS/s and an approximate bandwidth of about 6 MHz of I/Q data and 12 MHz of real data [*GNU Radio*]. Because of this limitation, standards like the 802.11b/g, which have 20-MHz channels, are not feasible using the USRP. On the other hand, because most of the signal processing is being done in the host device, the CPU's processing power often becomes an issue before the USB 2.0 device reaches its maximum throughput. For example, HDTV reception is one of the early proof-of-concept applications implemented on the USRP. Because of HDTV's 6-MHz channel bandwidth, the application was an ideal candidate to illustrate the USRP's capabilities. Although the USRP can successfully down-convert HDTV frames, the time to process the frames is significantly slower than real-time (40 seconds of processing for 1 second of data) using a personal computer running Linux as the host device [*Gilmore*, 2007].

2.4 GNU Radio

GNU Radio is a free collection of signal processing blocks that can be used for RF real-time applications [*GNU Radio*]. GNU Radio can act as a stand-alone software package or as a backend

to a hardware device [*Blossom*, 2007]. GNU Radio applications are written in both C++ and Python and programs are compiled and run on most general purpose processors (GPPs) and operating systems (e.g., Linux, Mac OSX, and Windows). In order to clarify the relation of this work to GNU Radio, this thesis references GNU Radio 3.2 when discussing GNU Radio in context. Control for the signal processing components and any time sensitive processing is done in C++.

Currently, GNU Radio is the primary software platform supporting the drivers for the USRP on a personal computer. The USRP's software-defined parameters (e.g., center frequency, PGA gain, interpolation factor, decimation factor, and some transmit and receive path multiplex options) can only be controlled using Python.

2.4.1 GNU Radio Flow Graphs

A GNU Radio flow graph is a group of signal processing blocks that are interconnected to form a communications system. When the flow graph blocks are interconnected, a GNU Radio flow graph, like an FFT plot of data from a file, an FM modulator, or a TDMA waveform, can be created. Figure 2-3 depicts the software organization of the GNU Radio flow graphs. The GNU Radio flow graphs are initialized within the primary Python thread, and the control portion of the primary Python thread refers to flow graph modifications (e.g., multiplier, filters, squelch level, etc.) during program execution. Any information (i.e., samples or bits) passed through a flow graph block is processed in the signal processing thread, which treats the signals as a continuous stream of data.

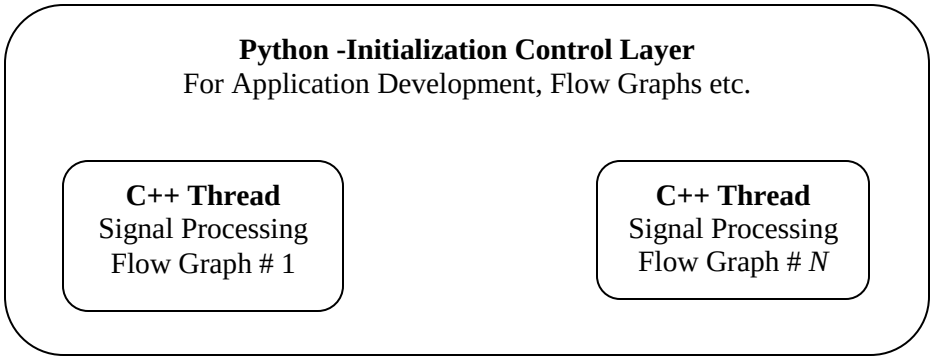


Figure 2-3 Organization of flow graphs in GNU Radio

In order to start the flow graph, each of the blocks must be interconnected using the input and output ports of the flow graph block. The first block in the flow graph, the source block, contains only output connections. The source block can be a sine wave, a set of samples stored in a file, a vector, or the USRP (for receiving), etc. The last block in the flow graph, the sink block contains only input connections. The sink block can be an output to a file, or a vector, or the USRP (for transmitting), etc. All other blocks contain both input and output connections.

2.4.2 Limitations of GNU Radio

Currently, the signal processing with GNU Radio flow graphs must use a continuous flow of samples from one block to the next, and if a block does not continue to receive data, the flow graph block stalls until more information is ready for processing. This becomes a great limitation for developers who are interested in MAC layer functionality for GNU Radio that does not inherently send continuous data through the system. The developers of GNU Radio have proposed extensions to the current architecture called message blocks ("m-blocks") that will allow for asynchronous use of data while still allowing users to continue to design and develop the classic GNU Radio blocks [*GNU Radio Architectural Changes*, 2007].

Another limitation of the current GNU Radio package is driver support for the USRP. The USRP drivers that are compatible with GNU Radio are written in Python, which is a limiting factor for the USRP because the C++ portion of GNU Radio flow graph blocks do not have access to the USRP's configurable parameters.

2.5 The Wireless Channel

The defining characteristic of a wireless mobile channel is the variation in channel strength with respect to time and frequency. Large-scale fading occurs due to signal path loss as a function of distance and shadowing by large objects such as buildings and hills. This occurs as the mobile device moves through a distance on the order of the cell size and is typically frequency independent. Small-scale fading occurs due to the constructive and destructive interference of the multiple signal paths between the transmitter and receiver as shown in Figure 2-4. This occurs at the spatial scale on the order of the carrier wavelength and it is typically frequency dependent.

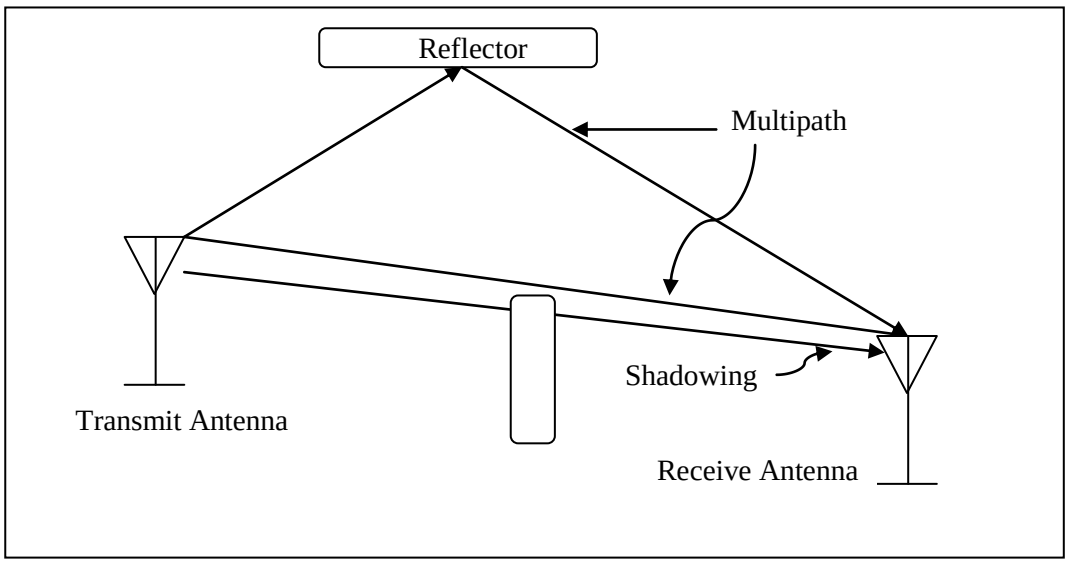


Figure 2-4 Fading and Shadowing

The received signal can be written as the sum of overall attenuation and propagation delay from the transmitter to the receiver along all the different paths. The overall attenuation is simply the product of the attenuation factors due to the antenna pattern of the transmitter and the receiver, the nature of the reflector, and a factor that is a function of the distance from the transmitting antenna to the reflector and from the reflector to the receive antenna.

Consider the transmission of baseband signal of envelope $\tilde{s}(t)$ at carrier frequency f_c., which is represented by

$$s(t) = \mathrm{Re}\{\tilde{s}(t)\exp(j2\pi f_c t)\}. \tag{2.1}$$

At the receiver antenna, the n^{th} plane wave arrives at angle θ_n and experiences Doppler shift $f_{D,n} = f_m \cos\theta_n$ propagation delay τ_n. If there are N propagation paths, the received bandpass signal is

$$r(t) = \mathrm{Re}\left[\sum_{n=1}^{N} C_n \exp^{j2\pi[(f_c + f_{D,n})(t-\tau_n)]}\tilde{s}(t-\tau_n)\right] = \mathrm{Re}\{\tilde{r}(t)e^{j2\pi f_c t}\}, \tag{2.2}$$

where the received complex envelope is

$$\tilde{r}(t) = \sum_{n=1}^{N} C_n e j\phi_n(t)\tilde{s}(t-\tau_n), \text{ here } \phi_n(t) = 2\pi\{f_{D,n}t - (f_c + f_{D,n})\tau_n\}. \tag{2.3}$$

By the superposition principle, (2.2) is represented as the impulse response for the fading multipath channel (i.e., the effect of mobile users arbitrarily moving reflectors and absorbers) and all of the complexities of solving Maxwell's equations, finally reduce to an input/output relation between transmit and receive antennas. This relationship is represented simply as the impulse response of a linear time-varying channel filter. The simplest probabilistic model for the channel filter taps is based on the assumption that there are a large number of statistically independent reflected and scattered paths with random amplitudes in the delay window corresponding to a single tap. Various methods of modeling multipath channel models include an IIR (Infinite Impulse Response) Filtering Method, an IFFT (Inverse Fast Fourier transform) Filtering Method,

and Sum of Sinusoids (SoS) Methods, Jakes' Model and Clarke's model, to name a few [*Stüber*, 2001]. In this work we are using Jakes' Model with Gaussian Doppler spectrum.

2.6 Towards Geolocation in Software-Defined Radio—A Review of Works in Progress

GNU Radio and the USRP were initially adopted by amateur radio enthusiasts, but the popularity of the open-source hardware/software has spread to universities as well. *Rahman et al.* [2009] are working on design and implementation of TOA and TDOA systems in GNU Radio Platform on USRP. *Ledeczi et al.* [2008] implemented baseline Radio Interferometry Positioning on the GNU Radio platform with USRP devices with implicit time synchronization and expect accuracy of the order of 1 °in angle estimation.

Chapter 3 Angle-of-Arrival

This chapter discusses the problem of AOA and different array-based techniques of estimation for AOA. We also discuss the chosen figures of merit to evaluate the performance of direction finding method.

3.1 1-D Angle-of-Arrival

The problem of interest here is estimation the DOAs of emitter signals impinging on a receiving array when a given finite data set **{***x*(*t*)**}** is observed over *t* = 1,2, ..., *N* [*Schmidt*, 1986]. Consider a uniform linear array (ULA) of *M* isotropic sensors that received *P* narrowband signals from distinct directions $\theta_1, \theta_2, \ldots \theta_P$, as shown in Figure 3.1. The $M \times 1$ output vector of the array, which is corrupted by additive noise, at the k^{th} snapshot can be expressed as

$$x(k) = \sum_{n=1}^{P} \mathbf{a}(\theta_i) s_i(k) + n(k), k = 0,1,\ldots, N-1, \tag{3.1}$$

where $s_i(k)$ is the scalar complex waveform referred to as the i^{th} signal; $n(k)$ is $M \times 1$ and is the complex noise vector; *N* and *P* denote the number of snapshots and the number of signals, respectively; and $\mathbf{a}(\theta_i)$ is the "steering vector" of the array toward direction θ_i and takes the following form

$$\mathbf{a}(\theta_i) = \left[1, e^{\left(j\frac{2\pi d}{\lambda}\sin\theta_1 \right)} \quad e^{\left(j\frac{2\pi * 2d}{\lambda}\sin\theta_i \right)} \ldots \quad e^{\left(j\frac{2\pi (M-1)d}{\lambda}\sin\theta_i \right)} \right]^T . \tag{3.2}$$

Here $\theta_i = \left(\frac{-\pi}{2}, \frac{\pi}{2} \right)$; *d* and λ are the inter-element spacing and the wavelength, respectively; and the superscript $(\cdot)^T$ denotes the transpose of matrix.

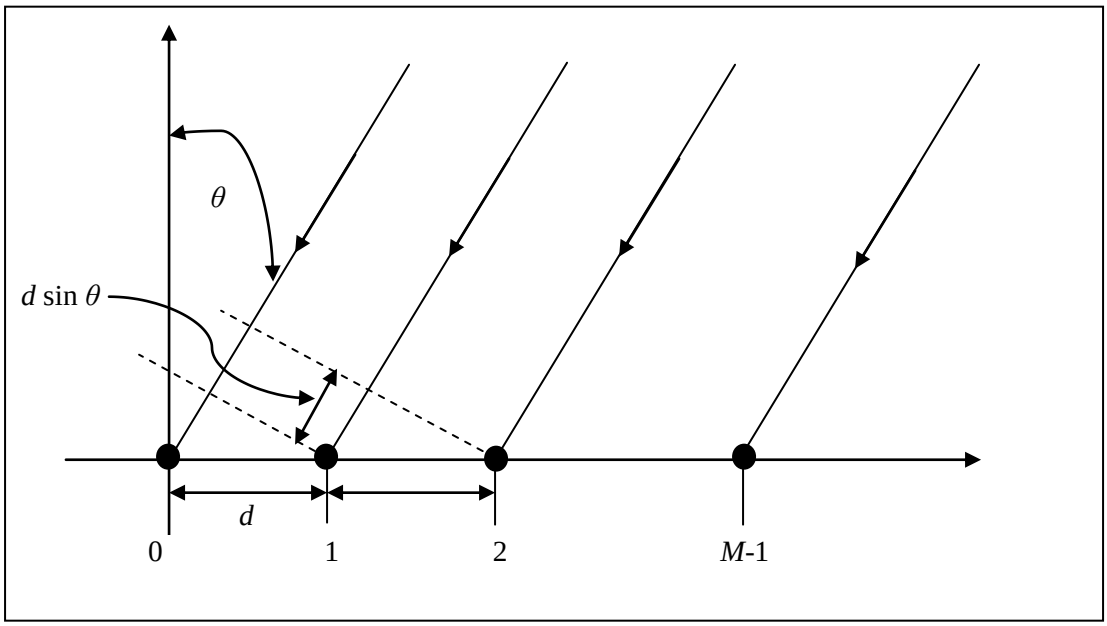


Figure 3-1 The problem of angle-of-arrival estimation

In matrix form, (3.1) becomes

$$\mathbf{x}(k) = \mathbf{A}(\theta)\mathbf{s}(k) + \mathbf{n}(k)\,, \tag{3.3}$$

where $\mathbf{A}(\theta) = \left[\mathbf{a}(\theta_1) \quad \mathbf{a}(\theta_2) \quad \cdots \quad \mathbf{a}(\theta_P)\right]^T$ and $\mathbf{s}(k) = \left[s_1(k) \quad s_2(k) \quad \cdots \quad s_P(k)\right]^T$ are the $M \times P$ steering matrix and the $P \times 1$ complex signal vector, respectively. Throughout this work we assume that $M > P$, i.e., the number of sensors or antennas is greater than the number of signals to be identified. Furthermore, the background noise is uncorrelated with the signals, and noise as well as signals are modeled as a stationary, spatially-temporally white, zero-mean, Gaussian complex random processes. Mathematically, the expectations are given as

$$E\left[\mathbf{n}(k)\mathbf{n}^{\mathrm{H}}(l)\right] = \sigma_n^2 \delta_{k,l}\mathbf{I}_M \text{ and } E\left[\mathbf{n}(k)\mathbf{n}^{\mathrm{T}}(l)\right] = \mathbf{0}\,, \tag{3.4}$$

where $\delta_{k,l}$ is the Kronecker delta, which is 1 for $k = l$ and 0 for $k \neq l$.

Various methods are proposed for estimating the number of signal sources detected by an antenna array (AA) [*Wax and Kailath*, 1985]. The majority of these methods are based on analysis of the likelihood function as well as the eigenvalues and eigenvectors of the sample

correlation matrix of signals received by an AA. The most well-known methods are the Akaike Information Criterion (AIC) [*Akaike*, 1974] and the Rissanen's Minimum Description Length (MDL) criterion [*Barron et al.*, 1998]. For this work, we have assumed that the number of signals is accurately known at the receiver side.

Under the assumption of Gaussian signals and noise, the output of the array is complex Gaussian with zero mean and the covariance matrix of the received signal is given as

$$\mathbf{R}_{xx} = E\left[\mathbf{x}(k)\mathbf{x}^{\mathrm{H}}(k)\right] = \mathbf{A}(\theta)\mathbf{R}_s\mathbf{A}^{\mathrm{H}}(\theta) + \sigma_n^2\mathbf{I}_n, \tag{3.5}$$

where, σ_n^2 is the noise variance and $\mathbf{R}_s$ is the signal covariance matrix $E[\mathbf{s}(k)\mathbf{s}^{\mathrm{H}}(k)]$. Practically, only a sample covariance matrix is available, that is, an estimate of $\mathbf{R}_{xx}$ based on a finite number (K) of data samples or snapshots. The sample covariance matrix is given as

$$\hat{\mathbf{R}}_x = \frac{1}{K}\sum_{j=1}^{K}\mathbf{x}(t_j)\mathbf{x}^{\mathrm{H}}(t_j). \tag{3.6}$$

For uncorrelated signals, the eigen decomposition of $\hat{\mathbf{R}}_x$ can be expressed as

$$\hat{\mathbf{R}}_x = \mathbf{V}_s\Lambda\mathbf{V}_{\mathrm{s}}^{\mathrm{H}} + \sigma_n^2\mathbf{V}_n\mathbf{V}_n^{\mathrm{H}} = \sum_{i=1}^{M}\lambda_i\mathbf{v}_i\mathbf{v}_i^{\mathrm{H}}, \tag{3.7}$$

where $\mathbf{V_s} = [\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_P]$ and $\mathbf{V}_n = [\mathbf{v}_{P+1}, \mathbf{v}_{P+2}, \cdots, \mathbf{v}_M]$ are the matrices of eigenvectors of signal and noise subspaces and $\Lambda = \mathrm{diag}[\lambda_1 \quad \lambda_2 \quad \cdots \quad \lambda_M]$ consists of eigenvalues corresponding to all the eigenvectors sorted in descending order such that $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{P+1} = \cdots = \lambda_M = \sigma_n^2$. Since the number P of sources is smaller than the number M of sensors, all the signal components are represented in the signal subspace spanned by the first P eigenvectors $\mathbf{V}_s = [\mathbf{v}_1, \mathbf{v}_2, \cdots, \mathbf{v}_P]$, and the remaining $M - K$ eigenvectors $\mathbf{V}_n = [\mathbf{v}_{P+1}, \mathbf{v}_{P+2}, \cdots, \mathbf{v}_M]$ represent the noise subspace. The signal and the noise subspaces are orthogonal to each other. This assumption of orthogonality is the basis of subspace-based estimation algorithms.

3.1.1 MUSIC

To apply the MUSIC estimator as a DOA peak locator [*Schmidt*, 1986]:

- Define the sub-matrix $\mathbf{N}_M$ representing the noise subspace of the given noisy signal.
- Exploit the MUSIC estimator $P_{\mathrm{MUSIC}}(\theta) = \dfrac{1}{\mathbf{a}_M(\theta)^{\mathrm{H}} \mathbf{N}_M \mathbf{N}_M^{\mathrm{H}} \mathbf{a}_M(\theta)}$ to produce peaks at the DOA locations, where $\mathbf{a}_M(\theta) = \begin{bmatrix} 1 & \exp(j\theta_1) & \cdots & \exp(j(M-1)\theta_1) \end{bmatrix}$.

3.1.2 ESPRIT

To apply the ESPIRIT estimator as a DOA parameters [*Roy and Kailath*, 1989; *Pauraj et al.*, 1986]:

- Define the sub-matrix $\mathbf{E}_s$ representing the signal subspace of the given noisy signal.
- Define $\mathbf{E}_{s_1}$ and $\mathbf{E}_{s_2}$ as $\mathbf{E}_{s_1} = \mathbf{E}_s[1{:}M-1,:]$ and $\mathbf{E}_{s_2} = \mathbf{E}_s[2{:}M,:]$
- Obtain $\overline{\boldsymbol{\psi}} = \mathbf{E}_{s_1} / \mathbf{E}_{s_1}$ in either a least squares sense (LS) or a total-least-squares (TLS) sense.
- Compute the eigenvalues $\overline{\mu}_i$, $i = 1, 2, \cdots, P$ of $\overline{\boldsymbol{\psi}}$.
- Obtain the DOA parameters by $\theta_i = \sin^{-1} \dfrac{\lambda \arg(\overline{\mu}_i)}{2\pi d}$ for $i = 1, 2 \ldots P$.

3.2 2-D MUSIC

The extension of 1-D angle-of-arrival methods to two-dimensional AOA (i.e., azimuth and elevation) is necessary for practical applications, but it introduces many new problems and challenges that do not occur in the "1-D" case. Specific examples include the need for so-called "pair matching" between two sets of independently obtained arrival angles, complexity reduction in approaches requiring a 2-D search, and non-planar array design. In addition, angular

measurements may be combined with other modalities, such as Doppler or delay, to aid 2-D direction finding or 3-D location determination and prediction.

For the 2-D (azimuth and elevation) DOA estimation problem, at least a planar array is required [*Harabi et al.*, 2009]. Diversity performance of several planar arrays has been a topic of research in various cases.

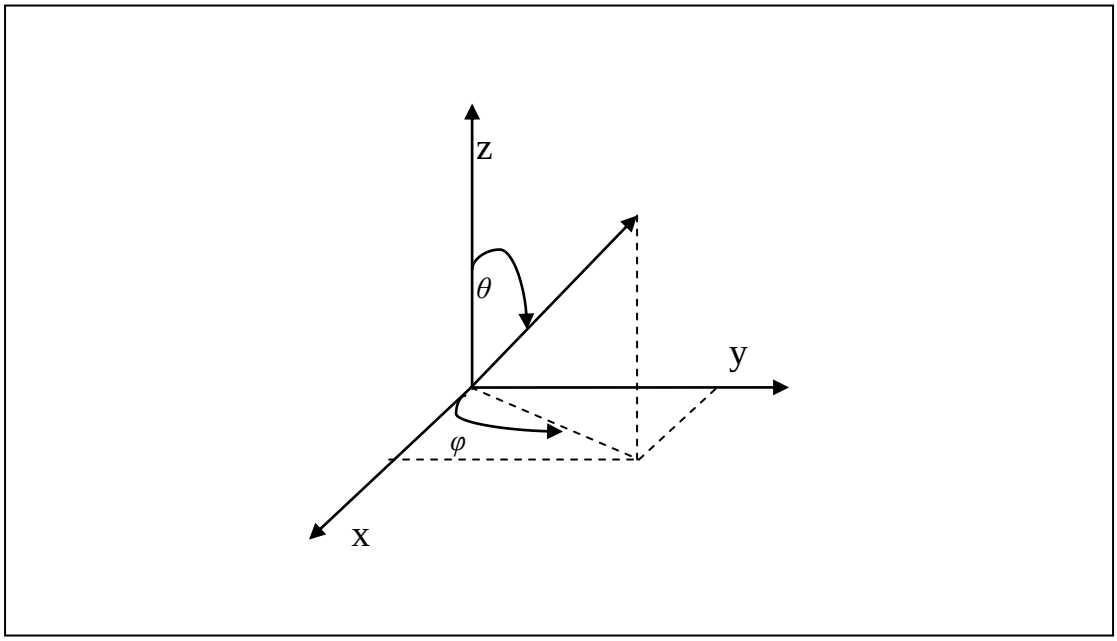


Figure 3-2 A 3D system showing a signal arriving from azimuth φ and elevation θ

Various geometries have been proposed for planar arrays. *Harabi et al.* [2009] performed a detailed study of various planar array topologies and conclusively showed that 2L arrays are the best option for 2-D angle-of-arrival measurements. The commonly used planar array geometries are shown in the Figure 3-3.

As long as a line-of-sight component exists in the channel, the angle-of-arrival approach can be used for direction-finding or position determination for cooperative or non-cooperative emitters. The received signal in a planar array is modeled as

$$\mathbf{x}(\mathrm{k}) = \mathbf{A}(\varphi, \theta)\mathbf{s}(\mathrm{k}) + \mathbf{n}(\mathrm{k}) \,, \tag{3.8}$$

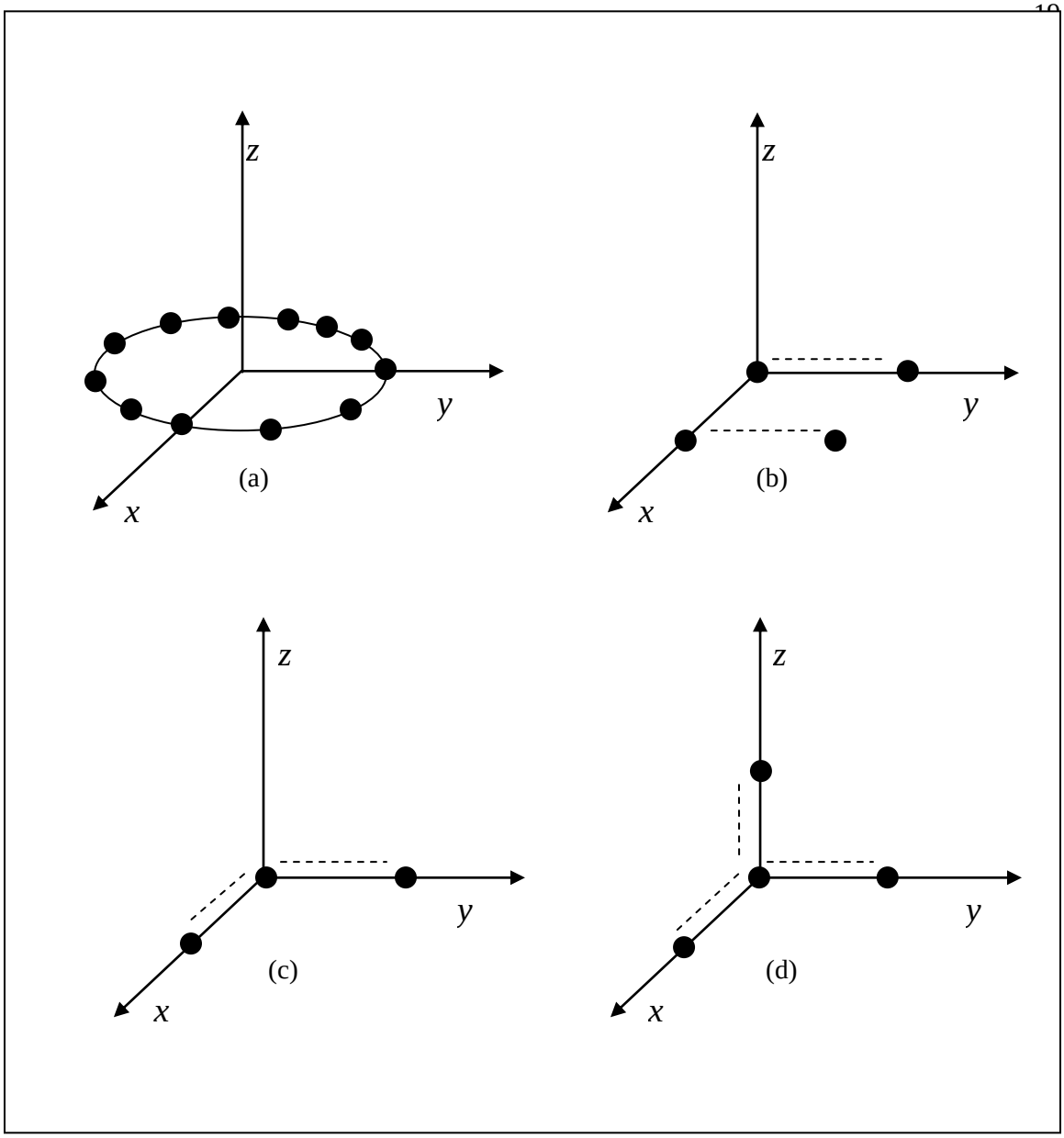


Figure 3-3 Various array configurations: (a) circular, (b) rectangular, (c) L-shaped, and (d) 2L-shaped

In Equation (3.8), $\mathbf{A}(\varphi,\theta)=\left[\mathbf{a}(\varphi_1,\theta_1)\quad \mathbf{a}(\varphi_2,\theta_2)\quad \cdots\quad \mathbf{a}(\varphi_P,\theta_P)\right]^T$ is the $M \times P$ steering matrix and $\mathbf{s}(k)=\left[\mathbf{s}_1(k)\quad \mathbf{s}_2(k)\quad \cdots\quad \mathbf{s}_P(k)\right]^T$ is the $P \times 1$ complex signal vector. Here, again, we assume that $M > P$, $\mathbf{a}_m(\varphi_k,\theta_k)=e^{j2\pi f\mathbf{d}_k^T(\varphi_k,\theta_k)\mathbf{r}_m/c}$ represents the delay between signals received on different antennas. Here, $\boldsymbol{d}_k(\varphi_k,\theta_k)$ is the unit vector pointing towards

source k and $\mathbf{r}_m = [x_m \quad y_m \quad z_m]^T$ is the position vector of sensor m that depends on the geometry of the antenna array

For a planar array, when the signal arrives at azimuth angle φ and elevation angle θ (refer to Fig. 3-2)

$$\mathbf{d}_k(\varphi_k, \theta_k) = [\cos\varphi \sin\theta \quad \sin\varphi \sin\theta \quad \cos\theta]. \tag{3.9}$$

Table 1 tabulates the values of $\mathbf{r}_m$ for four different array geometries of planar arrays considered in this work, namely the circular, rectangular planar, L-shaped, and 2L-shaped.

Table 1 Position Vector for Various Array Configurations

Array Geometry	**Position Vector**
Circular	$\boldsymbol{r}_{\text{circ},k} = [\rho\cos\varphi_k \quad \rho\cos\varphi_k \quad 0]$
Rectangular	$\boldsymbol{r}_{\text{circ},k} = [x_k \quad y_k \quad 0]$
L Shaped	The sensors are placed on the *x* and *y* axes. The sensors on the *x* axis have *y* and *z* co-ordinates equal to zero The sensors on *y* axis have *x* and *z* co-ordinates equal to zero. Origin is common for referencing purpose
2L Shaped	The sensors are placed on the *x*, *y*, and *z* axes. The sensors on the *x* axis have *y* and *z* co-ordinates equal to zero The sensors on the *y* axis have *x* and *z* co-ordinates equal to zero. The sensors on the *z* axis have *x* and *y* co-ordinates equal to zero. Origin is common for referencing purpose

3.2.1 Classical 2-D MUSIC Algorithm Implementation

The 2-D MUSIC algorithm, a logical extension of 1-D method, searches for AOA in azimuth and elevation. The 2-D MUSIC algorithm is implemented as follows:

Step 1: Use the forward iteration of Multi Stage Wiener Filters for separating the noise and signal subspaces. **T** is the matrix whose rows consist of orthogonal Wiener filters.

Step 2: Define the sub-matrix $\mathbf{N}_M(\theta) = \mathbf{T}[:, P+1{:}M]$ representing the signal subspace of the given noisy signal.

Step 3: Exploit the MUSIC estimator $\mathrm{P}_{\mathrm{MUSIC}}(\varphi,\theta)=\dfrac{\mathbf{a}_M^{\mathrm{H}}(\varphi,\theta)\mathbf{a}_M(\varphi,\theta)}{\mathbf{a}_M^{\mathrm{H}}(\varphi,\theta)\mathbf{N}_M(\varphi,\theta)\mathbf{N}_M^{\mathrm{H}}(\varphi,\theta)\mathbf{a}_M(\varphi,\theta)}$ to

produce peaks at the DOA locations,

$$\mathbf{a}_M(\varphi,\theta)=\lfloor 1 \quad \exp(j\varphi_1, j\theta_1) \quad \cdots \quad \exp(j\varphi_{M_L}, j\theta_{M_L})\rfloor.$$

Reduced rank filtering methods assume that the training data of one desired signal is well known. Based on the assumption, a Multi Stage Wiener Filter is employed for separating the noise subspace from signal subspace.

3.3 Wiener Filter

The Wiener filter (WF) is a well known approach to estimation of the unknown signal $d_0[n]$ from an observation $\mathbf{x}_0[n]$ and is optimal in the Minimum Mean Square Error (MMSE) sense. WFs are employed in many applications because they are easily implemented, are optimal in a Bayesian sense, and only rely on second-order statistics that can be estimated with partial knowledge of the transmitted signal.

There are various approaches to reduce the dimensions of the estimation problem, such as the well established Principal Component (PC) method [*Eckart and Young*, 1936; *Hoteling*, 1933]. The PC method transforms the observation vector to lower dimensionality via a matrix composed of eigenvectors belonging to principal eigenvalues. For this method, principal eigenvectors must be estimated and tracked. The major disadvantage of the PC method is that it only accounts for the statistics observation vector and not the relation of observed vector to the desired signal. To overcome this problem, *Goldstein and Reed* [1997] proposed the Cross-Spectral Metric (CSM) method that evolved from the Generalized Sidelobe Canceller (GSC) and incorporates the similarity of the cross correlation vector of the observation and the desired signal with the respective eigenvector. The CSM method chooses the eigenvectors that belong to the

largest CS metric involving $\mathbf{r}_{\boldsymbol{x}_0 d_0}$ as was also suggested by *Byerly and Roberts* [1989]. CSM does not select the principal eigenvectors, in general.

Goldstein and Reed [1997a] ultimately presented the MSNWF. The most important contribution of work by *Goldstein et al.* [1999] demonstrated that rank reduction techniques based on eigenvectors are suboptimum. MSNWF does not require computation of eigenvectors and is thus computationally advantageous. The MSNWF has been applied to a broad spectrum of radar signal processing problems.

The most important contribution to the facilitation of real-time implementation occurred when *Ricks and Goldstein* [2000] showed that the MSNWF can be implemented without blocking matrices. This further reduced computational complexity of MSNWF compared to full-rank RLS or PC-based reduced-rank adaptive filtering. They developed a lattice/modular MSNWF structure that facilitates efficient data-level implementation and provides a low complexity alternative to covariance level processing. The lattice structure has the advantage of avoiding formation of covariance matrix. There may not be enough sample support to form a reliable covariance matrix estimate, especially when the data vector is high-dimensional and/or the signal statistics are rapidly time-varying.

In this work, we use the Multi Stage Wiener Filter–based subspace reduction methods for estimation of signal subspace in the MUSIC [*Schmidt*, 1986] and ESPRIT [*Roy and Kailath*, 1989; *Huang et al*, 2005a; *Huang et al*, 2005b] on GNU Radio platform for 1-D angle-of-arrival estimation. We extend the reduced complexity subspace estimation method for 2-D MUSIC algorithm application.

In this section, we briefly review the theory of the Wiener filter in order to explain our notation. Before we discuss the lattice structure of MSNWF, we concentrate on the original WF approach.

3.3.1 Classical Wiener Filter

The Wiener filter (WF), $\mathbf{w}^{\mathrm{H}} \in C^{M}$, is a well known estimator of the desired signal $d_0[n] \in C$ from the observation data $\mathbf{x}(k) \in C$ in the minimum mean square error (MMSE) sense. The desired signal, assumed to be a zero-mean white Gaussian process, is estimated by applying the linear filter to the observation signal which is assumed to be a multivariate zero-mean Gaussian process. The error of the estimation $\varepsilon_0[n]$ can be written as

$$\varepsilon_0[n] = d_0[n] - \hat{d}_0[n] = d_0[n] - \mathbf{w}^{\mathrm{H}}\mathbf{x}_0[n]. \tag{3.10}$$

Here, $\hat{d}_0[n] = \mathbf{w}^{\mathrm{H}}\mathbf{x}_0[n]$ is the estimate obtained from Wiener filtering for the desired signal $d_0[n]$. The variance of the estimation error is the mean squared error

$$MSE_0 = E\left\{\left|\varepsilon_0[n]\right|^2\right\} = \sigma_{d_0}^2 - \mathbf{w}^{\mathrm{H}}\mathbf{r}_{\mathbf{x}_0,d_0} - \mathbf{r}^{\mathrm{H}}{}_{\mathbf{x}_0,d_0}\mathbf{w} + \mathbf{w}^{\mathrm{H}}\mathbf{R}_{\mathbf{x}_0}\mathbf{w}\,, \tag{3.11}$$

where $\mathbf{R}_{\mathbf{x_0}} = E\left\{\mathbf{x_0}[n]\mathbf{x}_0^{\mathrm{H}}[n]\right\} \in C^{N\times N}$ is the covariance matrix of observed signal and $\sigma_{d_0}^2 = E\left\{\left|d_0[n]\right|^2\right\}$ is the variance of the desired signal $d_0[n]$ and the cross correlation between the desired signal $d_0[n]$ and the observation signal $x_0[n]$ given as $\mathbf{r}_{\mathbf{x}_0,d_0} = E\left\{\mathbf{x_0}[n]\mathbf{d_0}^{*}[n]\right\}$. The Wiener filter is designed using following design criteria

$$\mathbf{w}_{wf} = \arg\min\, E\left\{[d(k) - \mathbf{w}^{\mathrm{H}}\mathbf{x}(k)]^2\right\}, \tag{3.12}$$

where $\mathbf{w}^{\mathrm{H}}\mathbf{x}(k)$ represents the estimate of the desired signal $d(k)$, and $\mathbf{w}_{wf}$ is $M \times 1$ complex, linear filter. Solving (3.12) leads to the Wiener–Hopf equation,

$$\mathbf{R}_x\mathbf{w}_{wf} = \mathbf{r}_{\mathbf{x}_0\mathrm{d}_0}\,, \tag{3.13}$$

and the Wiener Filter

$$\boldsymbol{w}_{wf} = \boldsymbol{R}_x^{-1}\mathbf{r}_{\mathbf{x_0}d_0} C^{N}\,, \tag{3.14}$$

where $\mathbf{R}_x = E[\mathbf{x}(k)\mathbf{x}^{\mathrm{H}}(k)]$ and $\mathbf{r}_{\mathbf{x},d} = E[\mathbf{x}(k)d^*(k)]$. The classical Wiener filter as shown in Figure 3-4, *i.e.*, $\mathbf{w}_{wf} = \mathbf{R}_x^{-1}\mathbf{r}_{\mathbf{x},d}$, is computationally intensive for large *M* since the inverse of the covariance matrix is involved.

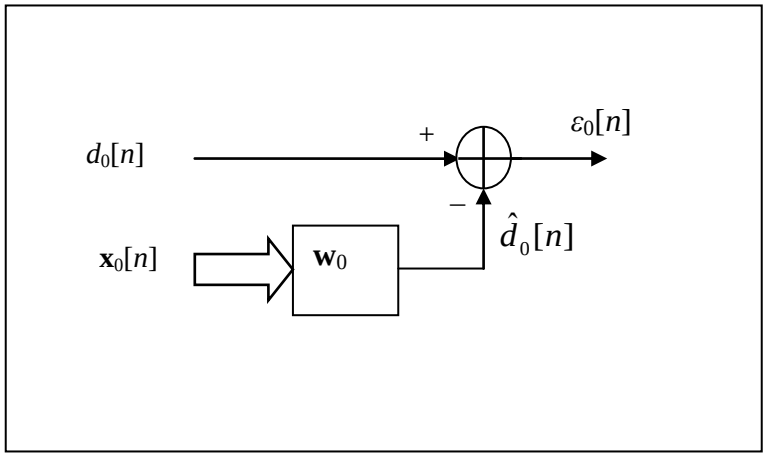


Figure 3-4 Wiener Filter

3.3.2 Multi-Stage Nested Wiener Filter

The Multi-Stage Nested Wiener Filter (MSNWF) [*Goldstein and Reed*, 1997a] provides an approximate solution of the Wiener–Hopf equation (3.12), without having to find the inverse or the eigen decomposition of the covariance matrix. The approximation for the Wiener filter is found by stopping the recursive algorithm after *D* steps, hence, the approximation lies in a *D*-dimensional subspace of C^N.

The first step of the MSNWF algorithm is to apply a full rank pre-filtering matrix $\mathbf{T}_1$ as shown in Figure 3-5 of the form

$$\mathbf{T}_1 = \begin{bmatrix} \mathbf{h}_1^{\mathrm{H}} \\ \mathbf{B}_1 \end{bmatrix} \in C^{N \times N} \tag{3.15}$$

to get the new observation signal

$$\mathbf{z}_1 = \mathbf{T}_1\mathbf{x}_0[n] = \begin{bmatrix} \mathbf{h}_1^{\mathrm{H}}\mathbf{x}_0[n] \\ \mathbf{B}_1\mathbf{x}_0[n] \end{bmatrix} = \begin{bmatrix} d_1[n] \\ \mathbf{x}_1[n] \end{bmatrix} \in C^{N\times N}. \tag{3.16}$$

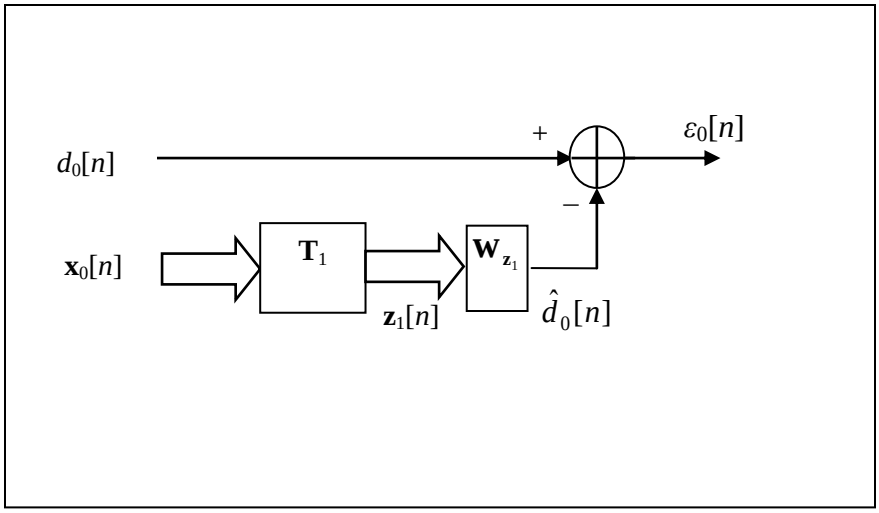


Figure 3-5 Wiener Filter with pre filtering matrix

It has been postulated and proven [*Goldstein et al.*, 1998] that

> "If the observation $x_0[n]$ to estimate $d_0[n]$ is pre-filtered by a full rank matrix **T** the Wiener filter $\mathbf{w}_{wf}$ to estimate $d_0[n]$ from $z_1[n]$ leads to the same estimate $\hat{d}_0[n]$ thus, the MMSE is unchanged."

Thus, $\mathbf{T}_1$ does not change the MSE of estimate of $\mathbf{d}_0[n]$. The rows of $\mathbf{B}_1$ are chosen to be orthogonal to $\mathbf{h}_1^{\mathrm{H}}$; therefore, $\mathbf{B}_1\mathbf{h}_1 = \mathbf{0}$ or $\mathbf{B}_1 = \text{null}\,(\mathbf{h}_1^{\mathrm{H}})^{\mathrm{H}}$.

The intuitive choice for the first row $\mathbf{h}_1^{\mathrm{H}}$ is the vector that, when applied to $\mathbf{x}_0[n]$, gives a scalar signal $d_1[n]$ that has maximum correlation with the desired signal $d_0[n]$. Without loss of generality, we assume that $\|\mathbf{h}_1\|_2 = 1$ and force $d_1[n]$ to be "in-phase" with $d_0[n]$, i.e., the correlation between $d_0[n]$ and $d_1[n]$ is real, which is motivated by the trivial case when $d_1[n] = d_0[n]$. This criterion, proposed by *Joham and Zoltowski* [2006], is different from the one proposed

in *Goldstein et al.* [1998] and does not optimize the absolute value of the correlation, thus preserving the phase information. With this consideration, *Joham and Zoltowski* produced the following optimization problem and the resulting normalized matched filter:

$$\mathbf{h}_1 = \arg\max_{\mathbf{h}} E\left\{\mathrm{Re}(d_1[n]d_0^*[n])\right\} = \arg\max_{\mathbf{h}} \frac{1}{2}(\mathbf{h}^{\mathrm{H}}\mathbf{r}_{\mathbf{x}_0 d_0} + \mathbf{r}_{\mathbf{x}_0 d_0}^{\mathrm{H}}\mathbf{h}), \text{ s.t. } \mathbf{h}^{\mathrm{H}}\mathbf{h} = 1, \quad (3.17)$$

$$\mathbf{h}_1 = \frac{\mathbf{r}_{\mathbf{x}_0 d_0}}{\left\|\mathbf{r}_{\mathbf{x}_0 d_0}^{\mathrm{H}}\right\|_2} \in C^N. \quad (3.18)$$

Note that $d_1[n]$ contains all information about $d_0[n]$, which can be found in $\mathbf{x}_0[n]$ since $d_0[n]$ is a scalar; thus, the information about $d_0[n]$ lies in a one dimensional subspace of C^N and, with the matched filter, we select only these portions of $\mathbf{x}_0[n]$ that are included in this subspace. Moreover, the second part of $\mathbf{z}_1[n]$, namely $\mathbf{x}_1[n]$, does not contain any information about $d_0[n]$ because of the orthogonality of $\mathbf{B}_1$ and $\mathbf{h}_1$.

Again, we have to solve the Wiener–Hopf equation of the new system obtained after prefiltering, which yields

$$\mathbf{w}_{\mathbf{z}_1} = \mathbf{R}_{z_1}^{-1}\mathbf{r}_{\mathbf{z}_1, d_0} \in C^N. \quad (3.19)$$

Here, the covariance matrix of $\mathbf{z}_1$ can be expressed from the statistics of $d_1[n]$ and $\mathbf{x}_1[n]$

$$\mathbf{R}_{\mathbf{z}_1} = \begin{bmatrix} \sigma_{d_1}^2 & \mathbf{r}_{\mathbf{x}_1, d_1}^{\mathrm{H}} \\ \mathbf{r}_{\mathbf{x}_1, d_1} & \mathbf{R}_{\mathbf{x}_1} \end{bmatrix} \in C^{N-1 \times N-1}. \quad (3.20)$$

This way, the covariance matrix of $\mathbf{z}_1$ can be expressed from the statistics of $d_1[n]$ and $\mathbf{x}_1[n]$ with variance of $d_1[n]$ $\sigma_{d_1}^2 = E\left\{\left|d_1[n]\right|^2\right\} = \mathbf{h}_1^{\mathrm{H}}\mathbf{R}_{\mathbf{x}_0}\mathbf{h}_1$ and cross correlation of $d_1[n]$ and $\mathbf{x}_1[n]$

$\mathbf{r}_{\mathbf{x}_1, d_1} = E\left\{\mathbf{x}_1[n]d_1^*[n]\right\} = \mathbf{B}_1\mathbf{R}_{\mathbf{x}_0}\mathbf{h}_1 \in C^{N-1}$ and the covariance matrix of $\mathbf{x}_1[n]$ is given as

$\mathbf{R}_{\mathbf{x}_1} = E\left\{\mathbf{x}_1[n]\mathbf{x}_1^{\mathrm{H}}[n]\right\} = \mathbf{B}_1\mathbf{R}_{\mathbf{x}_0}\mathbf{B}_1^{\mathrm{H}} \in C^{N-1 \times N-1}$.

The cross correlation vector of the pre-filtered observation signal $\mathbf{z}_1[n]$ and the desired signal $d_1[n]$ reveals the utility of the choice of pre-filtering:

$$\mathbf{r}_{\mathbf{z}_1,d_0} = \mathbf{T}_1\mathbf{r}_{\mathbf{x}_0,d_1} = \left\|\mathbf{r}_{\mathbf{x}_0,d_0}\right\|_2 e_1 \in R^N . \tag{3.21}$$

Here, e_1 denotes the unit norm vector with a one at i^{th} position. Therefore, the Wiener filter $\mathbf{w}_{z_1}$ of the pre-filtered signal $\mathbf{z}_1[n]$ is simply a weighted version of the first column of the inverse of the covariance matrix $\mathbf{R}_{\mathbf{z}_1}$ in (3.20). After applying the inversion lemma for partitioned matrices we end up with [*Goldstein and Reed*, 1997a; *Goldstein et al.*, 1998],

$$\mathbf{w}_{z_1} = \alpha_1 \begin{bmatrix} 1 \\ -\mathbf{R}_{\mathbf{x}_1}^{-1}\mathbf{r}_{\mathbf{x}_1 d_1} \end{bmatrix} \in C \text{, where } \alpha_1 = \left\|\mathbf{r}_{\mathbf{x}_0,d_0}\right\|_2 \left(\sigma_{d_1}^2 - \mathbf{r}_{\mathbf{x}_1,d_1}^{\mathrm{H}}\mathbf{R}_{\mathbf{x}_1}^{-1}\mathbf{r}_{\mathbf{x}_1,d_1}\right)^{-1}. \tag{3.22}$$

Equation (3.22) is the important fundamental equation for interpreting the theory of MSNWF.

The significant observation resulting from Equation 3.22 is that $\left(\sigma_{d_1}^2 - \mathbf{r}_{\mathbf{x}_1,d_1}^{\mathrm{H}}\mathbf{R}_{\mathbf{x}_1}^{-1}\mathbf{r}_{\mathbf{x}_1,d_1}\right)^{-1}$, when applied to $\mathbf{z}_1[n]$, gives the error signal $\varepsilon_1[n]$ of the Wiener filter, which estimates d_1[n] from $\mathbf{x_1}[n]$:

$$\varepsilon_1[n] = d[n] - \hat{d}[n] = d[n] - \mathbf{w}_1^{\mathrm{H}}\mathbf{x}_1[n] = [1 \quad -\mathbf{w}_1^{\mathrm{H}}]\mathbf{z}_1[n], \tag{3.23}$$

where $\mathbf{w}_1$ is the Wiener Filter

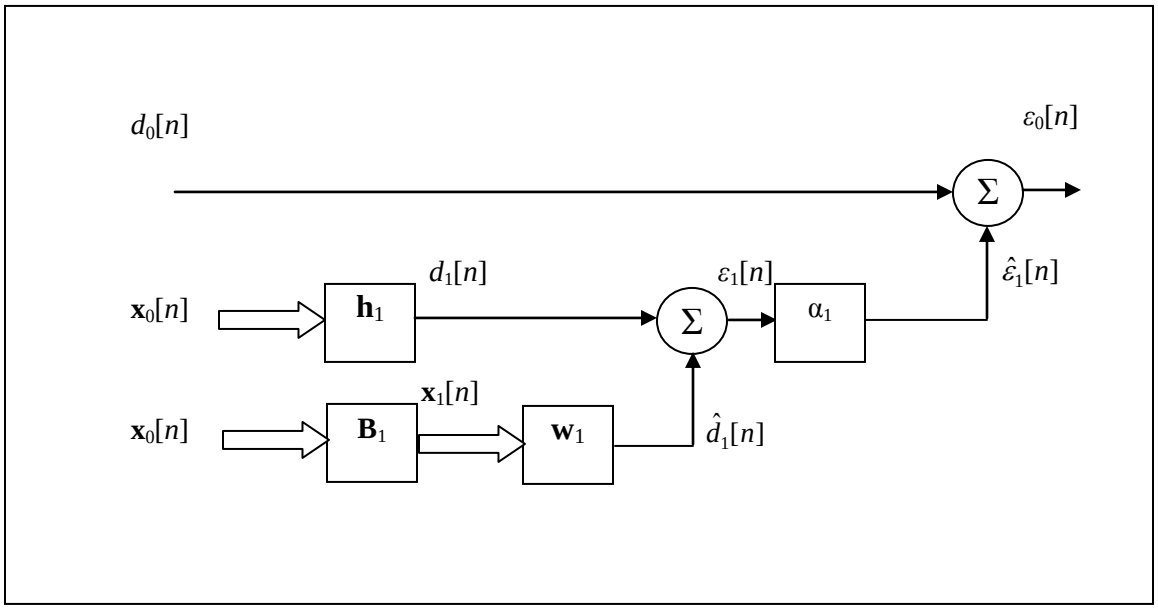
$$\mathbf{w}_1 = \mathbf{R}_{\mathbf{x}_1}^{-1}\mathbf{r}_{\mathbf{x}_1,d_1} \in C^{N-1}. \tag{3.24}$$

This observation immediately leads to the next step in the MSNWF development. In the second step, the output of the Wiener filter $\mathbf{w}_1$ with dimension $N - 1$ can be replaced by the weighted error signal $\varepsilon_2[n]$ of a Wiener filter $\mathbf{w}_2$ that estimates the output signal $d_2[n]$ of the matched filter $\mathbf{h}_2$ from the blocking matrix output $\mathbf{x}_2[n] = \mathbf{B}_2\mathbf{x}_1[n]$.

Note that, the factor α_1 is a scalar Wiener filter to estimate $d_0[n]$ from the scalar $\varepsilon_1[n]$ is the MMSE of the Wiener filter $\mathbf{w}_1$ and the cross correlation between the scalar observation signal $\mathbf{w}_1[n]$ and the desired signal $d_0[n]$ is the norm of the matched filter $\mathbf{r}_{\mathbf{x}_0,d_0}$, thus,

$$\alpha_1 = \sigma_{\varepsilon_1}^{-1}\mathbf{r}_{\varepsilon_1,d_0} = \left(\sigma_{d_1}^2 - \mathbf{r}_{\mathbf{x}_1,d_1}^{\mathrm{H}}\mathbf{R}_{\mathbf{x}_1}^{-1}\mathbf{r}_{\mathbf{x}_1,d_1}\right)^{-1}\left\|\mathbf{r}_{\mathbf{x}_0,d_0}\right\|_2 . \tag{3.25}$$

These two interpretations lead to the well known structure in Figure 3.6. The Wiener filter $\mathbf{w}_0$ is replaced by the scalar Wiener filter α_1, which estimates $d_0[n]$ from the error signal $\varepsilon_1[n]$ of the vector Wiener filter $\mathbf{w}_1$.



3-6 Multi Stage Wiener Filter after first stage

Equation 3.22 also suggests that the MSNWF can be created without knowledge of the covariance matrix $\mathbf{R}_{\mathbf{x}_0}$; therefore, the implementation of a MSNWF is simplified, because at each step the Wiener filter is replaced by the normalized matched filter and the next stage. Since the matched filter is simply the cross correlation between the new observation $\mathbf{x}_i[n]$ and the new desired signal $d_i[n]$ at each step only an estimation of this cross correlation is needed. Note, however, that estimating the entire cross correlations and estimating the covariance matrix lead to

the same estimate $\hat{d}_0[n]$ at the same computational expense. Figure 3-7 shows the four-stage nested Wiener filter.

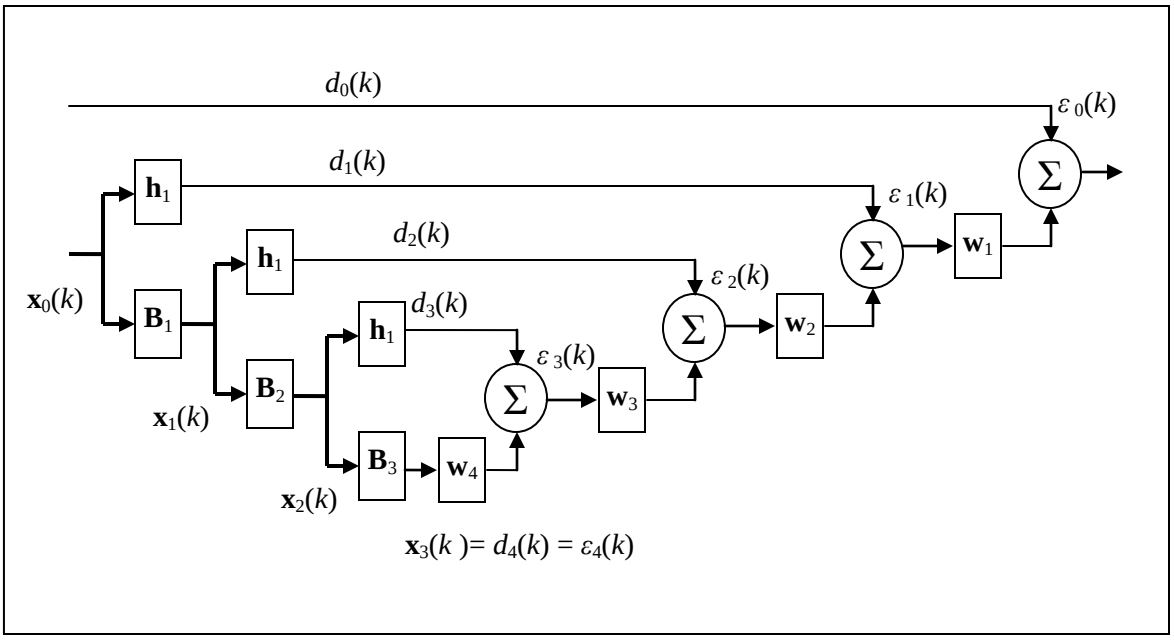


Figure 3-7 Lattice structure of four-stage nested Wiener filter

3.3.2 Data-Level Lattice Structure of Multi Stage Wiener Filter

The four-stage MSNWF structure depicted in Figure 3-7 has several drawbacks. For example, it requires a forward recursion to determine $\mathbf{h}_i$ followed by a backwards recursion to determine scalar weighting functions $\mathbf{w}_i$. The computational burden is further increased due the formation of blocking matrices and the change in scalar Weiner weights $\mathbf{w}_i$ each time a new stage is added. *Ricks and Goldstein* [2000] in an ingenuous work showed that the MSNWF can be implemented without blocking matrices as shown in Figure 3.8, further reducing the computational complexity of MSNWF relative to full-rank RLS or PC based reduced-rank adaptive filtering methods. *Ricks and Goldstein* also developed a lattice/modular MSNWF structure that facilitates efficient data-level implementation as an alternative to covariance level processing. Avoidance of covariance matrix formation reduces computationally complexity and facilitates real-time implementation.

At times there may not be enough sample support to form a reliable covariance matrix estimate, especially when data vector is high-dimensional and/or the signal statistics are rapidly time-varying.

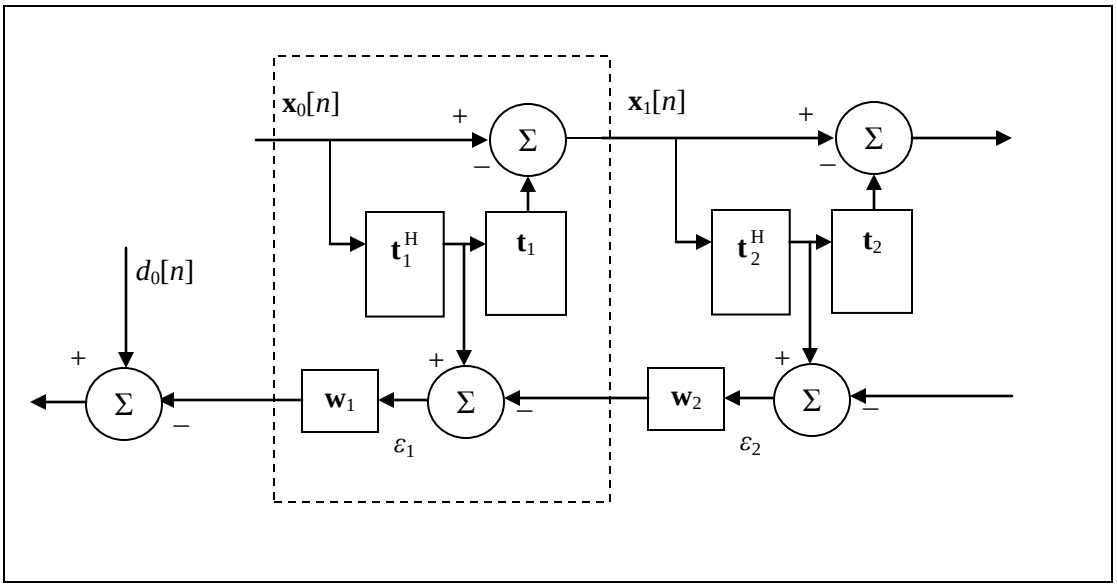


Figure 3-8 Lattice structure of Multi Stage Nested Wiener Filter

Joham and Zoltowski [2006] have proven that stopping MSNWF at stage D constrains $\mathbf{w}_0$ to lie in D-dimensional Krylov subspace

$$\mathbf{w}_0 \in \text{range}\{\mathbf{r}_{\mathbf{x}_0,d_0}, \mathbf{R}_{\mathbf{x}_0}\mathbf{r}_{\mathbf{x}_0,d_0}, \ldots, \mathbf{R}_{\mathbf{x}_0}^{D-1}\mathbf{r}_{\mathbf{x}_0,d_0}\}. \tag{3.26}$$

The MSWF based on the data-level lattice structure is given as follows [*Ricks and Goldstein*, 2000]:

Initialization (3.27)

$$d_0(k) = s_1(k), x_0(k) = x(k)$$

Forward Recursion

$$\mathbf{t}_i = \sum_{n=0}^{M-1} d_{i-1}^{*}[n]\mathbf{x}_i[n]$$
$$\mathbf{h}_i = \mathbf{t}_i / \|\mathbf{t}_i\|_2$$
$$d_i[n] = \mathbf{t}_i^H x_{i-1}[n], n = 0,1,\ldots M-1$$
$$\mathbf{x}_i[n] = \mathbf{x}_{i-1}[n] - d_i[n]\mathbf{h}_i$$
$$\varepsilon_D[n] = d_D[n]$$

Backward Recursion

$$w_{i+1} = \left\{\sum_{n=0}^{M-1} d_i[n]\varepsilon_{i+1}^{*}[n]\right\} \Bigg/ \left\{\sum_{n=0}^{M-1} \left|\varepsilon_{i+1}[n]\right|^2\right\}$$
$$\varepsilon_i[n] = d_i[n] - w_{i+1}\varepsilon_{i+1}[n], n = 0,1,\ldots,M-1$$
$$w_0^{(D)} = \sum_{i=1}^{D} (-1)^{i+1}\left\{\prod_{l=1}^{i} w_i\right\}\boldsymbol{h}_i$$

3.4 Low Complexity Signal and Noise Subspace Separation Method

It is shown in [*Huang*, 2004] that all the matched filters $\mathbf{h}_i$, $i = 1, 2\ldots D$ ($D \le P$) are contained in the column space of $\mathbf{A}(\theta)$, i.e., the orthogonal matched filters $\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_P$ span the signal subspace, namely

$$\text{span}\{\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_P\} = \text{col}\{\mathbf{A}(\theta)\}. \tag{3.28}$$

Since all the matched filters $\mathbf{h}_1$, $\mathbf{h}_2\ldots$ $\mathbf{h}_M$ are orthogonal for the special choice of the blocking matrix $\mathbf{B}_i = \mathbf{I} - \mathbf{h}_i\mathbf{h}_i^{\mathrm{H}}$, the matched filters after the P stage of the MSWF are orthogonal to the signal subspace, *i.e.*, $\mathbf{h}_i \perp \text{col}\{\mathbf{A}(\theta)\}$ for $i = P + 1, P + 2, \cdots; M$. Therefore, the last $M - P$ matched filters span the orthogonal complement of the signal subspace, namely the noise subspace

$$\text{span}\{\mathbf{h}_{P+1}, \mathbf{h}_{P+2}, \cdots, \mathbf{h}_M\} = \text{null}\{\mathbf{A}(\theta)\}. \tag{3.29}$$

The lattice structure of MSNWF indicates that the noise subspace can be readily obtained by performing the forward recursions of the MSWF. The MUSIC estimator based on the noise

subspace is exploited to produce peaks at the DOA locations. We first review the method for 1-D AOA estimation by [*Huang et al.*, 2005a; 2005b] before presenting an extension to 2-D AOA.

In the presence of multipath fading or heavy interferers or for coherent signals, the noise subspace estimated by multi stage Wiener filter or the Singular Value Decomposition (SVD) method may be incorrect. That is to say, the last $M - P$ matched filters may not span a noise subspace for the case where the signals are coherent. As a result, we must resort to the smoothing techniques to decorrelate the coherent signals and restore the order of covariance matrix.

For the spatial smoothing technique [*Shan et al.*, 1985], a linear array consisting of M sensors is subdivided into L subarrays. Thereby, the number of elements per subarray is $M_L = M - L + 1$, for $l = 1, 2 \ldots L$. Let the $M_L \times M$ matrix $\mathbf{J}_l$ be a selection matrix, which takes the following form

$$\mathbf{J}_l = \left[\mathbf{0}_{M_L \times (l-1)} \quad \vdots \quad \mathbf{I}_{M_L \times M_L} \quad \vdots \quad \mathbf{0}_{M_L \times (M-l-M_L+1)}\right]. \tag{3.30}$$

The selection matrix $\mathbf{J}_l$ is exploited to select part of the $M \times N$ observation data matrix

$$\overline{\mathbf{X}}_0 = \left[x_0(0) \quad x_0(1) \quad \cdots \quad x_0(N-1)\right]. \tag{3.31}$$

This associates with the l^{th} sub array. Hence, the spatially smoothed $M_L \times LN$ data matrix $\overline{\mathbf{X}}_0$ is constructed by

$$\overline{\mathbf{X}}_0 = \left[\mathbf{J}_1\mathbf{X}_0 \quad \mathbf{J}_2\mathbf{X}_0 \quad \cdots \quad \mathbf{J}_L\mathbf{X}_0\right] \in C^{M_L \times LN}. \tag{3.32}$$

The "smoothed" training signal vector should take the following form

$$\overline{\mathbf{d}}_0 = \left[d_0 \quad d_0 \quad \cdots \quad d_0\right] \in C^{LN}. \tag{3.33}$$

And, the i^{th} spatially smoothed pre-filter of the MSWF is given by

$$\hat{\mathbf{h}}_i = \frac{\hat{\mathbf{r}}_{\mathbf{x}_{i-1}} d_{i-1}}{\left\| \hat{\mathbf{r}}_{\mathbf{x}_{i-1}} d_{i-1} \right\|_2} = \frac{\hat{\mathbf{X}}_{i-1} \overline{d}^*_{i-1}}{\left\| \hat{\mathbf{X}}_{i-1} \overline{d}^*_{i-1} \right\|_2}. \tag{3.34}$$

Therefore, for coherent signals the low complexity methods for DOA estimation based on the MSWF are given as follows:

Step1: Apply spatial smoothing to the $M \times N$ observation data matrix $\mathbf{X}_0$, obtain the spatially smoothed $M_{\mathrm{L}} \times NL$ data matrix $\overline{\mathbf{X}}_0$.

Step2: Construct the spatially smoothed training data vector $\overline{\mathbf{d}}_0$

$$\overline{\mathbf{d}}_0 = \begin{bmatrix} d_0 & d_0 & \cdots & d_0 \end{bmatrix}$$

Step3: Perform the following recursions

For $i = 1,2,....,M_L$

$$\overline{\mathbf{h}}_i = \overline{\mathbf{X}}_{i-1}\overline{\mathbf{d}}_{i-1} \Big/ \left\| \overline{\mathbf{X}}_{i-1}\overline{\mathbf{d}}_{i-1} \right\|_2$$

$$\overline{\mathbf{d}}_i = \overline{\mathbf{h}}_i^{\mathrm{H}} \overline{\mathbf{X}}_{i-1}$$

$$\overline{\mathbf{X}}_i = \overline{\mathbf{X}}_{i-1} - \overline{\mathbf{h}}_i \overline{\mathbf{d}}_i$$

Obtain the subspace consisting of orthogonal wiener filters

$$\mathbf{T} = \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_1 & \cdots & \mathbf{h}_{M_L} \end{bmatrix}$$

Apply the 1-D MUSIC estimator or Root MUSIC estimator or ESPRIT estimator on the resulting subspaces to obtain an estimate of the angle-of-arrival.

3.4.1 Low Complexity 2-D MUSIC

The 2-D algorithm can be summarized as follows. Notice here that we are not performing smoothing on the faded data.

Perform the following recursions

For $i = 1,2,....,M$

$$\overline{\mathbf{h}}_i = \overline{\mathbf{X}}_{i-1}\overline{\mathbf{d}}_{i-1} \Big/ \left\| \overline{\mathbf{X}}_{i-1}\overline{\mathbf{d}}_{i-1} \right\|_2$$

$$\overline{\mathbf{d}}_i = \overline{\mathbf{h}}_i^{\mathrm{H}} \overline{\mathbf{X}}_{i-1}$$

$$\overline{\mathbf{X}}_i = \overline{\mathbf{X}}_{i-1} - \overline{\mathbf{h}}_i \overline{\mathbf{d}}_i$$

Obtain the subspace consisting of orthogonal wiener filters

$$\mathbf{T} = \begin{bmatrix} \mathbf{h}_1 & \mathbf{h}_1 & \cdots & \mathbf{h}_M \end{bmatrix}$$

Apply the 2-D MUSIC estimator or Root MUSIC estimator or ESPRIT estimator, or any other subspace-based estimator for that matter, on the resulting subspaces to obtain an estimate of the angle-of-arrival.

It should be noted here that the low-complexity algorithm given here avoids the formation of blocking matrices, and all the operations involve only complex matrix-vector products, thereby requiring the computational complexity of $O(MN)$ for each matched filter $\boldsymbol{h}_i$, $i \in \{1,2,\cdots,M\}$. To fulfill the estimation of the noise subspace, M stages of the MSWF are needed. Thus, the computational cost of the proposed method is only $O(M^2N)$ flops. However, the classical MUSIC method includes the estimation of the spatially smoothed covariance matrix and its eigen decomposition, which require $O(M^2N + M^2)$ flops. The classical ESPRIT method resorts to the estimate of the covariance matrix and its eigen decomposition, which require $O(M^2N + M^3)$ flops. Therefore, the low-complexity algorithm involving only complex matrix-vector products is more computationally efficient requiring only $O(M^2N)$ flops.

3.5 Figures of Metric for Performance Evaluation

Accuracy and resolution are the important figures of metric used to evaluate the performance of angle-of-arrival measurement system.

3.5.1 Accuracy

The Cramer Rao Bound (CRB) of the variance of direction estimation errors provides the relatively simple lower bound for useful characterization of the achievable accuracy of a

Direction Finding System. In general, it can be said that the error variance increases sharply when emitter separation is smaller than array beamwidth. Error variance is independent of beamwidth when the separation exceeds the beamwidth and is equal to the variance of single emitter case. It is expected that the error variance approached CRB at high enough SNR. The CRB forms the fundamental measure of performance evaluation in problems where the exact minimum-mean-square estimation error is difficult to evaluate. From our previous discussion of MSNWF, we can claim that the accuracy or MSE of MUSIC algorithms remains unchanged whether SVD or Eigen Analysis based methods are used for signal and noise space separation or MSNWF is used for the purpose.

For multi-parameter estimation, CRB inequality has the form.

$$MSE\{\hat{\theta}_i\} = \mathrm{var}(\hat{\theta}_i) \geq [\mathbf{J}^{-1}]_{ii} \tag{3.35}$$

It has been derived [*Tuncer and Friedlander*, 2009] that

$$CRB \approx \frac{1}{2N\left|\dot{\mathbf{a}}(\theta)\right|^2 SNR}. \tag{3.36}$$

Here, N is the number of snapshots and $\dot{\mathbf{a}}(\theta)$ is the derivative of array manifold with respect to θ. For M element circular array with a radius R, we have

$$CRB \approx \frac{1}{8\pi^2 N M/2 SNR}. \tag{3.37}$$

Of all the planar arrays, circular arrays have the worst performance. In general [*Baysal and Randolph*, 2003],

$$CRB = \left[\frac{\partial \mathrm{d}(\varphi,\theta)}{\partial\varphi} \quad \frac{\partial \mathrm{d}(\varphi,\theta)}{\partial\theta}\right]^T \mathbf{B} \left[\frac{\partial \mathrm{d}(\varphi,\theta)}{\partial\varphi} \quad \frac{\partial \mathrm{d}(\varphi,\theta)}{\partial\theta}\right] \cdot \frac{2N(2\pi f M\boldsymbol{\sigma}_s^2/c)^2}{\boldsymbol{\sigma}_n^2(\boldsymbol{\sigma}_n^2+\boldsymbol{\sigma}_s^2 M)}, \tag{3.38}$$

where

$$\mathbf{B} = \frac{1}{M}\sum_{m=1}^{M}(\mathbf{r}_m - \mathbf{r}_c)(\mathbf{r}_m - \mathbf{r}_c)^T. \tag{3.39}$$

$\boldsymbol{\sigma}_n^2$ and $\boldsymbol{\sigma}_s^2$ are the variances of noise and signal, respectively. *N* is the number of snapshots and $\underline{\mathbf{r}}_c$ is the geometrical mean of array manifold given as

$$\mathbf{r}_c = \frac{1}{M}\sum_{m=1}^{M}\mathbf{r}_m \ . \tag{3.40}$$

Thus, the Mean Squared Error of estimation of AOA depends on the variance of noise and signal, the number of sensors, and the number of snapshots. From the CRB expression, it appears that the only parameter that is going to change, according to the geometry of the antennas array, is the vector position $\mathbf{r}_m$.

3.5.2 Resolution

Estimation accuracy is not sufficient for characterizing the performance of a direction finding system when multiple signals are present. Resolution is the ability of system to distinguish between tightly spaced emitters. Resolution is closely related to number of distinct components in the composite signal received by an emitter. Accuracy is related to estimation and resolution is related to detection or decision. Also, the knowledge of only the number of emitters is not sufficient for meaningful results. Assume there are two signals present in a composite received noisy signal, one strong with high SNR and the other one weak with low SNR. The estimation algorithm used to evaluate the angles of arrival without considering the SNR information will produce meaningless information for the signal with low SNR. As a rule of thumb, the SNR required by MUSIC and its variations is inversely proportional to the fourth power of desired resolution limit.

Chapter 4 Simulation Results

Simulations were conducted to illustrate the performance of the proposed algorithm. We simulated the algorithm for 4-QAM (Quadrature Amplitude Modulation), 16-QAM, 64-QAM, 4 PSK (4-Phase Shift Keying), 8-PSK, and DPSK. There are some variations in the performance of algorithms in terms of accuracy as well as resolution. The variations can be expected because of different distribution functions of different modulation types. Here, we present results of QAM for 1-D algorithms and DQPSK for the 2-D algorithms.

4.1 1D MUSIC

The MSNWF Based methods for 1-D Angle-of-arrival were simulated in MATLAB to evaluate performance in the presence of an added white Gaussian noise channel, specifically, signal-to-noise ratios (SNR) in the range of 2 to 20. The number of antennas M is chosen to be 12, and the number of received signals $P = 3$. The modulation type was chosen as 4-QAM. For subspace smoothing, the size of smoothing matrix M_L was chosen to be 5. The performance of MSWF-based angle-of-arrival algorithms in presence of different SNR values was evaluated. Figure 4-1 shows that smoothing actually degrades the performance of algorithm—convergence is slow. In presence of multipath, smoothing gives better performance. Smoothing also increases the number of data points to be processed and thereby increases the overall computational burden. The angle estimation results for both algorithms are consistent. There was essentially no degradation in performance of the ESPRIT algorithm from SNR values of 2 to 45. The number of bits processed for estimation was 64.

Figure 4.1 shows the results of angle-of-arrival measurements using MUSIC and Figure 4.2 shows the results of angle-of-arrival measurements for ESPRIT algorithm. For the smoothing

case, the path delays are assumed to be $[0 \;\; 300\times10^{-9} \;\; 500\times10^{-9}]$ and path gains are assumed to be $[0 \;\; -5 \;\; -10]$. The angles of arrival are [30 60 70] in degrees.

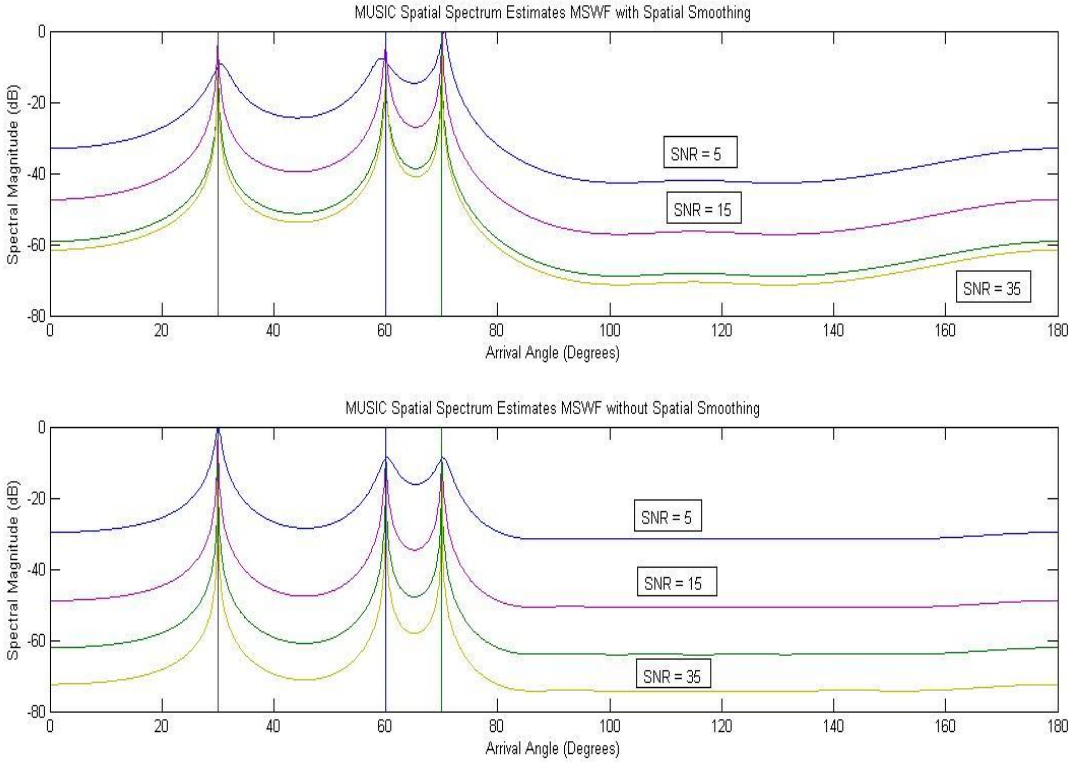


Figure 4-1Classical MUSIC angle-of-arrival with and without smoothing AWGN channel

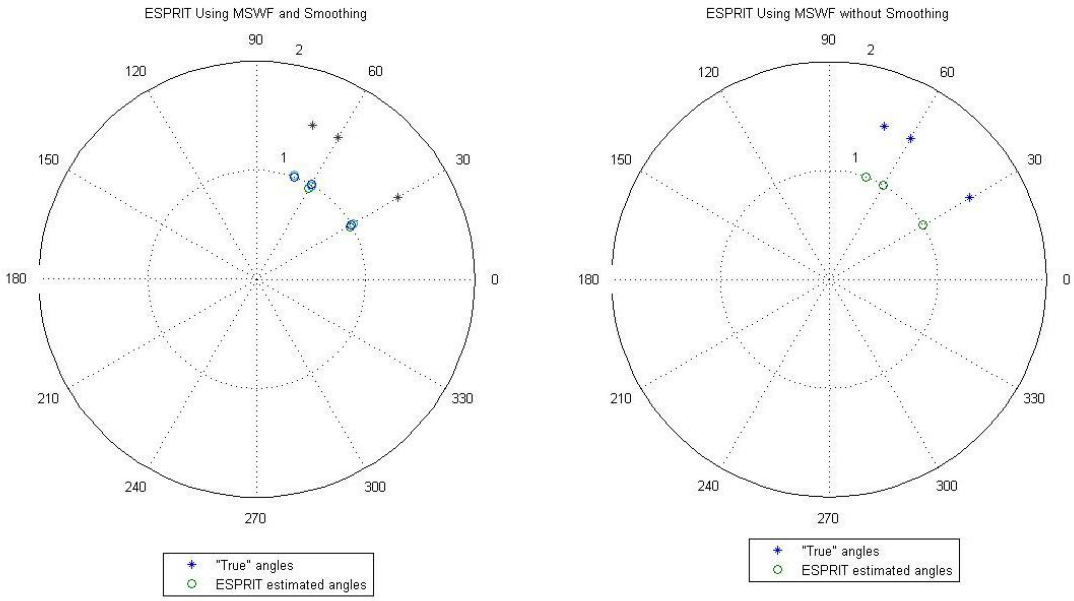


Figure 4-2 ESPRIT angle-of-arrival with and without smoothing AWGN channel

4.2 2-D MUSIC

The Wiener Filter Based Classical MUSIC method was simulated in MATLAB to evaluate performance in the presence of an added white Gaussian noise channel as well as multipath fading. The number of antennas M in the simulations was 16, the number of received signals P = 3. The performance was again evaluated in presence of multipath and added white Gaussian noise channels.

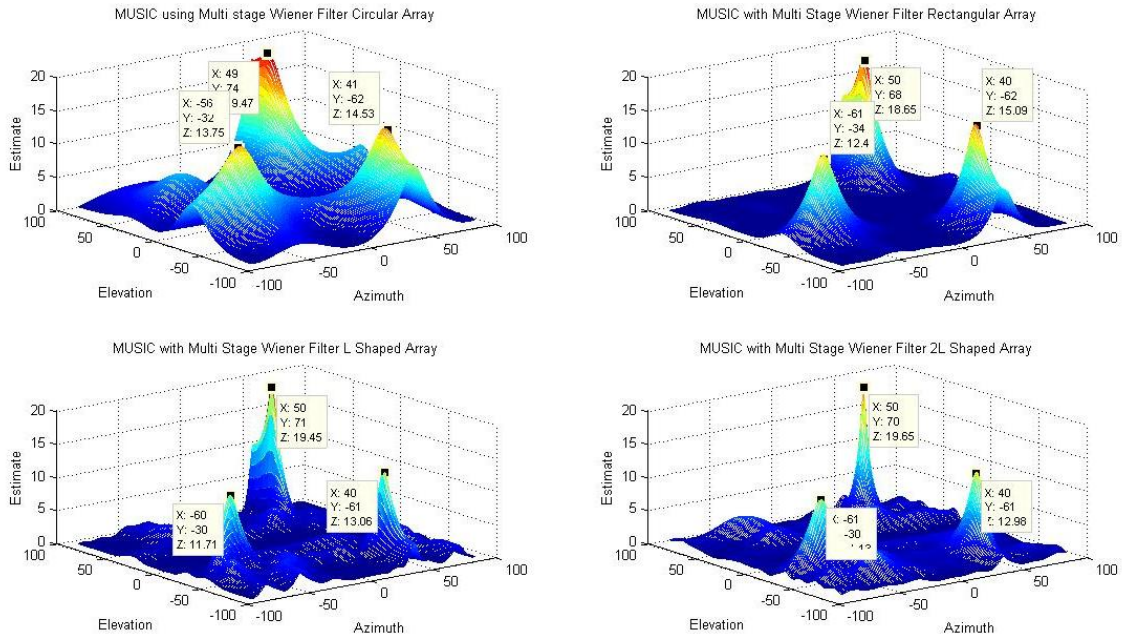


Figure 4-3 MUSIC with Multi Stage Wiener Filter for noise subspace estimation SNR = 15

Multipath fading was simulated for four path delays of [0 $5t_s$ $40t_s$ $80t_s$], where t_s is the sampling period. The sampling frequency is assumed to be 44 kHz and path gains for the different paths are [0 −3 −3 −5] dB, respectively. In the absence of multipath fading for the same array configurations and signal, slower convergence is observed.

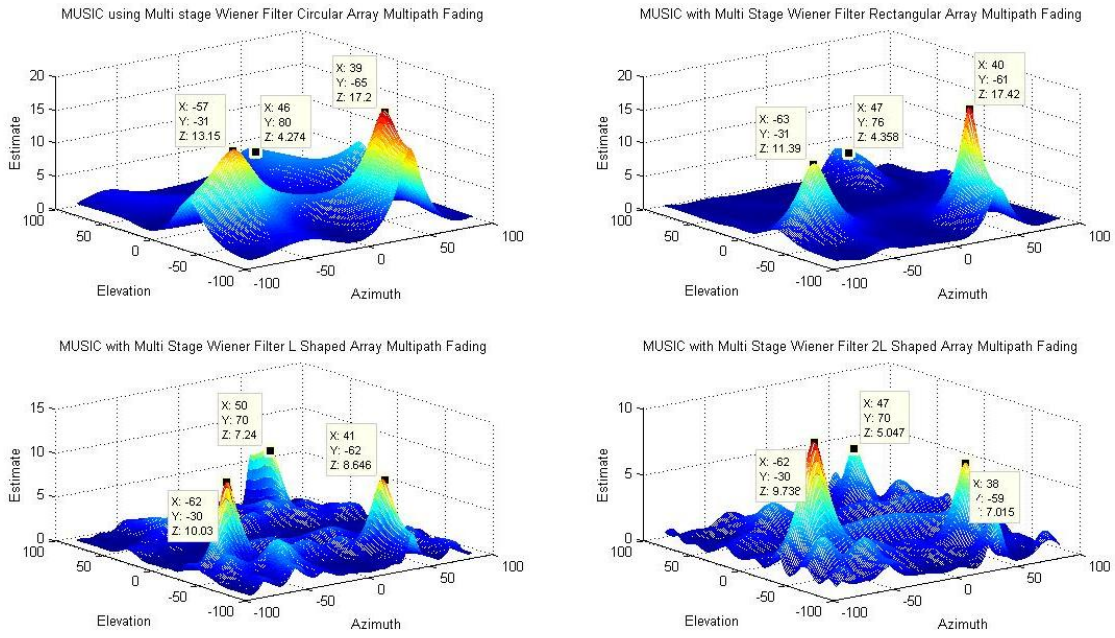


Figure 4-4 MUSIC with Multi Stage Wiener Filter SNR = 15 and multipath

Next, we observe the effect of variations in SNR on the estimation accuracy. It can be seen from Figure 4-5 through Figure 4-10 that the algorithm performance is poor at low SNR, but improves as the SNR increases. It is also likely that at very low SNR values, one or more of the signals might not be detected as shown in Figure 4-5.

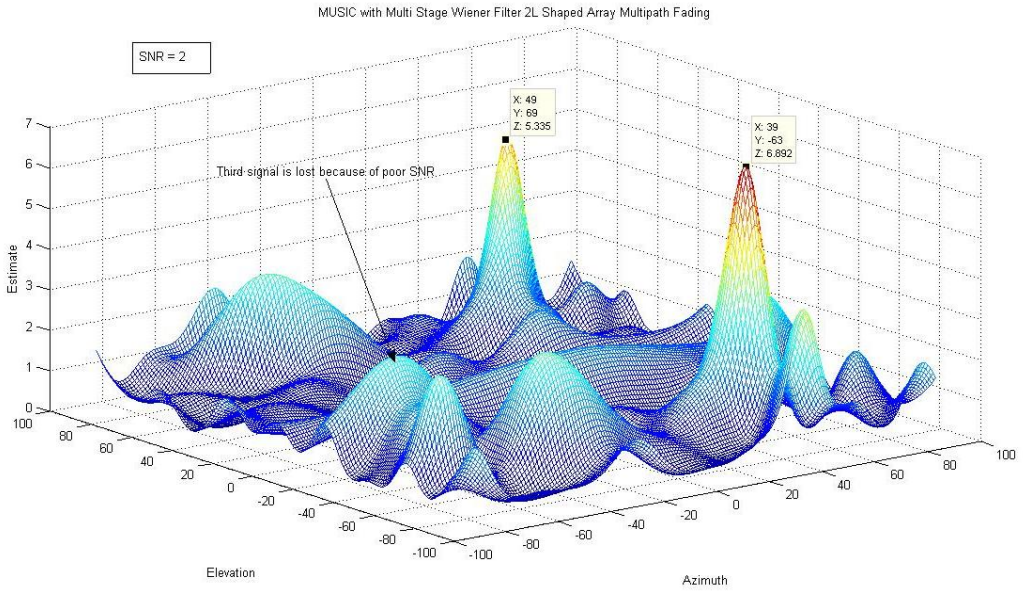


Figure 4-5 MSNWF for MUSIC in the presence of multipath fading SNR = 2

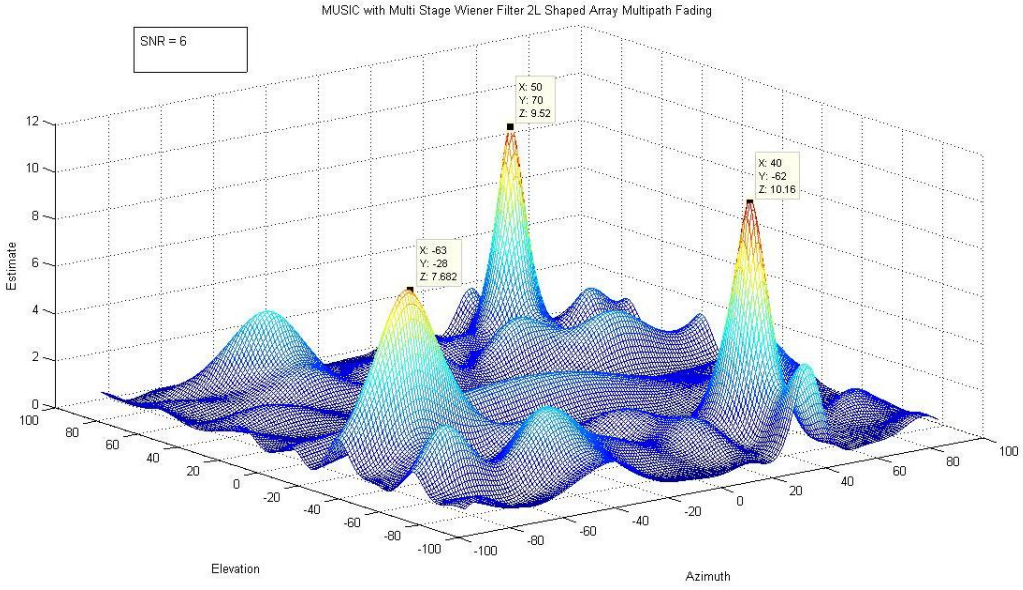


Figure 4-6 MSNWF for MUSIC in the presence of multipath fading SNR = 6

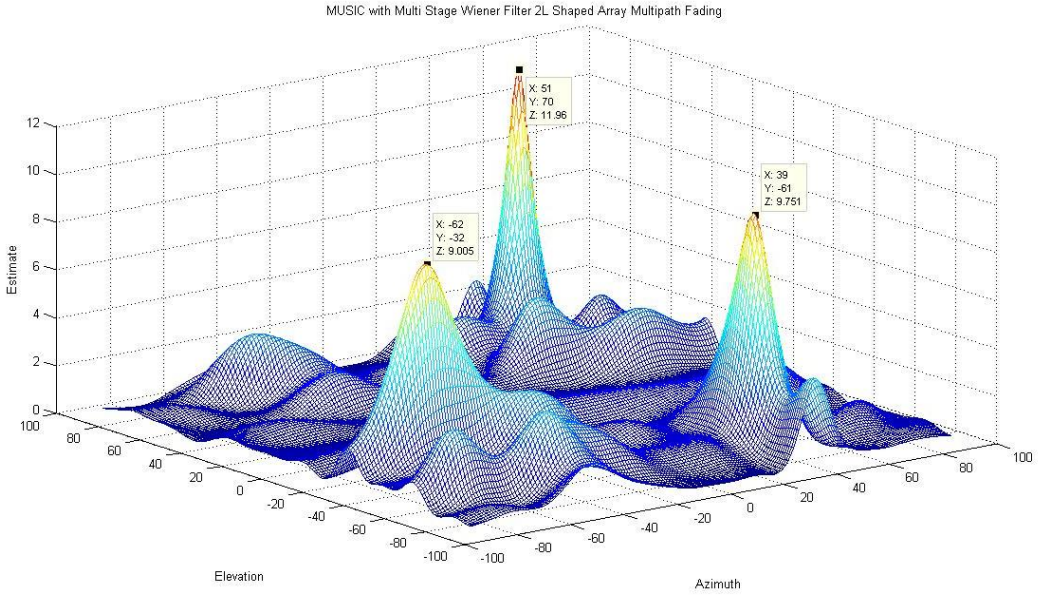


Figure 4-7 MSNWF for MUSIC in the presence of multipath fading SNR = 10

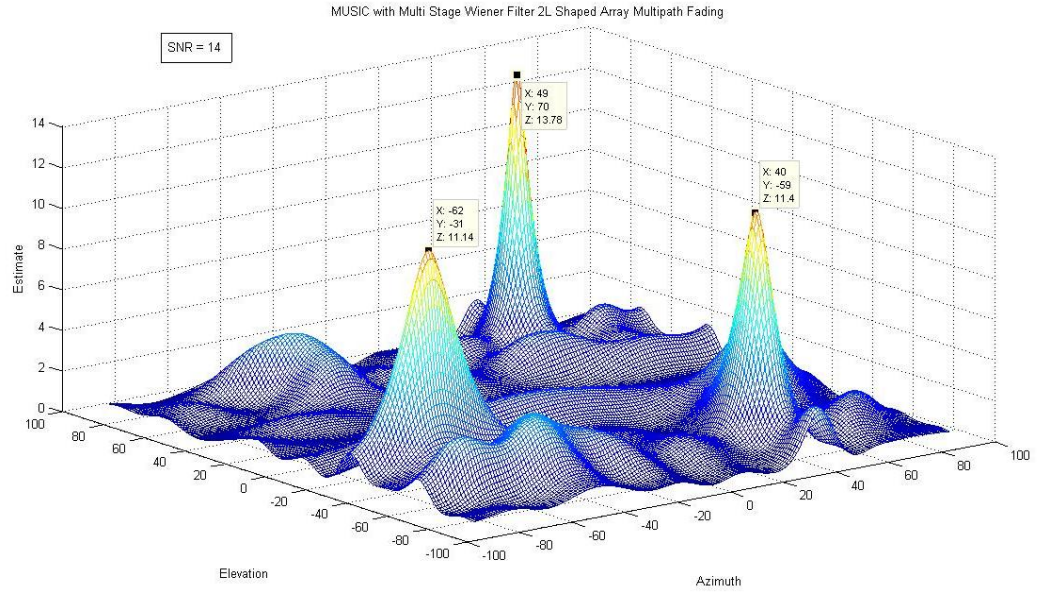


Figure 4-8 MSNWF for MUSIC in the presence of multipath fading SNR = 14

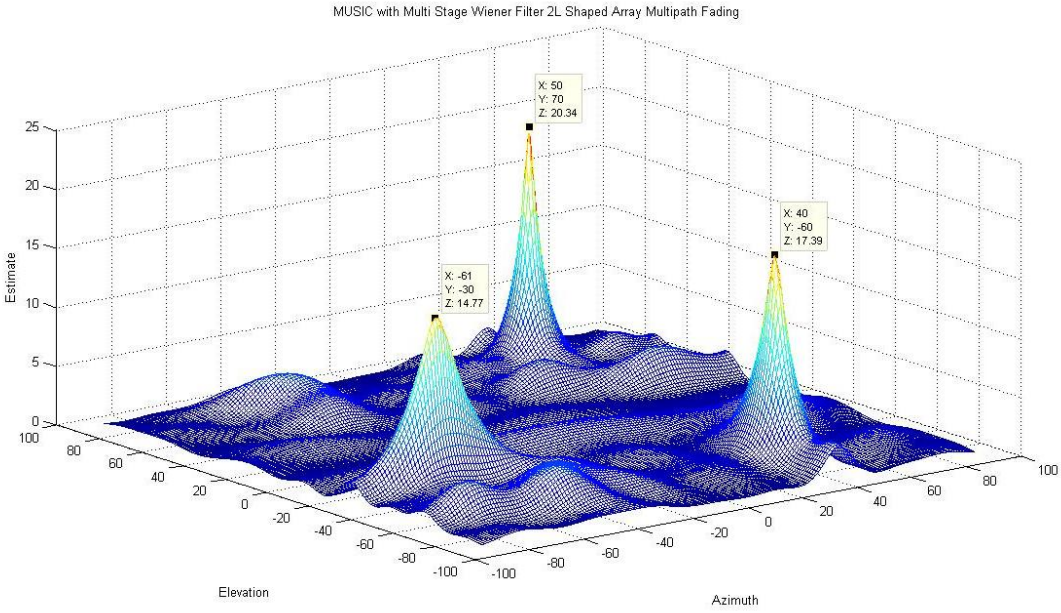


Figure 4-9 MSNWF for MUSIC in the presence of multipath fading SNR = 18

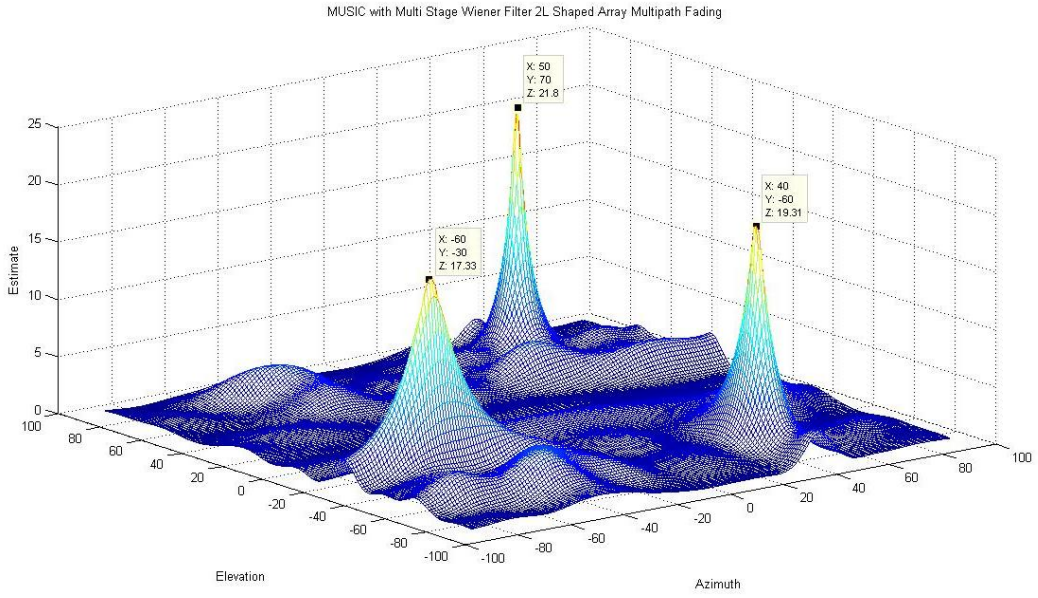


Figure 4-10 MSNWF for MUSIC in the presence of multipath fading SNR = 22

These simulations produce similar results when compared with the estimated values of angle-of-arrival using the conventional SVD. Next we compared the accuracy of Multi Stage

Wiener Filter–based method to the CRB and observed that, for 1024 sample points, the MSE of a 2L-shaped array matches closely with the lower bound predicted by CRB as shown in Figure 4-11. At higher values of SNR, the performance of the AOA method reaches the CRB limit. Thus, the performance of the extension of the 2-D MUSIC algorithm for angle-of-arrival estimation using MSWF for signal subspace estimation matches the performance of SVD based estimator, but the computational complexity is drastically reduced.

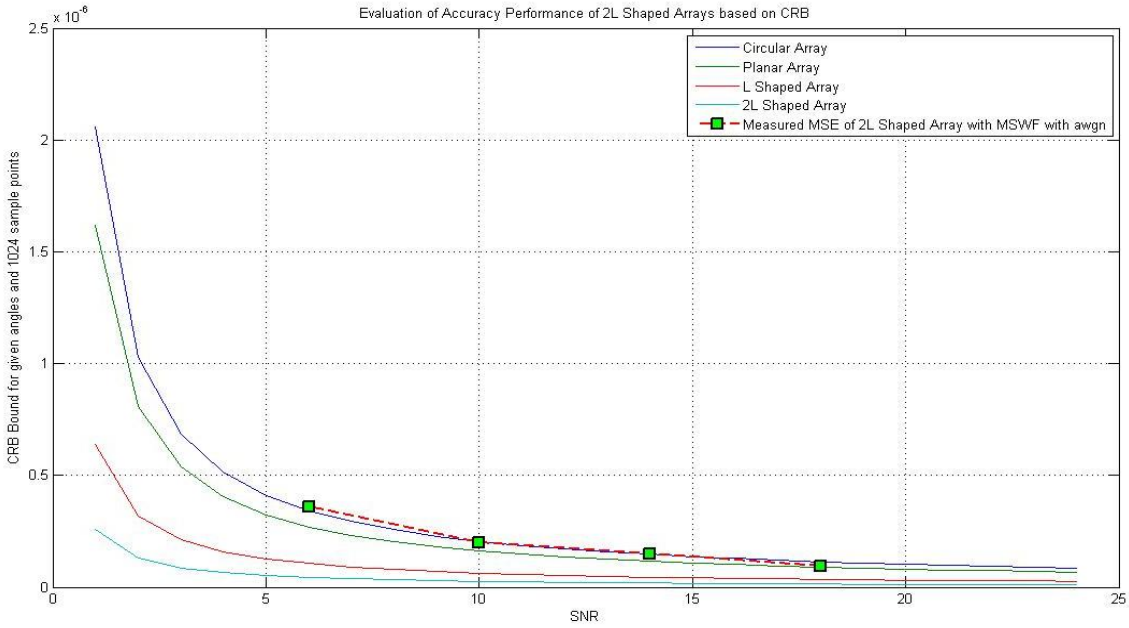


Figure 4-11 MSE and Cramer Rao Bounds comparison

Chapter 5 Hardware Implementation

Building upon the theory and MATLAB simulations of various angle-of-arrival techniques, GNU Radio and the USRP were used as the software-defined-radio platform testbed for comparing these algorithms. Use of the hardware testbed offers the advantage of evaluating the algorithms in an environment similar to what would be experienced operationally.

5.1 Testbed Configuration

The USRP board provides the RF front end and basic signal conditioning blocks and the GNU Radio software package provides all of the signal processing capabilities needed to create a radio platform. This integrated system is capable of transmitting or receiving any variety of waveforms and performing any required signal processing by interchanging daughter boards with different RF front ends. GNU Radio's structure allows for a combination of simplicity, flexibility, and power. The heart of the platform is a set of predefined C++ signal processing blocks connected through the Python scripting language. Rather than custom defining basic operations such as filters and mixers every time they are required, the included C++ blocks can be called within a Python script, pass the necessary arguments to customize their operation, and run in real-time in conjunction with the USRP. If a signal processing block is required that is not native to GNU Radio, a C++ block can be custom written to accomplish any task, and then called just like any other block using Python.

The tests for each algorithm were performed according to the data flow in Figure 5-1 [*Gardner,* 2010]. The USRP was used as a signal sink to record three unique streams of data. These data were then fed into GNU Radio and passed through a DQPSK modulator flowgraph, which performed the required symbol mapping, modulation, and up-conversion to RF, at which point white Gaussian noise was added. The data were sinked to a file rather than to a USRP

Board. The three RF signals were then processed to emulate the effects of being received on a 16-element antenna array. In the 1-D cases, the array was a ULA with $\lambda/2$ spacing, while in the 2-D cases a circular, planar rectangular, L-shaped, and 2L-shaped arrays were used. The "received signals" were then passed through GNU Radio processing blocks to handle the DQPSK signal reception, down-conversion, demodulation, and symbol extraction. Prior to data extraction in this receiver chain, the down-converted signal was processed through AOA algorithms. The results of each algorithm are then plotted using Python's Matplotlib library.

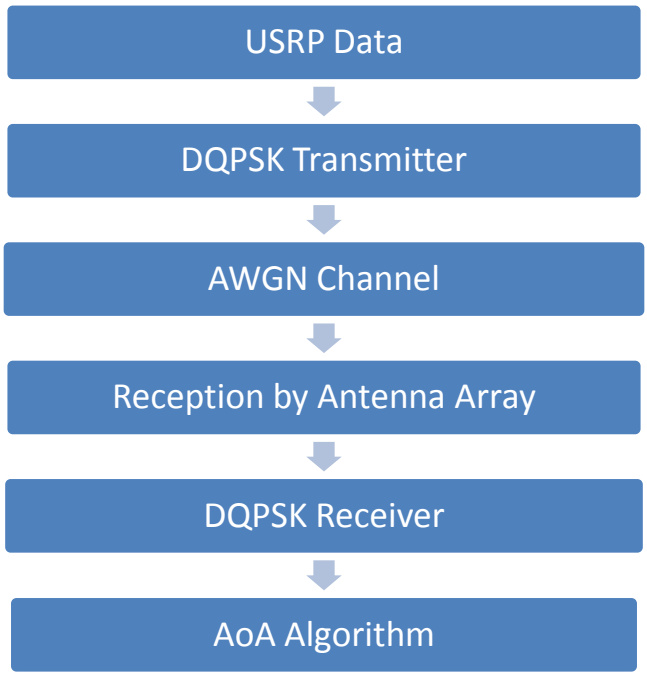


Figure 5-1: USRP and GNU Radio testbed configuration

5.2 Test Results

For both the 1-D ESPRIT and MUSIC algorithms, 12 element linear antenna array, i.e., $M = 12$ with $P = 3$ incident signals were used. Both of these algorithms rely on the separation of the signal subspace from the noise subspace in the observed signals. One simple way to accomplish

this in a MATLAB simulation is to use SVD, but, as described earlier, this is computationally complex. Instead, the MSWF approach was used on the USRP with GNU Radio since it is just as effective in separating the signal and noise subspaces with reduced computational effort.

Figures 5-2 and 5-3 show the 1-D ESPRIT and MUSIC results. Both methods converged on the three angles of interest: 37 °, 109 °, and 144 °. Figures 5-4 through 5-9 show the implementation of 2-D angle-of-arrival algorithms in GNU Radio platform using SVD method as well as the Multi Stage Wiener Filtering Method for separation of subspaces applied to various geometries.

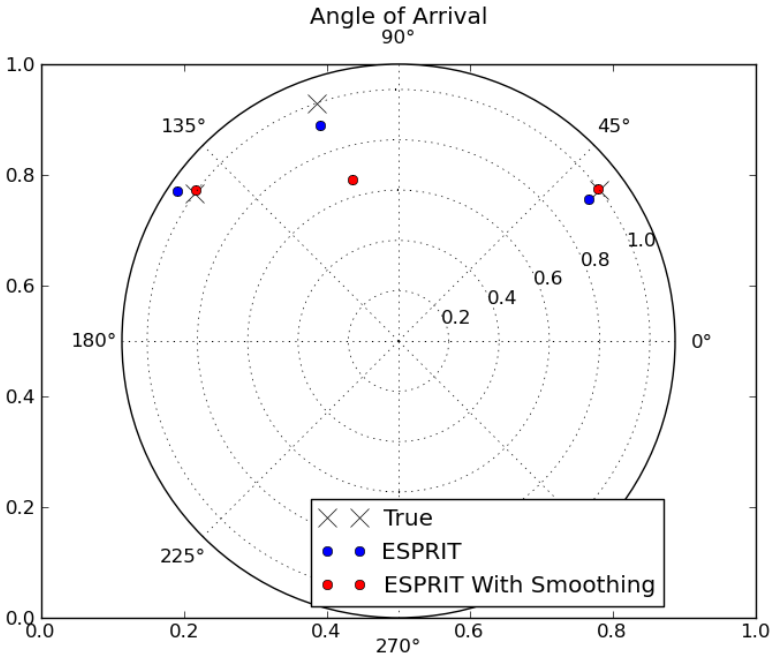


Figure 5-2: ESPRIT using MSWF with and without spatial smoothing

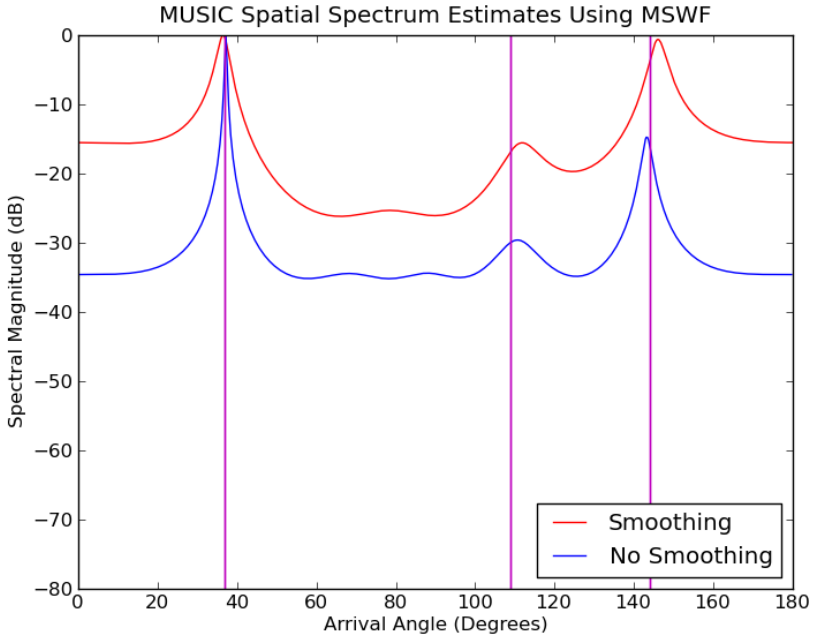


Figure 5-3: 1-D MUSIC using MSWF

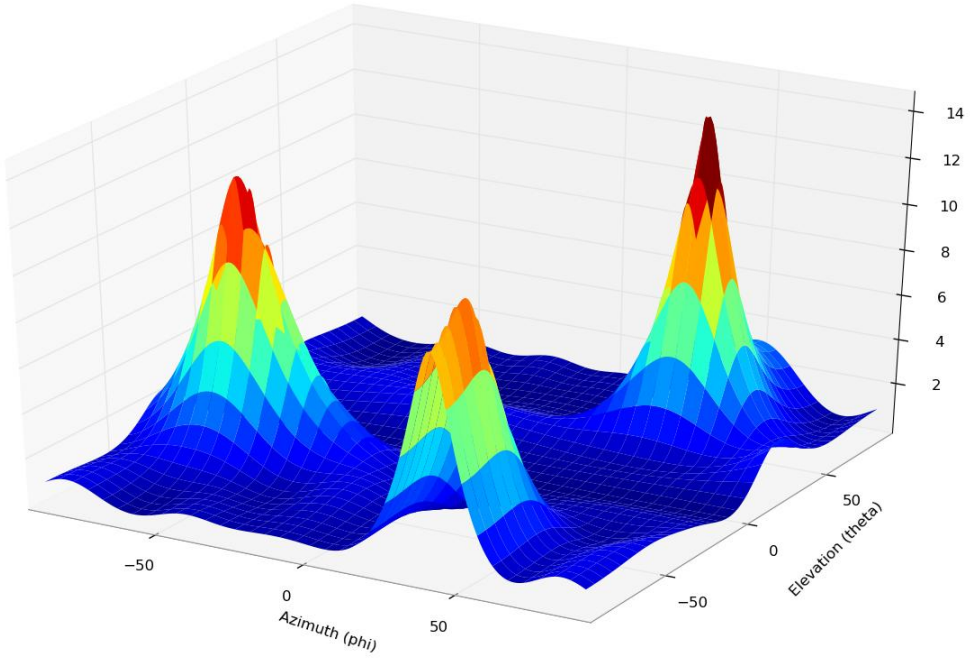


Figure 5-4 Circular array Multi Stage Wiener Filter

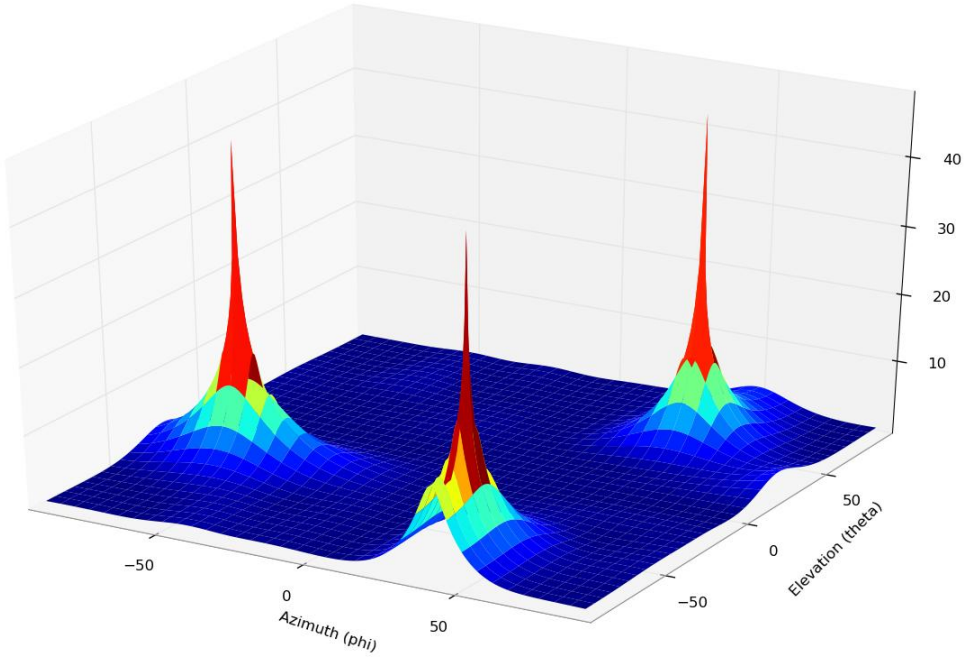


Figure 5-5 Circular array Singular Value Decomposition

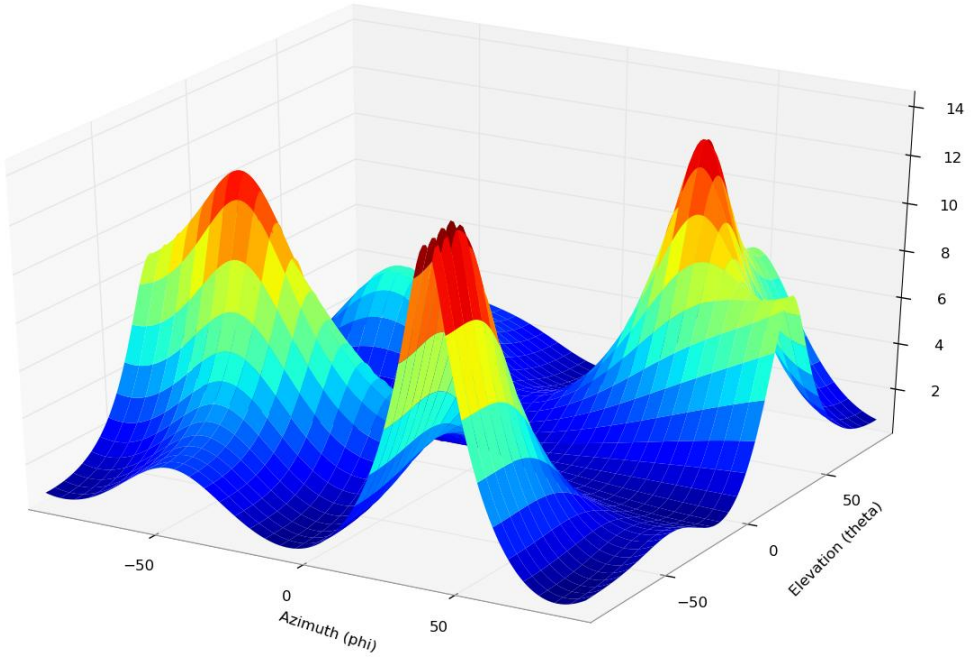


Figure 5-6 Rectangular array Multi Stage Wiener Filter

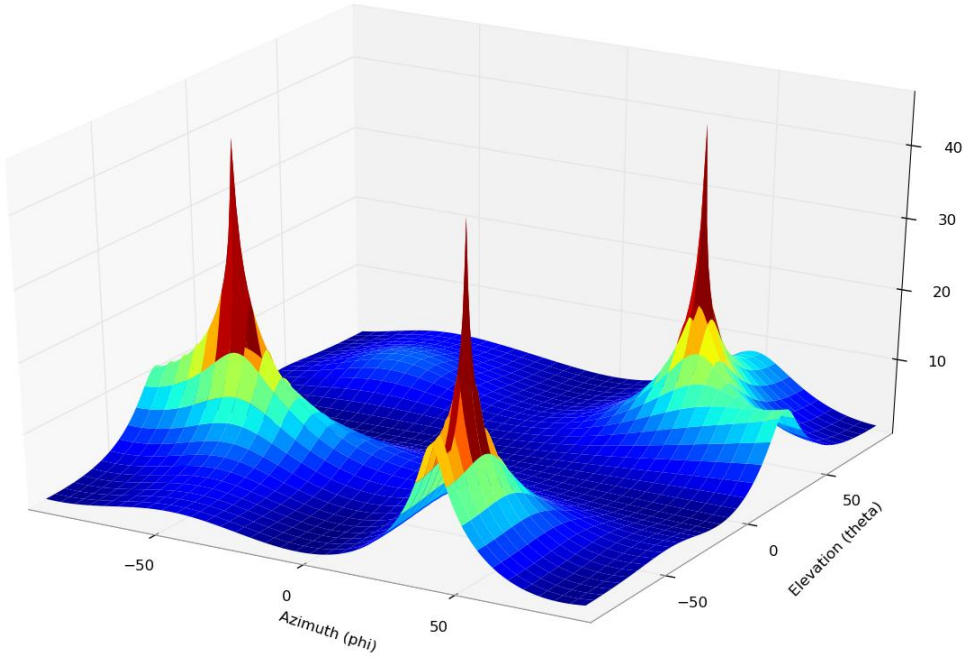


Figure 5-7 Rectangular array Singular Value Decomposition

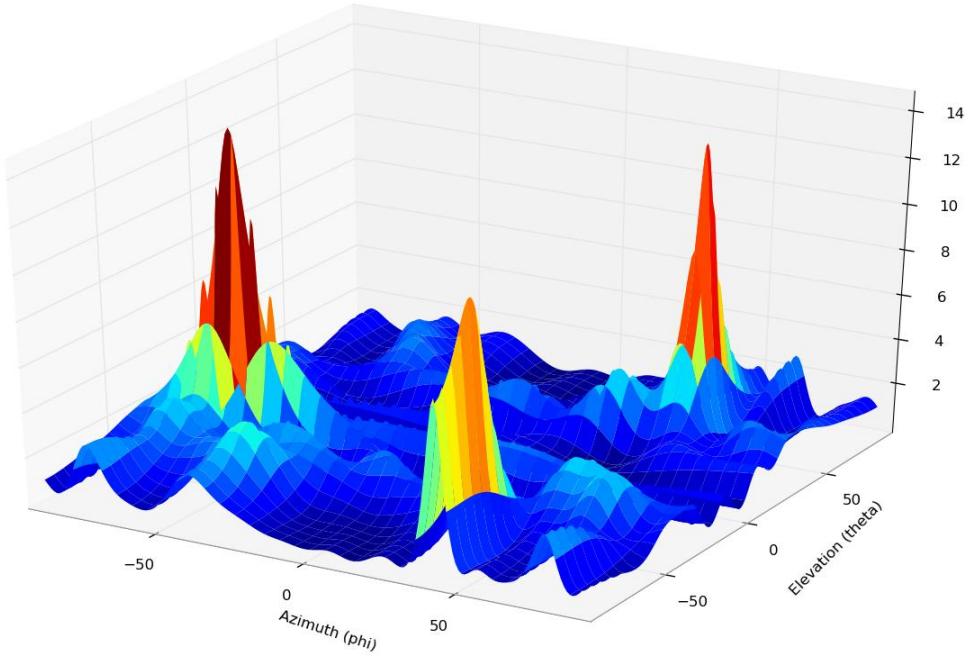


Figure 5-8 L-shaped array MSWF

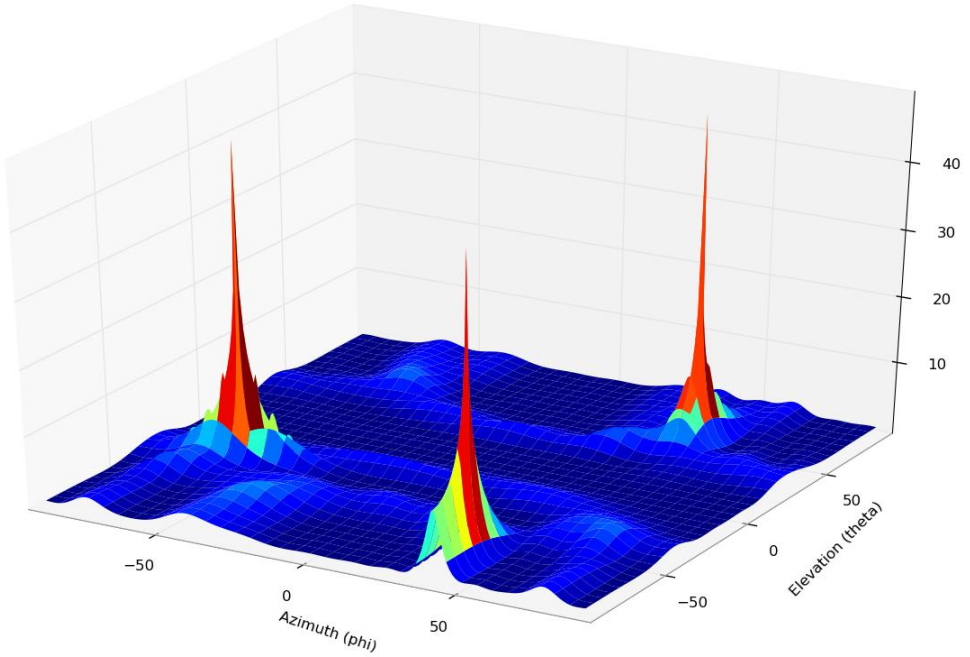


Figure 5-9 L-shaped array Singular Value Decomposition

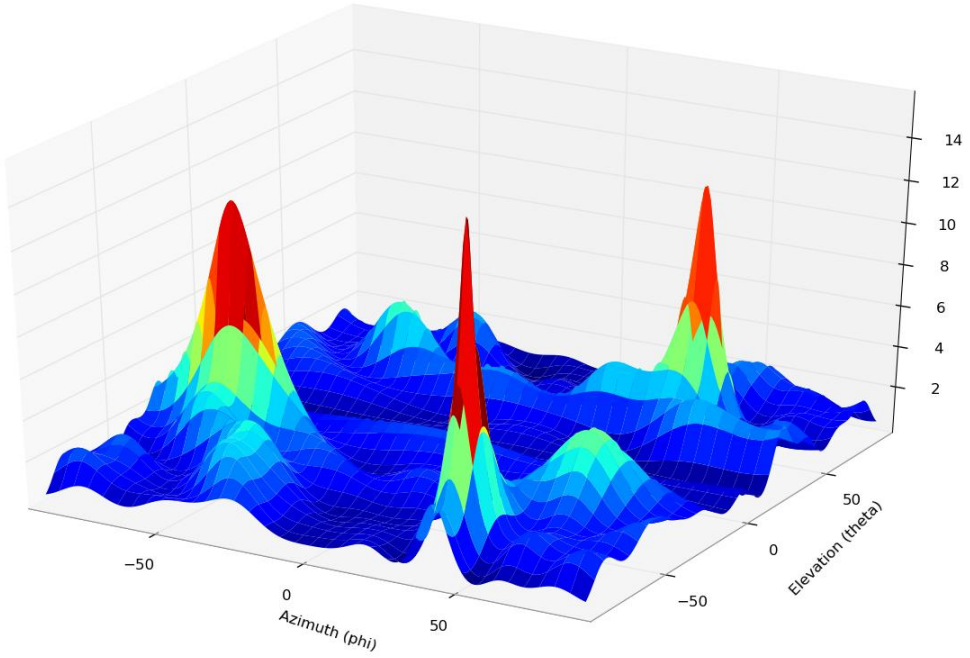


Figure 5-10 2L-shaped array Multi Stage Wiener Filter

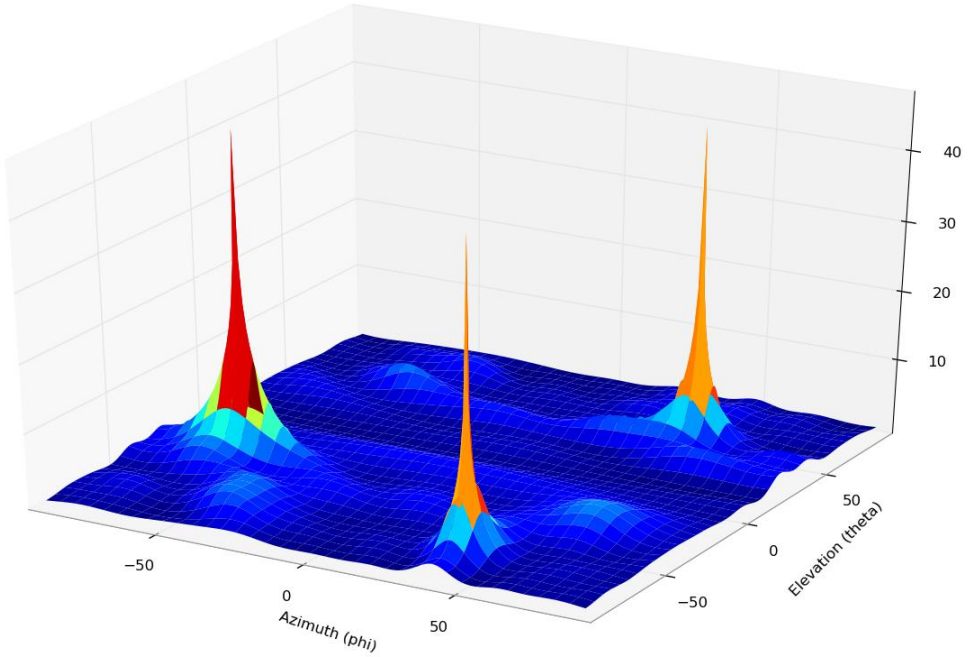


Figure 5-11 2L Shaped Array Singular Value Decomposition

The 2-D MUSIC test used an $M = 16$ antenna planar arrays with $P = 3$ incident signals at angles of $(-60°, 40°)$, $(-30°, -60°)$, and $(70°, 50°)$. It can be observed that 2L-shaped array clearly outperforms other types of arrays not just in terms of accuracy but also in terms of resolution.

Chapter 6 Conclusions and Future Work

AOA measurements can be advantageous in a diverse set of scenarios including, but not limited to, disaster response, military applications, and smart antenna array at base stations. To preserve stealth and locate the non cooperative transmitters, Angle-of-Arrival systems can be operationally deployed and Software-Defined Radio provides us with the flexible platform for eliminating the difficulties in implementation of such systems. We proposed a low complexity algorithm for AOA measurements, described the architectural implementation of such algorithm, and have shown the usefulness of the estimator with simulations as well as hardware implementation.

In a Cognitive Network (CN)—a network composed of elements that, through learning and reasoning, dynamically adapt to varying network conditions in order to optimize end-to-end performance of system—AOA is a very important parameter. A CN can intelligently select and adapt radio spectrum, transmission power, and antenna parameters to meet network requirements. Spectrum management and power management, in particular, have broad economic and policy implications, with interest from both the military and industry. Also, as Cognitive Radio Networks (CRNs) emerge as the promising next generation wireless technology that can ease the growing spectrum scarcity and support novel wireless applications; they will become bigger targets for diverse security threats, especially at the physical layer (PHY) spectrum sensing module. The knowledge of emitter locations not only helps to identify threats but can also be used for active interference cancelation. Knowledge of emitter locations also has a wide range of applications for first responders, disaster response teams, and military applications.

In this work we proposed a 2-D AOA method for geolocation of emitters. We also discussed the effectiveness of software-defined radio in evaluating the signal processing algorithms for the geolocation applications. The theory of signal parameter estimation has been a topic of research for decades now and is very well developed but there is much work yet to be

done in evaluating the knowledge of these signal parameters for CRN. Here, we present a few specific topics for designing a stand-alone, fully functional geolocation system. Real-world experimentation will identify further design problems, flesh out implementations details, and expose system limitations. This will require the use of open radio (e.g., USRP) and network (GNU Radio).

- Search-free methods to the 2-D problem—MUSIC is a computationally expensive algorithm because it searches across all the angles for peaks. Moving forward, we recommend further exploration into the search free methods, specifically Root MUSIC, ESPRIT, Root Rare, and Synthetic MUSIC.
- "Pair matching" approaches—Investigation of MUSIC-based algorithms would require further research into pair matching techniques to make sure that all antennas in arrays are matched. Studies on effects of array mismatch will reveal further difficulties in practical implementations of AOA algorithms. There are some ESPRIT-based 2-D AOA methods that do not require calibration of antenna arrays or pair matching. These methods should be further investigated.
- Smoothing techniques for improved robustness to multipath or identification of direct/non direct paths should be explored further for reliable application in the presence heavy interference.
- Approaches combining AOA with TOA, TDOA measurements, such as joint angle-Doppler, or angle-delay measurements can be more effective for precise geolocation.
- Application of 2-D AOA to spatial channel estimation, Joint Angle-Frequency Estimation or Multiple Input Multiple Output (MIMO) communication should be studied specifically for OFDM modulation because of the inherent immunity of Orthogonal Frequency Division Multiplexiung (OFDM) signals to multipath fading.

Bibliography

Akaike, H.: "A new look at the statistical model identification," *IEEE Trans. Automat. Contr*., vol. AC-19, pp. 716–23, 1974.

Barron A., J. Rissanen J, and B. Yu, "The minimum description length principle in coding and modeling," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2743–2760, 1998.

Baysal, Ü., and M. L. Randolph, "On the geometry of isotropic arrays," *IEEE Transactions on Signal Processing*, vol. 51, no. 6, pp. 1469–1478, 2003.

BBN Technologies Group, "GNU Radio Architectural Changes," BBN Technologies, 2006. [Online]. Available: http://acert.ir.bbn.com/downloads/adroit/gnuradio-architectural-enhancements-3.pdf. [Accessed September 7, 2009].

Blossom, Eric, "Exploring GNU Radio," Free Software Foundation, 2007. [Online]. Available: http://www.gnu.org/software/gnuradio/doc/exploring-gnuradio.html#usrp. [Accessed September 7, 2009].

Blossom, Eric, "GNU Radio – The GNU Software Radio," Free Software Foundation, 2007. [Online]. Available: http://www.gnu.org/software/gnuradio. [Accessed September 7, 2009].

Byerly, K.A., and R.A. Roberts, "Output power based partial adaptive array design," *Proc of 23rd Asilomar Conference on Signals, Systems and Computers*, vol. 2, pp. 576–580, 1989.

Eckart C., and G. Young, "The approximation of one matrix by another of lower rank," *Psychometrika*, vol. 1, pp. 211–218, 1936.

Edgewall Software, "USRP FPGA," GNU Radio, Free Software Foundation, 2007. [Online]. Available: http://gnuradio.org/trac/wiki/UsrpFPGA. [Accessed September 7, 2009].

Ettus Research LLC, Homepage, [Online]. Available: http://www.ettus.com. [Accessed September 7, 2009].

Gardner, C., "Software-defined radio techniques for geolocation of first responder transceivers," B.S. Honors Thesis, The Pennsylvania State University, University Park, 2010.

Gilmore J.: "HDTV and the USRP," list.gnu.org, Free Software Foundation, 2007. [Online]. Available: http://lists.gnu.org/archive/html/discuss-gnuradio/2004-07/msg00034.html. [Accessed September 7, 2009].

Goldstein, J., I. Reed, and L. Scharf, "A multistage representation of the Wiener filter based on orthogonal projections," *IEEE Transactions on Information Theory*, vol. 44, no. 7, pp. 2943–2959, 1998.

Goldstein, J. S., and I. S. Reed., "A new method of Wiener filtering and its application to interference mitigation for communications," *Proceedings of IEEE MILCOM,* vol. 3, pp. 1087–1091, Monterey, CA, Nov. 1997a.

Goldstein J. S., and I. S. Reed., "Subspace selection for partially adaptive sensor array processing," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 2, pp. 539–543, 1997b.

Goldstein, J. S., J. R. Guerci, and I. S Reed, "An optimal generalized theory of signal representation," *Proceedings of the Acoustics, Speech, and Signal Processing*, vol. 3, pp. 1357–1360, 1999.

Harabi, F., A. Gharsallah, and S. Marcos, "Three-dimensional antennas array for the estimation of direction of arrival," *IET Microwaves, Antennas & Propagation*, vol. 3, no. 5, pp. 843–849, 2009.

Hotelting, H., "Analysis of a complex of statistical variables into principal components," *Journal of Educational Psychology*, vol. 24, pp. 417–441, 1933.

Honig, M. L., and W. Xiao, "Performance of Reduced-Rank Linear Interference Suppression," *IEEE Transactions Information Theory*, vol. 47, no. 5, pp. 1928–1946, 2001.

Huang, L., S. Wu, D. Feng, and L. Zhang, "Low complexity method for signal subspace fitting," *IEE Electronics Letters*, vol. 40, no. 14, 2004.

Huang, L., S. Wu, and L. Zhang, "Low complexity ESPRIT method for direction finding," *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, pp. iv/929–iv/932, 2005a.

Huang, L., S. Wu, and L. Zhang, "A novel MUSIC algorithm for direction-of-arrival estimation without the estimate of covariance matrix and its eigendecomposition," *IEEE Vehicular Technology Conference*, vol. 1, pp. 16–19, 2005b.

Joham, M., and M. D. Zoltowski, "Interpretation of the multi-stage nested Wiener filter in the Krylov subspace framework", *Tech. Rept. TUM-LNS-TR-00-6. Munich University of Technology*, November 2006. *Also: Technical Report TR-ECE-00-51, Purdue University*.

Kay, S. M., *Modern Spectral Estimation Theory and Applications*, Prentice-Hall, Englewood Cliff, 1997.

Ledeczi, A., J. Sallai, X. Koutsoukos, P. Volgyesi, B. Kusy, and M. Maroti, "Towards precise indoor RF localization," *Proceedings of the 5th Workshop on Embedded Networked Sensors (HotEmNets '08)*, June 2008..

Mitola, J., "Software radios survey, critical evaluation and future directions," *IEEE AES Systems Magazine*, Vol. 8, No. 4, pp. 25-36, Apr. 1993.

Mitola, J., and G. Q. Maguire, "Cognitive radio: making software radios more personal," *Personal Communications, IEEE*, vol. 6, no. 4, pp. 13–18, 1999.

Mitola, J., "An integrated agent architecture for software defined radio," Ph.D. dissertation, KTH, Sweden, May 2000.

Patton, L., "A GNU Radio based software defined radar," Master's Thesis, Wright State University, 2007.

Pauraj, A., R. Roy, and T. Kailath, "A subspace rotation approach to signal parameter estimation," *Proceedings of IEEE*, vol. 74, pp. 1044–1045, 1986.

Rahman, M. A., S. Khadka, Iswandi, M. Gahadza, A. Haniz, M. Kim, and J. C. Takada, "Design and implementation of a cognitive radio based emergency sensor network in disaster area," Tokyo Institute of Technology: Institute of Electronics, Information and Communication Engineers, 2009.

Reed, J., *Software Radio: A Modern Approach to Radio Engineering*. Prentice Hall, 2002.

Ricks, D., and J. S. Goldstein, "Efficient implementation of multi-stage adaptive Weiner filters," Antenna Applications Symposium, Allerton Park, Illinois, 20–22 Sept. 2000.

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

Scaperoth, D. A, "Configurable SDR operation for cognitive radio applications using GNU Radio and the Universal Software Radio Peripheral," M. S. Thesis, Virginia Polytechnic Institute and State University, 2007.

Schmidt, R., "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, 1986.

Shan, T. J., M. Wax, and T. Kailath, "On spatial smoothing for directions of arrival estimation of coherent signals," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. no. 33, pp. 806–811, 1985.

Stoica, P., and Moses R., *Introduction to Spectral Analysis*, Prentice-Hall, Upper Saddle River, NJ, 1997.

Stüber, G. L., *Principles of Mobile Communication*, Kluwer Academic Publishers, 2001.

Tse, D., and P. Vishwanath, *Fundamentals of Wireless Communications*, Cambridge University Press, 2005.

Tuncer, T. E., and B. Friedlander, *Classical and Modern Direction of Arrival Estimation*, Elsevier Academic Press, 2009.

Witzgall, H. E., and J. S. Goldstein, "ROCK MUSIC non-unitary spectral estimation," *Technical Report ASE*-00-05-001, SAIC, May 2000.

Witzgall, H. E., and J. S. Goldstein, "Detection performance of the reduced-rank linear predictor ROCKET," *IEEE Transactions on Signal Processing*, vol. 51, no. 7, pp. 1731–1738, 2003.

Wax, M., and T. Kailath, "Detection of signals by information theoretic criteria," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 33, no. 2, pp. 387–392, 1985.