

# **A Low Cost, High Accuracy Radio Localization System for Tracking Wildlife**

**Russell Silva**

Electrical and Computer Engineering, Cornell University  
9/17/2018

Faculty Advisors—

**Dr. Julian Kapoor, Prof. Joe Skovira**

Behavior/Electrical and Computer Engineering, Cornell University

Funding Source: Semiconductor Research Corporation with Support from Intel

## **I. Abstract**

Animal Movement Research Using Phase-based Trilateration (AMRUPT) is a technology being designed for the localization of small animals in the field of ecology. This technology is proposed to suit ornithological studies involving individual- and species-level migratory patterns and social interactions. The system utilizes Phase Interferometry for use in estimating the angle of arrival (AoA) of radio signals. This direction-finding method has demonstrated resiliency in environments with high multipath interference and radio-frequency (RF) noise pollution, although performance scales strongly with the spatial scale of the receiver network. The system design of AMRUPT is forward compatible with phase difference of arrival ranging, a high precision localization approach developed for dense multipath indoor environments.

The embedded devices and RF modules chosen keep our system low-cost while meeting hardware requirements for high spatial resolution. The radio tag being developed to transmit signals to radio base stations will be at a suitable weight (0.011g) for the wildlife tracking of small bird species such as the Little Hermit (*Phaethornis longuemareus*). Inexpensive software defined radios (SDRs) are integrated into receiver basestations to obtain the radio-wave measurements required for AoA calculations. Each basestation consists of a four RTL-SDR network and each RTL-SDR in the network is connected to an antenna, an RF switch for synchronization purposes, and a Raspberry Pi. The Raspberry Pi serves as an embedded device which collects and analyzes the characteristics of the received radio signal. Each Raspberry Pi contains an open-source radio coding platform called GNU radio installed, equipped with the software protocols necessary for obtaining accurate AoA measurements.

## **II. Introduction**

The accurate and real-time localization of small-bodied animals in the field of ecology is imperative to determine individual- and species-level migratory patterns, social interactions, and other key behaviors. Many attempts have been made to determine the positioning of animals temporally and spatially in the past, but have been either inaccurate (errors over five meters) or have required constant manual human intervention. Since direction finding requires wireless telecommunication, measurements have been thwarted by multipath interference from vegetation, electromagnetic interference, or other environmental conditions. Our objective is to develop a cost effective and automated system to track animal movements within the range of five meters while taking into account expected causes of error. Our system consists of a receiver architecture that is built specifically for phase interferometry direction finding to facilitate accurate measurements from radio tags on tracked individuals.

### III. Materials and Methods: Review of Literature

Many different techniques have been explored to achieve localization. Transmitted signals received at antenna array elements can be quantized at receivers to provide phase difference information such as in phase interferometry [4,6]. However, phase based measurements can be skewed by multipath effects in the environment by constructive and destructive interference [6]. In this literature review, we discuss methods to increase multipath resiliency in AoA methods and examine the strengths and weaknesses of widely used localization methods.

#### III. i. Received Signal Strength

One widely used method to position RF sources is Received Signal Strength (RSS). RSS can be used for localization through several different algorithms [5]; however, these metrics have been found to be unreliable. This is because RSS values fluctuate based on the complexity of the environment and are susceptible to interference from other transmitting sources [1], [15]. The complexity of the environment is a significant drawback in RSS because a signal received at an antenna will likely be a vectorial combination of time-delayed signals, causing RSS instability.

#### III. ii. Time Difference of Arrival

Another localization method is time difference of arrival (TDOA), in which the position of a transmitter is determined from the time differences between the arrivals of a signal at receiver stations in a network [1]. TDOA systems are not as susceptible to multipath effects because a line of sight (LOS) signal always arrives at receivers before reflected copies of the LOS signal. However, obtaining precise positioning from close proximity transmitters in TDOA is difficult because nanosecond synchronization among distributed ground-nodes is required to compare lightspeed propagated signals. The “Thrifty” system in [1] surmounts this drawback in TDOA systems by using direct sequence spread spectrum (DSSS) techniques in order to align a received signal with a local template code at a receiver based on a Psuedo Random Noise (PRN) code. Since PRN codes have low auto-correlation (correlation between the transmitted signal and local template is very low until peaks are perfectly aligned), their utilization in signal detection and alignment allows the arrival times of LOS signals to be determined accurately in time and aggregated on a centralized time frame for comparison among distributed ground nodes. Additionally, the use of DSSS techniques does not only aid with network synchronization but makes TDOA more robust against noise/interference by spreading the transmitted signal’s bandwidth. Modulation schemes employed for DSSS and the cross correlation capabilities of PRN codes are described in III. v.

It is important to note that the system developed in [1] was tested in a LOS environment, and was not built to be versatile in dense multipath environments. This is most likely due to the fact

that copies of a LOS signal can arrive at receivers with a delay greater than the chip length (binary 1's and 0's in the PRN sequence), adding chips in the LOS PRN code [18]. Perhaps this system can be further optimized against this problem by integrating RAKE receivers, which induces a time diversity on the transmitted signal so that independent multipath components separated by more than a chip time can be resolved [18].

### III. iii. Angle of Arrival

A phase interferometry implementation that provides a good introduction to AoA calculation is a passive direction finding system in [4] developed for airborne signals. This system involves real time operation on multiple receivers driven by a common local oscillator (block diagram shown in Figure 1). Because this system was driven by a common local oscillator, the resulting intermediate frequency at each receiver could be compared without having to perform phase-offset synchronization. This system had an AoA accuracy with less than  $\pm 2.5$  degrees of error. Although the error rate is promising, this receiver system was designed to receive line of sight airborne signals. This does not account for the adverse effects of a cluttered environment where signals propagate near the ground, which will add additional factors to AoA error such as multipath interference.

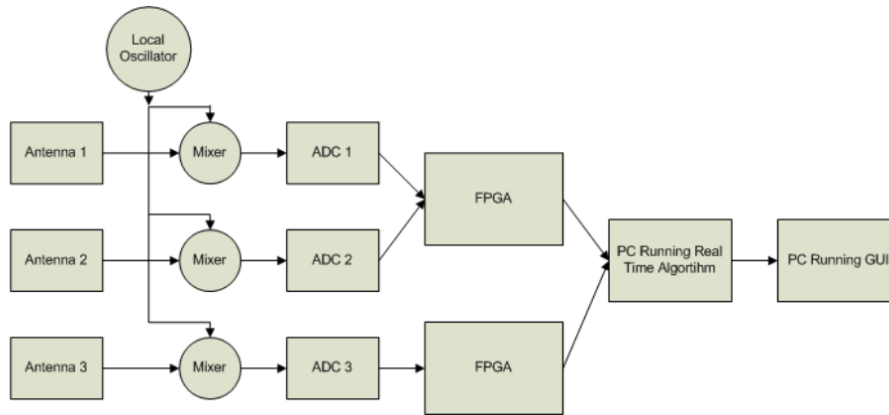


Figure 1: Block diagram for Phase Interferometry in Guerin, Jackson, and Kelly [4]

The direction of arrival (DOA) approach in [4], although refined through parameter simulations (i.e. the effect of antenna separation on AoA error), determines DOA solely based on the phase difference of antennas which could suffer from the effects of external noise (atmospheric, galactic, industrial) or system-inherent noise (antenna amplifiers, DF converts, A/D converters) [2]. The resolution of angle of arrival measurements in phase interferometry can be improved by using subspace techniques such as MUSIC and root MUSIC which are intended as a means of eliminating the effect of noise at an antenna array. In MUSIC and root MUSIC, a certain matrix notation is used to represent signals impinging on an antenna array which is entailed in Appendix

A. The MUSIC algorithm is described in Appendix B and the root MUSIC algorithm is described in Appendix C. In general, root MUSIC has a lower calibration error, sensitivity with respect to signal to noise ratio (SNR), and root mean square error (with respect to the estimated and measured DOA) than MUSIC [19]. Also, root MUSIC alleviates algorithm complexity. A drawback of root MUSIC is that it requires the use of uniform linear arrays [19]. Figure 2 displays the summarized steps of the MUSIC and root MUSIC algorithms.

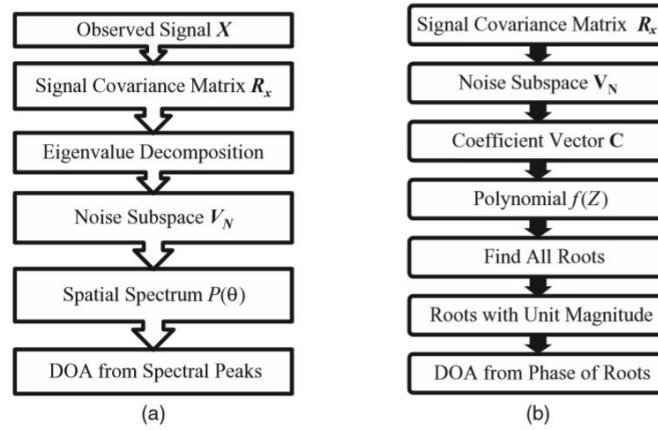


Figure 2: Block diagram of the (a) MUSIC and (b) root MUSIC algorithms [19]

Although accurate, MUSIC is a computationally expensive algorithm [14] and suffers from the effects of multipath interference [15]. [14] offers an alternative subspace technique, in which a modified switched beam system (SBS) is used. A SBS uses a fixed number of beams to scan the azimuthal plane. Even though SBSs such as Bartlett beamforming are computational inexpensive, they need low Signal to Noise ratios to work well, and they usually suffer low resolution as scanning the entire azimuthal plane will likely lead to multiple signal sources that are parallel to a steering vector. [14] proposes a novel cross-correlation based SBS (XSBS) technique which utilizes an omnidirectional signal beacon as a reference signal to compare signals within the angular range of interest. In this method, antenna weights (used in beamforming to scan the azimuthal plane) are all set to zero to receive the omnidirectional signal as stand alone antennas. Then, antennas are weighted according to the beamforming direction, and signals received at the weighted antenna array are cross correlated with the omnidirectional reference signal from the previous step. This method outperforms the MUSIC algorithm with more computational efficiency, but it is unclear how this method performs in a multipath environment.

In a multipath environment, the performance of MUSIC and root MUSIC degrades because refracted signals received at antennas will be correlated with the source signals themselves. This is a problem because separate signals in the signal subspace will be correlated, which goes against MUSIC's requirement for independent signal sources [8, 13]. A technique known as Spatial Smoothing can be used to remove the correlation of received signals by dividing the

antenna array into subarrays [8, 13]. If the number of antennas in the entire array is large enough, a noise subspace of uncorrelated vectors can be formed by averaging the correlation matrices of all subarrays and applying the averaged correlation matrix to the first step of MUSIC or root MUSIC. Nevertheless, spatial smoothing adds computational complexity and resolution scales with the number of antennas used. [13] proposes a new method, Covariance Differencing and Iterative Spatial Smoothing, to reduce the computation and antenna amount required by spatial smoothing. This technique subtracts an uncorrelated covariance matrix, corresponding to uncorrelated signals from resolved DoAs, from a covariance matrix computed with raw signals retrieved at an antenna array. This process is performed iteratively until no peaks appear in the MUSIC pseudospectrum, indicating an absence of correlated signals.

### III. iv. Phase Difference of Arrival

The last localization method that will be investigated is phase difference of arrival (PDOA). Much like TDOA, PDOA systems compute a differential distance metric between a pair of receivers instead of an absolute propagation distances. The fundamental equation for obtaining a differential distance measurement in PDOA is  $d = \frac{\varphi\lambda}{4\pi} + \frac{n\lambda}{2}$  where  $\varphi$  is the phase difference between two antennas,  $\lambda$  is the wavelength of the signal, and  $n$  is an ambiguous phase integer. Note that there is an extra factor of two integrated in the denominator of both terms. This is done to reflect that the RF wave travels double the tag to reader distance in a backscattering propagation.

To our current knowledge, only systems with radio frequency identification device (RFID) tags have implemented PDOA and most of these systems operate in indoor environments. RFID tags are highly desirable in PDOA systems, because the initial phase of the transmitted signal is known through conventional linear backscatter (a signal is transmitted to and reflected from a RFID tag) [15]. Moreover, the range over possible integers  $n$  in which a phase-integer disambiguation algorithm such as HMFCW [6, 12] would have to search for could be two orders of magnitude greater for outdoor environments with receivers spaced more than 100 meters apart. A solution to this drawback in outdoor PDOA systems is discussed in the Design Objectives section of this document.

To improve 3D localization in PDOA, Ma, Hui and Kan [6] proposes broadband harmonic backscatter and heuristic multi-frequency continuous wave (HMFCW) ranging. Broadband harmonic backscatter eliminates self-jamming in RFID readers by modulating the uplink response (tag to reader) on a second harmonic which is passively generated by non-linear devices inside the RFID tag. HMFCW ranging is used to resolve ambiguous phase cycle integers with maximum tolerance of multipath induced phase errors. [6] extends HMFCW ranging by using a genetic algorithm which selects an optimal frequency combination that maximizes phase error

tolerance. The larger the bandwidth of the multi-frequency transmission, the more robust the system will be to multi-path induced phase error.

Once the optimal frequencies are transmitted, and a phase integer  $n$  is determined reliably for each frequency, differential distances from a RFID tag to an antenna pair are computed and averaged for  $n_i$  and  $\lambda_i$  (in the fundamental PDOA equation) from indices 1 to  $K$  where  $K$  is the number of frequencies used for ranging. Ultimately, 3D positions are localized using optimized hyperboloid functions parameterized by differential distance measurements as shown in Figure 3.

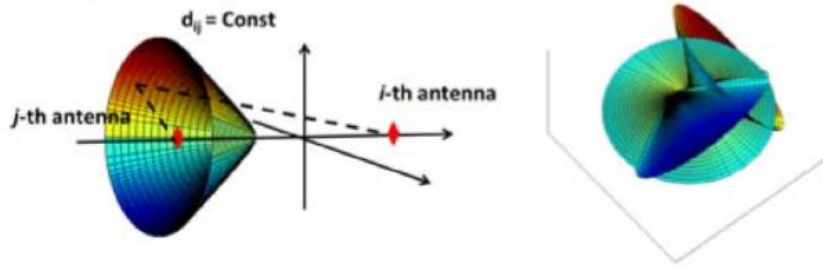


Figure 3: 3D localization by hyperboloid intersection [6]

This novel approach towards mitigating the effects of multipath interference, defined as the occurrence when radio waves reach a receiver via two or more paths, achieved localization with less than one-centimeter median error for 2D localization measurements.

A PDOA system with low-directivity receiving antennas in [12] uses AoA beamforming at antenna arrays to reject ranging measurements in transmission scenarios where multi-paths are stronger than LOS signals. The concept of coherence bandwidth is used, which classifies a range of frequencies which have a strong likelihood for amplitude and phase correlation. The coherence bandwidth is inversely proportional to the root-mean-square (RMS) multi-path delay spread of the transmission environment. For LOS signals stronger than multi-paths, two frequencies will be heavily correlated, independent of frequency separation. When Multi-paths are stronger than LOS signals, two frequencies will yield a large AoA gap when frequencies outside the coherence bandwidth are used. A graphic representation of this phenomenon is displayed in Figure 4.

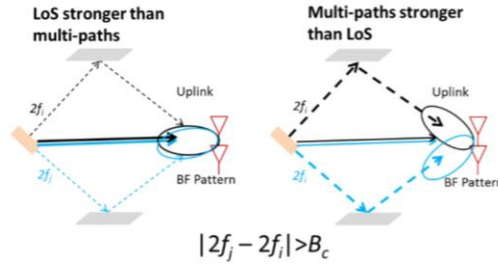


Figure 4: AoA Gap based on Strong LOS vs. Strong Multi-paths [12]

In [12], the maximum AoA gap is computed for frequencies within the HMFCW sequence. Every frequency pair spread further than the coherence bandwidth is considered for determining the maximum AoA gap. If the maximum AoA gap is above a certain threshold, the measurement is deemed unreliable and PDOA ranging is not performed. If this threshold is set large, measurements are more prone to be inaccurate. If this threshold is set small, reliable measurements may be rejected. [12] considers 30-40 degrees to be an optimal threshold.

### III. v. Synchronization of Multi-channel Receivers and Distributed Ground Nodes

It is important to note that measurements used in calculating AoAs are obtained from antennas connected to the same receiver, instead of comparing parameters (e.g. TOA, phase) among disconnected receivers (TDOA/PDOA). In AoA systems with ADC's sharing a common clock signal; sampling does not start at the same time, resulting in bulk delays [9]. These synchronization errors can be corrected by cross correlating samples based on measured channel delays. These delays would be determined by monitoring signal dispersion among channels when a common signal (e.g. white noise) is delivered simultaneously to each ADC in a multichannel receiver.

In TDOA and PDOA, measurements must be compared among distributed ground nodes with no hard connection. Therefore, Sample-of-Arrival (SOA) values must be used instead of raw TOA or POA values for differential measurements to have an accurate common time base among receivers. A SOA value represents the index of the sample at which a local template code within a receiver lines up with the PRN code of the incoming signal [1]. In this way, the measurements of an incoming signal can be referenced to precise time values and compared with a common time base at a central basestation. The DSSS techniques examined mainly follow a two-step procedure: use a modulation scheme to inject a PRN code into a carrier wave before transmission, and then detect and cross correlate a PRN sequence at a receiver via a matched filter.

Before a radio tag transmits a DSSS signal, it injects a PRN sequence into a carrier signal (also known as a baseband signal) using a modulation scheme. One of these modulation schemes is On-off keying, in which the carrier wave is modulated on and off by taking the linear combination of a PRN sequence, consisting of pulses/square waves, and the carrier wave. Each pulse/square wave in the sequence is called a "chip". The smaller the chip duration, the larger the bandwidth of the modulated signal by the Heisenberg-Gabor uncertainty principle [17]. The modulation of a carrier wave with a PRN sequence increases the bandwidth of the transmission because the square wave itself is a collection of many different sine waves at varying frequencies. In a PRN code with an equal amount of one and zero code bits, half of the signal's



energy will be contained in the narrowband component and the other half in the unmodulated carrier component [1]. The transmitter used in [1] modulated the carrier wave for OOK by rapidly toggling the supply voltage of the power amplifier from a MCU digital output pin.

When an incoming PRN modulated signal reaches a receiver, a matched filter continuously runs a cross correlation operation, shifting the samples of the signal against a template at a specified sampling rate (i.e. four samples per chip) [19]. After each shift operation, the samples of the signal and template are multiplied and then accumulated to a total amount which indicates the level of signal/template alignment. A detection of a PRN signal is confirmed when this total amount exceeds a specified threshold.

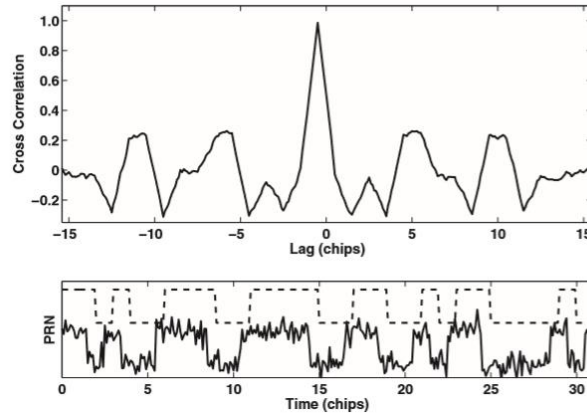


Figure 5: Time domain cross correlation of a PRN signal. [19]

The matched filter described above is extremely computationally expensive, because data streams of an incoming signal would have to be continuously cross correlated in the time domain to have prompt PRN detection and accurate SOA values. Since PRN codes can be represented as a complex spectrum of frequencies, the discrete fourier transform (DFT), at specified time intervals of the data stream (window size), can be compared against PRN time-shift sequences (i.e. 1, 1, 1, 0) for a time complexity of  $O(N \log N)$  instead of  $O(N^2)$  [19]. The window size of the DFT is selected to be the minimum length necessary to contain the data from one complete tag transmission in order to further reduce the computational load (a shorter time domain window size in the DFT will also yield a finer frequency domain resolution).

### III. vi. Low-cost Systems and Conclusion

In order to improve the cost effectiveness of direction finding, [1] has used low-cost RTL SDRs to extract in-phase and quadrature (I/Q) samples from incoming radio signals for TDOA and AoA calculations respectively. Direction Finding Implementations using RTL SDRs are promising alternatives to more expensive options by achieving up to 3.5m accuracy (under low multipath conditions) in TDOA [1] and by having an extensive hobbyist base with multiple Github repositories such as this one [10], demoed here. The advantages of having this repository

available to us is that it will provide us with a point of reference when implementing our code and hardware. This specific repository was a precursor to the RTL SDR system developed by Sam Whiting in [11].

The “Thrifty” system in [1] is promising, as ultra-wide bandwidth transmissions would be less susceptible to noise interference and signal refractions from a cluttered environment. However, this TDOA approach is limited by the effects of multipath interference, unless more complicated receiver hardware or software is used (i.e. RAKE receivers). The subspace smoothing methods for root MUSIC and MUSIC present powerful solutions to mitigate multipath interference. These AoA algorithms may be more suitable for close proximity (~100 meters) receivers outdoors than a TDOA system, which would be much more versatile with kilometer or greater receiver separations. A hybrid AoA/PDOA system with a combined coherent receiver and distributed ground node network could provide exceptional positioning for multi-frequency ranging within triangulated areas.

#### **IV. Materials and Methods: Design Requirements**

We have the following objectives in the design:

1. The receiver system is low-power and can track up to 50 lightweight and low-power radio tags
2. System architecture is resilient in cluttered environment (unsusceptible to multipath interference, electromagnetic interference, and other environmental conditions)
3. System is able to achieve two dimensional high spatial accuracy (error for triangulation results is limited within 5 meters) with a 100-300m distance between receivers
4. Forward compatibility: Must be compatible with and adaptable to a multi-frequency PDOA approach for future versions.
5. System is cost-efficient (almost all components are commercially off-the-shelf)

The first objective is to successfully track the locations of 50 individuals in the testing environment. We need to design the tags as lightweight as possible since the individuals are small in size and heavy tags may affect the individuals’ biological activities. To allow for the least human intervention possible during the tracking process, both the receivers and tags need to operate with minimal power consumption to increase automatic tracking period. In addition, both the transceivers (ground nodes) and tags (mobile nodes) follow a communication protocol in which the mobile nodes will go to sleep when they are not communicating with the ground nodes to reduce power consumption.

The communication protocol is an intended route for development, but has not yet been designed. It specifies that mobile nodes wake up every 5 minutes to prepare for data transmission to the ground nodes. The mobile node will receive a 5-second countdown signal once it wakes

up. As soon as the mobile node is verified to be within the receiver's range and has good link, it will be synchronized to global time before it is given a scheduled transmission time by the receiver or sent back to sleep again. If the mobile node is not within range of any receiver, it will go to sleep and wake up every 5 minutes to check whether it's within range again. The complexity of the ground-node to mobile-node communication protocol will be governed by how accurate our receivers are when taking angles of arrival measurements. If angles of arrival from at least three base stations intersect to a triangulation area of no more than 5 meter error (discussed further) over the specified tracking area, then tags will not have to be linked to different receivers depending on location and the same set of receivers can be used for all tags. The communication protocol will also be used for a multi-frequency system, which is a possibility in the future of this project.

Furthermore, the system must be able to obtain accurate results in a cluttered environment. We agreed that a real environment would have substantial multiple interference as there will be trees and rocks that can reflect a wireless signal. Multipath interference could result in multiple copies of a LOS signal with different DOAs. The effects of multipath interference and proposed solutions to the multipath problem are more amply discussed in the literature review section.

We agreed to set the tracking accuracy of our system to 5 meters because this is a minimum requirement to monitor the social interactions and movements of small mammal species. We intend to use a triangulation algorithm using high resolution AoAs to acquire this accuracy. Figure 6 illustrates error minimization with relation to AoA calculations. In addition, we plan to implement a multi-frequency PDOA system based on [6] in addition to AoA to increase tracking accuracy to sub 1-meter resolution.

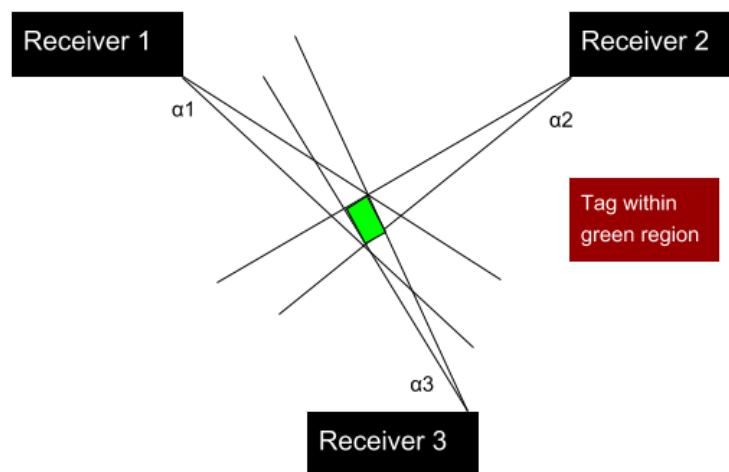


Figure 6: Error in triangulated area

To help visualize our 5-meter tracking accuracy objective, a line of more than five meters cannot be drawn within the triangulated area of error (green region in Figure 6) at any given time while tracking an individual radio tag. This area of error will be determined by  $\alpha_1$ ,  $\alpha_2$ , and  $\alpha_3$  (Figure 6) which resemble the angle of arrival error from receiver 1, 2, and 3 respectively.  $\alpha_1$ ,  $\alpha_2$ , and  $\alpha_3$  will be determined by phase difference errors from a transmitting RF signal to multiple antennas.

In order for the ground node network to track multiple tags continuously, we must have a system in mind to distinguish tags. Code-division multiple access (CDMA) is a very popular method for doing this due to the coding infrastructure. We plan to use a less complex time-division multiple access (TDMA) approach instead.

To address forward compatibility to the PDOA approach entailed in [6], the system must obtain reliable phase values at ground nodes in a distributed network. This will be accomplished by utilizing a radio tag that can modulate OOK signals, so that PRN codes can be used to obtain accurate SOA measurements at receivers. Once this is accomplished, the state of one RTL SDR within each coherent receiver will change back and forth from an antenna element used in MUSIC to a ground node that demodulates and collects phase SOA measurements. Receivers under this protocol will begin by triangulating an area for the radio tag, and then will proceed to hyperbolically determining a radio tag position using PDOA. This is done to reduce the parameter space over which the phase-integer disambiguation algorithm would need to search (e.g. AoA triangulation might produce a 10-meter area of high confidence for the location of a transmission, and the phase-based DOA ranging algorithm could then search within this area to further refine position estimates).

## V. Materials and Methods: Design Implementation

The entirety of the direction finding system consists of radio transmitters and receivers. This section will focus primarily on receiver design as the lightweight radio tags have already been developed. Please note that bi-weekly update reports written over the summer may compliment and add understanding to some aspects of the design implementation briefly discussed in this document. These reports can be found online [here](#).

### V. i. Receiver Architecture

We first discuss the architecture of each receiver basestation which consists of RTL SDRs to simplify wireless communication and improve the cost effectiveness of this project. The receiver architecture is outlined in Figure 7. Note that this includes 4 antennas connected to four individual RTL SDRs, so that a linear array of substantial size can be used in AoA subspace methods.

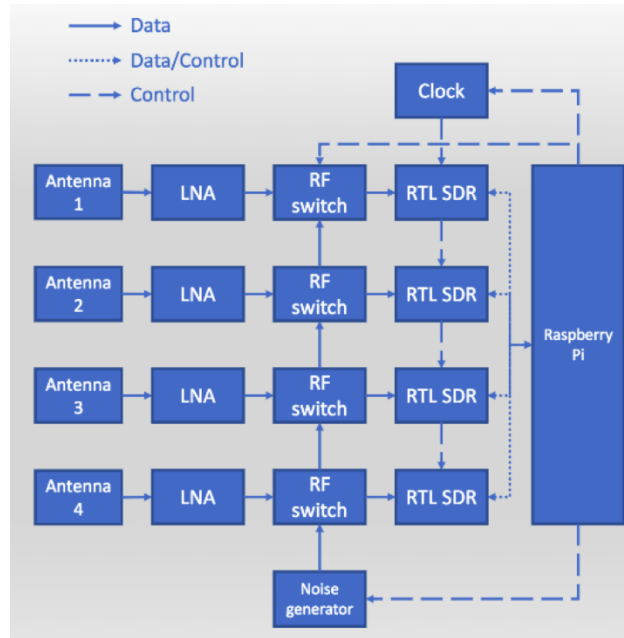


Figure 7: Receiver Architecture

Before choosing the RTL SDR and Raspberry Pi for the above architecture, we considered the following ideal specifications for a software defined radio and embedded device:

1. A sub 1-GHz device for VHF or UHF frequencies transmitted from radio tags. We choose a lower frequency band (relative to most RF applications) to mitigate multipath interference and better determine the phase difference of signals. Previous systems have

used ~150 MHz as the operating frequency of transmitters because of the impact of large trees on multipath interference.

2. A very high sample frequency during the analog to digital conversion of RF signals. This is essential for mitigating adverse effect from noise when determining accurate phase differences from radio waves moving at the speed of light. However, a sampling frequency above twice the radio frequency (constant in a non-frequency modulated signals) is not needed. Sampling rates are further discussed in section IV. Vii. “RF Wave Reconstruction and Matlab Simulation”.
3. Ample UART/I2C/SPI/GPIO connections for data logging and transfer
4. Contains every component necessary for receiving an RF signal from an external antenna – ADC, local oscillator, etc.
5. Extremely high RF sensitivity and blocking performance
6. Programmable and highly used by the public – helpful for finding more tutorials and readily available information on the device
7. Low power and low cost

From this list of specifications, the RTL SDR was chosen. An RTL SDR is a low-cost software defined radio which uses quadrature demodulation and receives frequencies from 500 kHz to 1.75 GHz. The RTL SDR has a maximum sample rate of 2.4 MS/s. At this sample rate, the RTL SDR does not drop any samples from data extraction. The RTL SDR uses a USB interface which can be connected directly with the Raspberry Pi. The receiving frequency and bandwidth of the RTL SDR can be dynamically adjusted from a terminal or software running on the Raspberry Pi.

Instead of having to manually hook up separate components together for the proposed receiver architecture, we ordered units from a company focused on multichannel RTL SDR integration called “coherent-receiver.com.” This company provides a 4-channel RTL SDR basestation with external components to coherently receive RF signals from radio tag sources. These external components include antenna switches and an integrated clock card. Figure 8 displays this device. The clock card is used to send a common clock signal to all four RTL SDRs. The common clock signal is a 28.8 GHz signal used to synchronize samples received at each RTL SDR. Even in systems with ADC’s sharing a common clock signal, sampling does not typically start at the same time (bulk delays) [9]. Therefore, a noise generator (which can be replaced by an external function generator) is utilized during noise switching. In the noise switching process, antenna switches synchronously transition from a common noise signal to individual antenna signals. This antenna switching can allow a cross correlation function to solve bulk delays by aligning channels based on the delays experienced between channels when the noise generator is on.

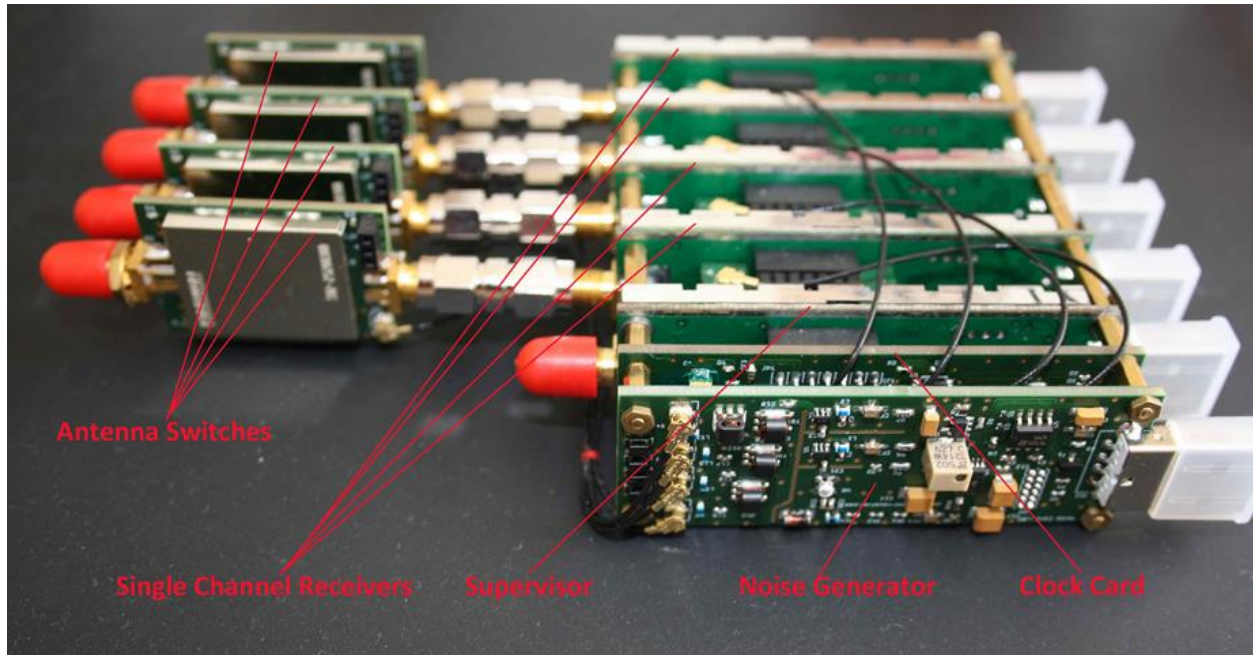


Figure 8. Coherent Receiver Components

In addition to the coherent receiver, low loss male to female SMA Connector wires are used to separate whip antennas connected to the RTL SDRs on the coherent receivers. The lengths of the connectors' insulated wires accommodate distances around half the wavelength of the incoming VHF RF signal (anywhere from 0.5 to 2.5 meters for VHF signals). In other words, the SMA connector wires will have lengths that are long enough so that antennas can be separated at half the wavelength of the incoming RF signals.

Each RTL SDR has a USB connection intended for an individual port on a Raspberry Pi. The Raspberry Pi serves as an embedded device which collects data samples from each RTL SDR.

#### V. ii. Overview of Software

The system's software primarily consists of sampling received RF signals at the RTL SDRs in the coherent receiver and performing digital signal processing (DSP) methods on these samples to generate accurate AoA measurements. All the DSP was implemented in c/python within a standard Raspbian operating system (OS).

In this system, we used a MUSIC implementation encapsulated into GNU Radio Custom Blocks developed by Ettus Research. GNU Radio is a free, open source software that provides signal processing blocks, and can display a streamlined interaction between these blocks in a GUI called GNU Radio Companion by displaying the program on a flowchart. GNU Radio was primarily used because it implements the high-speed ring buffering protocols needed to maintain synchronicity between high data rate (2 Megasamples per second) streams [22]. Ettus Research

is a National Instruments company that is the world’s leading supplier of software defined radio and publishes software that can be used with its products online. We choose to use custom blocks developed by Ettus Research because they are very well documented and have been empirically tested (see [16]). One of the major procedures of this summer’s work was developing our own version of root MUSIC/autocorrelation using the ETTUS modules as a basepoint for the software implementation outlined in Figure 9.

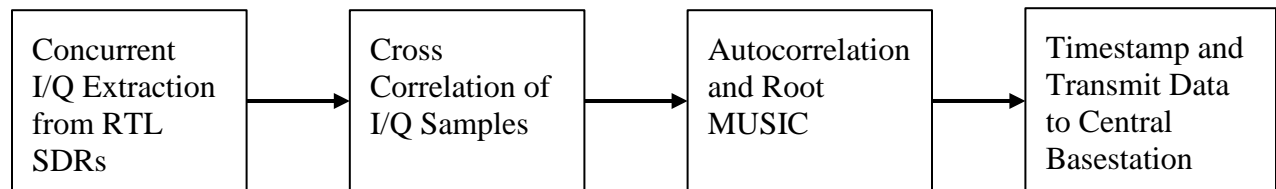


Figure 9: General software procedure for the proposed system, starting from the extraction of I/Q samples to AoA calculation and transmission.

The autocorrelation step of this procedure computes a covariance matrix using subspace smoothing. This covariance matrix is inputted into the root MUSIC algorithm (Appendix C). This procedure is explained in further detail in section V. iv. Software: Root Music with Four Antenna Elements.

### V. iii. Software: Phase Interferometry with Two Antenna Elements

The first program/flowchart implemented in GNU Radio used the phase difference of an incoming radio signal at two antennas to compute an angle of arrival [23]. The second program/flowchart (described in V. iv.) implemented root MUSIC which is described in section V. iv. The flowcharts along with the Python and C++ programs associated with the blocks within the flowcharts have been pushed to the AMRUPT GitHub [here](#). Please note that “alpha” and “normalized spacing” in the two-element flowchart and root MUSIC flowchart (in V. iv.) respectively have been corrected to 0.5 meters (each antenna in the array is spaced at half the wavelength for a 433.92 MHz signal).

The two-antenna element flowchart (based on Whiting et. al) is displayed in Figure 10. Two RTL-SDR sources are sent through virtual sinks which sends the raw I/Q samples to be used in cross correlation for timing offset corrections and phase difference analysis at the same time. The cross correlation is performed in the sample offset blocks on top which computes a delay between the data channels based on a 3 iteration FFT convolution in the frequency domain. The delay then applied by the delay blocks below. An important note is that the number of iterations specified in the sample offset blocks cannot be set too high, or the processing demands of this block will cause a loss of timing synchronicity especially when comparing more than 2 data streams.



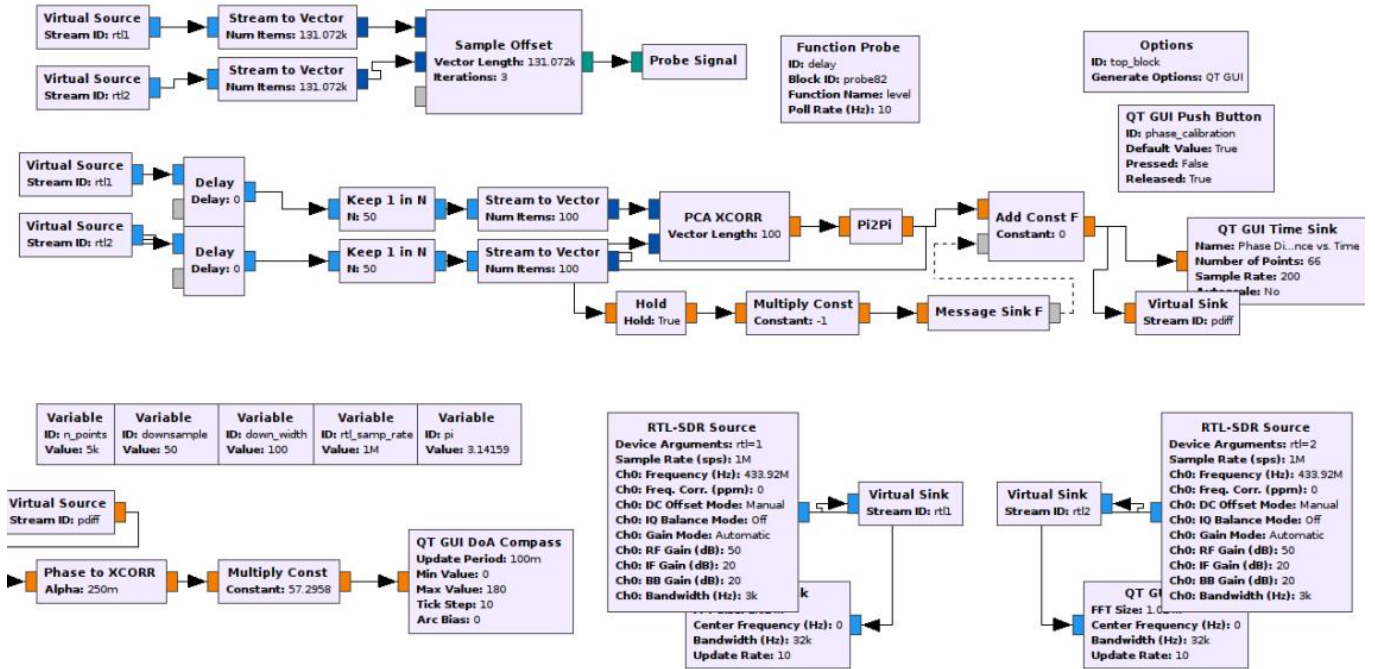


Figure 10. Two-antenna element AoA flowchart based on the phase difference between receivers

Once timing offsets are corrected for, the two RTL SDR datastreams are interpolated to include 1 out of every 50 samples for further DSP analysis. 100 samples are placed in a vector after this interpolation and sent to a block called “PCA XCORR.” The “PCA XCORR” block code [here](#) uses Capon Beamforming to compute a phase difference between the signals received by two RTL SDRs. This block was found to produce a very stable phase difference between two RTL SDRs when a common sinusoidal signal was supplied to the receivers (less than 0.1 radian variation). The “Pi2Pi” block wraps the float result from “PCA XCORR” into a range from  $-\pi$  to  $\pi$ .

Once a phase difference is reliably computed, a phase offset can be corrected by pushing a button named “phase\_calibration” on a GNU Radio GUI during runtime. The phase calibration button should be pressed when the common sinusoidal signal is being applied to both receivers. When the phase\_calibration button is pressed, the phase difference registered by “PCA XCORR” will be sent in a message format into the “Add Const F” block, which will continuously subtract the offset before the phase difference data stream is sent to the Angle of Arrival computation. The Angle of Arrival is computed by the “Phase to XCORR” block using the calculation  $DOA = \arccos(\text{phase} / (2 * \pi * \alpha))$  where  $\alpha$  is determined by the antenna spacing and the wavelength of the received signal. The flowchart has since been updated to include a float sink as well as a DoA Compass for displaying AoA values.

#### V. iv. Software: Root Music with Four Antenna Elements.

The second program/flowchart implemented Root MUSIC based on the GNU Radio modules developed in [24]. Root MUSIC is a subspace technique that estimates an angle of arrival based on the roots of a polynomial determined by the eigenvector analysis of a sensor array correlation matrix described in Appendix C. The root MUSIC flowchart is displayed in Figure 11. Timing offsets are computed using three sample offset blocks which send delay values respective to the RTL SDR combination (delay1, delay2, and delay3) variables listed as “Function Probe” hold these values.

After timing offsets are corrected between the RTL SDR channels, phase offsets are corrected using a similar procedure to the previous flowchart. Instead of directly altering the phase difference computed by PCA XCORR, the Multiply Exp block was used for phase offset correction was used so that the I/Q streams can be phase corrected instead of the phase difference values. This is done because the autocorrelation block takes in I/Q streams and not phase difference streams. The Multiply Exp block alters the phasors which changes the complex values of the signal directly by multiplying the phasor exponent by the phase offset. This method has been tested to yield stable phase difference results as the previous method.

During root MUSIC testing the following protocol would be performed on the user end for timing and phase offset correction. The RF signal generator was turned off and the noise source was turned on by a separate terminal before the start of the program. On startup, the program computes and applies timing delays between the channels, which would cause the phase difference display on the GUI with the noise source on to steady to a phase difference with +/- 0.2 radian jittering. Afterwards, the noise source was turned off, and a common sinusoidal signal source was fed into the first two receivers on the coherent receivers. The phase\_calibration1 button was pressed to correct for a phase offset when this common sinusoid is sent to RTL SDR 1&2. The same is then done for RTL SDR 1&3 and RTL SDR 1&4. Finally, the RF signal generator is turned off and the four antennas are connected to each receiver accordingly. The GUI Compass/float sink can then be monitored to record AoA values.

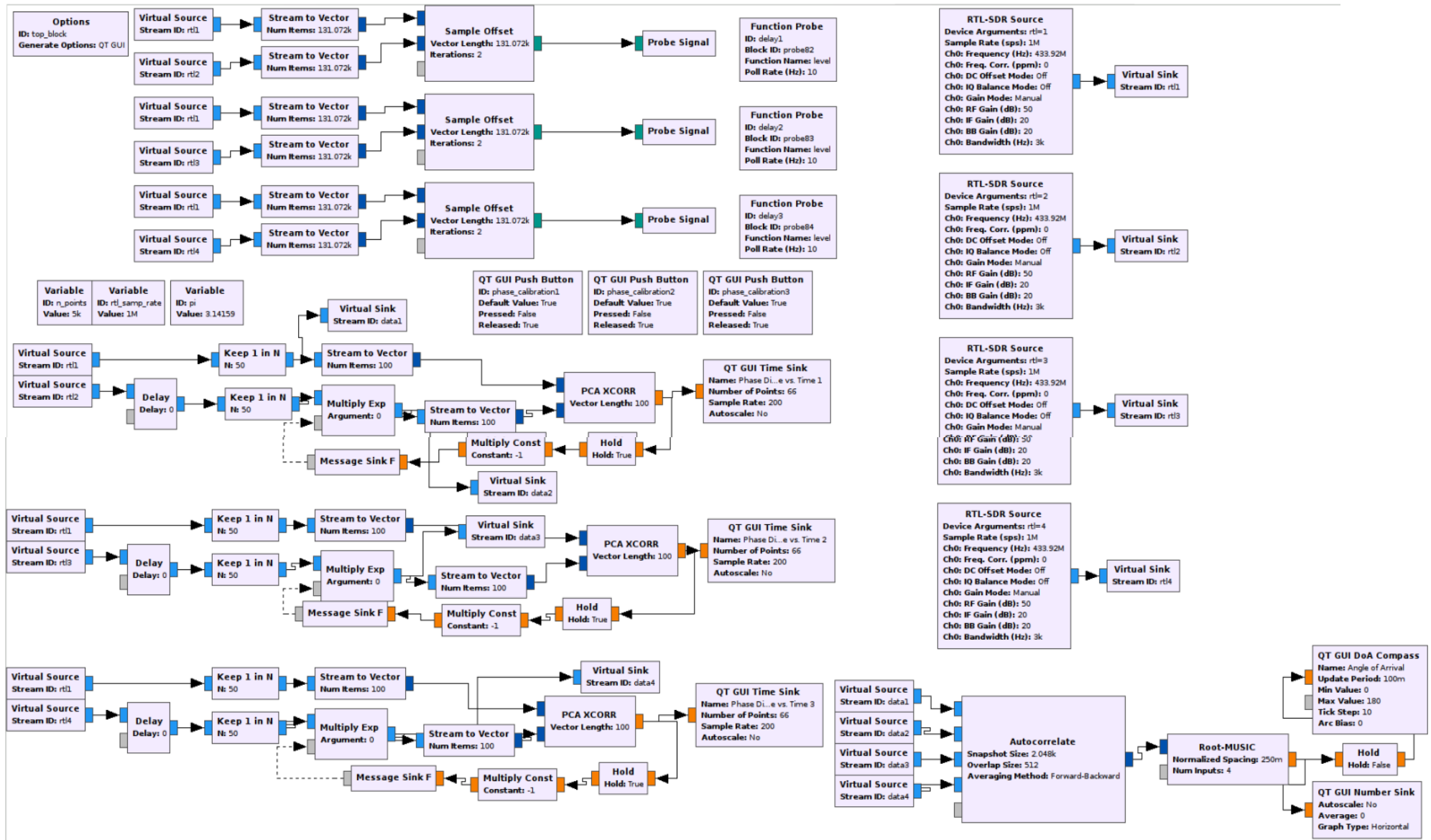


Figure 11. Root MUSIC AoA flowchart

## V. v. Raspberry Pi Datalogging and Triangulation

Once we obtain reliable AOA data from each raspberry pi basestation, we will append code to the root MUSIC program to store timestamped AoA measurements into a text file. The information in this text file will be transmitted to a central basestation for triangulation in either two ways:

1. Attaching a wireless mobile broadband modem to the Raspberry Pi similar to the procedure used in [1]. This procedure is relatively easy to implement and is cost effective; however, it comes with the cost of several drawbacks. The system clocks of several Raspberry Pi devices can differ up to 0.5 seconds which is detrimental to detection matching. This issue can be solved by using a dedicated beacon transmitter to pair detections with unique events, which can rectify timestamp offsets of up to 30 s [1]. The use of a usb broadband modem will also require mounting an external board to the Raspberry Pi with extra usb connections (the four usb ports on the Pi will be consumed with RTL SDR connections) and an internet connection, which may be unavailable in several wildlife tracking areas.
2. Connect long range RF modules (i.e. CC1310) to each Raspberry Pi ground node and a central basestation through an I2C or SPI connection. These modules would transmit the data with frequencies that do not interfere with the radio frequencies used in localizing the radio tags. The major drawback of this method is that transmissions to the central basestation would have to be coordinated so that multiple receivers do not transmit data to the central basestation at the same time. Also, data might be lost if environmental conditions prevent these transmissions from reaching the central basestation.

Once AoA measurements from separate basestations are reliably transmitted to the central basestation, triangulation will be performed using these measurements according to the pseudocode below:

```
class Coordinate:
    def __init__(self, x = 0.0, y = 0.0):
        self.x = x
        self.y = y

    #Finds intersection of two lines (y = mx + b)
    def intersect (m1, b1, m2, b2):
        intersection = Coordinate()
        xval = float(b2 - b1) / float(m1 - m2)
        yval = xval * m1 + b1
        intersection.x = xval
        intersection.y = yval
        return intersection
```

```

#Finds the centroid of the triangulated area
def centroid(vertex1, vertex2, vertex3):
    center = Coordinate()
    center.x = (vertex1.x + vertex2.x + vertex3.x) / 3.0
    center.y = (vertex1.y + vertex2.y + vertex3.y) / 3.0
    return center

#Parameters are decimal angle of arrivals calculated at
#each base station and coordinate positions of each
#base station.
def triangulated area(boa1, boa2, boa3, receiver1, receiver2, receiver3):
    #Converts angle degrees to slope
    slope1 = math.tan(boa1)
    slope2 = math.tan(boa2)
    slope3 = math.tan(boa3)

    b1 = receiver1.y - receiver1.x * slope1 #y = mx + b
    b2 = receiver2.y - receiver2.x * slope2
    b3 = receiver3.y - receiver3.x * slope3

    intersection1 = intersect(slope1, b1, slope2, b2)
    intersection2 = intersect(slope2, b2, slope3, b3)
    intersection3 = intersect(slope1, b1, slope3, b3)

    position = centroid(intersection1, intersection2, intersection3)
    return position

```

At the central basestation, a separate program will build a primary text file for the triangulation measurements computed. The program will also create a time and tag label for each measurement on file. In a future iteration of this system, a tag label will be determined by the approximate time the measurement was made (by referring to the TDMA schedule). There will be a real time clock onboard to monitor the time and help create the time and tag label for each measurement.

## VI. Results and Discussion

My original intention for this section was to quantify the level of accuracy and real time precision of the developed subspace Angle of Arrival (AoA) system for one receiver basestation. Due to inconsistent and disoriented Angle of Arrival measurements from the first round of testing, and the number of technical issues and fixes which have occurred, it is clear that the system is in a debugging stage rather than an experimental one.

This section is split into three parts. The first entails the planning and setup of the testing environment. The second entails the results of the tests and discusses issues which may have

undermined the accuracy and real time precision of AoA measurements. Possible solutions to these technical issues whether implemented or not yet implemented are entailed in the third.

## VI. i. Planning and Setup of the Testing Environment

The list of equipment for AoA tests includes the coherent receiver, usb hub, raspberry pi, ethernet cable, 8 male to female sma cables, 4 antennas, two laptops, one transmitter cc1310, a Back-UPS 450, an RF signal generator, and a testing rig for antenna array placement. The testing rig is shown in Figure 12. The top mast holding the antenna array is 6 feet above the ground and 5 feet wide. A platform was placed towards the bottom of the rig which held the coherent receiver, usb hub, raspberry pi, and ethernet cable connection. This platform was placed 2.5 feet above the ground and 3.5 feet away from the top mast. These metrics were chosen to eliminate RF interference from near field coupling [25]. To compensate for the low platform placement, two male to female sma cables had to be used for each antenna-receiver connection. The near field of the antenna array for 434 MHz would be one wavelength ( $\sim 0.7$  meters).



Figure 12: Testing Rig

One laptop was used to establish a VNC connection to a Raspberry Pi via ethernet so that on-site debugging could be performed when needed. Another laptop was used to control the CC1310 to easily set the frequency and Tx power of unmodulated transmissions with SmartRF Studio. Only one laptop is needed indoors if Wifi is present (the VNC connection to the Raspberry Pi can be made wirelessly).

The four RTL SDR radio receivers which were used for direction finding were held within the coherent receiver. The coherent receiver contained a noise card, supervisory RTL SDR, and an external connection to an RF Signal Generator to correct for timing and phase offsets between the RTL SDRs. Each RTL SDR was connected to input ports on a usb hub. The usb hub had one output port connected to the Raspberry Pi. The command “rtl\_test” was entered in the terminal before testing, to ensure that no samples were lost from the 4 to 1 usb connection.

The Back-UPS 450 was used to power electronic devices in outdoor environments. It had a 1-hour battery life when the RF signal generator, laptop, Raspberry Pi, usb hub, and coherent receiver were connected.

The testing rig would be setup in an outdoor and indoor environment for initial testing. The outdoor environment was a turfed baseball field next to Phillips Hall, Cornell University. The indoor environment was in the Duffield Hall atrium, Cornell University. The testing rig was placed in the middle of the baseball field for the outdoor test and was centered in an area with few surrounding objects in the indoor atrium.

## VI. ii. Testing Results and Improvements

I performed all the tests at a 433.92 MHz center frequency, 3k receiving bandwidth, and 50 RF gain at each receiver. I conducted the first test outdoors, and the second test indoors. From the results of these tests, I have already made several improvements to the system (to be described in section III. ii.). After making these improvements, a third indoor test was conducted. All the tests were performed with a fellow Cornell Student who helped carry equipment and hold the transmitter/computer. To properly test the AoA accuracy and real time precision of this system, calculated AoAs were to be recorded at different set measured angles and distances from the receiver basestation for both flowchart programs.

The following chart was prepared to record AoA results at multiple angles, at different distances prior to testing:

Angle of Arrival (2 element AoA or root MUSIC)

AoA RPi3	AoA Measured (Compass)	Distance
	0	~20 feet
	30	~20 feet
	60	~20 feet
	90	~20 feet
	120	~20 feet
	150	~20 feet
	180	~20 feet
	60	~40 feet
	90	~40 feet
	60	~60 feet
	90	~60 feet

Unfortunately, the AoAs for the two element AoA program could not be recorded because the phase difference would fluctuate very rapidly when moving the slightest amount along an imaginary line determined by the measured angle. This effect was recorded [here](#). Due, to this effect the AoA at a specific location was too unstable to record. Furthermore, impossible phase differences between the two antennas were registered for the test's specification. At 433.92 MHz, the maximum phase difference that can occur between two antennas spaced at 0.345



meters (quarter wavelength spacing) is a half radian. Phase differences from a half radian to a full radian were observed during testing.

The root MUSIC program generated NULL AoA values for 433.92 MHz (this was observed from the GUI compass and float sink). It is interesting to note that this did not happen at twice the frequency. A program based on the original MUSIC algorithm was briefly tested to see if it had the same effect, and it did not; although its results had the same instability issues as the two element AoA program. (This issue was recently fixed, see Solution in the Works 3 in the next section).

## VI. iii. Issues, Reconciliations, and Solutions

**Issue 1:** One of speculated possible issues of error had to do with a GUI Compass that displayed AoA values on the computer screen. The GUI Compass may have an instability issue with reported issues [here](#) and [here](#). Therefore, AoA values will be outputted to a GUI float sink in addition to the GUI Compass to see if there is disparity between the end AoA results.

**Reconciliation 1:** This issue has since been reconciled as the float sink and GUI compass display the same values, so the compass is likely not a problem. The float sink will still be used for further validation.

**Issue 2:** Constant sinusoidal amplitude disparity between receiver channels (independent of transmitter location). This was observed when a common sinusoidal signal was fed into each receiver as well.

**Solution 2:** The powering of the bias tee on each receiver (from the RTL source block code [here](#) – bias tee powered on line 233) fixed this constant sinusoidal amplitude disparity. I made this improvement because it is a necessary step according the coherent receiver [technical support instructions](#) and because an antenna signal would be blocked off from the bias tee routing more power to the noise input channel (even when the noise source is off). An interesting result of the bias tee power-up is the reduced the phase offset between the noise phase difference and the signal phase difference between receivers during phase offset correction which can be seen in the testing video [here](#).

**Issue 3:** Sinusoidal amplitude disparity dependent on transmitter position (amplitude disparity between the received waveforms displayed from each receiver when a 433.92 mHz tone was sent from the cc1310) and rapid phase difference fluctuation between any two antennas on the array when transmitter location is changed slightly. The phase difference between antenna elements should not change as the tag moves along the same angle-based line. Phase difference fluctuations were observed to go from  $-2\pi$  to  $2\pi$  radians in just a couple of meters along the



same angle-based line (angle-based line: an imaginary line corresponding to an angle of arrival). for all tests so far. Furthermore, impossible phase differences such as a  $2\pi$  phase difference is being registered for antennas spaced a quarter wavelength apart.

**Solution in the Works 3:** After speaking with Xiaonan Hui, a postdoctoral student in Dr. Kan's lab, we concluded that this issue is likely a multipath problem. This was initially difficult to pick up as the source of the issue, since the sinusoidal waves at receivers are periodic and very stable. However, this is the case for coherent reflected and line of sight signals from a single source, in which the reflected and line of sight phasors add to change the amplitude and phase of signal impinging on the antenna array, but not its period or waveform shape. Since the receiving antennas are omnidirectional, all reflected transmitted signals from all possible directions are included in this result.

The forward backwards smoothing in the ETTUS autocorrelation block does not directly mitigate multipath interference. It was my initial belief that the ETTUS autocorrelation block performed forward backwards spatial smoothing; however, it in fact performs forward backwards smoothing in time between samples. I recently programmed a forward spatial smoothing module [here](#) based off of [13] which averages the subarray matrices produced by several autocorrelation blocks.

Another way to minimize multipath is to reduce the gain from the transmitter, so that the LOS signal is more prominent among lower power reflected signals. The Tx gain can be set to lower dB values from smartRF studio during further testing.

During initial testing, the transmitter was near to a human (me) and a laptop, which could present a proximity multipath problem. The CC1310 has been placed at the end of a wooden plank (see Figure 13) and attached to a usb extension cable to minimize multipath interference from the transmitter. When the change was performed, root MUSIC did not produce anymore NULL AoA values. In future testing, a JSC to SMA connection will be made so that the orientation and polarization of the receiver and transmitter antennas are the same. The JSC to SMA connection will only be made after the CC1310 is opened with smartRF studio, because the CC1310 runs an example code at very high Tx power which overuses the power supply of the laptop usb port. This example code halts when the CC1310 is opened with smartRF studio.

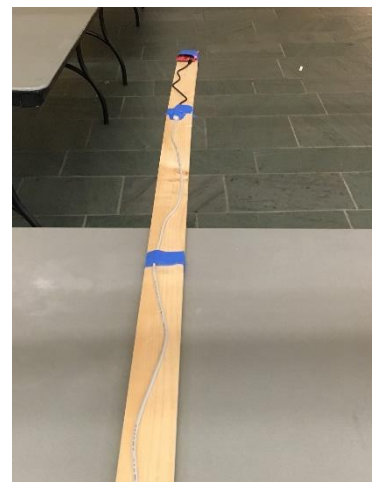


Figure 13: CC1310 Transmit Structure

I will check if our system produces reasonable AoA values using the changes mentioned in “Solution in the Works 3” during scheduled outdoor tests in the next couple of weeks. Further

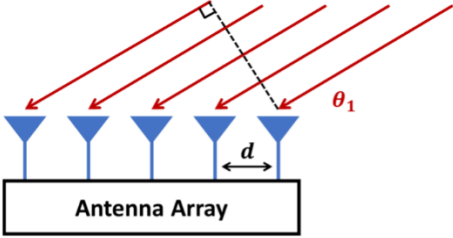
debugging outdoors will be conducted to see if the system works in an environment with much less multipath interference. It is important to note though that similar AoA systems have worked in indoor environments in close proximity to the antenna array (please see [here](#) and [here](#)), and our system is unable to generate comparable results indoors at this time. This fact may foreshadow similar technical errors for our system outdoors.

## VII. Conclusion

In order to accomplish the localization of small animals, we planned to develop a cost effective and automated system to track animal movements within the range of five meters while taking into account expected causes of error. Our proposed system consists of a receiver architecture that is built specifically for phase interferometry direction finding to facilitate accurate measurements from radio tags on tracked individuals. In order to accomplish this, a low weight radio tag is being developed to transmit signals to radio base stations. These tags will transmit sub 1-GHz UHF frequencies.

For the steps moving forward for this project, it would be a good idea to test the system outdoors in an open-field environment to see if it performs better with low multipath interference. Second, it seems better results have been obtained by experts such as Igor Sedov (CEO of coherent-receiver.com), Sam Whiting (obtained AoA from two antenna element phase difference in [11] and [23]), and Travis Collins (obtained AoA from root MUSIC in [16] and [24]). I think it would be a good time to record the system's performance in video and contact these experts on the central phase instability issue when a transmitter is used. Third, frequency-hopping can be implemented to find common angles of arrival among multiple frequencies and forward-backwards smoothing can be used to more fully mitigate the effects of multipath interference. Finally, we could move to single ground node TDOA or PDOA low-range tracking as most of the system infrastructure (hardware/extraction firmware) required for these direction finding systems have been implemented already.

### Appendix A. Matrix Notation for Received Signals at Antenna Array Elements [3, 7, 8]:



$$\begin{bmatrix} x_1(t)^T \\ x_2(t)^T \\ \vdots \\ x_M(t)^T \end{bmatrix} = [a(\theta_1) \quad a(\theta_2) \quad \dots \quad a(\theta_N)] \begin{bmatrix} s_1(t)^T \\ s_2(t)^T \\ \vdots \\ s_N(t)^T \end{bmatrix} + \begin{bmatrix} n_1(t)^T \\ n_2(t)^T \\ \vdots \\ n_M(t)^T \end{bmatrix}$$

$$X = AS + N$$

M - # of Antennas  
N - # of Signal Sources

**Steering vector:**  $a(\theta) = [1 \quad e^{j2\pi\tau(\theta)} \quad e^{j2\pi\tau(\theta) \cdot 2} \quad \dots \quad e^{j2\pi\tau(\theta)(M-1)}]^T$

Figure 14: Antenna Array Matrix Representation [7]

Each element in the array is spaced equally at distance  $d$  from each other with  $d$  less than or equal to half of the wavelength  $\lambda$  of the incoming signal.  $\lambda/2$  spacing is applied to avoid phase ambiguity (a phase difference at antennas spaced greater than  $\lambda/2$  can be  $\pm \pi n$  where  $n$  is the number of ambiguities).

As shown in the diagram, there is a radio wave which impinges upon each antenna array element with angle  $\theta_1$ . The received signal(s) at each antenna can be represented by a matrix “X” formed of vectors  $x_M(t)$  where the prefix M denotes the antenna number. The phase shift at each antenna of an incoming signal can be put into a  $M \times N$  matrix “A” formed of steering vectors (the number of steering vectors correspond to the number of signal sources). A steering vector can be used to represent the phase shift per antenna of an incoming signal, with  $\tau_\theta = d \cos \theta / \lambda$  terms derived from the extra distances a radio wave must travel to antennas based on the received angle of the signal. The  $\tau_\theta$  term in each array element in the steering vector is multiplied by the array element’s index in the exponent of the phasor. This is because the second array element has 1x the phase shift, the third array element has 2x the phase shift, and so on. Signals denoted by  $s_N(t)$  are represented in phasor notation as  $s_N(t) = A e^{j\omega t}$  where the prefix N denotes a signal number (there can be multiple incoming signals), A represents an amplitude, and  $\omega$  represents the frequency of the signal.  $n_M(t)$  terms are noise vectors which represent external or system-inherent effects on signals transferred into a digital signal processor.

## **Appendix B. Multiple Emitter Location and Signal Parameter Estimation [3, 7, 8]:**

Using the assumption that incident signals and noise vectors are uncorrelated, we can take the covariance “Rxx” of the  $X = AS + N$  matrix formulation from Appendix A and set it equal to  $ARssA^H + \sigma^2 I$  where the superscript H denotes the Hermitian transpose of a matrix. Rss is an  $N \times N$  matrix with rank equal to N if the source signals are independent. An important consideration here is if the signals are correlated, the signal subspace will have dependence, which disrupts the performance of the MUSIC algorithm. This often occurs under multipath interference in which refracted signals can be correlated with source signals. Methods such as subspace smoothing or iterative methods described in [13] can improve the resiliency of MUSIC under multipath conditions.

Following the  $ARssA^H + \sigma^2 I$  from before, A is a matrix composed of the steering vectors of the incoming signals impinging on the antenna array. It will have as many positive eigenvalues as signal sources (N) because each signal source will contribute an independent eigenvector in the signal subspace. The rest of the eigenvalues ( $M - N$  eigenvalues left) will correspond to eigenvectors in the noise subspace. The crucial step of MUSIC is partitioning the eigenvectors of positive eigenvalues into a signal subspace and the eigenvectors of zero eigenvalues into a noise subspace. Once a noise subspace is found, the Hermitian transpose of the subspace is multiplied

to a calculated steering vector (based on the positions of antennas in the array) for a range of all possible angles. This calculation will result in zero values for situations in which the noise subspace is orthogonal to the steering vector (representative of the signal subspace), which will occur precisely when the AoA of a signal source is aligned with the AoA of the chosen steering vector.

Step by step, the algorithm is as follows:

1. Calculate the covariance matrix of the vector of complex signals, the number of source signals must be known beforehand.
2. Perform an eigenvalue decomposition of the covariance matrix.
3. Sort the eigenvectors according to their eigenvalues in greatest to least order.
4. Place the last  $M - N$  eigenvectors into a noise subspace matrix  $Q$
5. Compute steering vector  $a(\theta)$  for all angles within  $[0, 180]$  based on antenna spacings and signal wavelength. (Can reduce computationally complexity here by focusing on a smaller angle range in a combined TDOA/PDOA approach).
6. Plot the pseudospectrum  $p(\theta) = \frac{1}{a^H(\theta)QQ^Ha(\theta)}$ , where the resulting plot has an x-axis in degrees and a y-axis in decibels. The peaks of the graph correspond to angle of arrivals of source signals.

A comparison of music against conventional beamforming, maximum likelihood, and maximum entropy methods is shown in Figure 15.

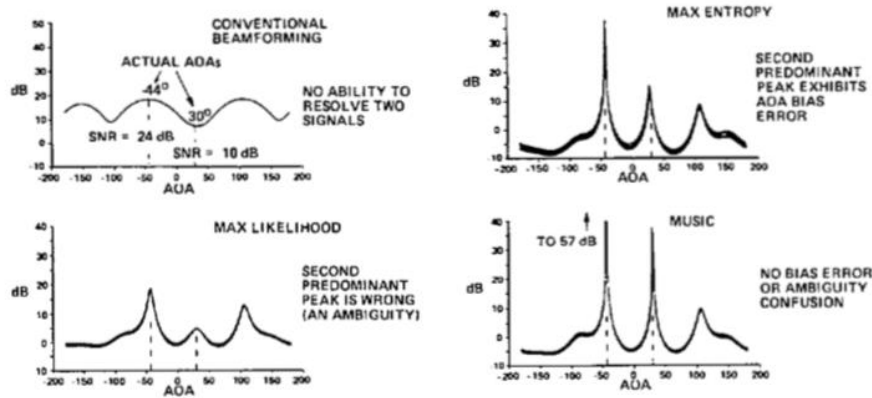


Figure 15: Azimuth-only performance in common AoA methods [3]

## Appendix C. Root Multiple Emitter Location and Signal Parameter Estimation [16, 21]:

Recall from Appendix B that the denominator of the pseudospectrum was  $a^H(\theta)QQ^Ha(\theta)$ , where  $a(\theta)$  is the steering vector (computed for an arbitrary number of angles within a direction finding range) and  $Q$  is the noise subspace retrieved from an eigenvalue decomposition of the covariance matrix. Also, recall the form of the steering vector from Appendix A, where each

element of the vector represents a phase shift at an antenna index. The denominator of the pseudospectrum can be expressed as in [16] where  $\mathbf{a}$  is  $\mathbf{v}$  and  $\mathbf{Q}$  is  $\mathbf{U}_N$ :

$$\begin{aligned} Q(z) &= \mathbf{v}^T (1/z) \mathbf{U}_N \mathbf{U}_N^H \mathbf{v}(z) \\ &\stackrel{(a)}{=} \sum_{m=0}^{N-1} \sum_{n=0}^{N-1} z^{-m} \mathbf{U}_N^2(m, n) z^n \\ &\stackrel{(b)}{=} \sum_{l=-N+1}^{N-1} u_l z^l. \end{aligned}$$

In b),  $u_l$  corresponds to the sum of elements under the noise subspace matrix's diagonal for  $l < 0$  and the sum of elements over the noise subspace matrix's diagonal for  $l > 0$ .  $l$  can take on any value between  $N - 1$  and  $N + 1$  where  $N$  corresponds to the number of antennas. Just imagine for now that  $N - 1$  and  $N + 1$  is  $M - 1$  and  $M + 1$  to avoid confusion with the explanations in Appendices A and B (where  $M$  is the number of antennas and  $N$  is the number of signal sources).

By multiplying each noise subspace sum with each element in the steering vector, and adding all of these results up, a polynomial of size  $M - 1$  will result. From this polynomial, there should be  $N$  roots that lie on the unit circle which will correspond to the AoAs of incoming signal sources. The reason for there being  $N$  roots on the unit circle ties into the orthogonality of noise and signal subspaces. This orthogonality means that a multiplication of a computed steering vector (for an angle of arrival of an incoming source signal, representing a vector from the signal subspace) and the noise subspace should result in a very tiny value if not zero. Furthermore, the roots of a polynomial are values that cause the polynomial to evaluate to zero.

Due to the presence of noise, the roots may not necessarily be directly on the unit circle. For this reason, the  $N$  (number of signal sources) closest roots to the unit circle are chosen to represent the angles of incoming source signals. Determining the closest roots to the unit circle can be done by taking the Euclidean distance of each root (for a real and complex value). The roots with Euclidean distances closest to 1 will be chosen.

Once the roots are determined, the AoA for each incoming signal is determined by the following expression:

$$\theta_n = \arcsin\left(\frac{\lambda}{2\pi d} \arg(z_d)\right)$$

Where  $z_d$  corresponds to an individual root out of  $N$  roots,  $\lambda$  is the wavelength of the signal, and  $d$  is the antenna separation. The function  $\arg$  refers to  $\tan^{-1}(\frac{y}{x})$  from the complex form  $x + iy$ .

## Appendix D. References

- [1] Krüger, S W. "An Inexpensive Hyperbolic Positioning System for Tracking Wildlife Using off-the-Shelf Hardware." May 2017.
- [2] Rohde & Schwarz "Introduction into Theory of Direction Finding," Radiomonitoring & Radiolocation, Catalog 2011/2012.
- [3] Schmidt, R. "Multiple Emitter Location and Signal Parameter Estimation." *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, 1986, pp. 276–280.
- [4] D. Guerin, S. Jackson, and J. Kelly, "Passive Direction Finding: A Phase Interferometry Direction Finding System for an Airborne Platform," Oct. 10, 2012. <https://web.wpi.edu/Pubs/E-project/Available/E-project-101012-211424/unrestricted/DirectionFindingPaper.pdf>.
- [5] D. Zhang, J. Ma, Q. Chen and L. M. Ni, "An RF-Based System for Tracking Transceiver-Free Objects," Fifth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom'07), White Plains, NY, 2007, pp. 135-144.
- [6] Y. Ma, X. Hui, and E. Kan, "3D Real-time Indoor Localization via Broadband Nonlinear Backscatter in Passive Devices with Centimeter Precision," Oct. 3, 2016. <https://dl.acm.org/citation.cfm?id=2973754>.
- [7] Wenguang Mao "Approaches for Angle of Arrival Estimation" <https://pdfs.semanticscholar.org/presentation/723e/3f7ce640f425671d4e0af7d23b82e04a402e.pdf>
- [8] Fan, H. Howard "Direction of Arrival Estimation (DOA) in Interference & Multipath Propagation," GIRD Systems, Inc.
- [9] Whiting, Sam, et al. "Time and Frequency Corrections in a Distributed Network Using GNURadio." 2017.
- [10] "Tejeez/rtl\_coherent." GitHub, 6 July 2016, [github.com/tejeez/rtl\\_coherent](https://github.com/tejeez/rtl_coherent)
- [11] Whiting, Sam, et al. "Time and Frequency Corrections in a Distributed Network Using GNURadio." 2017
- [12] Y. Ma, X. Hui, P. Sharma, E. Kan. "Indoor Passive Device Ranging by Low-directivity Antennas with Centimeter Precision."
- [13] E. M. Al-Ardi, R. M. Shubair, and M. E. Al-Mualla. "Direction of Arrival Estimation in a Multipath Environment: an Overview and a New Contribution," *Aces Journal*, Vol. 21, No.3, November 2006.
- [14] Badawy, Ahmed, et al. "A Simple Angle of Arrival Estimation System." *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, 2017.
- [15] Qiu, Lanxin, et al. "Multifrequency Phase Difference of Arrival Range Measurement: Principle, Implementation, and Evaluation." *International Journal of Distributed Sensor Networks*, vol. 11, no. 11, 2015, p. 715307.
- [16] "Phase Synchronization Capability of TwinRX Daughterboards and DoA Estimation" [https://github.com/EttusResearch/gr-doa/blob/master/docs/whitepaper/doa\\_whitepaper.pdf](https://github.com/EttusResearch/gr-doa/blob/master/docs/whitepaper/doa_whitepaper.pdf)
- [17] <http://www.its.caltech.edu/~matilde/GaborLocalization.pdf>
- [18] Goldsmith, Andrea. *Wireless Communications*. Cambridge University Press, 2004.
- [19] Zekavat, Reza, and Michael Buehrer. *Handbook of Position Location: Theory, Practice and Advances*. IEEE-Wiley, 2012.

- [20] Scher, Aaron “How to capture raw IQ data from a RTL-SDR dongle and FM demodulate with MATLAB”
- [21] H. K. Hwang, Z. Aliyazicioglu, M. Grice, and A. Yakovlev, “Direction of Arrival Estimation using a Root-MUSIC Algorithm” March, 2008.
- [22] Müller, Marcus. “Behind the Veil: A Peek at GNU Radio's Buffer Architecture.” *GNU Radio*, [www.gnuradio.org/blog/buffers](http://www.gnuradio.org/blog/buffers).
- [23] Whiting, Sam. “Samwhiting/Gnuradio-Doa.” *GitHub*, 2017, [github.com/samwhiting/gnuradio-doa](https://github.com/samwhiting/gnuradio-doa).
- [24] Collins, Travis. “EttusResearch/Gr-Doa.” *GitHub*, 2016, [github.com/EttusResearch/gr-doa](https://github.com/EttusResearch/gr-doa).
- [25] Wang, Hanfeng, et al. “Estimating Radio-Frequency Interference to an Antenna Due to Near-Field Coupling Using Decomposition Method Based on Reciprocity.” *IEEE Transactions on Electromagnetic Compatibility*, vol. 55, no. 6, 2013, pp. 1125–1131.