Distributed Coherent Mesh Beamforming: Algorithms and Implementation

by

Jacob Holtom

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

Approved March 2023 by the
Graduate Supervisory Committee:

Daniel W. Bliss, Chair
Ahmed Alkhateeb
Andrew Herschfelt
Nicolò Michelusi

ARIZONA STATE UNIVERSITY

May 2023

ABSTRACT

In this dissertation, I implement and demonstrate a distributed coherent mesh beamforming system, for wireless communications, that provides increased range, data rate, and robustness to interference. By using one or multiple distributed, locally-coherent meshes as antenna arrays, I develop an approach that realizes a performance improvement, related to the number of mesh elements, in signal-to-noise ratio over a traditional single-antenna to single-antenna link without interference. I further demonstrate that in the presence of interference, the signal-to-interference-plus-noise ratio improvement is significantly greater for a wide range of environments. I also discuss key performance bounds that drive system design decisions as well as techniques for robust distributed adaptive beamformer construction. I develop and implement an over-the-air distributed time and frequency synchronization algorithm to enable distributed coherence on software-defined radios. Finally, I implement the distributed coherent mesh beamforming system over-the-air on a network of software-defined radios and demonstrate both simulated and experimental results both with and without interference that achieve performance approaching the theoretical bounds.

ACKNOWLEDGMENTS

Thank you Dr. Daniel Bliss for the opportunity to work in the lab, teaching me, and enabling me to think about interesting problems.

Thank you Dr. Ahmed Alkhateeb, Dr. Nicolò Michelusi, Dr. Andrew Herschfelt for serving on my committee.

Thank you Dr. Christ Richmond for your advice and direction.

Thank you Dr. Owen Ma for thinking and working on the interesting problems with me.

Thank you Isabella Lenz, Shammi Doly, Adarsh Venkataramani and the rest of the lab for their input and hard work.

Thank you Mom and Dad for your support, encouragement, and setting an example of lifelong learning.

Finally and most of all, Camille, for your undying love and support in this endeavor and always.

TABLE OF CONTENTS

Page

LIST OF TABLES . . . vi

LIST OF FIGURES . . . vii

CHAPTER

1 INTRODUCTION . . . 1

1.1 Prior Work in Distributed Coherent Beamforming . . . 10

1.2 Contributions . . . 12

1.3 Notation . . . 13

2 BACKGROUND . . . 14

2.1 MIMO Channels . . . 15

2.1.1 Space-Time MIMO Channels . . . 16

2.2 Beamforming . . . 17

2.2.1 MMSE Beamforming . . . 19

2.3 Mesh Relay Systems . . . 21

2.3.1 Amplify and Forward . . . 21

2.3.2 Decode and Forward . . . 22

2.3.3 Space Time Block Codes and Other Approaches . . . 22

2.4 Software Defined Radio Systems and WISCANet . . . 22

2.4.1 Ettus USRP and UHD . . . 23

3 DISTRIBUTED MESH BEAMFORMING . . . 25

3.1 Adaptive Spatial Relay Beamformer ($1 \to N \to 1$) . . . 28

3.2 Adaptive Space-Time Relay Beamformer ($1 \to N \to 1$) . . . 35

3.3 Operating Regimes . . . 40

3.4 Performance Bounds . . . 42

CHAPTER Page

3.5 Performance Metrics . . . 47

3.5.1 SNR Gain . . . 47

3.5.2 Interference Rejection . . . 48

3.6 Practical Algorithm Implementation and Challenges . . . 50

3.6.1 Time-Frequency Synchronization . . . 50

3.6.2 Information Exchange and Limited Data Rates . . . 57

3.6.3 Outage Handling . . . 59

3.6.4 Covariance Matrices . . . 60

3.6.5 Internal Beamforming . . . 66

4 MULTIPLE STAGE RELAY BEAMFORMING . . . 67

4.1 Compound SIMO-MIMO-MISO Channel . . . 67

4.2 MIMO Beamforming Technique . . . 70

4.3 Compensated MIMO Beamforming Technique . . . 72

4.3.1 Internal Beamforming . . . 73

4.4 Results . . . 74

4.4.1 MATLAB Simulation . . . 74

4.4.2 WISCANet Experiments . . . 75

5 SIMULATION . . . 78

5.1 Results . . . 81

5.1.1 Mesh Relay Noise Dominant Regime . . . 82

5.1.2 Receiver Noise Dominant Regime . . . 91

6 EXPERIMENTS . . . 95

6.1 Pseudo-Real-Time WISCANet Experiments . . . 95

6.2 Real-Time Over-the-Air Experiment . . . 98

6.2.1 Waveform Design 99

6.2.2 Time Alignment Approach 101

6.2.3 Frequency Alignment 101

6.2.4 Beamformer Estimation and Expected Performance 104

6.3 Over-the-Air Real-Time Experimental Results 105

7 SUMMARY 109

REFERENCES 110

LIST OF TABLES

Table Page

1. Distributed Mesh System Simulation Parameters 74

LIST OF FIGURES

Figure Page

1. By Utilizing a Relay Mesh between Two Single Antenna Nodes, the Desired SNR Gain Is Realized. Interference Received at the Mesh Is Rejected through the Application of Spatial Nulls in Transmit Beamforming. The Mesh Receives the Signal at Its N Antennas, and Applies the Appropriate Beamforming Pre-Distortion Filter to the Signal before Retransmission. Finally, It Retransmits the Transformed Signal to the Receiving Node. The Cycle Is Then Repeated in the Reverse Direction. 5

2. By Utilizing Relay Meshes between Two Single Antenna Nodes, the Desired SNR Gain Is Realized. Interference Is Rejected at the Mesh A and Mesh B Receiving Stages through the Application of Spatial Nulls in the Transmit Beamforming. The Interference at the Final Receiver Is Overcome by the Increased Signal Power. The First Mesh Receives the Signal at Its N Antennas, and Applies the Appropriate Beamforming Pre-Distortion Filter to the Signal before Retransmission. The Second Mesh Receives the Retransmitted Signal at Its M Antennas, Where It Applies Its Beamforming Pre-Distortion Filter. Finally, the Second Mesh Retransmits the Transformed Signal to the Receiving Node. The Cycle Is Then Repeated in the Reverse Direction. 6

3. Single Mesh Topology: Transmitting Node to Distributed Mesh to Final Receiving Node in the Presence of an Interferer. 26

4. Dual Mesh Topology: Transmitter Node to Distributed Mesh to Another Distributed Mesh to Final Receiving Node in the Presence of Interferers at Each Relay Stage. 26

Figure Page

5. Interference Received at the Mesh Is Retransmitted along with the Signal of Interest, and Combining for SNR Gain and Interference Migitation Occurs at the Receiving User. 27

6. Operational Regimes of 1 to N to 1 Distributed-Coherent Beamforming Relay System. The Mesh Has a Total Power Limit that Depends on the Output Power of Each Element (Blue Dash). To close a Given Link, the Final Receiver Must Receive the Signal of Interest at Some Minimum Power Threshold (Yellow Dash). The Mesh Network Acts as a Bent-Pipe Relay, So Any Received Interference Will Occupy Some Portion of the Re-Transmitted Power (Red Bars) on Top of the Actual Signal of Interest (Green Bars). In Scenario 6a, the Noise Power at the Mesh Dominates the System, Limiting the Potential SNR Gain to a Factor of N. This Power Distribution Scenario Occurs When the Transmitter to Mesh Link Is Long or Interference Is Strong. In Scenario 6b, the Noise Power at the Receiver Dominates, Limiting the Potential SNR Gain to a Factor of N^2. This Power Distribution Scenario Happens When the Mesh to Receiver Link Is Long, as a Result the SNR Gain Will Dominate the Retransmitted Power. In the Intermediate Distance Scenario 6c, Neither Dominates and the Potential SNR Gain Is Limited Somewhere between N and N^2. This Power Distribution Occurs When the Links between the Transmitter and Mesh, Mesh and Receiver Are Both Longer, or of "intermediate Distance", the System Does Not Fall into One of the Previous Power Distribution Regimes, and Fails to close the Total Link with Insufficient Retransmitted Power Allocation. 41

Figure Page

7. Sources of Frequency Error that Result Throughout the Mesh Transmissions. ϵ_A Results from the Deviation from Nominal Center Frequency of the A Node. The Δs Arise from Motion between A and the Mesh and B Inducing Doppler Shifts. ϵ_M Arises from the Deviation from the Nominal Center Frequency of the Mesh Elements. ϵ_B Arises from the Deviation from the Nominal Center Frequency of the B Receiver. 51
8. Theoretical SNR Gain as a Function of Frequency Estimation Error Standard Deviation Parameterized by Update Rate 55
9. Theoretical INR Reduction as a Function of Frequency Estimation Error Standard Deviation Parameterized by Update Rate. 56
10. Progression of Covariance Estimate Eigenspectra over 3 Iterations at System Startup, Including Raw and Versions with Regularization Applied. 62
11. By Utilizing a Relay Meshes between Two Single Antenna Nodes, the Desired SNR Gain Is Realized. Interference Is Rejected at Each Stage through the Application of Spatial Nulls in the Transmit Beamforming. The First Mesh Receives the Signal at Its N Antennas, and Applies the Appropriate Beamforming Pre-Distortion Filter to the Signal before Retransmission. The Second Mesh Receives the Retransmitted Signal at Its M Antennas, Where It Applies Its Beamforming Pre-Distortion Filter. Finally, the Second Mesh Retransmits the Transformed Signal to the Receiving Node. The Cycle Is Then Repeated in the Reverse Direction. 68
12. Basic Mesh-Oriented MIMO Channel in the Presence of an Interferer. 70
13. Performance Results from 400 Trials of Simulation of the Internal Beamforming Multiple Stage Relay System 75

Figure Page

14. Experimental Validation of the Ratio of Beamformed-To-SISO SNR over Three Separate Runs for a Single Node to 4-Node Mesh to 4-Node Mesh to Single Node Network. The Channel Is Unknown for the First Trial of Each Run, So the Combining Is Incoherent. 77

15. Transmitter Power Spectral Density 82

16. Full Rate Mesh Relay Cross Correlation 83

17. Mesh Relay Cross Correlation Downsampled 84

18. Receiver Detected Training Sequences 85

19. Channel B Estimate 85

20. Channel B Estimate Downsampled 86

21. Averaged Covariance Estimate 87

22. Covariance Estimate Eigenspectrum 87

23. Computation Rate Beamforming Taps 88

24. Full Rate Beamforming Taps 88

25. Receiver Received Signal Power Spectral Density 89

26. Receiver Received Signal Power Spectral Density without Beamforming Optimization 90

27. Per Cycle System Performance Evaluation in the Mesh Noise Dominant Regime 92

28. INR Surface Evaluated around the Receiver Showing the Interference Combining at All Locations but the Receiver 93

29. Per Cycle System Performance Evaluation in the Receiver Noise Dominant Regime 94

Figure Page

30. SNR Surface Evaluated around the Receiver Showing the far Field Response of the Transmit Beamformer . . . 94

31. Experimental Validation of a Narrowband Beamformer Implementation of the Ratio of Beamformed-To-SISO SINR Given for a Single Node to 4-Node Mesh to Single Node Network with an Interferer Received at the Mesh. The Channel Is Unknown for the First Trial, So the Combining Is Incoherent. Note: The Measurement Is of Signal to Interference+Noise Ratio (SINR) Gain, but the Theoretical Limit Is Formulated as Signal to Noise Ratio (SNR) Gain. . . . 96

32. Experimental Validation of a Wideband Beamformer Implementation of the Ratio of Beamformed-To-SISO SINR Given for a Single Node to 4-Node Mesh to Single Node Network with an Interferer Received at the Mesh. The Channel Is Unknown for the First Trial, So the Combining Is Incoherent. Note: The Measurement Is of Signal to Interference+Noise Ratio (SINR) Gain, but the Theoretical Limit Is Formulated as Signal to Noise Ratio (SNR) Gain. . . . 97

33. System Designs . . . 98

34. Mesh and Transmitter and Interference Source . . . 99

35. Receiver Looking back at Mesh, Transmitter and Interference . . . 100

36. Transmitter to Mesh Waveform Structure . . . 100

37. Mesh Transmit to Receiver Waveform Structure . . . 100

38. Expected SNR Improvement Performance for Prototype Systems . . . 105

39. Expected INR Reduction Performance for Prototype Systems . . . 106

Figure Page

40. Over-The-Air Experimental Results. In Figures 40a and 40b, We Present the Over-The-Air Real-Time Distributed Coherent Mesh Beamforming SNR Improvement Performance for 4 and 8 Element Meshes Respectively. The Results Were Collected in the Receiver Noise Dominant Regime, with Performance within Approximately 1 DB of the Theoretical Limit. The First Data Points Are during the Initialization of the Frequency Correction, Accordingly They Achieve Incoherent Gain. In Figures 40c and 40d, We Present the Interference Mitigation Performance for 4 and 8 Element Meshes Respectively. These Measurements Were Made in the Mesh Noise Dominant Regime with a Signal to Interference Ratio of −7 DB Received at the Mesh. 108

Chapter 1

INTRODUCTION

In this dissertation, I implement and demonstrate a distributed coherent mesh beamforming system for communications that provides increased range, data rate, and robustness to interference. To achieve this, I leverage one or more distributed, locally-coherent meshes as antenna arrays. I then develop an approach that realizes signal-to-noise ratio (SNR) performance improvements related to the number of mesh elements. I further demonstrate that in the presence of interference, the approach enables the improvement of the signal-to-interference-plus-noise ratio for a wide range of environments. I will then discuss key practical system design decisions and techniques for robust distributed adaptive beamformer construction. I implement the required time and frequency synchronization to enable distributed coherent beamforming. Finally, I demonstrate a practical distributed coherent mesh beamforming system over-the-air on a network of software-defined radios and demonstrate both simulated and experimental results both with and without interference that achieve performance approaching the theoretical bounds.

Beamforming enables a system to take advantage of spatial diversity in distributed antenna arrays. This results in increased signal-to-noise ratio (SNR) and resistance to interference. Typically, beamforming is done with a wired antenna array, consequently transitioning to distributed coherent beamforming requires synchronization between the distributed elements. Other existing distributed transmit beamforming approaches do not directly consider robustness to interference in the channel. By considering robustness to interference in the beamforming formulation, I enable both SNR improvement and interference rejection capabilities. Recent advancements in distributed positioning and timing further enable the required synchronization between distributed transceivers. Additional advancements in low-cost antenna design, high-performance domain specific processor design [5, 27, 1], and efficient synchronization techniques [15, 17] have additionally contributed to enabling a new class of distributed-coherent RF systems [9, 16] that were previously considered prohibitively difficult or expensive. Among these systems are distributed beamforming networks, which enable extremely large beamforming apertures using arbitrarily and irregularly spaced elements at the cost of significant synchronization and processing requirements. While this approach can significantly outperform the intrinsic limitations of traditional beamforming arrays, it also introduces a new set of challenges and shortcomings.

It has been assumed possible, but highly difficult, to build practical distributed coherent array processing systems. A practical real-time distributed coherent system must be relatively low-cost and has independent RF processing chains. With these independent elements comes inherent noise - that is different between all elements - in many stages in the processing chain. For example, there are differences in frequency and in each systems sense of time. To further complicate matters, these differences are constantly changing as clocks drift and systems undergo mobility. The core challenge of real-time distributed coherent systems is to synchronize all differing quantities sufficiently to realize coherence amongst the nodes and perform coherent processing in real-time. This requires alignment of time, frequency and phase. No system wants to behave naturally in such a synchronous fashion as the natural chaos and noise of realizable systems fight this goal. Additionally, the typically selected components and designs do not favor implementing such a system. It is also computationally and spectral resource intensive to achieve coherence.

In this dissertation, I show that distributed coherence can be achieved in real time with a practical implementation. I present one set of techniques and approaches that address these difficult problems. The precision required of the synchronization is several orders of magnitude better than what currently employed communications systems operate with. I now address the three core synchronization challenges. Time synchronization has been well studied and shown feasible. There are practical implementations of distributed time synchronization with the potential for sufficient accuracy. However, they are not commonly employed for this purpose. Frequency synchronization has also been well studied. Sufficient frequency precision is challenging and not commonly done. Most existing systems do not require frequency alignment within a small fraction of a hertz. For example, traditional communications systems typically only estimate with precision on the order of tens or even hundreds of hertz. Phase alignment is also required for coherent processing. Often systems rely on position knowledge to compute the required phase shifts, instead I leverage waveform exploitation in a novel way for the defined scenario.

In addition, the proposed system requires real time coherence between the distributed systems. Not only must the system have sufficiently precise time, phase and frequency alignment, it must do so in real-time. The approach leverages transmit beamforming, and as a result, coherence must be realized over the air, not in post processing.

Existing approaches such as distributed coherent radar systems or radio astronomy systems, typically rebuild their coherence in post-processing. Other distributed relay systems, such as amplify and forward or barrage relays, typically rely on time sharing approaches to get multiple perspectives on the signal and rebuild the signal and improvement in post processing.

Using these advancements in distributed coherent systems, I realize more robust links in clusters of IoT devices, 6G radio systems or other size, weight and power constrained systems – even in potentially congested spectral environments. Complex distributed systems are becoming the dominant mode of operation, with many antennas, distributed base stations and/or remote radio heads. Battery life in mobile devices and Internet of Things (IoT) systems is a major obstacle, thus it is imperative to design systems that can maintain high data rates without a high power budget. Clusters of IoT systems distributed over long distances can utilize these techniques to reach previously unavailable base stations for reporting or tasking. The proposed approach can also improve the reliability and flexibility of proposed drone based communications relay systems. The developed algorithm can be used in distributed 5G/6G base stations to serve a greater area, number of users, or to reduce need for wired backhaul. The same approach could also be used in small satellite constellations to develop more robust long distance links among nodes and ground segments. In addition, spectral congestion is also increasing with the advent of massive IoT deployments and high density 5G systems. Developing interference tolerant and adaptive systems, such as the proposed distributed coherent adaptive beamforming algorithm, will help support and maintain the expected performance to all users.

I present a distributed mesh beamforming algorithm that takes advantage of many lower power transmit and receive elements to develop a higher performance link, with increased resistance to interference. As part of a team, I developed and implemented a novel set of techniques that fully exploit the distributed nodes to maximize link range, data rate, and robustness. By performing transmit beamforming from the mesh, the signal observed by the intended receiver is simultaneously amplified and the intereference observed at the mesh is nulled, that is, receive nulling is performed through transmit beamforming.

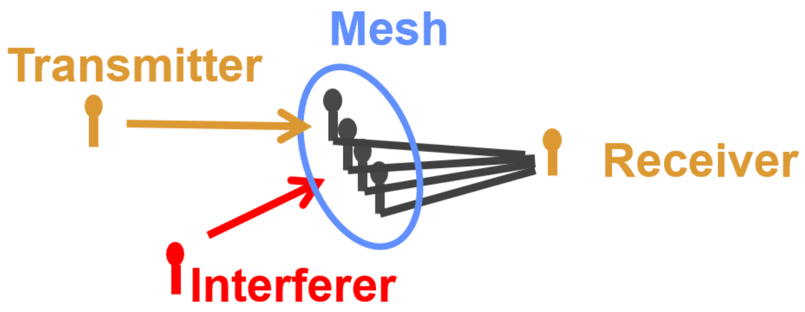


Figure 1: By utilizing a relay mesh between two single antenna nodes, the desired SNR gain is realized. Interference received at the mesh is rejected through the application of spatial nulls in transmit beamforming. The mesh receives the signal at its N antennas, and applies the appropriate beamforming pre-distortion filter to the signal before retransmission. Finally, it retransmits the transformed signal to the receiving node. The cycle is then repeated in the reverse direction.

I now will briefly discuss the novel distributed coherent mesh relay technique. To extend and improve the performance of a typical SISO communications link, locally-coherent meshes of distributed helper radios are employed, as depicted in the mesh of Figure 2. First, a signal is sent from the transmitter node. This original radio signal is potentially shifted in frequency, has training signals incorporated, and potentially is compressed. This signal is then distributed amongst the Mesh A nodes, and is potentially transformed again. An optimized set of distributed pre-distortion filters (one for each mesh node) are applied to each signal, operating as a wideband beamformer. This beamformer maximizes the SINR at the receiving Mesh B [45].

At Mesh B, the signal is received, potentially transformed, and transmitted to the final receiving node by using another wideband beamformer also implemented as a

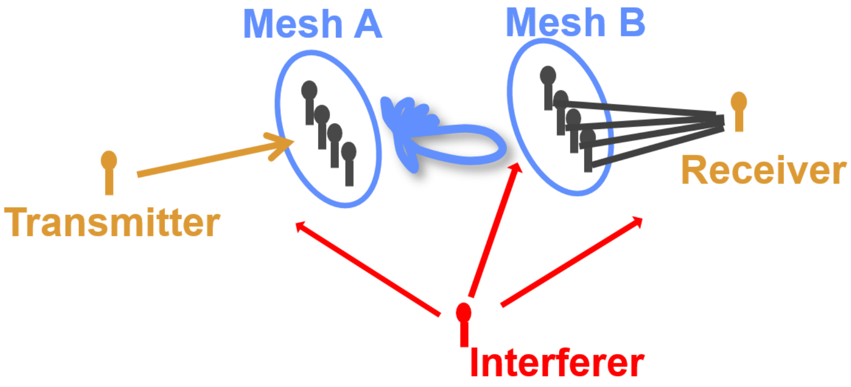


Figure 2: By utilizing relay meshes between two single antenna nodes, the desired SNR gain is realized. Interference is rejected at the Mesh A and Mesh B receiving stages through the application of spatial nulls in the transmit beamforming. The interference at the final receiver is overcome by the increased signal power. The first mesh receives the signal at its N antennas, and applies the appropriate beamforming pre-distortion filter to the signal before retransmission. The second mesh receives the retransmitted signal at its M antennas, where it applies its beamforming pre-distortion filter. Finally, the second mesh retransmits the transformed signal to the receiving node. The cycle is then repeated in the reverse direction.

set of pre-distortion filters. This Mesh-B-to-receiver beamforming implements what would typically be performed in a receive beamforming stage, including interference mitigation. However, traditional receive beamforming would require high-data-rate links between the Mesh B nodes and the final receiving node.

By implementing the function of the receive beamforming in the Mesh B to the receiving node link, the data rate requirements are significantly reduced at the expense of greater sensitivity to the Mesh B to receiving node channel estimation accuracy. Simultaneously, the link is duplicated in reverse by transmitting from the final node back to the starting node through the mesh network on different carrier frequencies. This approach is characterized by minimal sharing of information between meshs, and performing all stages of beamforming over-the-air.

In a traditional adaptive receive array processing problem, an array of antennas wired to a common processing unit observes a transmission and adjusts each stream of information using a set of filters that virtually adjust the array's beam pattern before summation. The discussed relay network does not exactly reflect this classic construction because each element of the receive array is disconnected from the point of summation, and in this relay network, the summation occurs over the air as a transmit beamformer. Phrased differently, the final receiving node is outfitted with a virtual antenna array and the transmission propagates through a virtual channel that contains the combined effects of the two stage transmission. Using a distributed network of elements cooperatively beamforming to relay signals to a destination allows us to realize robust links in clusters of reduced size, weight and power RF systems. Coherent beamforming by using a distributed array of antennas allows us to achieve significant power gains, which means we can close the same link with less transmission power. For a fixed average power, coherent transmit beamforming can improve signal of interest power over a single element relay by N^2, where N is the number of elements in the array. One factor of N comes from the increase in transmit power, and the other N is attributed to coherent combining. Additionally, beamforming allows the possibility to mitigate in-band interference. Accomplishing this through distributed elements permits simpler construction of the platforms, portability, flexibility in network configuration, and mobility.

I study an intelligent bent pipe space-time beamforming relay mesh network, depicted in the mesh of Figure 1, that receives some source signal from an origin transmitter along with jamming and noise, then applies a predistortion filter, and finally transmits to a terminating receiver. In a traditional receive beamforming scenario, data is received first at each element of an antenna array that is wired to a central processor. Postprocessing those data streams implements a virtual antenna pattern that ideally emphasizes a signal of interest but nulls interference. In a distributed network, array elements are physically disconnected, so signal combining can only be achieved by transmitting to a receiver. I predistort the relay signals, with an adaptive filter, such that the effects of receive beamforming are achieved at the terminating receiver despite it not being physically outfitted with an antenna array. From this perspective, traditional techniques, with minor modifications, can be used to construct an adaptive beamformer. However, the distributed nature of the system introduces immense practical challenges.

The challenges of successfully deploying distributed coherent beamforming include, but are not limited to, time-frequency synchronization, data starvation, data staleness, and computational load of beamformer construction. All of these factors intermingle together in complex ways that ultimately affect beamforming performance. Most fundamental to any distributed coherent system is time-frequency synchronization. I require carrier frequency alignment to be within a small fraction of a hertz across the locally-coherent mesh for robust operation. I employ data driven methods to construct the beamformer. The distributed nature of the system introduces heavy restrictions on how much observation data can be exchanged. In a practical system, the data backhaul must incur only a small cost of the operating resources, while maintaining sufficiently fast exchange to reduce data staleness. Lastly, processing must operate as fast as possible to provide robustness to the system.

The typical class of beamformer construction methods adjust the array beam pattern to point the main lobe toward an optimal direction relative to element positions and optionally minimize energy from undesired directions. Minimum variance distortionless response (MVDR) and minimum power distortionless response (MPDR) are examples of such techniques. Accurate node position information is required to achieve optimal performance. The sensitivity of these techniques to positioning accuracy is increased by the use of mobile, disconnected elements.

My methodology does not require explicit knowledge of the environment, rather, I exploit knowledge of the waveforms to estimate the required characteristics, which I will refer to as "waveform-exploitation". I construct an adaptive wideband beamformer using an minimum mean square error (MMSE) objective. I instead deploy known waveforms and observe the perturbations they undergo while passing through communications channel. Information about the precise relative positions of each receive and transmit element is embedded within the observations and implicitly used during construction.

All of these methods utilize a spatial, optionally spatiotemporal, covariance matrix to inform the placement of nulls in the beam pattern to reject interference. The propagation characteristics of interference sources are contained within this quantity. Thus data exchange of snapshots, used to estimate this quantity, among mesh elements is necessary to achieve interference rejection. Techniques that maximize the signal to interference plus noise ratio (SINR), including MVDR, rely upon an interference plus noise covariance, which may be an impractical quantity to obtain. Free-running communications signaling would not be permitted using these strategies. The MMSE and MPDR methods use covariance matrices that can include structure attributed to the signal of interest, but incur some risk of self-nulling. I employ the MMSE method to avoid sensitivities toward array element position knowledge and to more practically achieve interference rejection capabilities, as well as adaptations to alleviate the risk of self-nulling.

1.1 Prior Work in Distributed Coherent Beamforming

In this section, I will discuss prior research and efforts in both distributed mesh beamforming and in software defined radio networks. There are many different approaches to utilizing multiple elements in an RF system. However, I will focus on recent research in the sub-field of distributed beamforming and coherence.

Distributed coherent systems combine distributed elements via synchronization or other reconstruction techniques to achieve performance improvements. Distributed mesh beamforming refers to using distributed elements similarly to a classical antenna array. This is done by applying adaptive weightings to each of the distributed elements.

There are several different regimes in which distributed coherence operates. For example, radio astronomy systems with distributed receivers, commonly rebuild coherence through the use of extensive post processing, reducing requirements on receiver synchronization. However, positioning and timing applications require absolute coherence, having both cycle and phase synchronized [18]. The presented distributed beamforming system falls somewhere in between, requiring the correct phase, but the overall system synchronization can tolerate a few cycle slips. The design in [18] enables phase-accurate distributed coherence simultaneously with communications. This provides a basis for the presented distributed beamforming algorithm.

The feasibility of this class of systems is established in [35], including cases with higher phase variance. There is a tradeoff established between poor synchronization and the maximum gain that can be achieved. In a following paper, further challenges are discussed and progress in lab demonstrations of such systems is shown [30].

Another approach presented in [7] achieves the desired SNR gain through combining coding decisions, and shared hard decisions. However, this approach does not consider robustness to interference.

In [40], the feasibility of the final transmit beamforming stage is established, with a set of low size, weight, power, and cost radios performing transmit beamforming to substantially increase range and robust communications to a distant receiver.

In [4], the signal-to-interference-plus-noise ratio (SINR) loss associated with performing the channel estimation simultaneously with the data transmission is characterized for distributed transmit beamforming.

In [3], an algorithm that enables timing synchronization to approach fundamental bounds is presented. It also discusses the enablement of distributed coherent beamforming techniques.

Most recently, researchers have implemented practical systems on common off the shelf (COTS) hardware that achieve some of the goals and demonstrate feasibility. In [34], a simple version using 3 distributed nodes achieve frequency synchronization through limited feedback from the receiver fed into an Extended Kalman Filter (EKF). Beamforming is achieved through 1 bit feedback used to tune phase. This system resulted in SNR gain approximately 1.5 dB away from the N^2 (where N is the number of mesh antenna elements) bound on SNR improvement. However, it required a long convergence time.

Similar results are achieved in [36, 31], with a key difference being that a reference LO signal was transmitted over the air to the nodes in the virtual array and used for synchronization.

Time synchronization was explored and demonstrated over the air in [46] again using COTS radios. They brought their systems within about 5 Hz and less than $\frac{1}{16}$ of a sample residual timing offset, sufficient to achieve SNR gain.

In [14, 13], these SNR gain focused systems are extended yet again into mobile platforms. The beam is shaped using a controlled guide transceiver to assist in phase adjustment feedback. Frequency synchronization is achieved through feedback from the destination receiver. Some bounds and distributions of beamformer phase estimates are explored.

1.2 Contributions

In this report, I make the following contributions in the field of distributed coherent communications:

1. Discuss distributed coherent mesh beamforming techniques and solutions,
2. Develop a space-time distributed beamformer that mitigates interference,
3. Discuss performance considerations, fundamental bounds and practical challenges,
4. Present solutions and approaches to address practical challenges and enable implementation a real-world system,
5. Develop a data exchange and real-time distributed coherent processing platform,
6. Develop a real-time distributed frequency synchronization subsystem,
7. Experimentally demonstrate real-time synchronization and distributed coherent beamforming, including interference mitigation with 8 elements.

In addition, Dr. Daniel Bliss, Dr. Owen Ma, Dr. Andrew Herschfelt and Isabella Lenz were key contributors to the distributed coherent beamforming research and demonstrations.

1.3 Notation

In this document, I will use the following notation. Let $(\cdot)^\dagger$ denote the hermitian conjugate, let $(\cdot)^T$ denote the transpose, and let $(\cdot)^*$ denote the complex conjugate. I use the underline $\underline{\cdot}$ to indicate a row vector and $\| \cdot \|$ to indicate the L2 norm. Let $\odot$ be the Hadamard (element-wise) vector or matrix product. Let $(\cdot)^{\circ -1}$ denote the Hadamard (element-wise) inverse of a vector or matrix. Let $(\cdot) \oslash (\cdot)$ denote the Hadamard (element-wise) division of two vectors or matrices. Let the form $1 \rightarrow x \rightarrow \cdots \rightarrow 1$ denote a distributed beamforming system configuration, where x is the number of antennas in that stage of mesh. For example, a one-stage distributed coherent mesh beamforming system with a single relay stage and N mesh nodes in that relay would be denoted as $1 \rightarrow N \rightarrow 1$. Accordingly, $1 \rightarrow N \rightarrow M \rightarrow 1$ refers to a two-stage distributed coherent mesh beamforming system with two relay stages with N mesh nodes in the first and M in the second.

Chapter 2

BACKGROUND

Multiple antenna systems are positioned to take advantage of spatial diversity through the use of a multiplicity of transmit and receive elements. Spatial diversity arises from the multiplicity of paths between the transmitting and receiving systems and can be utilized to improve links over an equivalent single input, single output (SISO) system. By having multiple antennas at both the transmit and receiving sides, a system can gain robustness, increased data rate, or resistance to interference [45]. Beamforming techniques can be utilized to realize coherent combining of desired signals, by considering the set of transmit antennas as an array.

There are many approaches and methodologies to exploiting the multiplicity of elements and channel paths. Some common ones will be mentioned quickly now, and the most relevant will be discussed briefly in a following section.

Space-Time coding approaches present one path to leverage multiple antenna systems. A space-time trellis code distributes a traditional trellis code across multiple antennas and time slots, thus granting both diversity and coding gain. A space-time block code distributes a block of data across antennas and time slots, but only produces the spatial diversity gain.

Another approach is to perform so-called "blind" beamforming, where weights on an antenna array are estimated from collected environmental data. Thus, there is no need to know any parameters of the transmitter, receiver, or interference positioning. The opposite approach is "informed" beamforming (or nonblind) where the element locations are known. This is seen in phased arrays such as an Active Electronically Scanned Array (AESA), which is scanned across at various angles.

There are other types of diversity, such as time, frequency or cooperative diversity that can be exploited by communications systems to achieve similar gains [6].

2.1 MIMO Channels

multiple-input, multiple-output (MIMO) channels arise from the use of multiple transmit and receive elements. The often discussed single-input multiple-output (SIMO) and multiple-input single-output (MISO) channels are simply degenerate versions of the MIMO channel, where one of the dimensions is reduced to 1. The SIMO channel with a single transmitter with multiple receivers, and the MISO with multiple transmitters and a single receiver.

A signal is considered to be narrowband when the channel between each of the transmit and receive elements can be characterized by a single magnitude and phase shift. This holds when the bandwidth B of the signal is small compared to the inverse of the channel delay spread (Δt). It is often referred to as flat-fading, since the same complex attenuation can be used across frequencies, as opposed to a frequency-selective channel, where the complex attenuation varies with frequency. The elements in the flat-fading channel matrix $\mathbf{H} \in \mathbb{C}^{n_r \times n_t}$ contain the complex attenuation from each transmitter to each receiver, where n_r is the number of receivers used and n_t the number of transmitters.

A received signal $z(t) \in \mathbb{C}^{n_r \times 1}$ is written as

$$\mathbf{z}(t) = \mathbf{H}\,\mathbf{s}(t) + \mathbf{n}(t). \tag{2.1}$$

Since processing is performed with discrete numbers of samples, we rewrite for n_s

samples as

$$\mathbf{Z} = \mathbf{H}\,\mathbf{S} + \mathbf{N}, \tag{2.2}$$

where the received signal is given by $\mathbf{Z} \in \mathbb{C}^{n_r \times n_s}$ and the transmitted signal by $\mathbf{S} \in \mathbb{C}^{n_t \times n_s}$. The noise here includes external interference and is given by $\mathbf{N} \in \mathbb{C}^{n_r \times n_s}$. This implies that the channel is static over the time period spanned by the n_s samples. This can all be extended to a wideband (dispersive channel) by instead of using single complex numbers to represent each element to element channel, instead representing each element to element channel as a filter of appropriate length.

2.1.1 Space-Time MIMO Channels

The previously discussed MIMO channels are narrowband, or flat-fading models. They can be quickly extended to either a time-varying (wideband) model ($\mathbf{H}(l)$) or frequency-selective formulation ($\mathbf{H}(f)$), which are equivalent modulo a transformation in analysis domain. The previously single tap filters are now filters of some length L related to the channel coherence time. The received signal can then be written as $\mathbf{z}[n] = \sum_{l=0}^{L} \mathbf{H}[l]\mathbf{s}[n-l] + \mathbf{n}[n]$. It can sometimes be convenient to convert this discrete convolution into a single matrix multiplication. This can be achieved by rewriting the time-varying collection of channel responses into a single channel matrix with the following construction:

$$\mathbf{H}_{augmented} = \begin{bmatrix} \mathbf{H}[0] & \mathbf{H}[1] & \dots & \mathbf{H}[L] & \mathbf{0} & \dots & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{H}[0] & \mathbf{H}[1] & \dots & \mathbf{H}[L] & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \ddots & \ddots & \dots & \ddots & \ddots & \dots & \vdots \\ \mathbf{0} & \dots & \dots & \dots & \mathbf{H}[0] & \dots & \dots & \mathbf{H}[L] \end{bmatrix} \tag{2.3}$$

We can then conveniently reuse the previous received signal formulation, now for wideband channels and discretely sampled $z[t] \in \mathbb{C}^{n_r \times 1}$ being written as

$$\mathbf{z}[n] = \mathbf{H}_{augmented}\,\mathbf{s}[n] + \mathbf{n}[n]. \tag{2.4}$$

2.2 Beamforming

An array of antenna elements can be used to "shape" a received or transmitted signal, by shifting received signals to coherently combine at a single point. This then can improve the SNR of a received signal, and/or be used to mitigate interference. It is assumed that the delay spread introduced by the channel across the antenna elements will be resolvable. Practically, this is not always true, complicating the problem. It is also assumed that there is no scattering.

It is assumed that there is a signal, or multiple signals located in some region of a space-time field. Noise and possibly interference are also located in some region of the same space-time field. In the systems and problems of interest, these two regions have at least some overlap. Typically, the field is spatially filtered, so that a signal from a particular angle, or set of angles, is strengthened by constructive interference, and noise and interference from other angles are rejected through destructive interference.

Array design is determined by two aspects, the first being geometry. The geometry sets basic constraints on performance and behavior. For example, a linear array can only resolve one angular component.

The second is the design of the complex weights of the data at each receiver. The choice of these weights determines the spatial filtering characteristics of the array for a given geometry. These weights can also be chosen in post-processing to achieve different effects on the signal.

Take a plane wave incident on the array. The characteristic direction information of the plane wave can be found within the wavevector, $\mathbf{k} \in \mathbb{R}^{3\times 1}$. The wave vector points along the direction of propagation and has magnitude $||\mathbf{k}|| = \frac{2\pi}{\lambda}$. The complex representation of the propagated plane wave as a function of time and position (in other words, the solution of the wave equation) can be written as

$$\psi(\mathbf{x}, t) = ae^{-i\omega t + i\mathbf{k}\cdot\mathbf{x}} \tag{2.5}$$

The complex amplitude attenuation is given by $a \in \mathbb{C}$, and the phase of the attenuation is defined by the position and phase of the source.

Under narrowband conditions, the received signal at the array is then an attenuated and delayed version of the transmitted signal,

$$z_m(t) \approx e^{i\mathbf{k}^T\mathbf{x}_m}\, s(t - \tau_0), \tag{2.6}$$

where $\mathbf{x} \in \mathbb{R}^{3\times 1}$ is the distance from some origin.

The $e^{i\mathbf{k}\cdot\mathbf{x}_m}$ term can be understood geometrically by recognizing that the phase difference at each of the antennas is the result of the relatively time delay between the plane wave arriving at each of the antennas. This delay is proportional to the relative position of the antennas along the direction of the wavevector. This delay relative to the origin for each antenna can be written as

$$\Delta t_m = \frac{\frac{\mathbf{k}}{||\mathbf{k}||} \cdot \mathbf{x}_m}{c} \tag{2.7}$$

A complex steering vector is developed that when applied to the signal to be transmitted or its received signal, will create the desired beamforming effect. This steering vector is dependent upon the angles at which the received signal is incident upon the array. This receive beamforming $\mathbf{w} \in \mathbb{C}^{n_r x 1}$ contains complex coefficients that modify the phases and amplitudes of the n_r signals received by the array. It then sums them after the applied shifts, to produce the single received data stream. This beamformer is the construction of the spatial filtering discussed earlier. There are a variety of ways to construct this beamforming, from either physical parameters, statistical parameters or simply to sweep a directed beam across the array.

Adaptive beamforming is one such method, with the concept being to sense the environment, and adjust the beamformer to improve the system performance. Flexibility is typically limited to just estimating parameters at the receive side rather than adapting full systems such as in cognitive radios. Typically the system design goal is to increase the gain with regard to the signal of interest and reject interference. Consider the signal $\mathbf{S} \in \mathbb{C}^{n_t \times n_s}$ at complex baseband, which will be known at the receiver, and the received signal as $\mathbf{Z} \in \mathbb{C}^{n_r \times n_s}$. The signal $\mathbf{S}$ is commonly referred to as the training sequence or pilot signal. Let $\mathbf{S}$ also be the normalized version of the transmitted signal $\mathbf{X} \in \mathbb{C}^{n_t \times n_s}$, so that $\mathbf{S}$ is given as $\mathbf{S} = \sqrt{\frac{P_0}{n_t}}\mathbf{X}$, where P_0 is the total noise-normalized power.

2.2.1 MMSE Beamforming

By considering the beamforming matrix $\mathbf{W} \in \mathbb{C}^{n_r \times n_t}$ as a linear operator, an estimate of the transmitted signal can be written as $\hat{\mathbf{X}} = \mathbf{W}^{\dagger}\mathbf{Z}$. One common method is to perform minimum mean-squared error (MMSE) spatial processing. First, consider the error matrix $\mathbf{E} \in \mathbb{C}^{n_t \times n_s}$ given by $\mathbf{E} = \mathbf{W}^{\dagger}\mathbf{Z} - \mathbf{X}$. The mean-squared error between the beamformer output and the transmitted signal is given as $E[||\mathbf{E}||_F^2] = E[||\mathbf{W}^{\dagger}\mathbf{Z} - \mathbf{X}||_F^2] = \mathrm{tr}\{E[(\mathbf{W}^{\dagger}\mathbf{Z} - \mathbf{X})(\mathbf{W}^{\dagger}\mathbf{Z} - \mathbf{X})^{\dagger}]\}$.

Next, to minimize the above mean-squared error, the derivative with respect to an unknown parameter α of the beamformer $\mathbf{W}$ will be set equal to 0,

$$\begin{aligned}
\frac{\partial}{\partial\alpha}E[||\mathbf{E}||_F^2] &= 0 && (2.8)\\
&= \frac{\partial}{\partial\alpha}\text{tr}\{E[(\mathbf{W}^\dagger\mathbf{Z}-\mathbf{X})(\mathbf{W}^\dagger\mathbf{Z}-\mathbf{X})^\dagger]\} && (2.9)\\
&= \text{tr}\{E[\frac{\partial}{\partial\alpha}(\mathbf{W}^\dagger\mathbf{Z}-\mathbf{X})(\mathbf{W}^\dagger\mathbf{Z}-\mathbf{X})^\dagger]\} && (2.10)\\
&= \text{tr}\{E[(\frac{\partial}{\partial\alpha}\mathbf{W})\,\mathbf{Z}(\mathbf{W}^\dagger\mathbf{Z}-\mathbf{X})^\dagger]\} + h.c. && (2.11)\\
&= \text{tr}\{(E[\mathbf{Z}\mathbf{Z}^\dagger]\mathbf{W} - E[\mathbf{Z}\mathbf{X}^\dagger])\frac{\partial}{\partial\alpha}\mathbf{W}\} + h.c. && (2.12)
\end{aligned}$$

It becomes evident that this equation is only satisfied if for all beamformers $\mathbf{W}$ the argument of the trace is zero. Thus, $E[\mathbf{Z}\mathbf{Z}^\dagger]\mathbf{W} - E[\mathbf{Z}\mathbf{X}^\dagger] = 0$, yielding a beamforming matrix $\mathbf{W} = E[\mathbf{Z}\mathbf{Z}^\dagger]^{-1}E[\mathbf{Z}\mathbf{X}^\dagger]$.

Practically, these expectations can't be evaluated exactly, thus they are typically approximated over some number of samples n_s. If $n_s \ll n_r$ and $n_s \ll n_t$, then the expectations are well approximated as

$$\begin{aligned}
E[\mathbf{Z}\mathbf{Z}^\dagger] &\approx \mathbf{Z}\mathbf{Z}^\dagger && (2.13)\\
E[\mathbf{Z}\mathbf{X}^\dagger] &\approx \mathbf{Z}\mathbf{X}^\dagger. && (2.14)
\end{aligned}$$

Combining the relationships with the previously calculated beamformer, the set of approximate MMSE beamformers are found in the columns of $\mathbf{W}$ given by $\mathbf{W} \approx (\mathbf{Z}\mathbf{Z}^\dagger)^{-1}\mathbf{Z}\mathbf{X}^\dagger$. This solution also happens to be the least-squared error solution. This is just one example of a method to determine a beamformer for use with an array. It is also an example of a so-called "blind" beamformer, where no knowledge of element positions, or other external input knowledge is used, rather just training samples and environmental sensing. This solution is also readily extendable to dispersive channel (wideband) formulations.

2.3 Mesh Relay Systems

There exist a multiplicity of approaches to leverage distributed relays for performance improvement. I will now briefly discuss a few of the most relevant.

2.3.1 Amplify and Forward

Amplify-and-Forward (AF) approaches utilize a relay to receive a message from the transmitter, amplify the message, and retransmit towards the destination with greater power than previously realized at the receiver. This can yield incoherent combining gains. There are many protocols put forward to leverage Amplify-And-Forward relay principles. They typically leverage time slots shared between relays and the source, such as the non-orthogonal AF (NAF) protocol which implements a half-duplex system [23]. This approach is characterized by two phases. In the first, the source sends the first signal to both the receiver and relay system. In the second phase, the source sends a second signal to the receiver, and the relay amplify-and-forwards the signal received in the first phase. This results in a multiple copies of the received signal at the receiver, displaced in time, which can then be leveraged to achieve diversity gain. Full-Duplex AF approaches also exist, such as the linear relaying or dual-hop AF approaches discussed in [23]. However, these approaches result in self-interference challenges.

2.3.2 Decode and Forward

Decode-and-Forward techniques employ relay nodes that can decode the message, re-encode and modulate and retransmit. This improves the retransmitted SNR as it no longer retransmits noise. However, performance is now limited by the ability of the relay to successfully decode, and if there are any errors, they will be propogated and the message lost or degraded at the destination. Decode-and-Forward techniques can also take advantage of distributed codes that make better use of the distributed relay resources [44].

2.3.3 Space Time Block Codes and Other Approaches

Space-Time Codes can be applied across the distributed relay to realize diversity gains. The signal from the transmitter is received at the relay, and each element in the relay applies a portion of the space-time block code [44]. Similarly, there are systems known as barrage relays, that apply a random time and/or phase shifts at each relay element. This results in multipath-like diversity at the receiver, allowing a well designed receiver to recover the shifts and realize diversity gain.

2.4 Software Defined Radio Systems and WISCANet

Commercial radio systems are often implemented on FPGAs or ASICs. These platforms offer high efficiency and performance, but are time-consuming to develop for, require specialized hardware knowledge, and typically have substantial non-recurring engineering costs. These solutions, while favorable for mass-produced technologies, are neither viable nor accessible for conducting research on novel RF techniques. Commerical SDR platforms have made advanced over-the-air RF research more accessible and affordable, but common software environments and tools often don't scale well to network applications or high-performance techniques. While there are many excellent open-source tools to support these SDRs, these solutions are typically insufficient for intermediate to advanced RF research.

WISCANet addresses these challenges by providing a scalable control architecture for commercial SDR products. WISCANet allows users to quickly develop on networks of SDRs without specialized hardware knowledge, and test over-the-air applications with minimal hardware configuration. By abstracting the hardware controls away from the user, WISCANet allows a non-expert user to deploy an OTA application by simply defining a baseband processing chain in a high level language. This technology reduces transition time between system design and OTA deployment, accelerates debugging and validation processes, and makes OTA experimentation more accessible to users that are not radio hardware experts. WISCANet is a hardware-agnostic control software that automatically configures and controls a software-defined radio (SDR) network [20, 21]. I leverage the fundamentals of WISCANet for the over-the-air system demonstration in this work. WISCANet leverages the Ettus USRP family of software-defined radios.

2.4.1 Ettus USRP and UHD

The Universal Software Radio Peripheral (USRP) is a product series developed by Ettus Research [11]. This product line consists of embedded system-on-chip (SOC) radios (E series); Universal Serial Bus (USB) radios (B series); Ethernet radios (N series); and wide bandwidth radios (X series). Each model targets different frequency ranges, sampling rates, and processing requirements [22]. The USRP Hardware Driver (UHD) is an open-source C++ library and API for systems leveraging the USRP, which can be used to build radio applications without understanding the full details of the underlying hardware [43].

Chapter 3

DISTRIBUTED MESH BEAMFORMING

The distributed coherent mesh relay network is comprised of radios that are designated either as initiating transceivers (the transmitter and receiver on either end of the relay), or mesh nodes. All radios, both initiating transceivers and mesh nodes, are single antenna systems. Initiating transceiver nodes are radios through which users can interface with the network. The system propagates the signal of interest (also referred to as the payload) through the transmitter. The initial transmitter potentially augments that signal to facilitate propagation throughout the network. The initial transmitter node forwards this modified signal to a collection of distributed, locally-coherent relay nodes. The relay nodes pre-distort and augment the received signal. Then, the relay nodes cooperate to forward the pre-distorted and augmented signal to either another mesh or a receiving node. If another mesh is the target set of receivers, then the process repeats. Otherwise, the receiving node effectively removes any remaining augmentations to isolate the signal of interest and outputs it for further decoding or use. The radios and algorithms that comprise this network are designed to be low size, weight, power, and cost, while being able to overcome long transmit distances and interference sources. A robust communications link can be developed given the constraints by leveraging beamforming to introduce significant receive power gain while mitigating interference.

The network acts as a “bent-pipe" by treating each mesh as if they were antenna arrays, where each node is one element. The unique aspect of this approach is that

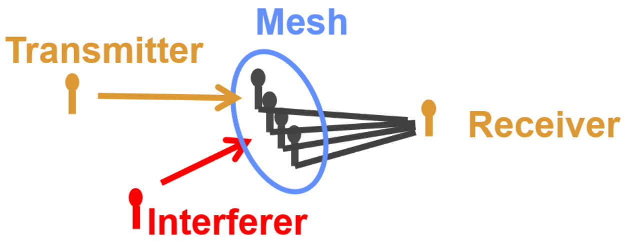


Figure 3: Single mesh topology: transmitting node to distributed mesh to final receiving node in the presence of an interferer.

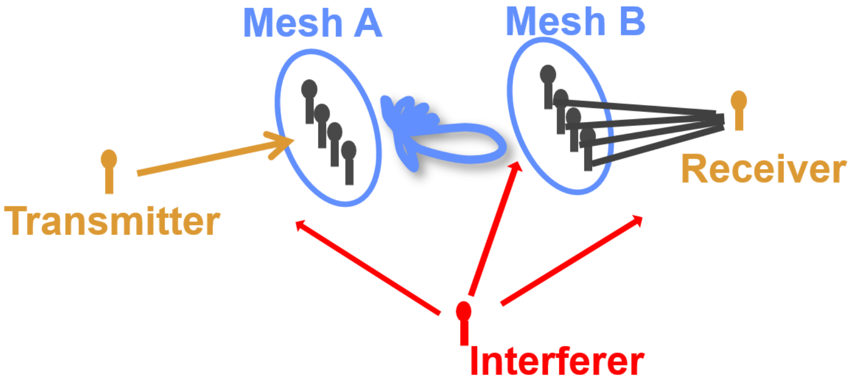


Figure 4: Dual mesh topology: transmitter node to distributed mesh to another distributed mesh to final receiving node in the presence of interferers at each relay stage.

predistortion filters are applied to the the relay signals such that the signal of interest coherently combines and the interference is rejected only upon arrival at the receiving node antenna. In other words, receive beamforming is accomplished through transmit beamforming. Thus, any receiver not co-located with the receiving node will not see the increased signal of interest power or rejected interference, and, in fact, will see incoherent combining of the interference.

Bidirectional communications can be supported through either time or frequency-division duplexing, where a single antenna is devoted to each direction. The algorithm used is the same in both directions, but must have two independent realizations of the

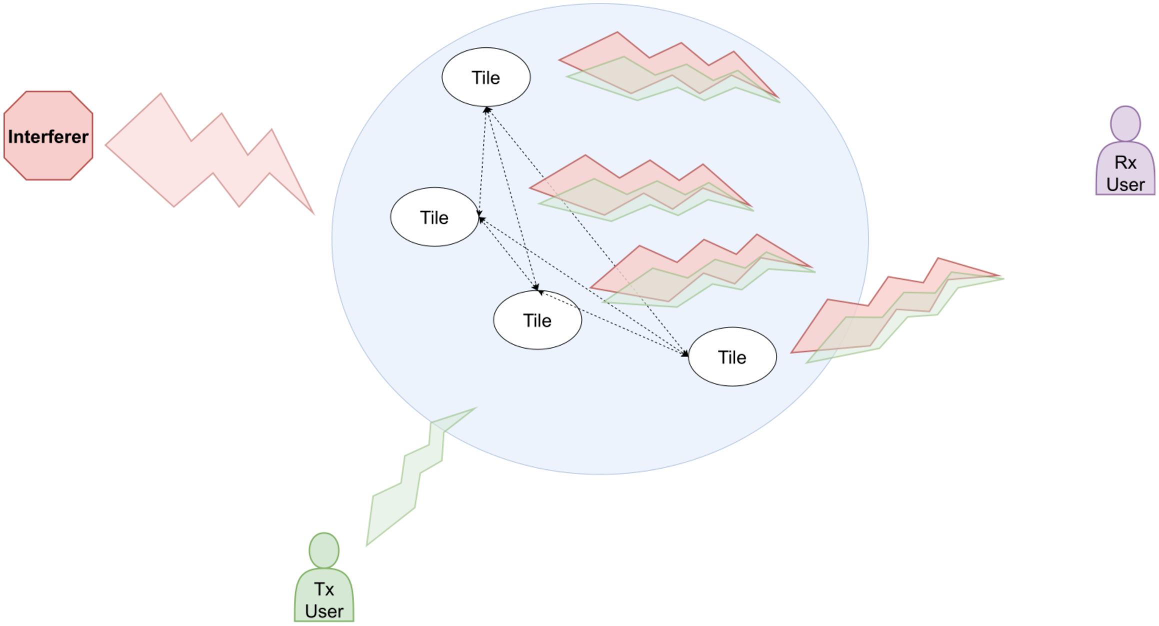


Figure 5: Interference received at the mesh is retransmitted along with the signal of interest, and combining for SNR gain and interference migitation occurs at the receiving user.

beamformer because, with the introduction of interference, reciprocity is not observed. Optionally, to support a bidirectional communications system, underlay waveforms could be leveraged at each stage to supply training information and some information feedback. The only augmentation an initaiting transceiver performs is the addition of a low-power underlay and potentially a center frequency translation from the input signal, acting effectively as a smart antenna. Mesh nodes would then process their incoming streaming information at baseband by filtering out most of the underlay and applying the required predistortion filter to implement beamforming. Mesh nodes then prepare the retransmission by adding the appropriate additional underlay and center frequency shift. This approach offers waveform agnosticism and some level of computational simplicity, which allows nodes to be flexible and attritable. There are no operations that decode the signal of interest and the computations for building the beamformer are “generic" further enabling the consideration of the system as a “smart" antenna or “bent pipe".

Some system and network infrastructure is necessary to support such operations. The relay system requires the support of an intramesh data exchange network to supply the mesh relay nodes the information needed to compute the beamformers. Additionally, a a time-frequency synchronization background process is required to facilitate distributed coherent beamforming.

In the remainder of this chapter, I discuss a compound channel extension of the traditional spatial and spatio-temporal MMSE beamforming problems. I first discuss the spatial (flat fading channel, narrowband) model to introduce the nuances. Using the space-only models enables quicker understanding of the beamformer's construction and its performance bounds. I then introduce the spatio-temporal (dispersive channel, wideband) form. In the following section, I discuss adjustments and adaptations to the space-time beamformer, so that this system is more practical to implement. Lastly, I discuss methods for measuring the performance of such a system.

3.1 Adaptive Spatial Relay Beamformer ($1 \rightarrow N \rightarrow 1$)

The traditional spatial MMSE receiver beamformer is described here, and then adapted. Note that for this system, the "receive" beamforming is implemented in the mesh to the final receiving node by using a novel transmit beamforming approach, as depicted in Figures 4 and 3. Typically, an array of receive antennas would be wired to a central processing system [45].

For a receive beamforming system processing N_{m_A} receive antennas and n_s received training samples $\underline{\mathbf{s}}_{tr} \in \mathbb{C}^{1 \times n_s}$, propogation channels to each element $\mathbf{h} \in \mathbb{C}^{N_{m_A} \times 1}$, with

$\mathbf{J}$ being the interference channel(s), $\mathbf{T}$ the transmitted interference signal(s), and $\mathbf{N}$ the received complex noise at each element, the narrowband beamformer $\mathbf{w}_{rx} \in \mathbb{C}^{N_{m_A} \times 1}$ is given by

$$\begin{aligned}
\mathbf{w}_{rx} &= \mathbf{C}^{-1}\,\boldsymbol{\rho} \qquad &(3.1)\\
\mathbf{C} &= \frac{1}{n_s}\,E[(\mathbf{h}\,\underline{\mathbf{s}}_{tr} + \mathbf{J}\,\mathbf{T} + \mathbf{N})\,(\mathbf{h}\,\underline{\mathbf{s}}_{tr} + \mathbf{J}\,\mathbf{T} + \mathbf{N})^{\dagger}] \qquad &(3.2)\\
\boldsymbol{\rho} &= \frac{1}{n_s}\,E[\mathbf{Z}_{tr}\,\underline{\mathbf{s}}_{tr}^{\dagger}] \qquad &(3.3)
\end{aligned}$$

where $\mathbf{J}\,\mathbf{T} + \mathbf{N}$ indicates the interference and noise contributions. Practically, the expectations over all realizations cannot be taken, so estimates must be made by using the training samples like so:

$$\begin{aligned}
\mathbf{w}_{rx} &= \mathbf{C}^{-1}\,\boldsymbol{\rho} \qquad &(3.4)\\
&\approx (\mathbf{Z}_{tr}\,\mathbf{Z}_{tr}^{\dagger})^{-1}\,\mathbf{Z}_{tr}\,\underline{\mathbf{s}}_{tr}^{\dagger} \qquad &(3.5)
\end{aligned}$$

The transmitted data is then recovered from the received data by applying the beamformer like so:

$$\hat{\underline{\mathbf{s}}}_{data} = \mathbf{w}_{rx}^{\dagger}\,\mathbf{Z}_{data}\,, \tag{3.6}$$

It is worth noting that the covariance estimate $\mathbf{C}$ may or may not include the signal of interest. The resulting performance is unchanged and the beamformer values dare the same up to a known shift in scale [45].

For the distributed system, there are no wired connections between the mesh and the final receiving node. Consequently, this receive beamforming is implemented through the use of transmit beamforming.

The beamformer used to perform transmit beamforming to the receiver node must be adjusted with the appropriate complex gains such that

$$\mathbf{w}_{rx} = \mathbf{w} \odot \mathbf{h}_B^* \tag{3.7}$$

$$\mathbf{w} = \mathbf{w}_{rx} \oslash \mathbf{h}_B^* \tag{3.8}$$

where $\mathbf{w} \in \mathbb{C}^{N_{m_A} \times 1}$ is the applied transmit beamformer and $\mathbf{h}_{m \to B} \in \mathbb{C}^{N_{m_A} \times 1}$ is the channel between the elements of the mesh and the final receiving node. The $\mathbf{w}_{rx}$ is the receive beamformer discussed previously. To correctly evaluate performance, the transmitted sum power across the antennae must be equal to the initially received sum power across the mesh antennae.

The distributed mesh beamforming system exploits knowledge of training waveforms to construct the beamformer. A known training waveform is added to the payload signal. A beamformer suitable for the payload signal can be constructed by observing how the environment affects the training data. This beamformer minimizes the mean square error between the training data and the reception at the terminal receiving node [19].

We have come to this solution somewhat intuitively by seeing the situation as a receiver beamformer with "noisy" and unequal length wires connecting each antenna that must be compensated for. We can also construct the beamformer from first principles by constructing an appropriate objective and optimizing. This will become important for the spatio-temporal version as the "noisy" wire compensation cannot be so neatly unfolded.

I define the signal model as follows. The initial transmitter node first transmits the training waveform $\underline{\mathbf{s}}$ to the N_{m_A} mesh nodes. The signal $\underline{\mathbf{s}}$ passes through the n^{th} path of channel A (the channels from transmitted to mesh), $\mathbf{h}_A$. Each node also receives an observation of a interfering signal $\underline{\mathbf{j}}$ that has passed through the n^{th} path of $\mathbf{h}_J$. Lastly, the mesh accumulates some complex noise $\underline{\mathbf{q}}_n$. The signal model for the reception at mesh node n is given by

$$\underline{\mathbf{z}}_n = h_{A,n}\underline{\mathbf{s}} + h_{J,n}\underline{\mathbf{j}} + \underline{\mathbf{q}}_n \,. \tag{3.9}$$

It is convenient to define and refer to the quantity

$$\mathbf{Z} = \begin{bmatrix} \underline{\mathbf{z}}_1 \\ \vdots \\ \underline{\mathbf{z}}_N \end{bmatrix} , \tag{3.10}$$

which compiles all observations. Each mesh node applies a predistortion transmit filter to the reception before retransmitting. In this case, the filter is purely spatial, so a simple amplitude and phase adjustment is applied. Mesh relay node n transmits

$$w_n^* \underline{\mathbf{z}}_n \,. \tag{3.11}$$

The n^{th} relay signal passes through the n^{th} path of channel B (the channels from mesh to receiver) $\underline{\mathbf{h}}_B$. The final target receiver node reception is modeled as

$$\begin{aligned} \underline{\mathbf{g}} &= \sum_{n=1}^{N} h_{B,n} w_n^* \left(h_{A,n}\underline{\mathbf{s}} + h_{J,n}\underline{\mathbf{j}} + \underline{\mathbf{q}}_n \right) + \underline{\mathbf{q}}_B && (3.12) \\ &= \sum_{n=1}^{N} h_{B,n} w_n^* \underline{\mathbf{z}}_n + \underline{\mathbf{q}}_B && (3.13) \\ &= \mathbf{w}^\dagger \mathrm{diag}(\underline{\mathbf{h}}_B)\mathbf{Z} + \underline{\mathbf{q}}_B && (3.14) \\ &= \mathbf{w}^\dagger \mathbf{Y} + \underline{\mathbf{q}}_B \,, && (3.15) \end{aligned}$$

where

$$\mathbf{Y} = \begin{bmatrix} \underline{\mathbf{y}}_1 \\ \vdots \\ \underline{\mathbf{y}}_N \end{bmatrix} = \begin{bmatrix} h_{B,1}\underline{\mathbf{z}}_1 \\ \vdots \\ h_{B,N}\underline{\mathbf{z}}_N \end{bmatrix} \,. \tag{3.16}$$

Matrix $\mathbf{Y}$ contains in its rows each node's re-transmitted signal having passed through the various paths of the second channel. This is effectively a hypothetical version of the completed re-transmission based on an estimate of channel B.

I now find a beamformer that minimizes the error between the final reception $\underline{\mathbf{y}}$ and $\underline{\mathbf{s}}$. Only $w_n^* \underline{\mathbf{z}}_n$ is transmitted, so $\mathbf{w}$ must account for $\underline{\mathbf{h}}_B$ ahead of time. The compound channel MMSE problem is expressed as

$$\min_{\mathbf{w}} \quad \|\mathbf{w}^\dagger \mathbf{Y} - \underline{\mathbf{s}}\|^2 \,. \tag{3.17}$$

The setup of this minimization problem is unconstrained, however a real system's hardware has a physical constraint on re-transmitted output power, which represents a path of future study. The original adaptive receive beamformer discussed occurs when $\underline{\mathbf{h}}_B = \underline{\mathbf{1}}$, which indicates that the final channels are perfectly wired to the same processing unit.

To solve the minimization, I expand the quadratic term in Equation 3.17, take the derivative while leveraging Wirtinger derivatives because $\mathbf{w}$ is complex, set the result to zero, and solve for $\mathbf{w}$ as follows:

$$\begin{aligned}
\left(\mathbf{w}^\dagger \mathbf{Y} - \underline{\mathbf{s}}\right)\left(\mathbf{w}^\dagger \mathbf{Y} - \underline{\mathbf{s}}\right)^\dagger &= \mathbf{w}^\dagger \mathbf{Y}\mathbf{Y}^\dagger \mathbf{w} - \underline{\mathbf{s}}\mathbf{Y}^\dagger \mathbf{w} - \mathbf{w}^\dagger \mathbf{Y}\underline{\mathbf{s}}^\dagger + \underline{\mathbf{s}}\underline{\mathbf{s}}^\dagger && (3.18)\\
0 &= \frac{\partial}{\partial \mathbf{w}^\dagger}\left[\mathbf{w}^\dagger \mathbf{Y}\mathbf{Y}^\dagger \mathbf{w} - \underline{\mathbf{s}}\mathbf{Y}^\dagger \mathbf{w} - \mathbf{w}^\dagger \mathbf{Y}\underline{\mathbf{s}}^\dagger + \underline{\mathbf{s}}\underline{\mathbf{s}}^\dagger\right] && (3.19)\\
&= \mathbf{Y}\mathbf{Y}^\dagger \mathbf{w} - \mathbf{Y}\underline{\mathbf{s}}^\dagger && (3.20)\\
\mathbf{Y}\mathbf{Y}^\dagger \mathbf{w} &= \mathbf{Y}\underline{\mathbf{s}}^\dagger && (3.21)\\
\mathbf{w} &= \left(\mathbf{Y}\mathbf{Y}^\dagger\right)^{-1} \mathbf{Y}\underline{\mathbf{s}}^\dagger && (3.22)\\
\mathbf{w} &= \mathbf{C}^{-1}\mathbf{r}\,. && (3.23)
\end{aligned}$$

Only in the spatial model, by rearranging some terms within the MMSE beamformer expression we find the following form:

$$\mathbf{w} = \Big((\mathrm{diag}(\underline{\mathbf{h}}_B)\mathbf{Z}) \, (\mathrm{diag}(\underline{\mathbf{h}}_B)\mathbf{Z})^{\dagger} \Big)^{-1} \mathrm{diag}(\underline{\mathbf{h}}_B)\mathbf{Z}\mathbf{s}^{\dagger} \tag{3.24}$$

$$= \mathrm{diag}(\underline{\mathbf{h}}_B^{*})^{-1} \, (\mathbf{Z}\mathbf{Z})^{-1} \, \mathbf{Z}\underline{\mathbf{s}}^{\dagger} \, . \tag{3.25}$$

This form confirms the intuitive approach we took to constructing the transmit beamformer previously, with each element of the typical receive beamformer being “backed off" by $h_{B,n}^{*}$.

The form $\mathbf{C}^{-1}\mathbf{r}$ is common to many adaptive beamformer constructions. The beamformer is constructed by multiplying the inverse data covariance matrix with the cross-correlation vector. Coherent combining properties are provided by the cross-correlation $\mathbf{r}$, which essentially quantifies the combined effects of channel B and channel A. The term $\mathbf{r} = \mathbf{Z}\mathbf{s}^{\dagger}$ differs from the least squares estimate of channel A by the normalization term, $\frac{1}{|\mathbf{s}|^2}$. Interference mitigation properties are provided by $\mathbf{C}^{-1}$. The matrix $\mathbf{C}$ is essentially comprised of an uncorrelated noise component and a sum of rank-1 components that correspond to each source in the environment. This matrix is the only term in the solution that contains information about the interference source's propagation characteristics. Information about the signal of interest is contained within $\mathbf{C}$ as well, which leads to concerns of self-nulling.

Without loss of generality, consider a scenario without interference and $\underline{\mathbf{h}}_B = \underline{\mathbf{1}}$. In the limit, the covariance matrix is given by

$$\mathbf{h}_A\mathbf{h}_A^{\dagger} + \sigma^2\mathbf{I} \, , \tag{3.26}$$

where I have assumed the noise power across nodes is equal with strength σ^2, and the cross-correlation converges to and the correlation converges to

$$\mathbf{h}_A|s|^2 \, . \tag{3.27}$$

Utilizing the Woodbury matrix identity and applying the resulting inverse to $\mathbf{r}$ reveals that the presence of $\mathbf{h}_A\mathbf{h}_A^\dagger$ within $\mathbf{C}$ only introduces a scaling factor on $\mathbf{w}$ granted $\mathbf{C}$ and $\mathbf{r}$ are well estimated. The beamformer is given by

$$\frac{1}{\sigma^2}\left(\mathbf{I} - \frac{1}{\sigma^2 + |\mathbf{h}_A|^2}\mathbf{h}_A\mathbf{h}_A^\dagger\right)\left(\mathbf{h}_A|s|^2\right) \ . \tag{3.28}$$

This result is the same as the SINR maximizing beamformer, differing by only the scaling factor. The scaling washes away as the signals inevitably need to be rescaled to maximize their occupancy of the digital-to-analog converter (DAC) dynamic range without clipping. Although the ceiling of performance is the same, there is still a risk of self-nulling. According to equation 3.28, inaccuracies in the estimate of $\mathbf{r}$ or the $\mathbf{h}_A\mathbf{h}_A^\dagger$ component of $\mathbf{C}$ causes performance degradation through self-nulling. A similar exercise can be performed including the interfering component to observe how inaccuracies in covariance estimation and thus a misrepresentation of the interference characteristics affects nulling performance.

Creation of the beamformer depends on having prior knowledge of channel B $\underline{\mathbf{h}}_B$, whereas the incoming channel to the mesh is implicitly estimated along with the interference properties. Practically, this quantity must be estimated, and possibly predicted and tracked to compensate for any delay in application of the estimated beamformer. The performance of the channel estimate scales directly with the number of samples used to integrate, assuming the SNR is a fixed quantity. Mesh relay nodes incorporate additional training sequences to the relay signal to sound the channel. The channel sounding waveform must not be processed by the beamforming filter. The receiver can then estimate all of the incoming channels from the mesh and provide this information back to the mesh for computation of the beamformer. Each mesh node must use an independent sequence $\underline{\mathbf{s}}$, for a CDMA style channel estimation approach, or the same sequence but separated in time (TDMA). The TDMA solution introduces issues with alignment of the channels in time across the mesh, increasing requirements on timing precision. The CDMA solution requires far more samples to yield an estimate of sufficient quality due to the interference of the other mesh elements as true synchronous CDMA is nigh impossible in this distributed configuration. One possible solution is the least squares channel estimator, where h is the complex-valued channel in question [6]:

$$\min_{h} \quad (\underline{\mathbf{y}} - h\underline{\mathbf{s}})(\underline{\mathbf{y}} - h\underline{\mathbf{s}})^{\dagger} \, . \tag{3.29}$$

The solution is found by taking the complex derivative with respect to h^*, setting the derivative to 0, and solving for h results in the following expression:

$$0 = \frac{\partial}{\partial h^{\dagger}} \left[\underline{\mathbf{y}}\underline{\mathbf{y}}^{\dagger} - \underline{\mathbf{y}}\underline{\mathbf{s}}^{\dagger} h^{\dagger} - h\underline{\mathbf{s}}\underline{\mathbf{y}}^{\dagger} + h\underline{\mathbf{s}}(h\underline{\mathbf{s}})^{\dagger} \right] \tag{3.30}$$

$$0 = -\underline{\mathbf{y}}\underline{\mathbf{s}}^{\dagger} + h\underline{\mathbf{s}}\underline{\mathbf{s}}^{\dagger} \tag{3.31}$$

$$\underline{\mathbf{y}}\underline{\mathbf{s}}^{\dagger} = h\underline{\mathbf{s}}\underline{\mathbf{s}}^{\dagger} \tag{3.32}$$

$$\hat{h} = \underline{\mathbf{y}}\underline{\mathbf{s}}^{\dagger}(\underline{\mathbf{s}}\underline{\mathbf{s}}^{\dagger})^{-1} \, . \tag{3.33}$$

3.2 Adaptive Space-Time Relay Beamformer ($1 \rightarrow N \rightarrow 1$)

To build a practical system, a beamforming filter that can handle fading (dispersive channels) must be constructed. Practically, as the elements in the mesh grow further and further apart, delay between each of them and the transmitter and receiver will vary. This must be compensated for. The minimum number of taps required can be determined by looking at the worst case delay spread: $n_{w_{min}} = \lceil 2\frac{D_{max}}{c} f_s \rceil$. D_{max} is the maximum pairwise distance between any two mesh elements, c the speed of light, and f_s the sampling rate required for the desired signal bandwidth. The filter length of the beamformer ($\mathbf{w}$) n_w ultimately is a design parameter that must be set long enough to handle a maximum expected delay spread, which could increase over the worst case from system geometry due to motion or channel scatterers. It is recommend to add a few additional taps to ensure robust operation and improve the quality of the final solution. I now construct the spatio-temporal version of the beamformer.

The mesh node n dispersive reception model is given by

$$\underline{\mathbf{z}}_n = \underline{\mathbf{h}}_{A,n} * \underline{\mathbf{s}} + \mathbf{h}_{J,n} * \underline{\mathbf{j}} + \underline{\mathbf{q}}_n \, , \tag{3.34}$$

where the $*$ operation denotes a convolution. Each channel is now represented by a complex finite impulse response (FIR) filter. Mesh relay node n applies a predistortion transmit filter of length n_w taps to the reception and transmits

$$\mathbf{w}_n^\dagger * \underline{\mathbf{z}}_n \, . \tag{3.35}$$

The n^{th} relay signal passes through the n^{th} path of channel B, $\underline{\mathbf{h}}_{B,n}$. The final reception at the target receiver is modeled as

$$\begin{aligned}
\underline{\mathbf{g}} &= \sum_{n=1}^{N} \underline{\mathbf{h}}_{B,n} * \mathbf{w}_n^\dagger * \underline{\mathbf{z}}_n + \underline{\mathbf{q}}_B && (3.36)\\
&= \sum_{n=1}^{N} \mathbf{w}_n^\dagger * \underline{\mathbf{y}}_n + \underline{\mathbf{q}}_B && (3.37)\\
&= \sum_{n=1}^{N} \mathbf{w}_n^\dagger \tilde{\mathbf{Y}}_n + \underline{\mathbf{q}}_B && (3.38)\\
&= \mathbf{w}^\dagger \tilde{\mathbf{Y}} + \underline{\mathbf{q}}_B \, , && (3.39)
\end{aligned}$$

where $\underline{\mathbf{y}}_n$ represents the n^{th} re-transmission through its corresponding path in channel B. Define the n^{th} re-transmission through its corresponding channel B as

$$\underline{\mathbf{y}}_n = \underline{\mathbf{h}}_{B,n} * \underline{\mathbf{z}}_n \, . \tag{3.40}$$

The optimization problem is of the same structure, with multiplications becoming convolutions with the introduction of temporal filters. To find the space-time relay beamformer, minimize

$$\left\| \sum_n (w_n * h_{\mathrm{B},n} * z_n)(t) - s(t) \right\|^2 \tag{3.41}$$

where n indexes the nodes in the mesh network.

This can be rewritten in vector-matrix form as:

$$\left\| \mathbf{w}^\dagger \mathbf{H}_{\mathrm{BST}} \mathbf{Z}_{\mathrm{ST}} - \underline{\mathbf{s}} \right\|^2 . \tag{3.42}$$

The derivation is the same as the spatial beamformer, with the exception $\mathbf{H}_{\mathrm{BST}}$ represents the convolution of channel B into $\mathbf{Z}_{\mathrm{ST}}$, instead of the previous single-tap channel, when forming the hypothetical reception matrix $\tilde{\mathbf{Y}} = \mathbf{H}_{\mathrm{BST}} \mathbf{Z}_{\mathrm{ST}}$. The solution is

$$\mathbf{w} = ((\mathbf{H}_{\mathrm{BST}} \mathbf{Z}_{\mathrm{ST}})(\mathbf{H}_{\mathrm{BST}} \mathbf{Z}_{\mathrm{ST}})^\dagger)^{-1} (\mathbf{H}_{\mathrm{BST}} \mathbf{Z}_{\mathrm{ST}}) \underline{\mathbf{s}}^\dagger . \tag{3.43}$$

Restructuring $\mathbf{w}$ yields the impulse responses of the beamforming pre-distortion filters. This form reduces to the spatial only case if the number of taps used is 1. The matrices are set up to perform the convolutions through matrix multiplication, and also include the temporal augmentations.

$$\mathbf{H}_{\mathrm{BST}} = \begin{bmatrix} \mathbf{H}_{\mathrm{B},1} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_{\mathrm{B},2} & & \\ \vdots & & \ddots & \\ \mathbf{0} & & & \mathbf{H}_{\mathrm{B},n_r} \end{bmatrix} \tag{3.44}$$

$$\mathbf{H}_{\mathrm{B},n} = \begin{bmatrix} h_{\mathrm{B},n}(1) & \dots & h_{\mathrm{B},n}(n_h) & 0 & \dots & 0 \\ 0 & h_{\mathrm{B},n}(1) & \dots & h_{\mathrm{B},n}(n_h) & & \\ \vdots & & \ddots & \ddots & \ddots & \\ 0 & & & h_{\mathrm{B},n}(1) & \dots & h_{\mathrm{B},n}(n_h) \end{bmatrix} \tag{3.45}$$

$\mathbf{H}_{\text{BST}}$ has dimensions of $n_r n_w \times (n_h + n_w - 1)n_r$, where n_h is the number taps in the final channel and n_w is the number of taps in the beamformer. Note that this matrix tends to not be square.

$$\mathbf{Z}_{\text{ST}} = \begin{bmatrix} \mathbf{Z}_1 \\ \mathbf{Z}_2 \\ \vdots \\ \mathbf{Z}_{n_r} \end{bmatrix} \tag{3.46}$$

$\mathbf{Z}_{\text{ST}}$ has dimensions of $(n_h + n_w - 1)n_r \times (n_s + n_h + n_w - 1)$

$$\mathbf{Z}_n = \begin{bmatrix} z_n(1) & \dots & z_n(n_s) & 0 & \dots & 0 \\ 0 & z_n(1) & \dots & z_n(n_s) & & \\ \vdots & & \ddots & \ddots & \ddots & \\ 0 & & & z_n(1) & \dots & z_n(n_s) \end{bmatrix} \tag{3.47}$$

We can also define another matrix $\tilde{\mathbf{Y}} = \mathbf{H}_{\text{BST}}\mathbf{Z}_{\text{ST}}$, which is can be viewed as being formed of the following components:

$$\tilde{\mathbf{Y}}_n = \begin{bmatrix} \underline{\mathbf{y}}_n[0] & \underline{\mathbf{y}}_n[1] & \underline{\mathbf{y}}_n[2] & \dots \\ 0 & \underline{\mathbf{y}}_n[0] & \underline{\mathbf{y}}_n[1] & \ddots \\ 0 & 0 & \underline{\mathbf{y}}_n[0] & \ddots \\ \vdots & \ddots & \ddots & \ddots \end{bmatrix}, \tag{3.48}$$

and then stacked according to the structure,

$$\tilde{\mathbf{Y}} = \begin{bmatrix} \tilde{\mathbf{Y}}_1 \\ \vdots \\ \tilde{\mathbf{Y}}_N \end{bmatrix}. \tag{3.49}$$

Lastly, $\mathbf{w}$ contains all N_{m_A} beamforming filters within it. Given the structuring of $\tilde{\mathbf{Y}}$, individual filters can be parsed out by reading out groups of sequential coefficients, where the length is the specified number of taps.

$$\mathbf{w} = \begin{bmatrix} \mathbf{w}_1 \\ \vdots \\ \mathbf{w}_{N_{m_A}} \end{bmatrix} \tag{3.50}$$

Each $\mathbf{w}_n$ must be complex conjugated before handing them off to be convolved with $\underline{\mathbf{z}}_n$ as a result of the problem setup.

Again, a mesh to receiver channel estimate must be provided, however, each channel estimate is a temporal filter. A least squares channel estimator using a tapped delay line model can be formulated as follows [6]:

$$\min_{\underline{\mathbf{h}}} \quad (\underline{\mathbf{y}} - \underline{\mathbf{h}}\mathbf{S})(\underline{\mathbf{y}} - \underline{\mathbf{h}}\mathbf{S})^{\dagger} \tag{3.51}$$

can be used here, where the number of shifts is the number of desired delays (taps) to estimate:

$$\tilde{\mathbf{S}} = \begin{bmatrix} \underline{\mathbf{s}}[0] & \underline{\mathbf{s}}[1] & \underline{\mathbf{s}}[2] & \dots \\ 0 & \underline{\mathbf{s}}[0] & \underline{\mathbf{s}}[1] & \ddots \\ 0 & 0 & \underline{\mathbf{s}}[0] & \ddots \\ \vdots & \ddots & \ddots & \ddots \end{bmatrix}. \tag{3.52}$$

Taking the complex derivative with respect to $\underline{\mathbf{h}}^{\dagger}$ while leveraging Wirtinger derivatives, setting the result to 0, and rearranging the expression produces the channel estimate as shown in the following sequence:

$$0 = \frac{\partial}{\partial \underline{\mathbf{h}}^\dagger} \left[\underline{\mathbf{y}}\underline{\mathbf{y}}^\dagger - \underline{\mathbf{y}}\mathbf{S}^\dagger \underline{\mathbf{h}}^\dagger - \underline{\mathbf{h}}\mathbf{S}\underline{\mathbf{y}}^\dagger + \underline{\mathbf{h}}\mathbf{S}(\underline{\mathbf{h}}\mathbf{S})^\dagger \right] \tag{3.53}$$

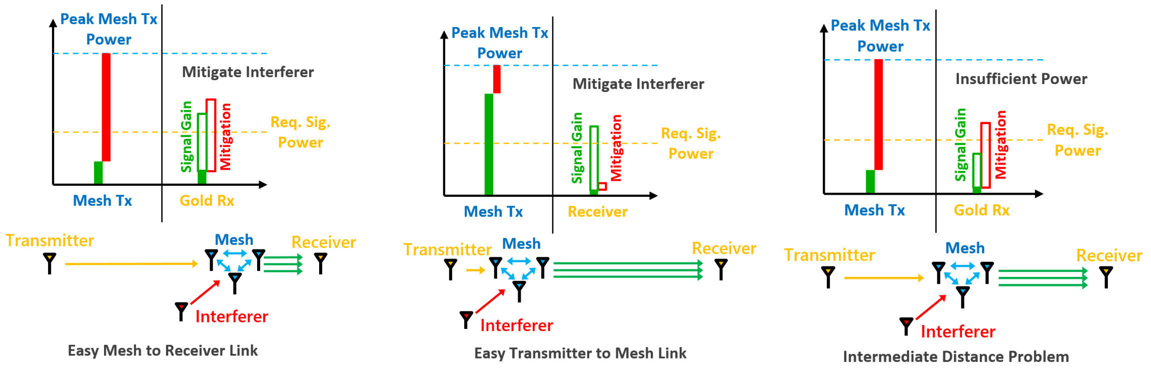
$$0 = -\underline{\mathbf{y}}\mathbf{S}^\dagger + \underline{\mathbf{h}}\mathbf{S}\mathbf{S}^\dagger \tag{3.54}$$

$$\underline{\mathbf{y}}\mathbf{S}^\dagger = \underline{\mathbf{h}}\mathbf{S}\mathbf{S}^\dagger \tag{3.55}$$

$$\underline{\mathbf{h}} = \underline{\mathbf{y}}\mathbf{S}^\dagger(\mathbf{S}\mathbf{S}^\dagger)^{-1} \,. \tag{3.56}$$

3.3 Operating Regimes

As a result of the bent-pipe relay approach, the SNR gain observed is dependent upon whether or not the retransmitted combined noise from the mesh nodes dominates the noise at the final receiver. Two operating extremes exist. When the final receiver's noise dominates, the SNR gain is N^2. When the combined retransmitted mesh noise dominates, the SNR gain is simply N. The power gain of the signal of interest scales by N^2 regardless, only the dominating factor of the noise changes. These scenarios arise when the relayed signal arrives at one receiver stage with a higher average SNR compared to the other. A smooth transition between the two regimes can be achieved by adjusting the received power of the predistorted relay signal. However, in certain scenarios or network configurations, the approach is insufficient. While the previously discussed system and MMSE solutions enable powerful performance enhancements compared to single-antenna links, the approach is insufficient in certain network configurations and geometries. The key operating regimes of the system are shown in Figure 6. The "intermediate distance problem" occurs when there is insufficient realized receive power at one of the stages of the relay system, despite having realized beamforming gain. This occurs due to constraints on element transmit



(a) Mitigation dominated regime.

(b) Range extension dominated regime.

(c) Intermediate distance regime.

Figure 6: Operational regimes of 1 to N to 1 distributed-coherent beamforming relay system. The mesh has a total power limit that depends on the output power of each element (blue dash). To close a given link, the final receiver must receive the signal of interest at some minimum power threshold (yellow dash). The mesh network acts as a bent-pipe relay, so any received interference will occupy some portion of the re-transmitted power (red bars) on top of the actual signal of interest (green bars). In scenario 6a, the noise power at the mesh dominates the system, limiting the potential SNR gain to a factor of N. This power distribution scenario occurs when the Transmitter to Mesh link is long or interference is strong. In scenario 6b, the noise power at the receiver dominates, limiting the potential SNR gain to a factor of N^2. This power distribution scenario happens when the Mesh to Receiver link is long, as a result the SNR gain will dominate the retransmitted power. In the intermediate distance scenario 6c, neither dominates and the potential SNR gain is limited somewhere between N and N^2. This power distribution occurs when the links between the Transmitter and Mesh, Mesh and Receiver are both longer, or of "intermediate distance", the system does not fall into one of the previous power distribution regimes, and fails to close the total link with insufficient retransmitted power allocation.

power, when the channels both have low SNR, and/or an interferer is present. As a result of the transmit beamforming approach, the interference power consumes a share of each mesh element's transmit power. When this interference power dominates the retransmitted power and both links have low SNR, the final received power can fall below the required signal power threshold despite beamforming gain.

The scaling performance of this system assumes that antennas have relatively similar access to the channel. An antenna that is significantly shadowed will not contribute. To be "shadowed" is to have a non-dominant line-of-sight component, and the dominant components arise from scattering. When spatially mitigating a large number of interferers, a portion of the signal of interest is lost for each interferer mitigated. If only a small number of the available spatial degrees of freedom are employed to mitigate interferers, the effective loss of the signal of interest is typically small. However, as the number of interferers mitigated increases, loss of the signal of interest also increases.

3.4 Performance Bounds

The performance of the system can partially be assessed through the signal to noise ratio (SNR) improvement or SNR gain it achieves. Here, I derive some bounds on the achievable gain. Consider the narrowband model without interference. I renormalize the beamformer $\mathbf{w}$ as given by equation (3.25) before application such that its L2 norm is $\sqrt{N}$. I denote this beamformer as $\mathbf{w}'$, which is proportionally related to $\text{diag}(\underline{\mathbf{h}}_B^*)^{-1}\mathbf{h}_A$ by the constant c. The narrowband beamformed reception is given by

$$\begin{aligned} \underline{\mathbf{y}} &= (\mathbf{w}'^{\dagger} \odot \underline{\mathbf{h}}_B)\mathbf{h}_A\underline{\mathbf{s}} + \text{diag}(\mathbf{w}'^{\dagger} \odot \underline{\mathbf{h}}_B)\mathbf{Q} + \underline{\mathbf{q}}_B && (3.57) \\ &= c\mathbf{h}_A^{\dagger}\mathbf{h}_A\underline{\mathbf{s}} + c\,\text{diag}(\mathbf{h}_A^{\dagger})\mathbf{Q} + \underline{\mathbf{q}}_B\,. && (3.58) \end{aligned}$$

First consider that each element of $\mathbf{h}_A$ and $\underline{\mathbf{h}}_B$ is identically, independently distributed (i.i.d.) according to $\mathcal{CN}(0, \mathbf{I})$. The expectation of the reception is

$$\begin{aligned}
&\mathrm{E}\left[c\mathbf{h}_A^\dagger \mathbf{h}_A \underline{\mathbf{s}} + c\,\mathrm{diag}(\mathbf{h}_A^\dagger)\mathbf{Q} + \underline{\mathbf{q}}_B\right] && (3.59)\\
=&c\left(\sum_{n=1}^{N} \mathrm{E}\left[|h_{A,n}|^2\right]\right)\underline{\mathbf{s}} && (3.60)\\
=&cN\underline{\mathbf{s}}\,. && (3.61)
\end{aligned}$$

The power of the signal of interest received at the final receiver is therefore

$$c^2N^2\|\underline{\mathbf{s}}\|^2\,. \tag{3.62}$$

The aggregated noise power is given by

$$\begin{aligned}
&\mathrm{E}\left[(c\,\mathrm{diag}(\mathbf{h}_A^\dagger)\mathbf{Q} + \underline{\mathbf{q}}_B)(c\,\mathrm{diag}(\mathbf{h}_A^\dagger)\mathbf{Q} + \underline{\mathbf{q}}_B)^\dagger\right] && (3.63)\\
=&\mathrm{E}\left[c\,\mathrm{diag}(\mathbf{h}_A^\dagger)\mathbf{Q}\mathbf{Q}^\dagger\mathrm{diag}(\mathbf{h}_A)c\right] + \sigma_B^2 && (3.64)\\
=&c^2\sum_{n=1}^{N} \mathrm{E}\left[|h_{A,n}|^2|\underline{\mathbf{q}}_n|^2\right] + \sigma_B^2 && (3.65)\\
=&c^2N\sigma_M^2 + \sigma_B^2\,. && (3.66)
\end{aligned}$$

The signal to noise ratio (SNR) of the beamformed signal is

$$\mathrm{SNR}_{\mathrm{MISO}} = \frac{c^2N^2\|\underline{\mathbf{s}}\|^2}{c^2N\sigma_M^2 + \sigma_B^2}\,. \tag{3.67}$$

This is a multiple-input-single-output SNR. Notice that the mesh noise incoherently combines and the signal of interest coherently combines. The noise accumulated throughout the relay chain may affect the achieved gain. This is a consequence of the bent-pipe forwarding strategy.

Measuring the gain that the mesh network offers should be performed in reference to an average single-input-single-output (SISO) SNR. Consider a scenario where a single mesh node forwards the signal using the same beamformer. The reception is given by

$$\begin{aligned}
\underline{\mathbf{y}}_n &= (\hat{w}_n^* h_{B,n}) h_{A,n}\underline{\mathbf{s}} + (\hat{w}_n^* h_{B,n})\underline{\mathbf{q}}_{M,n} + \underline{\mathbf{q}}_B && (3.68)\\
&= (c(h_{A,n}/h_{B,n}^*)^* h_{B,n}) h_{A,n}\underline{\mathbf{s}} + (c(h_{A,n}/h_{B,n}^*)^* h_{B,n})\underline{\mathbf{q}}_{M,n} + \underline{\mathbf{q}}_B && (3.69)\\
&= c|h_{A,n}|^2\underline{\mathbf{s}} + c h_{A,n}^*\underline{\mathbf{q}}_{M,n} + \underline{\mathbf{q}}_B\,. && (3.70)
\end{aligned}$$

Using knowledge of the supposed distribution of $h_{A,n}$ and $\mathrm{E}\left[|h_{A,n}|^2\right] = 1$. The SNR along path n is given by

$$\mathrm{SNR}_{\mathrm{SISO,n}} = \frac{c^2\|\underline{\mathbf{s}}\|^2}{c^2\sigma_M^2 + \sigma_B^2}\,. \tag{3.71}$$

The average $\mathrm{SNR}_{\mathrm{SISO}}$ can be expressed by the simple average of $\mathrm{SNR}_{\mathrm{SISO,n}}$

$$\mathrm{SNR}_{\mathrm{SISO}} = \frac{1}{N}\sum_{n=1}^{N}\mathrm{SNR}_{\mathrm{SISO,n}}\,. \tag{3.72}$$

In this case $\mathrm{SNR}_{\mathrm{SISO}} = \frac{c^2\|\underline{\mathbf{s}}\|^2}{c^2\sigma_M^2+\sigma_B^2}$.

A more realistic model to operate within is a Rician channel, in which there is a distinct line of sight component with a scattering component attached. Generating a narrow band Rician channel with such characteristics involves evaluating

$$h_{A,n} = \sqrt{\frac{K}{1+K}}e^{i\phi} + \sqrt{\frac{1}{1+K}}\sigma_s\,, \tag{3.73}$$

where K is a factor that controls the relative strength between line of sight and scattering components, σ_s represents the scattering component, which is distributed by $\mathcal{CN}(0, \mathbf{I})$. The magnitude of $h_{A,n}$ is drawn from a Rician distribution. Complex values $h_{A,n}$ are distributed by $\mathcal{CN}\left(\sqrt{\frac{K}{1+K}}e^{i\phi}, \frac{1}{1+K}\mathbf{I}\right)$. The SNR derivations leveraged the expected value of the channel's square magnitude. For a Rician channel, $\mathrm{E}[|h_{A,n}^2|] = 1$, which is the same as the previously used complex Gaussian channel. Consequently, the expressions for $\mathrm{SNR}_{\mathrm{MISO}}$ and $\mathrm{SNR}_{\mathrm{SISO}}$ remain the same if using Rician channels since the expectations have the same value. Although the expected behavior of using Gaussian channels is more or less the same as Rician channels, on a per draw basis the former can have significant imbalances in attenuation, which can prevent the system from consistently achieving the performance bound as it effectively denies equal channel access across all mesh nodes.

This system is designed for operation in line of sight dominant channels with relatively similar channel access to each mesh node. There exist two extremes of achievable SNR gains: final receiver noise dominance and mesh noise dominance. This phenomenon is a consequence of using the bent-pipe architecture. Under receiver noise dominance, the bound is N^2 SNR gain, however if the mesh noise dominates, the bound becomes N SNR gain.

Consider first when the noise at the final receiver is significantly larger than the incoherently accumulated noise power at any of the mesh nodes (ie $\sigma_B^2 \gg c^2\sigma_M^2$). The approximate average SISO SNR, MISO SNR, and resulting gain are

$$\mathrm{SNR}_{\mathrm{MISO}} \approx \frac{c^2 N^2 \|\underline{\mathbf{s}}\|^2}{\sigma_B^2} \tag{3.74}$$

$$\mathrm{SNR}_{\mathrm{SISO}} \approx \frac{c^2 \|\underline{\mathbf{s}}\|^2}{\sigma_B^2} \tag{3.75}$$

$$\mathrm{Gain} = \frac{\mathrm{SNR}_{\mathrm{MISO}}}{\mathrm{SNR}_{\mathrm{SISO}}} \tag{3.76}$$

$$= N^2 \,. \tag{3.77}$$

A scenario that yields this performance is when the mesh is close to the originating node but far away from the terminal receiver node (assuming the mesh and the initial node transmit with the same average power). An example is shown in Figure 6b. As a result, the transmission is received with significantly better SNR first. The signals must then endure a significant amount of attenuation propagating from the mesh to the final receiver.

Consider, however, if the accumulated mesh noise is significantly larger than the noise at the final receiver (ie $c^2 N\sigma_M^2 \gg \sigma_B^2$). The approximate average SISO SNR, MISO SNR, and resulting gain are

$$\mathrm{SNR}_{\mathrm{MISO}} \approx \frac{c^2 N^2 \|\underline{\mathbf{s}}\|^2}{c^2 N \sigma_M^2} \tag{3.78}$$

$$\mathrm{SNR}_{\mathrm{SISO}} \approx \frac{c^2 \|\underline{\mathbf{s}}\|^2}{c^2 \sigma_M^2} \tag{3.79}$$

$$\mathrm{Gain} = \frac{\mathrm{SNR}_{\mathrm{MISO}}}{\mathrm{SNR}_{\mathrm{SISO}}} \tag{3.80}$$

$$= N\,. \tag{3.81}$$

A scenario that yields this operating regime is when the mesh is farther away from the transmitting node than the receiving node. Naturally, if one transmission direction results in a receiver noise dominant operating regime, then the opposite direction operates within the opposite extreme. An example is shown in Figure 6a. In these cases, the signal is first received by the mesh with poor SNR. The mesh forwarded signal is then received with good SNR, but the noise floor is actually set by the aggregated mesh node noise floors.

There exists a smooth transition between the two regimes, where SNR improvement between N and N^2 is achievable. This transition can be swept through, forming a surface, by varying the SNRs at each reception stage. One factor that can affect the actual operating regime is the presence of additional signals in the bent pipe. The bent pipe essentially forwards along whatever signals were received, including interference. Interference and noise consume the limited power available for strictly sending the payload signal. Additional power needs to be given up when underlays are added to the relay signal. This effectively decreases the receive SNR at the next reception point and moves the system's operating point. There is a range of scenarios where despite effective beamforming, there is not enough SINR to close the link simply due to a decrease in the effective retransmit power of the signal of interest.

3.5 Performance Metrics

The SNR gain bound is well known to be N^2 with N of the improvement from the addition of N elements with the same power, and N from the coherent combining [35]. The interference rejection performance is dependent upon the quality of the estimates, and is proportional to the number of elements in the mesh. I now discuss how to evaluate against these metrics in a practical system.

3.5.1 SNR Gain

As discussed in the previous bounds section, SNR gain achieved by the mesh should be measured by computing the ratio of the MISO beamformed payload signal SNR to the average SISO payload signal. The average SISO SNR should be performed by only having one mesh node forward at a time. Failing to satisfy that condition would invalidate the measurement. To make a proper comparison, the average SISO SNR should be the same during the MISO SNR measurement as was measured during the SISO tests. Practically, this is easiest to measure without interference in the environment.

3.5.2 Interference Rejection

Ultimately performance should be assessed by measuring the INR before applying an optimal beamformer and after. Each node experiences interference. They receive the interference sources at some interference to noise ratio (INR). If the array is wired together, without beamforming, there is some form of gain simply by summing the data. It is not clear what level of coherency that summation is. This is at the mercy of the environment. This idea also applies in the mesh relay case. However, the mesh to final receiver channel affects how the forwarded interference combines. Consequently, the baseline INR measured can vary.

As a result, there are several reference points that can be used for the evaluation of interference rejection performance. Similar to SISO SNR measurements, the average SISO INR that arrives at the final receiver can be measured. From that point, a perfect incoherently combined MISO INR can be set, which is the average SISO INR multiplied by the number of nodes. The INR that results without beamforming should be compared to the perfect incoherently combined INR. If the interferer arrives at the final receiver with gain greater than the perfect incoherent gain, then does the system get to take credit for that reduction? There is a difference between this fluke and an interferer simply emitting more power. Consider if the INR happens to be lower than perfect incoherent gain without beamforming. Say when the optimal beamformer is applied 10 to 15 dB reduction but the interference power is below the noise floor and the payload is received with a high enough SINR to be decoded. That would mean some rejection capability was not credited toward the beamformer. If an interferer arrives naturally at a low enough INR (such as an interferer near 0 dB at the receiver) before beamforming, it is difficult to measure a meaningful delta as the beamformer is not doing any work. During algorithm development and using simulations to validate the design, both the actual measured INR and the theoretical perfect incoherent combination INR are referenced to assess performance. Measuring INR is another issue though. In simulation this can be done rather easily. Experimentally this can be quite difficult. One method is to measure bit error rate curves with different interferer powers and measure the point where the curve begins to move detrimentally as the interferer power level at which the system falls apart. The null depth can be then be backed out from other system parameters and measurements.

3.6 Practical Algorithm Implementation and Challenges

The solutions discussed previously must be adapted for a practical hardware implementation and the particular application of the algorithm. The first is the distributed nature of the computation. Implementing a distributed system introduces many complications, which may not be immediately clear through the previously presented equations. Two major assumptions, time-frequency synchronization and data availability, are immediately broken as a result. Achieving these two requirements ultimately drive many aspects of the algorithm adaptation and system design choices.

3.6.1 Time-Frequency Synchronization

Assumed in the spatiotemporal MMSE beamformer construction is the use of data, where each stream has been captured starting at a precise global time, and the transmission and reception center frequencies are identical across all elements, as is typical in a classic beamforming system. However, the system has nodes that are distributed, meaning the radios have independent clock sources that may not agree in their sense of time, including the rate of progression. Differences in the clock's rate of progression introduce deviations from the nominal center frequency. Additionally, each source and receiver within the environment may be mobile, which will introduce doppler shifts, which are seen as additional frequency errors [6]. A depiction of the network labeled with all the possible frequency offsets that can be introduced is shown in Fig. 7. If frequency alignment is poor enough, poor channel estimates (correlations) may be observed. More importantly, the beamformed payload signal will not be able to maintain coherence. If timing alignment is poor, then the geometry of the network and sources is inaccurately represented through the estimates. To realize the desired beamforming gain and interference rejection at the receiver, the mesh's sense of time and center frequencies must be aligned. Synchronization is a requirement no matter what distributed coherent beamforming algorithm or technique is used. This approach does not require the use of GPS to provide any synchronization.

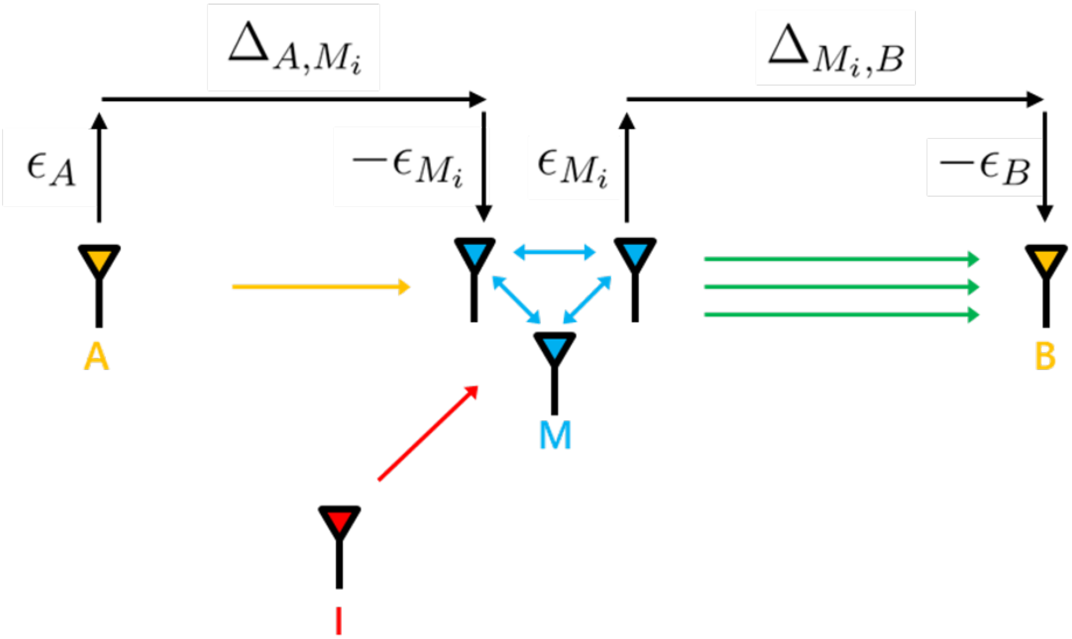


Figure 7: Sources of frequency error that result throughout the mesh transmissions. ϵ_A results from the deviation from nominal center frequency of the A node. The Δs arise from motion between A and the mesh and B inducing doppler shifts. ϵ_M arises from the deviation from the nominal center frequency of the mesh elements. ϵ_B arises from the deviation from the nominal center frequency of the B receiver.

The discussed spatiotemporal MMSE beamformer construction explicitly corrects for time and phase offsets. However, mesh nodes need to be sufficiently synchronized in time so that they can accurately measure time difference of arrival (TDOA) information. They must also signal toward the final receiver in a way that the channel estimator can also accurately determine TDOA information. A common sense of time should be established and can be done by declaring a reference mesh node for all others to align to. Beamformer construction is somewhat robust to misalignments between each mesh's sense of global time from a true global timeline if certain conditions are met. Time alignment should be accurate within one chip or less of the signal of interest. Mislignment is more tolerable when only SNR gain is desired than when effective interference rejection is desired. Lastly, any misalignments must be stable and jitter minimally. A mechanism to coordinate the timing between the mesh nodes is required. Algorithms such as CHP2 [41, 18] can be leveraged to achieve sufficient timing synchronization.

The spatiotemporal MMSE beamformer construction presented does not explicitly account for frequency offsets. It can implicitly make minor frequency adjustments, but the range of offsets for which this is effective is extremely limited. The beamformer construction can be augmented in frequency as well to explicitly correct center frequency alignments at the expense of much greater computational complexity. A separate mechanism is needed to align the transmissions and receptions. Each node in the network must correct for its own offset from some reference center frequency. Typically a node in the relay network is determined to be the reference. On reception each mesh node needs to correct for its offset from the initial transmitter. An independent pre-correction needs to be applied toward each mesh node's transmission toward the final receiver, such that signals arrive correctly centered after down-conversion. According to this strategy, the beamformer computation must occur as if the mesh to receiver offset does not exist. Depending on the channel B estimation strategy a phase ramp correction may need to be applied to the estimate before feedback.

Estimators are implemented at the mesh and initial and final transceiver nodes to inform the necessary corrections for Doppler shifts and offsets attributed to oscillator mismatches. It is important to distinguish that frequency offset estimation error is the concerning metric that affects performance, not the size of the frequency offset. For instance, assume that all radios were driven by nominally performing clocks. There is still a frequency offset that can occur if the nodes are moving. Correcting for effects due to smooth predictable motion is simple. Erratic motion is difficult to track, which can increase the estimation error variance. In the case of imperfect clocks, discontinuous clock disciplining causes complications. The system requires stable, continuous phase at the output frequency synthesizers. This makes TCXOs very undesirable as they frequently phase reset. The beamformer needs to be updated regularly to adapt to the dynamics of the environment and the system. However, the accuracy of the frequency offset estimator is the main component that dictates how often updates occur. There is latency between offset measurement and the correction application, so any residual error will cause excess phase rotation within the application period, which will cause the beamformed signal to become incoherent. If the estimator's accuracy is precise and the clocks extremely stable, then there is less of a need to update often. On the other hand, updating often places less pressure on the estimator to be precise. In my designs, I err on the side of updating more frequently to provide some margin for estimator performance, or to support motion.

Some performance bounds related to the information update latency and residual frequency distribution are now presented. For an N-element antenna array SNR Gain as a function of the update interval τ and the frequency error variance σ^2 is given by [2, 32]:

$$(N^2 - N)(e^{-4(\pi\tau)^2\sigma^2}) + N\,. \tag{3.82}$$

Isabella Lenz derived the INR reduction parameterized in the same way, and with permission, I reproduce it here. Consider the uncorrupted signal vector $\mathbf{v} = \mathbf{1}$ and the frequency error corrupted received signal vector $\mathbf{v}'$, at each element i be $v'_i = e^{j2\pi\delta_i\tau}$. Then, define a projection operator, $P_v = \mathbb{I} - \mathbf{v}(\mathbf{v}'^\dagger\mathbf{v}')\mathbf{v} = \mathbb{I} - \frac{\mathbf{1}\mathbf{1}^\dagger}{N}$. Let the residual frequency error δ_i be distributed as $\delta_i \sim \mathcal{N}(0,\, \sigma^2)$, and the time-frequency product, $2\pi\delta_i\tau \sim \mathcal{N}(0,\, (2\pi\tau)^2\sigma^2)$. Start by defining the power reduction realized at the receiver γ:

$$\begin{aligned}
\gamma &= \frac{||P_v\mathbf{v}'||^2}{||\mathbf{v}'||^2} && (3.83)\\
&= \frac{\mathbf{v}'^\dagger P_v P_v \mathbf{v}'}{\mathbf{v}'^\dagger\mathbf{v}'} && (3.84)\\
&= \frac{\mathbf{v}'^\dagger P_v \mathbf{v}'}{N} && (3.85)\\
&= \frac{1}{N}\mathbf{v}'^\dagger(\mathbb{I} - \frac{\mathbf{1}\mathbf{1}^\dagger}{N})\mathbf{v}' && (3.86)\\
&= \frac{1}{N}(N - \mathbf{v}'^\dagger\frac{\mathbf{1}\mathbf{1}^\dagger}{N}\mathbf{v}') && (3.87)\\
&= 1 - \frac{||\mathbf{v}'\mathbf{1}||^2}{N^2} && (3.88)\\
&= 1 - \frac{||\sum_{i=1}^{N} e^{j2\pi\delta_i\tau}||^2}{N^2} && (3.89)\\
&= 1 - \frac{|H|^2}{N^2}\,. && (3.90)
\end{aligned}$$

Then taking the expectation yields the expected parameterization:

$$\begin{aligned}
\mathbb{E}[\gamma] &= 1 - \frac{\mathbb{E}[|H|^2]}{N^2} && (3.91)\\
&= 1 - \frac{(N^2 - N)(e^{-4(\pi\tau)^2\sigma^2}) + N}{N^2} && (3.92)\\
&= 1 - (1 - \frac{1}{N})e^{-4(\pi\tau)^2\sigma^2} + \frac{1}{N} && (3.93)\\
&= 1 - e^{-4(\pi\tau)^2\sigma^2} - \frac{e^{-4(\pi\tau)^2\sigma^2}}{N} + \frac{1}{N} && (3.94)\\
&= \frac{N+1}{N} - \frac{N+1}{N}e^{-4(\pi\tau)^2\sigma^2} && (3.95)\\
&= \frac{N+1}{N}(1 - e^{-4(\pi\tau)^2\sigma^2})\,. && (3.96)
\end{aligned}$$

A family of curves parameterized by the update rate describing SNR gain as a function of the error variance is shown in Fig. 8. A similar family of curves describing INR reduction performance is shown in Fig. 9. It should be observed that achieving SNR gain is somewhat tolerant to errors, but achieving effective INR reduction is much less tolerant. The adage that beams are wide and nulls are narrow holds in the context of frequency error and information latency for distributed coherent beamforming. These two graphs allow us to determine how quickly the beamformer needs to update given some estimator performance.

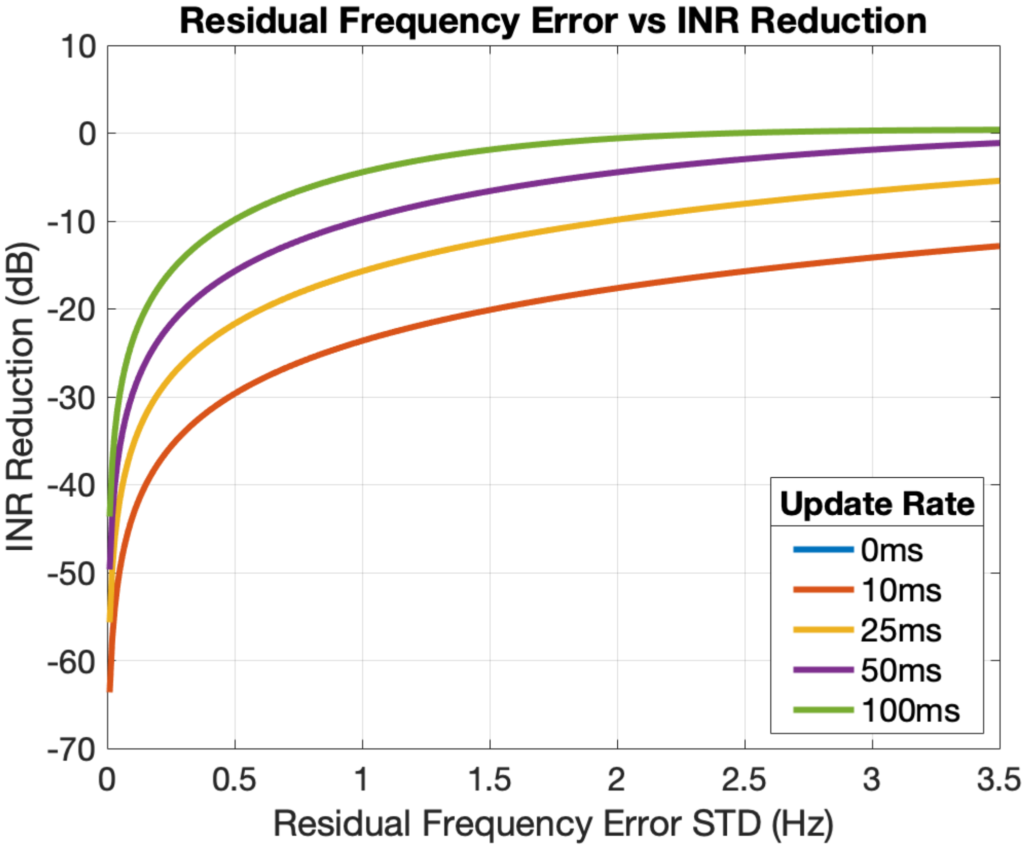


Figure 8: Theoretical SNR gain as a function of frequency estimation error standard deviation parameterized by update rate

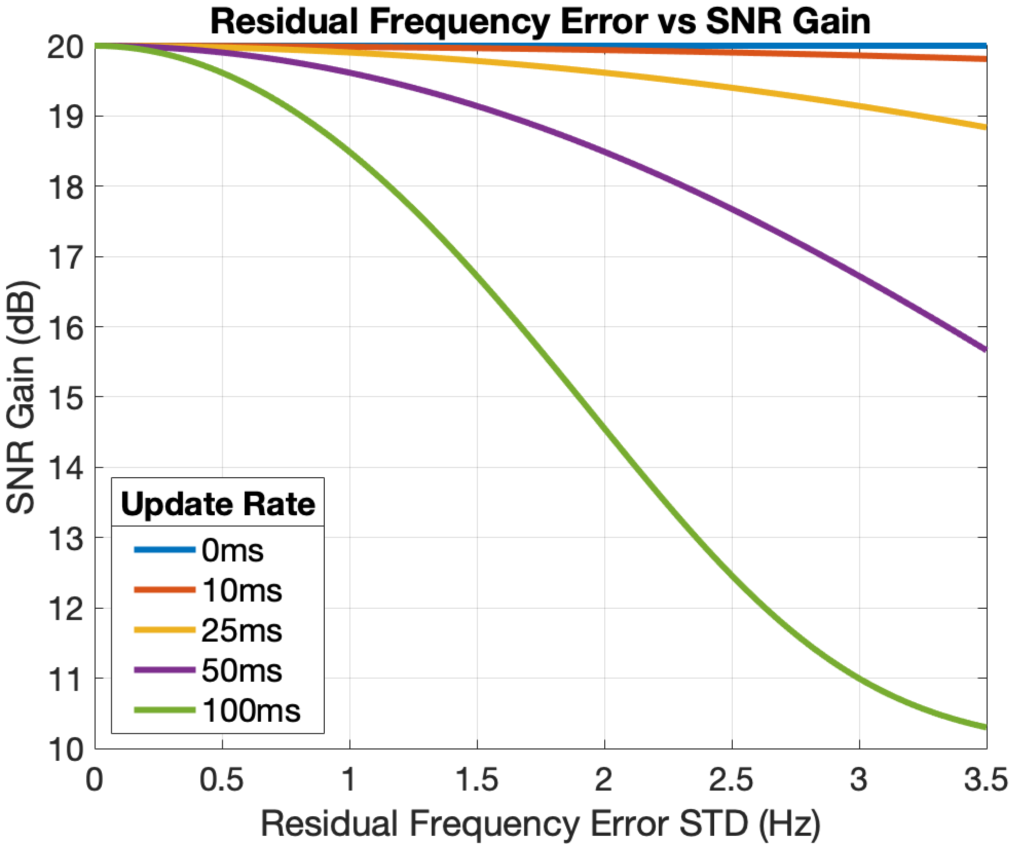


Figure 9: Theoretical INR reduction as a function of frequency estimation error standard deviation parameterized by update rate.

The estimator performance limits can be evaluated by constructing the Cramer-Rao lower bound (CRLB) for frequency offset estimation. A common initial point of reference is the CRLB for a sinusoidal tone in white noise, which is given as: $\mathrm{var}(\hat{f}) \geq \frac{12\,f_s^2}{(2\pi)^2\eta N(N^2-1)} \approx \frac{12}{(2\pi)^2}\frac{1}{\mathrm{ISNR}}\frac{1}{T^2}$, where η is the SNR, N is the integration length in samples, f_s is the sampling rate, ISNR is the integrated SNR, and T is the time interval over which the samples were taken [26]. The CRLB can be parameterized according to the integrated SNR (ISNR) of the observation. Whether or not an estimator can achieve the lower bound is dependent upon its construction. Estimator performance that enables sufficient SNR gain and INR reduction should be targeted.

The ISNR associated with the desired estimator performance can be used to inform waveform design and the signaling strategy. Several candidate estimators are discussed in [38, 39, 10, 46, 42]. The required solution is often system and waveform specific.

3.6.2 Information Exchange and Limited Data Rates

Provided sufficient time-frequency alignment, the efficacy of the beamformer depends on the system's ability to accurately estimate the spatiotemporal data covariance matrix and the cross-correlation given a period of time where the statistics are considered stationary. Estimation performance is ultimately tied to the length of training information and observation data available.

The beamformer computation requires the observations of all mesh nodes be compiled on to some processing agent. This information must be exchanged through an intramesh data exchange network. Because all mesh nodes are distributed, there exists a non-trivial limitation on the amount of information that can be feasibly exchanged. The network rate can increase very quickly with the adjustment of certain parameters.

The "minimum" number of samples needed to achieve some estimation accuracy of the covariance matrix depends on its dimensionality. More samples are needed as the number of dimensions increases. The number of dimensions is dependent upon the number of nodes and the number of beamforming filter taps used. The estimation accuracy is tied to how many samples of each $\underline{\mathbf{z}}_n$ can be exchanged across the network, but how effective they are depends on the dimensionality of the covariance matrix being estimated. The dimension of the covariance in question is $N \cdot n_w$, or the number of elements in the mesh times the number of taps used in the beamformer. This means that the network rate scales with the maximum distance between any two mesh elements, and the number of mesh elements.

For the cross-correlation (channel estimate) components of the beamformer, to exchange sufficent samples for a quality estimate is impractical, especially when interference is present. However, if the estimate is made locally, exchanging the correlation results becomes tractable. This reduces the amount of data transfer to depend on the number of beamformer taps used. It however places slightly more computational burden on each mesh element.

Thus, for information exchange strictly between mesh nodes, each node needs to communicate their local covariance estimation samples and correlation samples, thus making the data rate highly dependent on the number of taps used, and number of mesh elements. In addition to these pieces, additional information is required to be transmitted. These include information required to support time-frequency synchronization between mesh nodes, and even the transmitter identity.

Inherent to estimating both the covariance matrix and the cross-correlation vector is the ability to also estimate the mesh to final receiver channel well. Similar to the cross-correlation, the estimation quality depends on the number of samples needed to overcome a significant interference at the final receiver node. However, only the estimates themselves are required to be transmitted back to the mesh. The amount of data required to be sent only depends on the number of nodes supported and the number of taps used. Given some fixed number of mesh nodes, the number of taps must be judiciously selected for feasible support.

The most expensive piece of data to send are the “raw" samples used for covariance estimation. And under some scenarios, the mesh to final receiver channel estimate is relatively expensive to transmit, typically as a function of the receivers distance from the mesh. Ultimately, the number of taps specified for the beamforming filter drives the required data rates the most.

actical set of mesh geometries, an impractically large covariance matrix tends to result, which would require an infeasible number of samples to be exchanged. Its construction and inversion would place an immense computational burden on the system as well.

Data rates can potentially be reduced further by tracking estimates or beamformers and sharing compressed vector updates. It may also be possible to efficiently implement a distributed algorithm that tracks the optimal beamformer (for example RMS, LMS, etc...) or to track the subspace of the beamformer itself and then share compressed updates to that vector space. There are other promising approaches such as Grassmanian source compression that could be leveraged on the raw channel estimates.

3.6.3 Outage Handling

The mesh is a network of distributed nodes so there is a risk of individual elements dropping out of the network either temporarily or indefinitely. Firstly, it is advisable to have all nodes compute the beamformer. Given the same data has been exchanged at every step, the computations and the results will be the same. Declaring a central node to perform computations puts the system at risk of a single point of failure. There is only a marginal increase in overhead to exchange all the necessary information to all nodes.

When a node loses intra-mesh communications, the most recently available data is used to compute the beamformer. As long as the data is not stale beyond some threshold, then using the most recent measurements should result in acceptable performance degradation. If the data were too stale, then the observations would poorly inform the beamformer of the correct adjustments. Even worse, there would be a risk of self-nulling. In the event a node cannot reliably exchange information, and the number of failed attempts has exceeded some limit, that node should not contribute toward signal relaying until stable communications can be established again. If a node is forced to dropout due to unreliable exchange or is put out of commission, then the computation must zero out the observations corresponding to that node. If a node has been forced to drop out, then it must recognize this situation and stop itself from relaying the payload waveform. Zeroing out the contributions will effectively remove it from the construction of the beamformer without affecting the other elements computational results. For example, a system designed for operation with 10 nodes and an inflexible computational chain can operate with fewer as long as their data contributions are simply set to 0.

3.6.4 Covariance Matrices

Self-interference and consequently self-nulling arises from the presence of the signal in the construction of the covariance matrix. The effects can be reduced through careful application of covariance matrix augmentations. I also now address augmentations to help support limited numbers of estimation samples used in the construction of the matrix. Assuming that the discussed information exchange has occured, computation can proceed according to the discussed spatiotemporal MMSE beamformer solution. First, each data vector is convolved with the channel estimate to form a hypothetical reception and stored as

$$\hat{\mathbf{Y}} = \begin{bmatrix} \underline{\hat{\mathbf{y}}}_1 \\ \vdots \\ \underline{\hat{\mathbf{y}}}_N \end{bmatrix} = \begin{bmatrix} \underline{\hat{\mathbf{h}}}_{B,1} * \underline{\hat{\mathbf{z}}}_1 \\ \vdots \\ \underline{\hat{\mathbf{h}}}_{B,N} * \underline{\hat{\mathbf{z}}}_N \end{bmatrix} . \tag{3.97}$$

A spatiotemporal data matrix $\tilde{\mathbf{Y}}$ can be built from $\hat{\mathbf{Y}}$ in the following way:

$$\tilde{\mathbf{Y}}_n = \begin{bmatrix} \underline{\hat{\mathbf{y}}}_n[0] & \underline{\hat{\mathbf{y}}}_n[1] & \underline{\hat{\mathbf{y}}}_n[2] & \dots \\ 0 & \underline{\hat{\mathbf{y}}}_n[0] & \underline{\hat{\mathbf{y}}}_n[1] & \ddots \\ 0 & 0 & \underline{\hat{\mathbf{y}}}_n[0] & \ddots \\ \vdots & \ddots & \ddots & \ddots \end{bmatrix} \tag{3.98}$$

$$\tilde{\mathbf{Y}} = \begin{bmatrix} \tilde{\mathbf{Y}}_1 \\ \vdots \\ \tilde{\mathbf{Y}}_N \end{bmatrix} . \tag{3.99}$$

Note that the organization corresponds to that of $\hat{\mathbf{r}}$. Practically, in an implementation of the computational chain, this spatiotemporal data matrix should never be explicitly formed, as it has structure that can be leveraged when computing the covariance.

The spatiotemporal covariance matrix is now constructed by evaluating

$$\mathbf{C} = \tilde{\mathbf{Y}}\tilde{\mathbf{Y}}^{\dagger} . \tag{3.100}$$

Assume that this estimation is explicitly sample starved, due to limited network bandwidth between the mesh nodes to share sufficient samples. Fewer samples of $\underline{\hat{\mathbf{z}}}_n$ than there are dimensions in $\mathbf{C}$ are exchanged, which means that latter is poorly averaged by itself. Furthermore, $\mathbf{C}$ will almost certainly be poorly conditioned. The first tool I deploy in addressing this ill-conditioning is the employment of averaging.

Specifically, I employ an exponential moving average. This procedure can be expressed as

$$\mathbf{C}_t = (1-b)\mathbf{C}_{t-1} + b\mathbf{C}\,, \tag{3.101}$$

where b is a forgetting factor, $\mathbf{C}$ is the most recent covariance computation, $\mathbf{C}_{t-1}$ is the average from the previous time step, and $\mathbf{C}_t$ is the current average. The forgetting factor b weights the history of all $\mathbf{C}$ such that the average depends most on more recent estimates. Over time, the influence of a previous estimate has on the average decays. Employing this type of averaging instead of a moving average due to computational simplicity as well its ability to adapt to a dynamic environment. From startup, the system requires a few iterations of training to produce a usable covariance matrix. When nodes of the network have high mobility, then the pseudo-stationary assumption is violated over a shorter period of time. In these situations, the averaging mechanism must be less reliant on the past. Consequently, the covariance matrix is at risk of being poorly estimated once again. I discuss a method to circumvent this issue toward the end of this subsection.

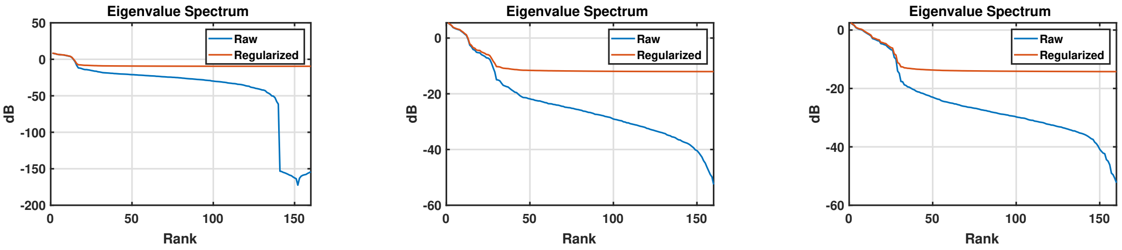


Figure 10: Progression of Covariance Estimate Eigenspectra over 3 iterations at system startup, including raw and versions with regularization applied.

Additional tools are needed to further stabilize the computation and make the algorithm robust. In sample starved scenarios, the covariance matrix tends to be poorly conditioned. Spatiotemporal data matrices tend to contain plenty of redundancy which exacerbates the ill-conditioning. When the covariance matrix is poorly conditioned, its inverse will introduce heavy distortions in the beamformer. One common technique that permits the construction of a sensible beamformer despite these scenarios is L2 regularization, otherwise known as diagonal loading. L2 regularization introduces a penalty term into the MMSE objective function. The new optimization problem is given by

$$\min_{\mathbf{w}} \quad \mathbf{w}^{\dagger}\mathbf{C}_t\mathbf{w} - \mathbf{w}^{\dagger}\mathbf{r} - \mathbf{r}^{\dagger}\mathbf{w} + \|\underline{\mathbf{s}}\|^2 + d\|\mathbf{w}\|^2 \,, \tag{3.102}$$

where d is a parameter set by the user. The result of taking the derivative, setting it equal to zero, and solving for $\mathbf{w}$ yields the following closed form solution:

$$\mathbf{w} = \left(\mathbf{C}_t + d\mathbf{I}\right)^{-1} \hat{\mathbf{r}} \,. \tag{3.103}$$

The parameter d is the loading level, which is related to the penalty weight for increasing the norm of $\mathbf{w}$. Diagonal loading constrains the white noise gain, often the most poorly estimated component of the covariance matrix [25, 37, 8, 28].

The loading level d must be judiciously set. Increasing d too much causes the covariance matrix to appear as a scaled identity matrix, in which case the beamformer will be proportional to $\mathbf{r}$. Consequently, excessive loading destroys any interference mitigation properties. Setting d loading level too low may fail to stabilize the computation, producing nonsensical filters. The appropriate amount of diagonal loading is highly dependent on the scenario but is often referenced from either the maximum eigenvalue or the system noise floor. The former method is expressed by

$$\alpha\lambda_1 \,, \tag{3.104}$$

where α scales down the maximum eigenvalue λ_1. A more practical approach is to use a proxy measure of the maximum eigenvalue instead. The simplest alternative is

$$d_1 = \alpha \frac{\mathrm{tr}(\mathbf{C})}{T_w}\,, \tag{3.105}$$

where the trace is divided by T_w to account for the redundancy introduced by using the spatiotemporal variant. The latter is expressed by

$$d_2 = \beta \max_n |\underline{\hat{\mathbf{h}}}_{B,n}|^2 \sigma_n^2\,, \tag{3.106}$$

where β is a parameter that scales up the highest noise floor. The noise floors considered are specifically the noise floor of $\underline{\hat{\mathbf{z}}}_n$, σ_n^2, with an adjustment for the channel estimate that has been applied to the data.

The primary factor dictating how the diagonal loading level should be set is the Signal-to-Interference Ratio (SIR) observed at the mesh. Given restrictions on computational capabilities and data exchange, this is difficult to measure in real time on a practical system. With an appropriate constant α, the trace-based method is appropriate for a wide range of high-SIR scenarios. With an appropriate constant β, the noise-based method is appropriate for a wide range of low SIR cases that the other method may not be able to support. Both loading methods are considered, and a system can switch between them to robustly and automatically adapt to the environment. Mesh relay nodes switch between the two methods according to the following rule:

$$d = \max\{d_1, d_2\}\,. \tag{3.107}$$

As the SIR changes, so will d_1, but d_2 will remain constant. The switching rule sets a lower limit on what the loading level d can be.

An additional technique that can be used to add robustness is covariance matrix tapering. Covariance matrix tapering applies a phase dither in time and space to increase the extent of every emitter represented. The main lobe and nulls of the beamformer are widened as a results [47, 12, 33, 49]. Sidelobes of the beamformer radiation pattern are also suppressed and any error in the beamforming targets statistics are compensated for by the wider mainlobe. This helps to ensure closer to optimal SNR gain in a dynamic and poorly estimated environment.

Poor estimation quality means the interferers are inaccurately represented in space. Mismatches between the estimate and truth will therefor yield poor interference mitigation. Widening the nulls provides a better opportunity to mitigate the interferers. Sidelobes of the beam forming pattern are also suppressed, and any error in the beamforming targets statistics are compensated by the wider main lobe. This helps to insure closer to optimal SINR gain in a dynamic, poorly estimated environment.

Using tapering and diagonal loading together can be a powerful combination for regularization and null widening. The diagonal loading factor fits easily neatly into the formulation of the taper and can be thought of as a special case of covariance tapering.

$$T_{m,n} = \text{sinc}((m-n) * \delta/\pi) \tag{3.108}$$

$$\mathbf{T}_{\text{DLMZ}} = (T_{\text{MZ}_n} \otimes 1_{n_{taps} \times n_{taps}}) \odot (1_{N \times N} \otimes T_{\text{MZ}_\tau}) + d\mathbf{I} \tag{3.109}$$

$$\mathbf{T}_{\text{MZ}_n} = [\text{sinc}((m-n)\Delta_n)] \tag{3.110}$$

$$\mathbf{T}_{\text{MZ}_\tau} = [\text{sinc}((m-n)\Delta_\tau)] \tag{3.111}$$

3.6.5 Internal Beamforming

While the previously discussed system and MMSE solutions enable powerful performance enhancements compared to single-antenna links, the approach is insufficient in certain network configurations and geometries. The "intermediate distance problem" occurs when there is insufficient realized receive power at one of the stages of the relay system, despite having realized beamforming gain. This issue can be seen in Figure 6. This occurs due to constraints on element transmit power, when the channels both have low SNR, and/or an interferer is present. As a result of the transmit beamforming approach, the interference power consumes a share of each mesh element's transmit power. When this interference power dominates the re-transmitted power and both links have low SNR, the final received power can fall below the required signal power threshold - despite beamforming gain.

Internal beamforming utilizes a surrogate receiver element to perform an intermediate beamforming stage at the mesh to mitigate interference spatially before retransmission to the receiver. This is performed by first using $N-1$ elements of the mesh to spatially mitigate the interference using the MMSE transmit beamforming process described previously. These $N-1$ elements transmit to the N^{th} element acting as a surrogate final receiver. The signal is then re-transmitted from the surrogate final receiver to the other elements of the mesh with significantly reduced interference power. By relaying internal to the mesh, the second link is shortened, such that it falls into the mesh noise dominant case, allowing it to null interference and regain power allocation to be used for the signal of interest. Then, returning to the original mesh, the system now operates in the receiver noise dominant case, allowing the desired SNR gain to be realized.

Chapter 4

MULTIPLE STAGE RELAY BEAMFORMING

In this chapter, I will introduce the models for a multiple stage relay distributed coherent mesh beamforming system. The multiple stage relay system shown in Figure 11 is designed to operate in the limit of low SNR and a highly correlated channel. Each of the stages channel's statistics and operating regime is critical to achieving the expected SNR improvement performance of N^2M. This scaling assumes that the Mesh A to Mesh B MIMO channel, has a nontrivial low-rank component. For line-of-sight channels, this is clearly satisfied, however, there may be a limitation in the case of terrestrial-to-terrestrial links with particularly rich scattering. I will outline optimal solutions for each stage and how they can be used in conjunction to achieve the multi-stage relay SNR improvement bound without interference. Further, I will discuss algorithm augmentations that can enable the desired performance in the presence of interferers at multiple stages, as well as simulations supporting the claims. Finally, I will present experimental work and results for the system without interference.

4.1 Compound SIMO-MIMO-MISO Channel

For the compound problem indicated in Figure 11 , I have to consider multiple communications propagation channels: $\mathbf{h}_{A\rightarrow m_A} \in \mathbb{C}^{n_A\times 1}$, $\mathbf{H}_{m_A\rightarrow m_B} \in \mathbb{C}^{n_B\times n_A}$, and $\mathbf{h}_{m_B\rightarrow B} \in \mathbb{C}^{n_B\times 1}$. Additionally, I have to consider the effect of interference at each stage. Let n_A be the number of elements in Mesh A (the first mesh) and n_B the number

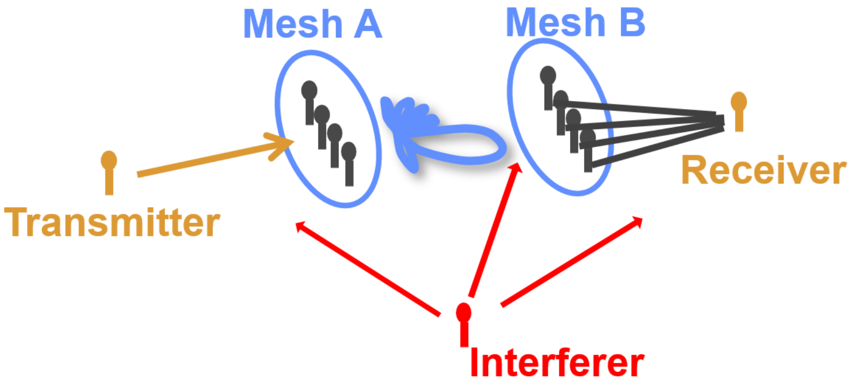


Figure 11: By utilizing a relay meshes between two single antenna nodes, the desired SNR gain is realized. Interference is rejected at each stage through the application of spatial nulls in the transmit beamforming. The first mesh receives the signal at its N antennas, and applies the appropriate beamforming pre-distortion filter to the signal before retransmission. The second mesh receives the retransmitted signal at its M antennas, where it applies its beamforming pre-distortion filter. Finally, the second mesh retransmits the transformed signal to the receiving node. The cycle is then repeated in the reverse direction.

of elements in Mesh B (the second mesh). If I assume a simple bent-pipe approach, such that the received signal is transformed without decoding and retransmitted, this interference will be forwarded and become relevant at the next stage. Any potential interference at each reception stage is given by $\mathbf{J}_{m_A}\,\mathbf{T}_{m_A}$, $\mathbf{J}_{m_B}\,\mathbf{T}_{m_B}$, and $\mathbf{j}_B\,\underline{\mathbf{t}}_B$, where $\mathbf{J}$ or $\mathbf{j}$ is used to represent the interference propagation channel and $\mathbf{T}$ or $\underline{\mathbf{t}}$ is used to indicate the interference time-domain signal.

The optimization problem is to maximize the SINR at the final receiving node. I will approach this by optimizing the capacity at each stage of the network individually. I now construct the received signal at each stage. The received signal $\mathbf{Z}_A \in \mathbb{C}^{n_A \times n_s}$ at Mesh A is given by

$$\mathbf{Z}_A = \mathbf{h}_{A\to m_A}\,\underline{\mathbf{s}}_A + \mathbf{J}_{m_A}\,\mathbf{T}_{m_A} + \mathbf{N}_{m_A} \tag{4.1}$$

where n_s is the number of received samples. By assuming that the baseband signal is not modified and is retransmitted through a beamformed bent-pipe approach, the reception $\mathbf{Z}_B \in \mathbb{C}^{n_B \times n_s}$ at Mesh B is then given by

$$\mathbf{Z}_B = \mathbf{H}_{m_A \to m_B} (\mathbf{w}_A \, \underline{\mathbf{1}} \odot \mathbf{Z}_A) + \mathbf{J}_{m_B} \, \mathbf{T}_{m_B} + \mathbf{N}_{m_B} \,, \tag{4.2}$$

and at the final receiving node the received signal $\underline{\mathbf{z}} \in \mathbb{C}^{1 \times n_s}$ is given by

$$\underline{\mathbf{z}} = (\mathbf{w}_B \odot \mathbf{h}^*_{m_B \to B})^\dagger \, \mathbf{Z}_B + \mathbf{j}_B \, \underline{\mathbf{t}}_B + \underline{\mathbf{n}}_B \,. \tag{4.3}$$

I represent the Mesh A transmit beamformer weights with $\mathbf{w}_A$ and the Mesh B transmit beamformer weights with $\mathbf{w}_B$. By combining these equations, I have a compound channel given by

$$\begin{aligned} \underline{\mathbf{z}} = (\mathbf{w}_B \odot \mathbf{h}^*_{m_B \to B})^\dagger \, \{\mathbf{H}_{m_A \to m_B} [(\mathbf{w}_A \, \underline{\mathbf{1}}) \odot (\mathbf{h}_{A \to m_A} \, \underline{\mathbf{s}}_A + \\ \mathbf{J}_{m_A} \, \mathbf{T}_{m_A} + \mathbf{N}_{m_A})] + \mathbf{J}_{m_B} \, \mathbf{T}_{m_B} + \mathbf{N}_{m_B}\} + \mathbf{j}_B \, \underline{\mathbf{t}}_B + \underline{\mathbf{n}}_B \,. \end{aligned} \tag{4.4}$$

By separating the propagation of the signal and propagation of the interference-plus-noise, I construct the SINR at the receiver, which is given by

$$\begin{aligned} \text{SINR} &= \frac{\|\hat{\underline{\mathbf{s}}}\|^2}{\|\hat{\underline{\mathbf{n}}}\|^2} \qquad (4.5) \\ \hat{\underline{\mathbf{s}}} &= (\mathbf{w}_B \odot \mathbf{h}^*_{m_B \to B})^\dagger \, \{\mathbf{H}_{m_A \to m_B} [(\mathbf{w}_A \, \underline{\mathbf{1}}) \odot (\mathbf{h}_{A \to m_A} \, \underline{\mathbf{s}}_A)]\}, \\ \hat{\underline{\mathbf{n}}} &= (\mathbf{w}_B \odot \mathbf{h}^*_{m_B \to B})^\dagger \, \{\mathbf{H}_{m_A \to m_B} [(\mathbf{w}_A \, \underline{\mathbf{1}}) \odot (\mathbf{J}_{m_A} \, \mathbf{T}_{m_A} \\ &\qquad + \mathbf{N}_{m_A})] + \mathbf{J}_{m_B} \, \mathbf{T}_{m_B} + \mathbf{N}_{m_B}\} + \mathbf{j}_B \, \underline{\mathbf{t}}_B + \underline{\mathbf{n}}_B \,, \end{aligned}$$

where $\underline{\mathbf{1}}$ is a row vector containing all 1s. Given the compound SINR, I can calculate the mesh relay system channel capacity C (b/s). This capacity is given by

$$C = B \, \log_2(1 + \text{SINR}) \,, \tag{4.6}$$

where B is the signal bandwidth observed at passband.

4.2 MIMO Beamforming Technique

Now consider a narrowband (flat-fading, nondispersive) MIMO system, notionally depicted in Figure 12 with perfect channel knowledge at both Mesh A (transmitter - CSIT) and Mesh B (receiver - CSIR).

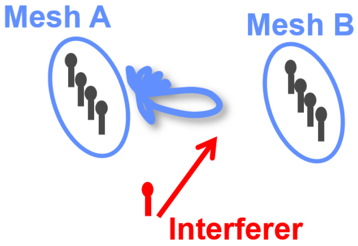


Figure 12: Basic mesh-oriented MIMO channel in the presence of an interferer.

The fundamental channel capacity in terms of spectral efficiency c (b/s/Hz) of the compound system is given by

$$c = \log_2 |\mathbf{I} + \mathbf{R}_{m_B}^{-1/2}\, \mathbf{H}_{m_A \to m_B}\, \mathbf{P}_{m_A}\, \mathbf{H}_{m_A \to m_B}^{\dagger}\, \mathbf{R}_{m_B}^{-1/2}| \,, \tag{4.7}$$

where $\mathbf{I}$ is the identity matrix, $\mathbf{R}_{m_B} \in \mathbb{C}^{n_B \times n_B}$, given by expectation

$$\mathbf{R}_{m_B} = \frac{1}{n_s} E[(\mathbf{J}_{m_B}\, \mathbf{T}_{m_B} + \mathbf{N}_{m_B})\, (\mathbf{J}_{m_B}\, \mathbf{T}_{m_B} + \mathbf{N}_{m_B})^{\dagger}] \,, \tag{4.8}$$

is the interference-plus-noise spatial covariance matrix, $\mathbf{H}_{m_A \to m_B}$ is the channel matrix that contains all the complex attenuations between each transmit and receive antenna, and $\mathbf{P}_{m_A}$ is the transmit covariance matrix [45]. The optimal solution to maximize channel capacity of Equation (4.7) is given by spatial water filling [45] when SNR is high. The specific solution is a function of the singular value spectrum of the whitened channel matrix $\mathbf{R}_{m_B}^{-1/2}\, \mathbf{H}_{m_A \to m_B}$ and the received SINR. In the rich channel and high SINR regime, I have solutions that are represented by transmit covariance proportional to the identity matrix. Space-Time Block Codes typically produce such a transmit covariance matrix. In the lower power and more correlated channel limit, the optimal solution is to use a rank-1 covariance, which is given by transmit beamforming. In the case of wideband (frequency-selective fading, dispersive channels) MIMO beamforming this solution becomes a rank that is equal to the number of taps in the filters [45, 4, 24]. The optimal transmit beamformer is given by evaluating the singular value decomposition (SVD) of the whitened channel matrix and using the dominant right-hand singular vector $\mathbf{v}_1$ so that the resulting transmit covariance matrix is given by

$$\mathbf{P} \propto \mathbf{v}_1 \, \mathbf{v}_1^\dagger \,. \tag{4.9}$$

The SVD is given by

$$\mathbf{U} \, \boldsymbol{\Sigma} \, \mathbf{V}^\dagger = \mathbf{R}_{m_B}^{-1/2} \, \mathbf{H}_{m_A \to m_B} \tag{4.10}$$

$$\mathbf{V} = [\mathbf{v}_1 \quad \mathbf{v}_2 \quad \mathbf{v}_3 \cdots] \tag{4.11}$$

$$\mathbf{U} = [\mathbf{u}_1 \quad \mathbf{u}_2 \quad \mathbf{u}_3 \cdots] \,, \tag{4.12}$$

where the columns of $\mathbf{U}$ contain the optimal matched receive beamformers, the diagonal matrix $\boldsymbol{\Sigma}$ contains the singular values, and columns of $\mathbf{V}$ contain the optimal transmit beamformers [29]. In simple line-of-sight environments, the beamforming effect of the transmit beamformer may look like a traditional, if sparse, beam pattern. However, in more complicated environments, it may be difficult to interpret the beam pattern from a traditional beamforming perspective. In practice, directly estimating the interference-plus-noise covariance required is challenging with limited or no exchange of IQ samples between mesh elements.

Although it is tempting to invoke reciprocity in evaluating the bi-directional version of this system, since $\mathbf{H}_{m_B \to m_A} = \mathbf{H}^T_{m_A \to m_B}$, it is worth noting that the whitened channel matrix $(\mathbf{R}^{-1/2}_{m_B}\mathbf{H}_{m_A \to m_B})$ does not observe reciprocity.

4.3 Compensated MIMO Beamforming Technique

In the prior section, I discussed the optimal MIMO beamforming technique in isolation. Typically, an array of receive antennas would be wired to a central processing system [45]. However, the distributed system does not have wired connections between Mesh B and the final receiving node. Since the system operates through over-the-air coherent combining, to use the previous beamformer formulations, I must compensate them for the effects of the channel from transmitter to Mesh A $(\mathbf{h}_{A \to m_A})$ as well as the effects of the channel from Mesh B to the receiver $(\mathbf{h}_{m_B \to B})$. The channels can be considered as imperfect wires with some amplitude scaling and phase rotation as if the combining occurred locally through wires. Consequently,the receive beamforming is implemented as transmit beamforming the signal to the final single node with appropriate complex gains such that

$$\mathbf{u}_1^* = \mathbf{w}_B \odot \mathbf{h}_{m_B \to B}\,, \tag{4.13}$$

where $\mathbf{w}_B \in \mathbb{C}^{n_B \times 1}$ is the actual applied beamformer taking into account effects of $\mathbf{h}_{m_B \to B} \in \mathbb{C}^{n_B \times 1}$, the channel between the elements of Mesh B and the final receiving node.

This is achieved in the flat-fading MIMO beamforming case by taking the dominant right and left singular vectors and dividing by the appropriate channel tap as shown in Equations (4.14) and (4.15) for the Mesh A and Mesh B beamformers respectively.

$$\mathbf{w}_A = \text{diag}(\mathbf{h}_{A \to m_A})^{-1} \, \mathbf{v}_1 \tag{4.14}$$

$$\mathbf{w}_B = \text{diag}(\mathbf{h}_{m_B \to B})^{-1} \, \mathbf{u}_1^* \tag{4.15}$$

Mitigating interference incident on Mesh A continues to be difficult. There are several possible solutions, such as an constrained minimization approach with the existing beamformer to minimize the impact of the interference received at Mesh A. It may also be possible to whiten the MIMO channel matrix again by the interference plus noise covariance at Mesh A, however this would require other constraints which are beyond the scope of this work. I now leverage the previously discussed internal mesh beamforming as a solution to this challenge.

4.3.1 Internal Beamforming

The multi-stage relay system is able to mitigate interference incident at Mesh B through the whitening of the MIMO channel matrix; however, this same approach does not work for interference incident at Mesh A.

Internal beamforming utilizes a surrogate receiver element to perform an intermediate beamforming stage at Mesh A to mitigate interference spatially before retransmission to Mesh B. This is performed by first using $n_A - 1$ elements of Mesh A to spatially mitigate the interference through an MMSE transmit beamforming process. These $n_A - 1$ elements transmit to the n_A^{th} element acting as a surrogate receiver. The signal is then re-transmitted from the surrogate receiver to the other elements of the mesh with significantly reduced interference power. By relaying internal to the mesh, the second link is shortened, such that it falls into the mesh noise dominant case, allowing it to null interference and regain power allocation to be used for the signal of interest. Then, returning to the original mesh, the system now operates in the secondary mesh receiver noise dominant case, allowing the desired SNR gain to be realized.

4.4 Results

4.4.1 MATLAB Simulation

I simulate the aforementioned compound 1 to n_A to n_B to 1 system with and without internal beamforming utilized. The theoretical SNR gain for the simulated system as compared to an equivalent SISO link is approximately 18 dB. The internal beamforming algorithm is implemented at Mesh A using the MMSE transmit beamforming with $n_A - 1$ elements.

Table 1: Distributed Mesh System Simulation Parameters

n_A	4
n_B	4
Average SNR at Mesh A Elements	20 dB
Average SNR at Mesh B Elements	0 dB
Average SNR at Receiver	20 dB
INR at Mesh A	20 dB
INR at Mesh B	20 dB

I simulate with interference present at different stages, and the results with and without internal beamforming applied. Note that this is a version of the previously discussed intermediate distance problem, thereby also limiting the realizable SNR gain.

The simulated system performance is shown in Figure 13. It can be seen that

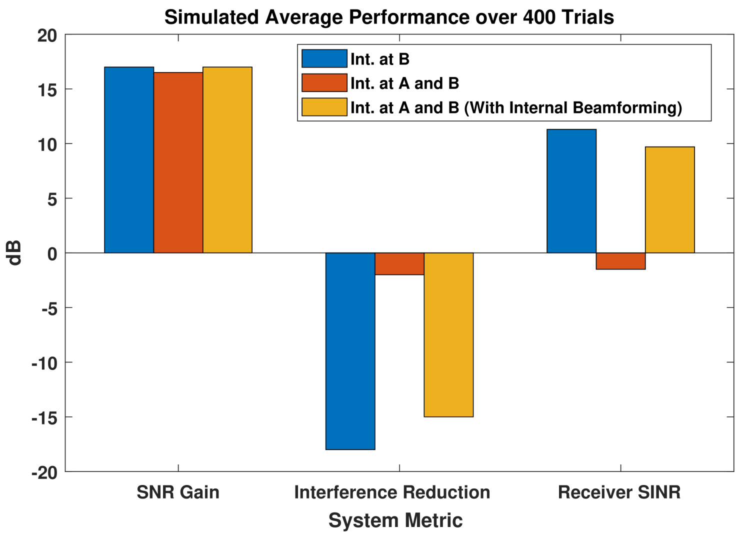


Figure 13: Performance Results from 400 trials of simulation of the internal beamforming multiple stage relay system

when interference is incident at both Mesh A and Mesh B, the interference reduction performance drops off significantly as the interferer power at Mesh A gets forwarded through and dominates the system. When internal beamforming is applied, the system regains close to the prior performance with some small penalty of any residual interference that was not completely mitigated in the internal beamforming step.

4.4.2 WISCANet Experiments

This experimental setup was a single node to four-node mesh to four-node mesh to single node (1-4-4-1). The procedure implemented follows the discussed “bent pipe" model. The meshes function as relays that pre-distort the received signal before retransmitting such that the signals coherently combine at the next receiving stage. Each node within the mesh extracts the signal to be forwarded, adds its unique training sequence, applies a beamformer weight, and sends this waveform over the air.

I used the WISCANet Software Defined Radio Network [48, 20] to prototype the algorithm in under two weeks, and substantial extensions which will be described in Chapter 7. This proof of concept uses sets of NI Ettus X310 radios as surrogates for the single and mesh nodes. This experiment operates at 907 MHz in the ISM band. I ran these experiments by using approximately 1 MHz of bandwidth. Genie channel feedback is facilitated by utilizing WISCANet's stop-and-go functionality and transferring information between radios before the next step in the transmission cascade.

Now consider the scenario in which there is a single node to 4-node mesh to 4-node mesh to single node network (1-4-4-1), similar to the network depicted in Figure 1. In Figure 14, I present the experimentally measured ratio of the coherent-distributed-network SNR improvement over the average SISO link. I provide an evaluation of the ratio of beamformed-to-SISO SNR over three separate runs. Both meshes provide beamforming but are otherwise sampled signal bent-pipe relays. In the first trial of each run, the channel is not known; thus, signal power is incoherently combined. I expect an improvement of about 6 dB in this incoherent case. Because the distribution of the channel attenuations is not ideal and SNR estimates are imperfect, the SNR ratio is slightly less. For trial numbers 2, 3, 4, and 5 the system has channel estimates, and the SNR gain averages near the ideal behavior of $4^3 = 18$ dB. Again, measurement fluctuations and channel non-idealities cause slight variations.

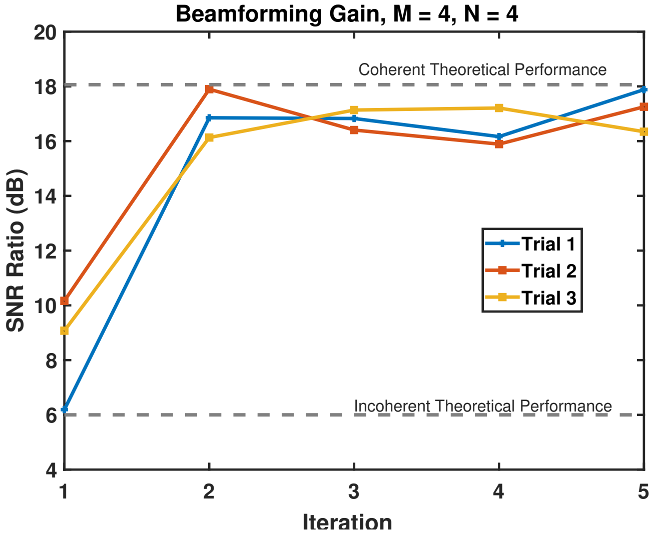


Figure 14: Experimental validation of the ratio of beamformed-to-SISO SNR over three separate runs for a single node to 4-node mesh to 4-node mesh to single node network. The channel is unknown for the first trial of each run, so the combining is incoherent.

Chapter 5

SIMULATION

I constructed a MATLAB simulation to explore the performance of the algorithm and augmentations in practical environments and with system limitations. It is fully parameterized enabling efficient parameter space exploration. The simulation platform enables rapid iteration and experimentation with algorithmic adaptations. The MATLAB code can also call the C++ beamformer implementation used in the real-time implementation for validation and testing.

The simulation loop starts with the transmission of a fresh training waveform from the initial transmitter. Each cycle tracks the propagation of an epoch of the transmitter's data stream through all stages of the relay. The cycle ends once this stream has been received and processed by the final receiver. Cycles are split up into a transmitter to mesh stage and a mesh to receiver stage. In between stages, the beamformer is computed, although in a practical implementation, the filters are constructed in parallel with signal relaying. The beamfomer computed using the current epoch is used on the next propogation cycle. All simulations will be performed with 10 simulated mesh relay nodes.

The simulation uses an underlay waveform for channel estimation and correlation estimation. This approach was selected for exploration into the idea of leveraging the algorithm for smart-antenna type systems. This type of system augments an existing system's capabilities without modification to its waveform or underlying operations. The affects and limitations of this approach on transmit beamforming performance are explored in [29]. The underlay waveform consumes eight times the spectral support than the payload waveform. I oversample the payload waveform by a factor of 2 and compute the beamformer at an oversampling factor of 1.5.

In the simulation, I set an update rate of 25 ms to target a system that can operate in environments with mobility and imperfect frequency correction. The initial transmitter's underlay sensing signal repeats every 25 ms, so that information is availble to construct a new beamformer at the same interval. Channel B sounding sequences are sent on a 2.5 ms interval to support the use of 10 tiles in the same update rate.

A practical system would require the use of feedback waveforms to communicate channel sounding and other estimation data products. In the simulation, I leverage a genie channel, and so do not directly simulate the signals required to communicate information through the mesh. Accordingly, I am able to shorten transmissions and the analysis window to reduce computational load. In addition, this assumption allows easy sweeping across the number of samples that need to be transmitted across the network enabling analysis of network rate limitations on the system performance.

The interference is a Gaussian signal that spans the entire frame's duration. I perform the analysis with the interference occupying the same spectral support of the payload waveform to assess the interference rejection capabilities solely attributed to the beamformer. The signal is upsampled and filtered to tightly confine energy within the spectral support of the payload waveform. Of course, in reality there is no control over interference.

A key step in bent pipe processing is normalizing the relay signals to meet the required transmission power that ensures sufficient SINR and meets expected SNR gain performance. In simulation, the receive filtered bent pipe signals are rescaled based on the L2 norm such that the average power across elements is unity, while maintaining the power relationships required for the unconstrained MMSE solution. The beamforming filters are normalized in the same way prior to resampling. The channel sounding underlays are normalized as if they were transmitting simultaneously with unity power. After ensuring a common reference power in all of the component signals, each can be weighted directly according to the prescribed underlay to signal ratio (USR) prior to summing. The proper USR could not be guaranteed without these intermediate normalization steps. This normalization scheme is best suited for determining nominal performance of the algorithm in simulation. However, the optimization problem I solve does not impose any constraints on the maximum power. The algorithm only implicitly imposes an average power constraint.

The algorithm is tested using several mesh node arrangements, including a linear array with a max distance of 200 m and random positions within a bounded box generated according to a uniform distribution. The simulation supports positioning in 3 dimensions, but the network is typically placed in the XY plane. The mesh is centered about the origin. I place interferers into the environment at some distance from the origin but at uniformly random angles of arrival. To validate the methodology and set a performance baseline, the simulation is ran with all nodes static. Even with static nodes, frequency offsets can still occur. Specifically, I model residuals from candidate frequency estimation and correction algorithms. To establish nominal performance, I simulate the system under time-frequency aligned conditions.

Wideband channels are generated based on a Rician distribution that also include the effects of relative time offsets, differences in attenuation, and phase delays dictated by the generated geometry. Impulse responses are directly generated at the system sampling rate. First, each channel for a particular transmission stage is created time aligned with each other and the sampling lattice. This structure is given by

$$\underline{\mathbf{h}}_{Rician} = \frac{\underset{n}{\max} D_n}{D_m} \left(\sqrt{\frac{K}{1+K}} \begin{bmatrix} e^{i2\pi f_c/cD_m} & 0 & \ldots & 0 \end{bmatrix} + \sqrt{\frac{1}{1+K}} \underline{\mathbf{h}}_{scatter} \right) , \tag{5.1}$$

where the fraction $\frac{\underset{n}{\max} D_n}{D_m}$ represents a relative attenuation based on the distances between a initiator node and the m^{th} node D_m. The scattering component $\underline{\mathbf{h}}_{scatter}$ is a sequence of i.i.d. complex Gaussian samples weighted by a taper that decreases the receive power of signals traveling through longer paths. Parameter K modulates the line of sight power relative to the scattering response power. Then, the impulse responses are zero padded such that relative time offsets (rounded to the nearest whole delay) of the line of sight component are included. Afterward, each impulse response is passed through a fractional delay filter to complete the time shift and accurately embed TDOA information within the channels. For baseline testing purposes, the channel impulse responses are static between loops.

5.1 Results

These simulation results use practical system parameters such as 125 covariance samples and 12000 cross-correlation/channel estimation samples. The maximum mesh pairwise distance is 200 meters. The underlay to signal ratio (USR) for long feedback link communications is -6 dB and the USR for short feedback link communications is -10 dB. I use an interferer, with 20 dB INR incident at the mesh, that occupies the same spectrum as the signal of interest. In the mesh noise dominant case, the transmitter to mesh has a 0 dB signal of interest SNR, and the mesh to final receiver has a 20 dB signal of interest SNR. This operating regime is the most difficult case for a practical system with limited resources to address.

5.1.1 Mesh Relay Noise Dominant Regime

The initial transmitter transmits a 5 MHz payload signal with a 40 MHz underlay, where the USR is -6 dB. Both spectra are clearly visible in the power spectral density (PSD) of the transmission plotted in Fig. 15. The underlay-to-payload bandwidth ratio is approximately a factor of 8.

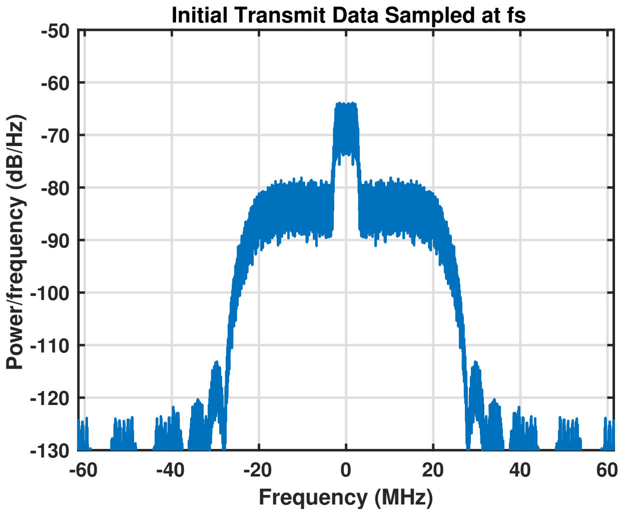


Figure 15: Transmitter Power Spectral Density

Below is the cross-correlation that each mesh node computes blind to each other. The cross-correlation plotted in Fig. 16 is of the first stage transmission. Significant noise and interference power can be detected. The peaks of the cross-correlation are less than 10 dB above the correlations associated resulting when the underlay is not aligned with the data. Clearly, running an anti-aliasing filter through this sequence would cause misaligned correlations to overwhelm the integration. Thus, postprocessing is needed.

I apply a tight window around the peaks of interest, which suppresses the contri-

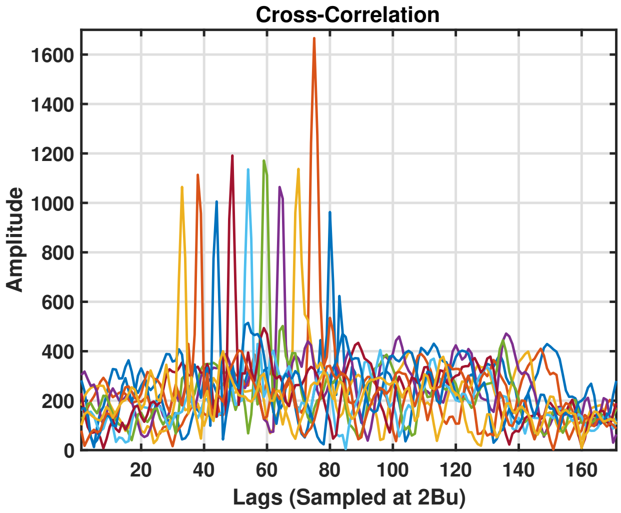


Figure 16: Full Rate Mesh Relay Cross Correlation

butions from misaligned correlations to the anti-alias filter. A side effect of this step is that it imposes a stronger requirement that the channels are line-of-sight dominance. I assume that signals within the bandwidth do not experience strong multipath and are significantly line-of-sight dominant.

The target sampling rate is 7.5 MHz. From 80 MHz, the downsampling factor is $3/32$. One sampling interval at 7.5 MHz corresponds to $10\frac{2}{3}$ samples of a signal sampled at 80 MHz. I observe that the delay spread shown in the above cross-correlation is within approximately 11 samples. Consequently, the delay spread at 7.5 MHz is expected to be approximately 1 sample. After resampling, some peaks should fall between samples, so there should be energy split between lags. After windowing, running the result through an anti-alias filter, and decimating, the result is shown in Fig. 17. Notice that there is a single peak in each of the downsampled results. There is a taper that results from decimating the sinc shape of the anti-alias filter.

I perform cross-correlation to detect the arriving TDMA channel B underlay. The cross-correlation associated with each mesh relay node's transmission is depicted in

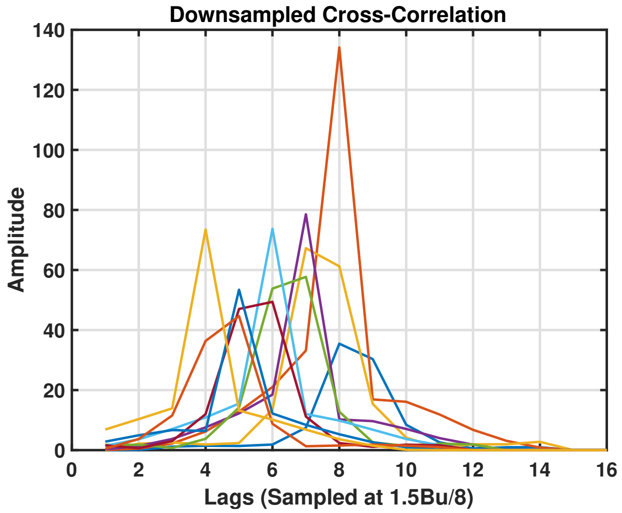


Figure 17: Mesh Relay Cross Correlation Downsampled

Fig. 18. In simulation, long data buffers are kept that are unrealistic to maintain. However, this is not a large departure from detecting within a sequence of smaller windows. One other point of departure is that each mesh node uses its own training sequence in the underlay, so that when the final receiver employs detection, only one peak results from each training sequence over the entire sounding cycle. It is demonstrated here that the sounding underlay arrives roughly according to the same interval as each node takes turns transmitting. Observe one peak correponding to each node. This cross-correlation contains strong peaks, confirming that the second stage transmission is a strong SNR link.

The time of arrivals are recorded and realigned assuming the ideal channel sensing transmission sequencing. The result of this realignment is depicted in Fig. 19. A short window of the cross-correlation centered around each detection is used. This effectively implements the same windowing technique. The aligned cross-correlations are zero padded.

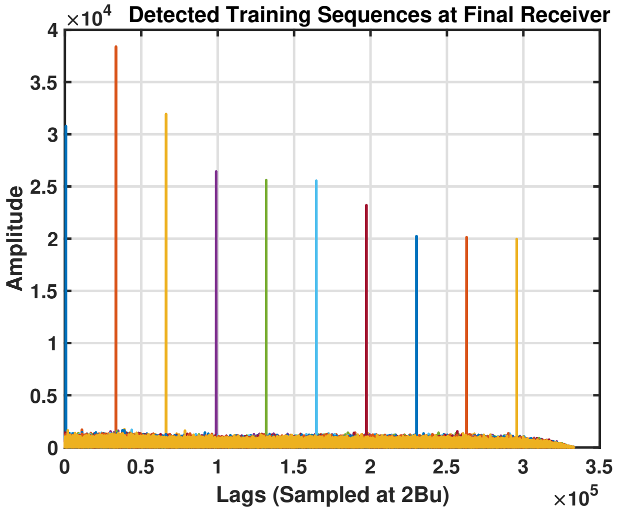


Figure 18: Receiver Detected Training Sequences

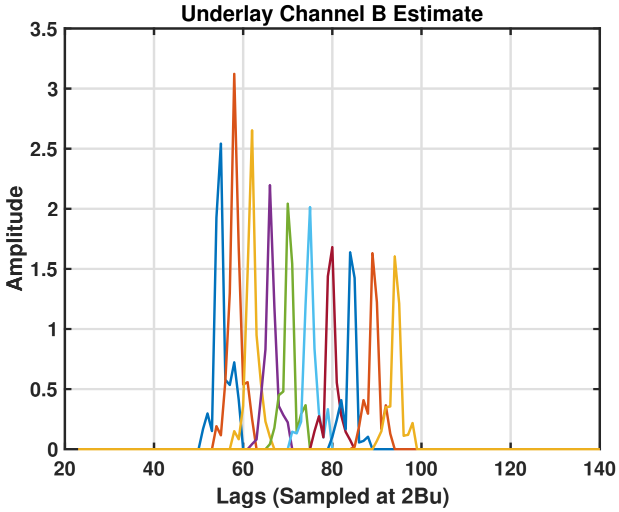


Figure 19: Channel B Estimate

The downsampled result is shown in Fig. 20. Notice that the delay spread after realignment spans approximately 10 samples here. Naturally in the downsampled version, energy should be seen to be distributed across taps depending on which part of the coarser lattice the 10 peaks map to.

Excerpts of the captures $\underline{\mathbf{z}}_n$ corresponding to the cross-correlation estimates made

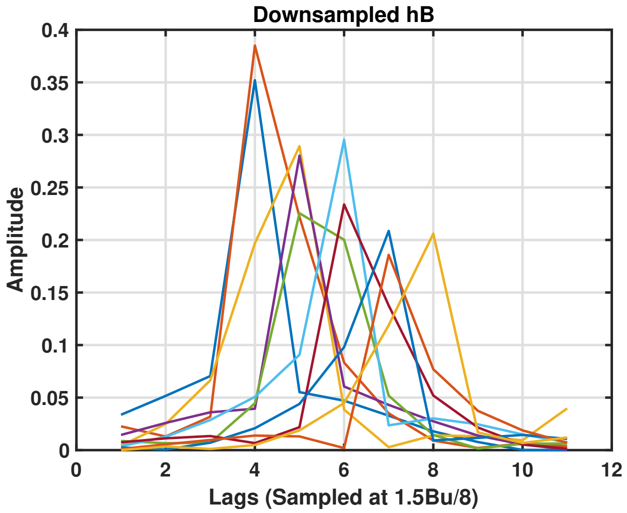


Figure 20: Channel B Estimate Downsampled

are then downsampled to a sampling rate of 7.5 MHz. The resulting length of each $\hat{\underline{\mathbf{z}}}_n$ is 125 samples. A covariance matrix corresponding to that cycle is created. That covariance matrix is incorporated to the average. A corresponding eigenvalue distribution is plotted in the Fig. 22. That averaged covariance matrix undergoes regularization. Its corresponding eigenvalue spectrum is also plotted in Fig. 21. Responses corresponding to the interference and signal of interest are visible in both eigenvalue spectra.

The beamformer is constructed by inverting the covariance matrix and applying it to the cross-correlation vector. Each filter is parsed from the resulting vector. The result is normalized based on the L2 norm to best assess performance. The filters are plotted together in Fig. 23. Within the filter is a structure that effectively corrects for timing misalignments introduced by the compound channels.

This filters are then upsampled to the system sample rate to be used in the

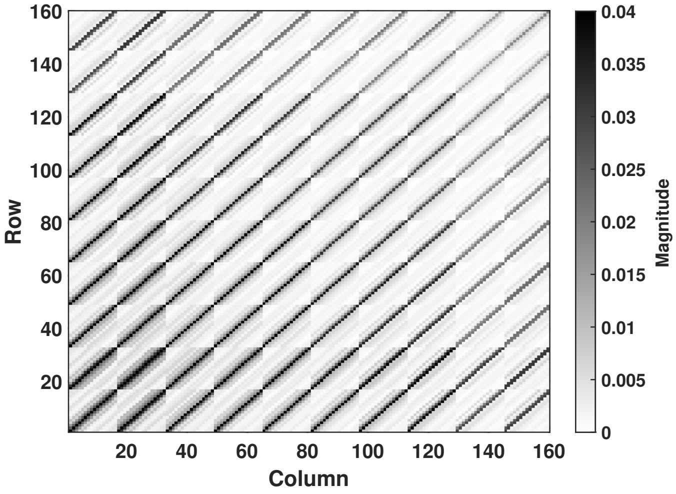


Figure 21: Averaged Covariance Estimate

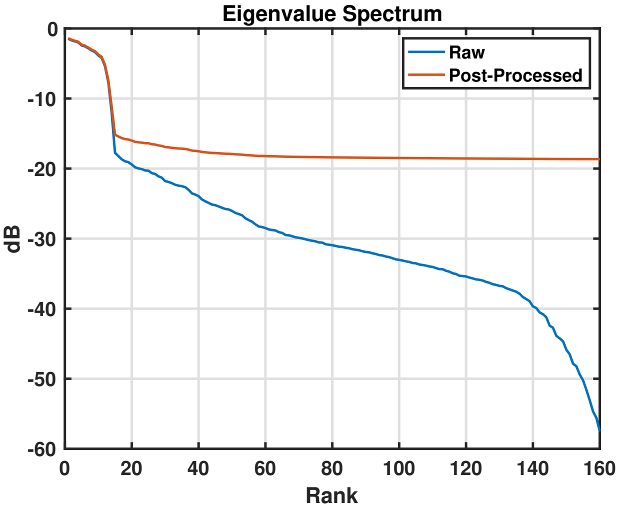


Figure 22: Covariance Estimate Eigenspectrum

bent pipe processing path. The result is depicted in Fig. 24. Notice more sinc-like structures as opposed to peak characteristics. This is attributed to the anti-alias filter. Additionally, it indicates that these filters affect a subset of the system's spectral support as it implies some low-pass behavior.

The received PSD at the final receiver immediately following receiver filtering is

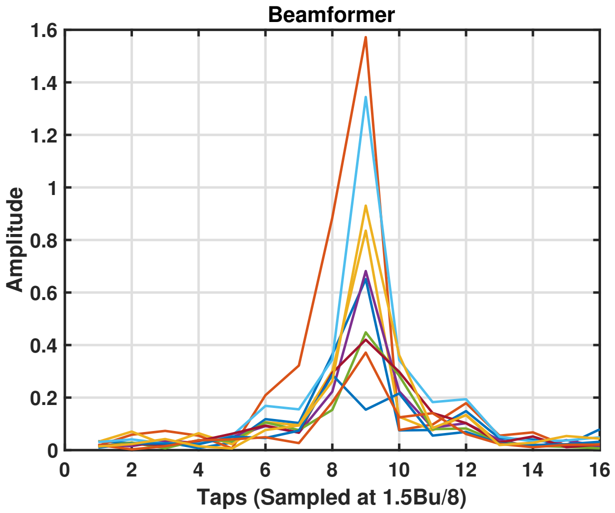


Figure 23: Computation Rate Beamforming Taps

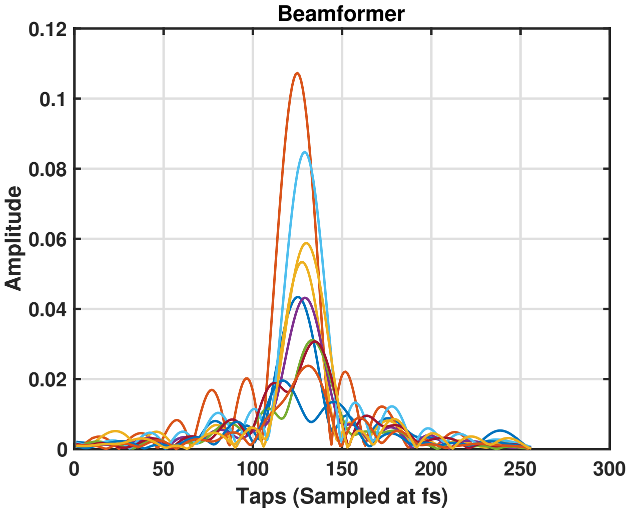


Figure 24: Full Rate Beamforming Taps

depicted Fig. 25. There is a clear structure attributed to underlay and information that passed through the bent pipe. The received signal of interest is quiet relative to the underlay because the interferer took up a significant amount of power and was rejected.

The PSD of the final reception on the first cycle (without beamforming) is dis-

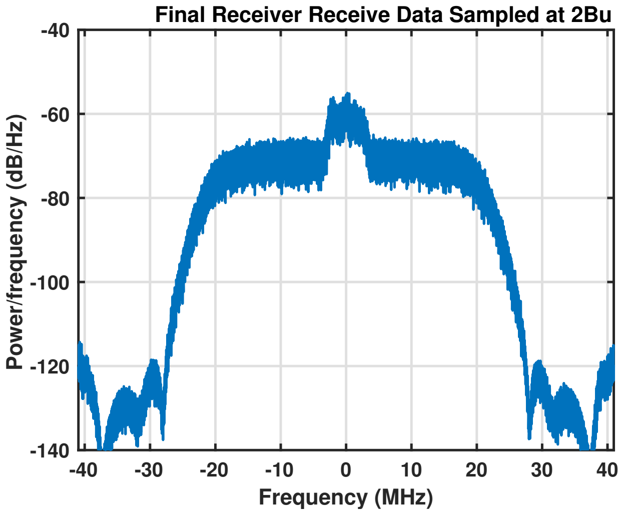


Figure 25: Receiver Received Signal Power Spectral Density

played in Fig. 26. Without beamforming, more energy is present in the frequency band that the beamformer is effecting. This excess energy is attributed toward the interference. Additionally, significant frequency selective fading is observed because the non-optimized beamformer does not correct for the significant multi-path that naturally results from transmitting to mesh nodes and onward to the final receiver. When comparing Fig. 26 and Fig. 25, observe that the beamformer has fixed the signal of interest's spectrum.

Several metrics must be tracked to completely assess performance of the algorithm. These metrics include the SNR Gain, INR, and SINR measured over sequential beamformer construction cycles. I compare the measured performance with theoretical bounds. I compute the theoretical SNR gain that can be achieved. Additionally, I compute the theoretical INR that results from exact incoherent combination. From that exact incoherent INR, I can project a target INR that the system must achieve through beamforming. This target is set 20 dB below the perfect incoherent INR. All metrics are plotted in Fig. 27. One cycle represents the construction of one set of

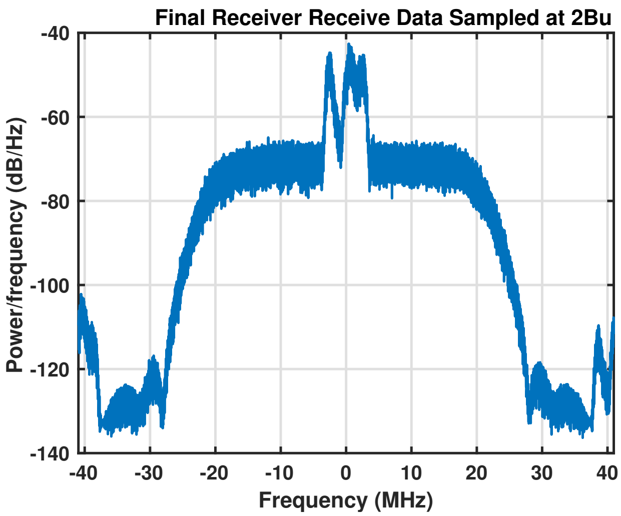


Figure 26: Receiver Received Signal Power Spectral Density without beamforming optimization

beamforming filters. In other words, the initial transmitter has sent a new training sequence, which prompts the start of a new construction.

Mesh relay nodes do not have all the necessary information to compute a proper beamformer on the first cycle, so I expect performance to be poor. On the next cycle, a proper channel B estimate has been made available to the mesh. From that point on, the estimate quality may vary depending on the SNR gain and interference rejection performance that can be achieved, which affects the UINR at the final receiver, and therefore its estimation quality. As long as the mesh has sufficiently accurate channel B information, ideal SNR gain can be achieved right away.

On the other hand, the mesh requires several cycles worth of $\underline{\mathbf{z}}_n$ data to be incorporated into the covariance average before a useful covariance matrix can be obtained. Consequently, several iterations need to occur before the desired interference rejection capabilities can be realized.

First consider the algorithm's performance in a simulated mesh noise dominance case. The SNR gain bound is 12.5 dB and the theoretical INR resulting from perfect incoherent combination is 19 dB. Notice that in the first cycle the SNR Gain does not improve at all over the average SISO SNR. Additionally, the measured INR is close to the theoretical computation. On the second cycle, the expected SNR gain is achieved, albeit ∼1 dB off from theoretical. In fact this performance is essentially maintained through the rest of the trial.

The INR falls over time after the first cycle. The INR hits the target rejection performance at cycle 5. The INR wavers around 0 dB and eventually drops below it, meaning the beamformer has buried the interference below the noise floor. Thus, it is demonstrated in Fig. 27 that the algorithm achieves the desired performance in terms of payload waveform gain and interferer suppression.

In Figure 28, I compute the INR as seen at many points in space around the receiver and mesh nodes. This algorithm is characterized by the interfence null only being realized at one point in space, and the interference incoherently combining during retransmission at all other points.

5.1.2 Receiver Noise Dominant Regime

The performance of the algorithm under the final receiver noise dominant regime is summarized in Fig. 29. The simulation that yielded this result used new randomly thrown initator positions, but the initial transmitter is 200 m away and the final receiver is 50 Km away. The expected result in this situation is primarily to achieve 20 dB of SNR gain. The INR should be reduced to the noise floor as well. The SNR gain begins at the perfect incoherent combining bound, which is given by the number of nodes present in the mesh. On the first cycle, the INR is naturally below the

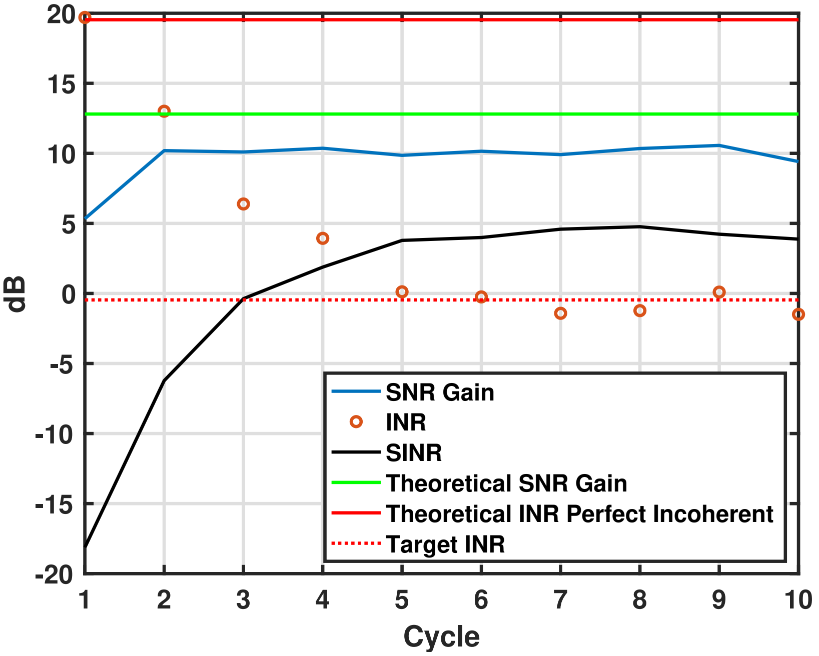


Figure 27: Per Cycle System Performance Evaluation in the Mesh Noise Dominant Regime

noise floor, approximately 7 dB below the perfect incoherent combining theoretical INR. The beamformer does not need to work as hard to suppress the interference as the received signal of interest power is easy to increase to maximize the SINR. The beamformer simply needs to avoid helping the interference at this point. Beyond the first cycle, the beamformer successfully increases the SNR gain toward the theoretical bound and the INR is held below the noise power, save for cycle 6 when there is a lapse in efficacy. The SINR increases from 0 dB in the first cycle to an average of 11 dB for the remainder of the simulation.

In Figure 30, I show the payload SNR computed at many points around the receiver. This shows the effective far-field array response of the virtual array created by the distributed coherent mesh. The side-lobe performance is generally about -15 dB

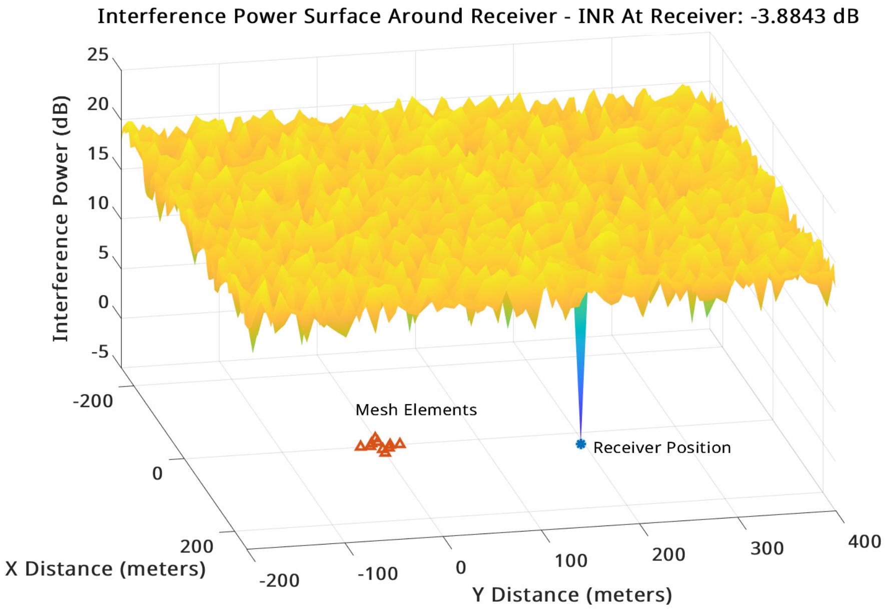


Figure 28: INR Surface evaluated around the receiver showing the interference combining at all locations but the receiver

from the peak. There are several strong sidelobes at about −7 dB from the peak gain. There will be some spatial aliasing as a result of the distributed array configuration.

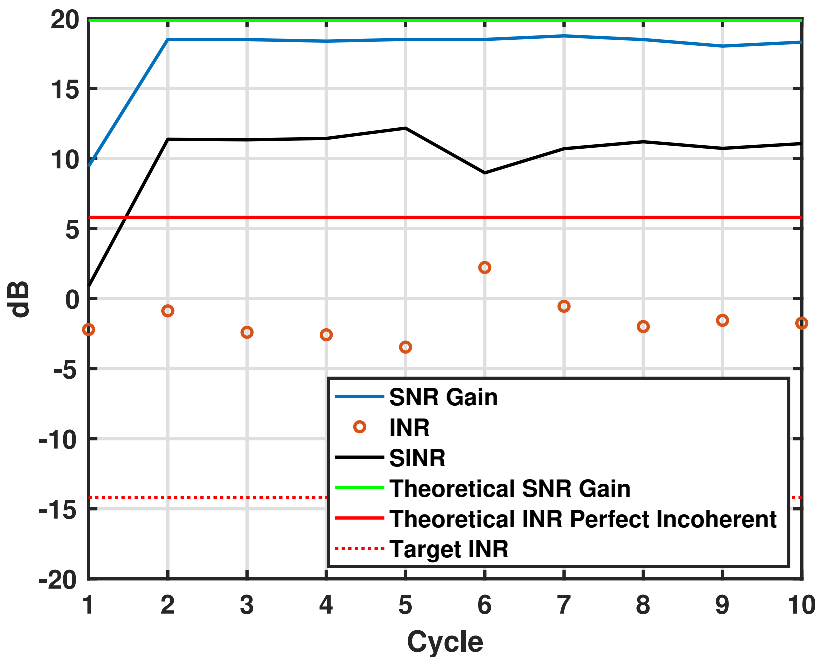


Figure 29: Per Cycle System Performance Evaluation in the Receiver Noise Dominant Regime

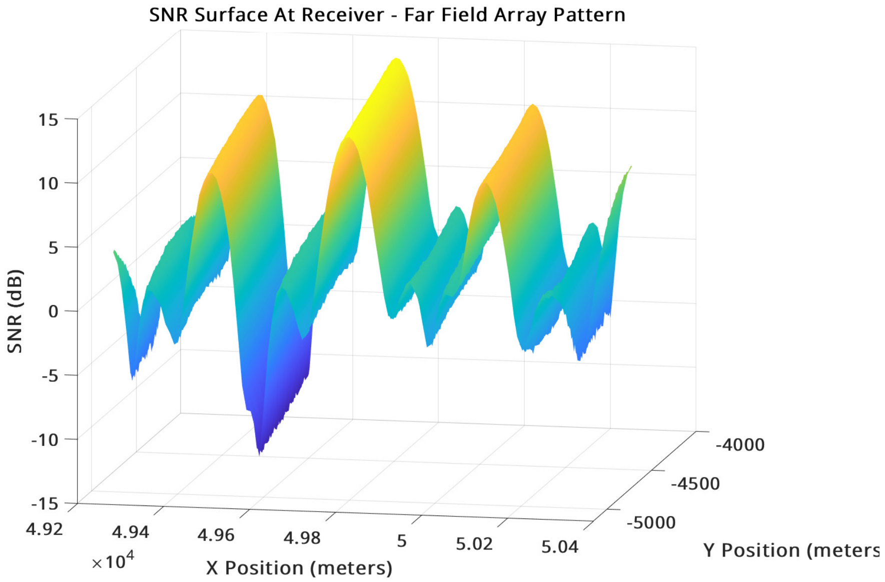


Figure 30: SNR Surface evaluated around the receiver showing the far field response of the transmit beamformer

Chapter 6

EXPERIMENTS

6.1 Pseudo-Real-Time WISCANet Experiments

I demonstrate my novel approach through a small-scale proof of concept experiment. The procedure implemented follows the “bent pipe" model. The meshes function essentially as relays that affect the received signal before retransmitting such that the signals coherently combine at the next receiving stage. Each node within the mesh extracts the signal to be forwarded, adds its unique training sequence, applies a beamformer weight, and sends this waveform over the air.

I used the WISCANet Software Defined Radio Network [48, 20] to prototype the algorithm in under two weeks. WISCANet has been upgraded to support $M \times N$ phase coherent transmit and receive [20]. This proof of concept uses sets of NI Ettus X310 radios as surrogates for the single and mesh nodes. This experiment operates at 907 MHz in the ISM band. I ran these experiments by using approximately 1 MHz of bandwidth. Genie channel feedback is facilitated by utilizing WISCANet's stop-and-go functionality and transferring information between radios before the next step in the transmission cascade. This configuration leverages shared 10 MHz and 1 PPS signals between each element to ensure synchronization in time and frequency.

I first consider the scenario in which there is a single node to 4-node mesh to single node network (1-4-1). In Figure 31, I present the experimentally measured ratio of the coherent-distributed-network SINR improvement over the average SISO link. I provide an evaluation of the ratio of beamformed-to-SISO SINR over three separate runs. The mesh provides beamforming but is otherwise a sampled signal bent-pipe relay. In the first trial of each run, the channel is not known; thus, signal power is incoherently combined. I expect an improvement of about 6 dB in this case. Because of the introduction of interference, a smaller SINR gain than the expected 6 dB is achieved. For trial numbers 2, 3, and 4, the system has the channel estimates, and the SINR gain averages above the ideal SNR performance of $4^2 = 12$ dB. This achievement above (but near) the theoretical SNR performance is due to the interference nulling effects of the beamformer.

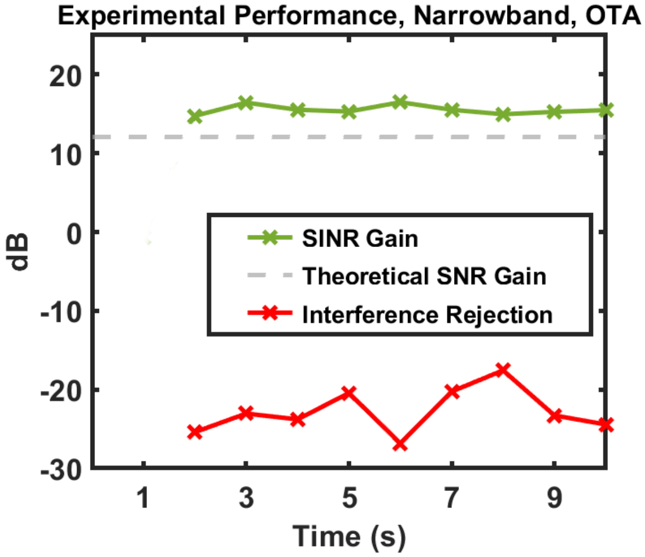


Figure 31: Experimental validation of a narrowband beamformer implementation of the ratio of beamformed-to-SISO SINR given for a single node to 4-node mesh to single node network with an interferer received at the mesh. The channel is unknown for the first trial, so the combining is incoherent. Note: The measurement is of Signal to Interference+Noise Ratio (SINR) gain, but the theoretical limit is formulated as Signal to Noise Ratio (SNR) Gain.

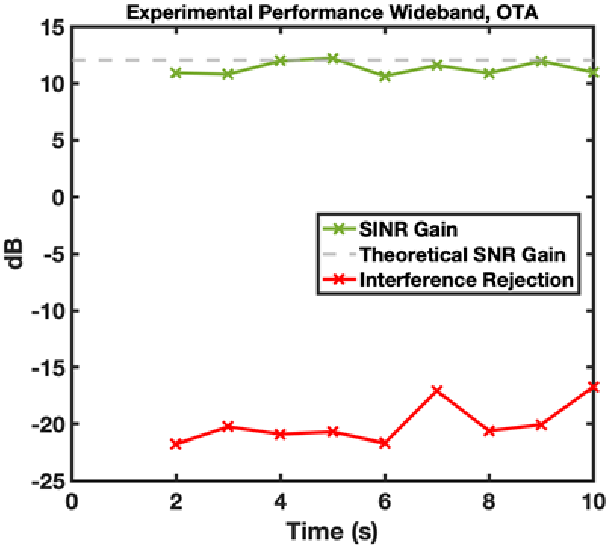
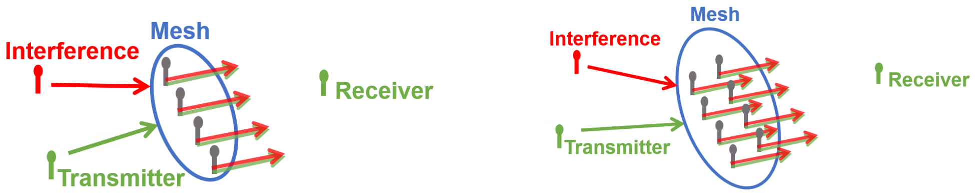


Figure 32: Experimental validation of a wideband beamformer implementation of the ratio of beamformed-to-SISO SINR given for a single node to 4-node mesh to single node network with an interferer received at the mesh. The channel is unknown for the first trial, so the combining is incoherent. Note: The measurement is of Signal to Interference+Noise Ratio (SINR) gain, but the theoretical limit is formulated as Signal to Noise Ratio (SNR) Gain.

6.2 Real-Time Over-the-Air Experiment

I developed a real-time over-the-air distributed coherent beamforming system using Ettus USRP X310 software defined radios. I implemented both the spatial and spatio-temporal algorithms discussed previously. I constructed two prototype distributed coherent mesh beamforming systems, one with 4 elements and one with 8 elements as shown in Figures 33a and 33b. They share the same software that can be configured for arbitrary numbers of elements, as well as being configurable for many other system parameters and design decisions.



(a) 4 Element System Design

(b) 8 Element System Design

Figure 33: System Designs

The system is written in C++ using the UHD library to interface with the radios. The Arrayfire and Armadillo libraries are used to accelerate computations and enable the system to meet timing deadlines. This system is optimized for fast superscalar processors, such as the Intel i7 and i9 families or AMD Ryzen 7 or 9 series. ZeroMQ is used to facilitate backchannels between the mosaic and final receiver. It is very low latency and run in a separate thread to prevent blocking. The system uses a total loop (transmitter to final receiver) time of 120 ms. This is effectively real-time, enabling robustness to channel motion and a dynamic environment. Each burst stage (Transmitter to Mesh, and Mesh to Receiver) takes 60 ms. The constructed prototype system in the lab is shown in Figure 34.

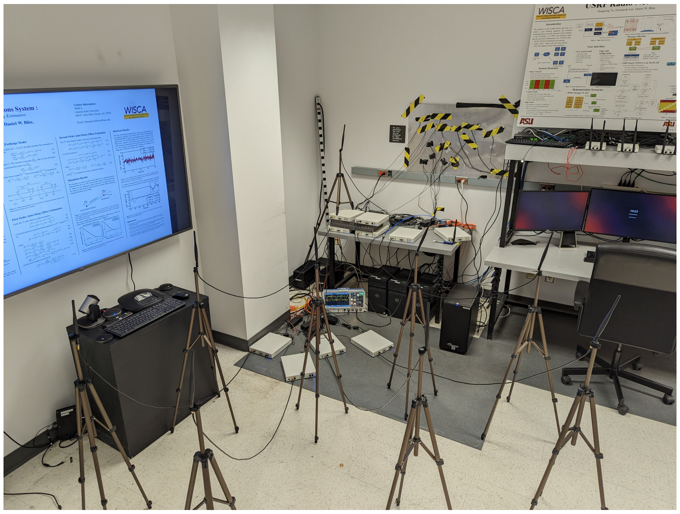


Figure 34: Mesh and Transmitter and Interference Source

6.2.1 Waveform Design

The transmitter node sends a training waveform that is used for estimation of the transmitter to each mesh tile channel. This same waveform is also used for estimating frequency offsets from the transmitter to the mosaic. This waveform structure is shown in Figure 36.

The training sequence is designed to be of sufficient length to yield the required integrated SNR for channel estimation and the phase estimation that feeds the frequency estimation algorithm. The transmit to mosaic receive stage is able to use the data from its reception immediately, and so the delay in information use for those frequency estimates and channel estimates is the guard interval, which is typically very small, on the order of microseconds. The waveform from the mesh transmit to receiver is shown in Figure 37.

For the second stage, the training is transmitted in CDMA fashion, so careful

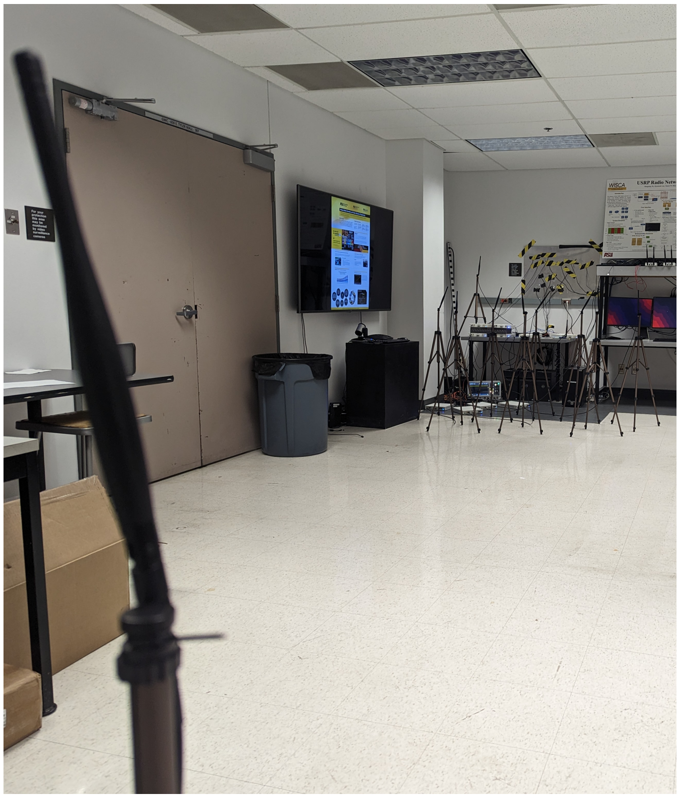


Figure 35: Receiver looking back at Mesh, Transmitter and Interference

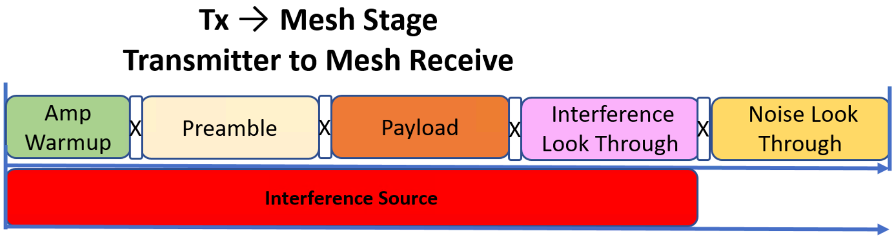


Figure 36: Transmitter to Mesh Waveform Structure

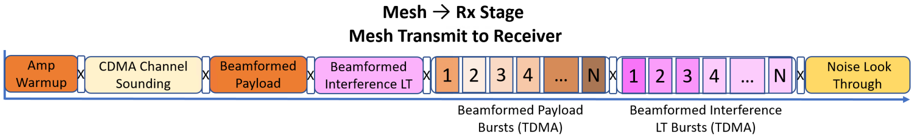

Mesh → Rx Stage
Mesh Transmit to Receiver
Amp Warmup
X
CDMA Channel Sounding
X
Beamformed Payload
X
Beamformed Interference LT
X
1
2
3
4
...
N
X
1
2
3
4
...
N
X
Noise Look Through
Beamformed Payload Bursts (TDMA)
Beamformed Interference LT Bursts (TDMA)

Figure 37: Mesh Transmit to Receiver Waveform Structure

waveform design is important. It is imperative to consider the wideband nature (asynchronous CDMA) and the interference, so carefully selecting training sequences to be approximately orthogonal is required. The lookthrough periods are leveraged to easily estimate system performance. The SISO retransmissions also allow easy performance comparison.

6.2.2 Time Alignment Approach

My approach does not require GPS for time or frequency synchronization. However, I leverage it to provide a coarse synchronization to accelerate our experimental development and reduce the search space. GPS-Disciplined Oscillators (GPSDOs) from Jackson Labs (specifically the LC-XO) are used to provide a synchronous GPS timestamp to each node in the system. The 10 MHz references are unlocked from the GPS constellation and the GPS disciplining disabled, by leveraging modifications we made to the UHD drivers. This forces the reference clock on each node to free-run, and in turn now all the clocks in the X310 (including synthesizers) are free-running, ensuring the continuous phase requirement. The sample clocks are also all synchronized to the same clock, ensuring that the sample times are all within the same sample, with minimal subsample misalignment.

6.2.3 Frequency Alignment

Frequency estimates are performed using phase estimates from the training sequence. They are made by comparing the estimated phase from burst to burst, so the time period over which the estimate is made is driven by the burst length and total loop time of the system. The quality of this frequency estimator is paramount to the systems operation, and so many other design decisions and parameters are driven by ensuring the estimator variance is sufficiently low.

A frequency estimate can be made by computing

$$\hat{f}_1 = \frac{\hat{\phi}_2 - \hat{\phi}_1}{2\pi T_1} \,. \tag{6.1}$$

Phases are estimated modulo 2π, creating ambiguities in the frequency estimates. Ambiguous solutions occur at $\pm\frac{n}{T}$, since the phase estimate $\hat{\phi}_2$ can't be distinguished from $\hat{\phi}_2 \pm 2\pi n, n \in \mathbb{Z}$.

Disambiguation can be done by

$$\hat{f}_2 = \frac{\left(\hat{\phi}_3 - \hat{\phi}_1\right) \pm 2\pi n}{2\pi T_2}, \, n = \arg\min \hat{f}_2 - \hat{f}_1 \,. \tag{6.2}$$

Refinement across multiple bursts is performed using the described disambiguation techinques. When running at a 120 ms loop-time, I perform this successive refinement process 4 times, first over 120ms burst to burst time, then 240 ms, 360 ms and 480 ms. This approach yields acceptable estimator variance, with standard deviations typically $\sigma < 0.1$ Hz. I also average the last 5 fully-refined estimates to further improve estimator stability. This further enforces the expectation that the local oscillators do not drift quickly.

This training burst transmission and computation will consume the majority of the computation time. The other challenge is the time it takes to feed estimates around the network. The frequency and channel estimates are made at the final receiver element, and must be fed back to the mosaic for use in computations and application. This is the key information delay that requires tracking and prediction for the future. The frequency offset and channel estimates from the mosaic to final receiver must be fedback before use. This ensures a minimum of a loop period before they can be applied. The frequency estimate must be of sufficient quality, and the local oscillators stable in their progression to predict the resulting phase offset from the frequency error at the future time. The system's sense of time and sampling clock must also be stable enough to support this predicted digital correction. The channel coherence time must also be greater than this information delay, or the system is able to sufficiently predict its change into the future.

The frequency corrections are all applied digitally. The following signals must be considered: first, the signal sent from the initial transmitter, $s(t)$, and next, the reception at each of the mesh elements, $y_n(t) = h_{A,n} \cdot s(t) \cdot \exp(2\pi j f_{\text{offset}_{A,n}} t) + \eta(t)$. Recognize that the propogation channel introduces a constant phase shift, and the local oscillator error, synthesizer error and doppler introduce an additional phase progression. For the reception at the mesh tiles, the frequency offset can be estimated and corrected immediately at the tile.

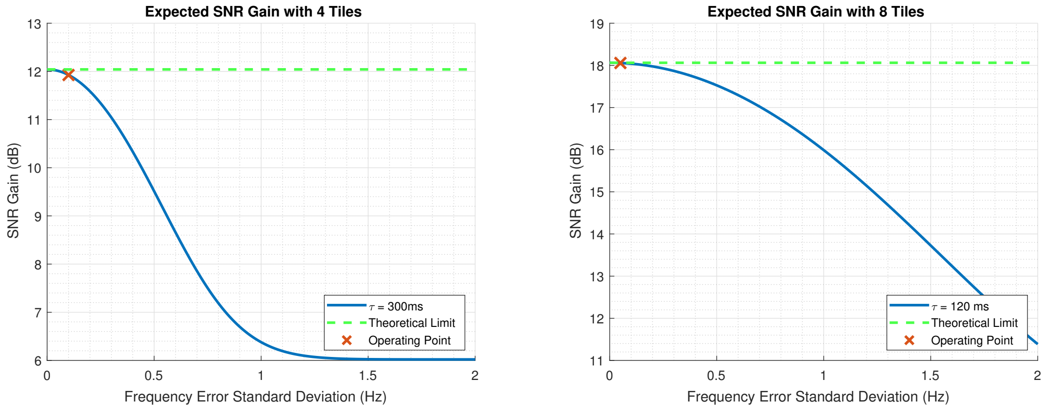
In the signal model, I consider the time t to be the global time according to the wall clock. Since the inverse phase ramp to correct for the frequency offset will be applied at a future time, the system must compensate for that difference. Thus, an arbitrary decision is made to set $t = t_a + \alpha$. t_a is the local burst time starting at 0 and counting up monotonically for the length of the burst. The timestamp of the first sample in the burst is denoted α. The SDR platform allows access to the value of α, which I leverage to compute the global time for each sample. The mesh then estimates $\hat{f}_{\text{offset}_{A,n}}$ and applies the compensation $\exp(-2\pi j \hat{f}_{\text{offset}_{A,n}} t)$. Once this application is complete, the mesh receiver can then estimate the channel without the additional confounding frequency error. There will be some residual error, but, with sufficient estimate quality for the frequency, the residual will be small.

The signal is then ready for retransmission. The mesh tile will predict the phase ramp that will be present at the final receiving node between its local clock and the receivers. This is done by compensating the value of α by adding the time between bursts, so that the transmission will be aligned with the predicted future phase offset. The tile then applies a similar phase ramp predistortion to the transmitted beamformed payload to compensate for the mesh tile to receiver frequency offsets. This frequency correction predistortion is only applied to the components that are desired to be coherent, using the same sampling counting technique described previously. It is critical that the training waveform is not corrected, else the channel estimates and frequency estimation and tracking will be thrown off. If the estimates are correctly propagated and accounted for, then continuous correction is possible. However, I choose not to continuously correct because I am not disciplining the system synthesizer or clock, and instead digitally correct waveform segments.

6.2.4 Beamformer Estimation and Expected Performance

The spatial formulation of the $1 - N - 1$ beamformer is computed with the shared data products. The payload signal has the computed beamformer applied, through convolution. This beamformed payload is then retransmitted with the aforementioned frequency correction ramp.

In the prior sections, I discussed the designed update rate, and expected performance of the frequency offset estimator. With these numbers, I can generate expected performance targets for the system. For the 4 tile system, with an update rate of $\tau = 300$ ms and a frequency estimator variance of ≈ 0.01 Hz, the maximum SNR improvement is 11.93 dB and the maximum INR Reduction is -13.6 dB. This is a performance loss of about 0.12 dB from the theoretical maximum for SNR improvement. For the 8 tile system, with an update rate of $\tau = 120$ ms and a frequency estimator variance of ≈ 0.0025 Hz, the maximum SNR improvement is 17.93 dB and the maximum INR Reduction is -27.49 dB. This is a performance loss of about 0.13 dB from the theoretical maximum for SNR improvement. The variance of the gain performance is expected to decrease as $\frac{1}{N}$, where N is the number of elements in the mesh, assuming all contributions are independent. The expected performance curves and operating points for these two systems are shown in Figures 38 and 39.



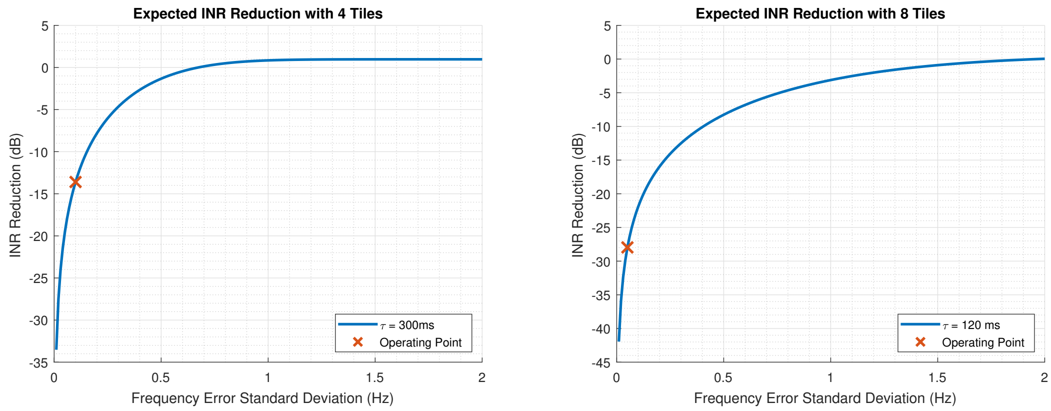
(a) 4 Tile Expected SNR Improvement Performance

(b) 8 Tile Expected SNR Improvement Performance

Figure 38: Expected SNR Improvement Performance for Prototype Systems

6.3 Over-the-Air Real-Time Experimental Results

The system with 4 tiles performs within about 1 dB of the expected SNR improvement bound. The 8 tile system performs within about 0.5 dB of the SNR improvement bound. SNR estimator variance contributes to fluctuation is in the gain estimates. The interference rejection with 4 tiles was approximately -12 dB and with 8 tiles,



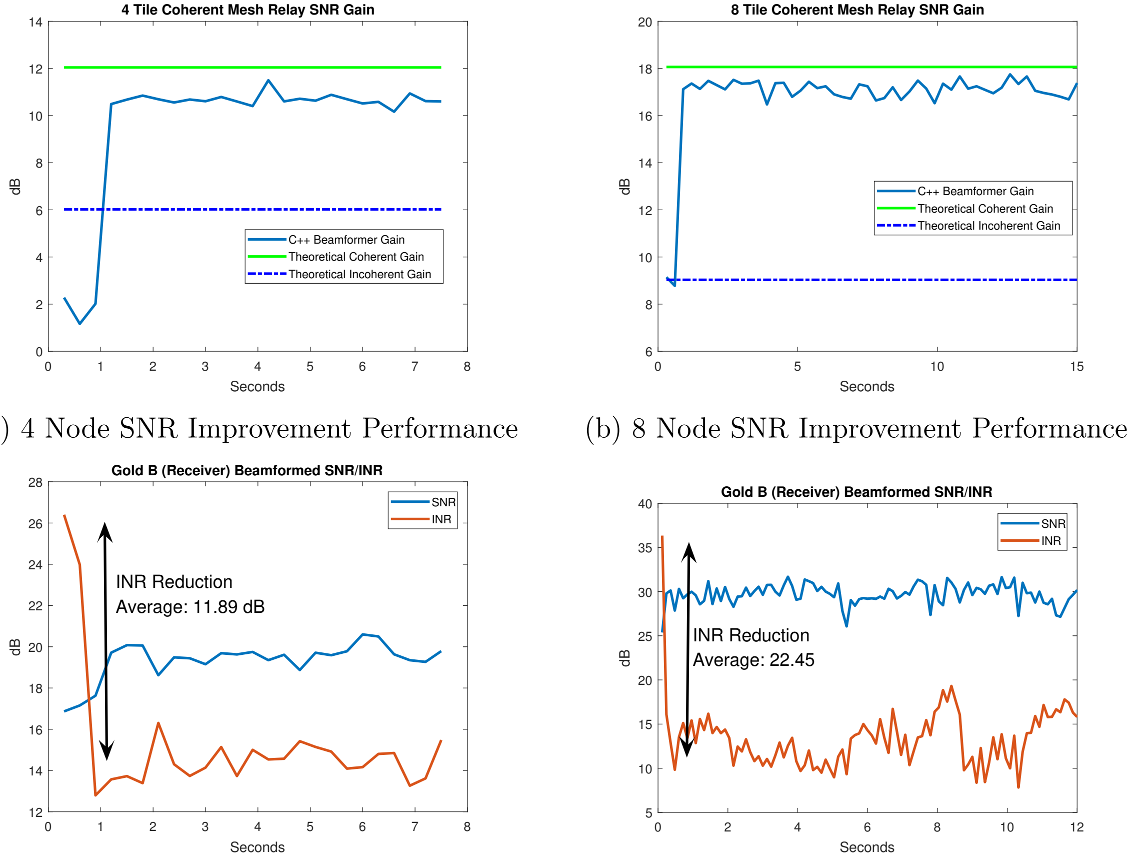
(a) 4 Tile Expected INR Reduction Performance

(b) 8 Tile Expected INR Reduction Performance

Figure 39: Expected INR Reduction Performance for Prototype Systems

approximately −22 dB. While performing interference rejection, the SNR gain for the 4 tile system was an average of 6.3 dB and for the 8 tile system, an average of 13.3 dB gain. These nulls are limited by the estimation quality of the covariance matrix achievable when sample starved as a consequence of constraining feedback rates. They are also limited by the frequency correction performance of the system. This is the first time, to the knowledge of the authors, that interference rejection over the air in a distributed coherent system has been achieved. The SNR improvement results were achieved in the receiver noise dominance regime, where the SNR improvement is most critical to system performance. The interference rejection results were achieved in the mesh noise dominance regime, where the interference rejection capability is most critical.

In the SNR improvement dominated regime, the experimental system achieves approximately 77% of the theoretical gain with the 8 element system. As seen in Figures 40a and 40b, the coherent gain improvement is stable over time, and the algorithms are able to compensate for motion, oscillator variability and the natural noise of an over-the-air distributed beamforming system. The interference rejection performance on average is 22.45 dB. This is about 5 dB from the theoretical limit for a system with the specifications I constructed. I attribute the system not reaching the theoretical limit firstly to our frequency estimator variance being higher than the 0.0025 Hz specification, and secondly to the reduced sample support used to estimate the covariance matrix for inversion.



(a) 4 Node SNR Improvement Performance

(b) 8 Node SNR Improvement Performance

(c) 4 Node Interference Mitigation Performance

(d) 8 Node Interference Mitigation Performance

Figure 40: Over-The-Air Experimental Results. In Figures 40a and 40b, we present the over-the-air real-time distributed coherent mesh beamforming SNR improvement performance for 4 and 8 element meshes respectively. The results were collected in the receiver noise dominant regime, with performance within approximately 1 dB of the theoretical limit. The first data points are during the initialization of the frequency correction, accordingly they achieve incoherent gain. In Figures 40c and 40d, we present the interference mitigation performance for 4 and 8 element meshes respectively. These measurements were made in the mesh noise dominant regime with a signal to interference ratio of −7 dB received at the mesh.

Chapter 7

SUMMARY

In this work, I discussed the fundamentals of distributed mesh beamforming systems and approaches to practically implement them. Additionally, I have outlined the algorithms required to support this distributed communications topology and augmentations to support practical and implementable systems. I then demonstrated the feasibility of the novel approach in both SNR improvement and interference reduction capabilities in simulation and in a psuedo-real-time OTA experiment thus validating the proposed algorithm and adaptations. Next, I implemented real-time intramesh synchronization in timing and frequency by trading coherence ambiguities for system resources. I further extended WISCANet to be able to run in real-time and enable distributed coherent system prototyping. Finally, I developed and demonstrated a practical real-time laboratory experiment that fully implements all synchronization and beamforming capabilities and showed a practical way to achieve the expected performance.

REFERENCES

[1] Samet E Arda et al. "DS3: A System-Level Domain-Specific System-on-Chip Simulation Framework". In: *IEEE Trans. on Computers* (2020). arXiv preprint arXiv:2003.09016.

[2] O. Bakr et al. "Interference suppression in the presence of quantization errors". In: *Allerton Conference on Communication, Control, and Computing* (Oct. 2009), pp. 1161–1168.

[3] Patrick Bidigare et al. "Attaining fundamental bounds on timing synchronization". In: *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 2012, pp. 5229–5232. DOI: 10.1109/ICASSP.2012.6289099.

[4] D. W. Bliss, S. Kraut, and A. Agaskar. "Transmit and receive space-time-frequency adaptive processing for cooperative distributed MIMO communications". In: *IEEE International Conference onAcoustics, Speech and Signal Processing (ICASSP)*. 2012, pp. 5221–5224.

[5] Daniel W Bliss. "Development of Next Gen Processors for RF Systems". In: *DARPA Electronics Resurgence Initiative Summit*. 2020.

[6] Daniel W. Bliss. *Modern Communications: A Systematic Introduction*. Cambridge University Press, 2021. DOI: 10.1017/9781108980685.

[7] D. Richard Brown III et al. "Distributed Reception with Hard Decision Exchanges". In: *IEEE Transactions on Wireless Communications* 13.6 (2014), pp. 3406–3418. DOI: 10.1109/TWC.2014.051314.131763.

[8] B.D. Carlson. "Covariance matrix estimation errors and diagonal loading in adaptive arrays". In: *IEEE Transactions on Aerospace and Electronic Systems* 24.4 (1988), pp. 397–401. DOI: 10.1109/7.7181.

[9] Alex R Chiriyath et al. "Technological Advances to Facilitate Spectral Convergence". In: *2021 55th Asilomar Conference on Signals, Systems, and Computers*. IEEE. 2021, pp. 623–628.

[10] W.G. Cowley. "Phase and frequency estimation for PSK packets: bounds and algorithms". In: *IEEE Transactions on Communications* 44.1 (1996), pp. 26–28. DOI: 10.1109/26.476092.

[11] *Ettus Research - the leader in Software Defined Radio (SDR)*. https://www.ettus.com/. Accessed: 2021-01-21.

[12] J.R. Guerci and J.S. Bergin. “Principal components, covariance matrix tapers, and the subspace leakage problem”. In: *IEEE Transactions on Aerospace and Electronic Systems* 38.1 (2002), pp. 152–162. DOI: 10.1109/7.993236.

[13] Samer Hanna and Danijela Cabric. “Distributed Transmit Beamforming: Design and Demonstration from the Lab to UAVs”. In: *IEEE Transactions on Wireless Communications* (2022), pp. 1–1. DOI: 10.1109/TWC.2022.3198102.

[14] Samer Hanna, Enes Krijestorac, and Danijela Cabric. “Destination-Feedback Free Distributed Transmit Beamforming Using Guided Directionality”. In: *IEEE Transactions on Mobile Computing* (2022), pp. 1–13. DOI: 10.1109/TMC.2022.3188602.

[15] Andrew Herschfelt. “Simultaneous Positioning and Communications: Hybrid Radio Architecture, Estimation Techniques, and Experimental Validation”. PhD thesis. Arizona State University, 2019.

[16] Andrew Herschfelt. “Vehicular RF Convergence: Simultaneous Radar, Communications, and PNT for Urban Air Mobility and Automotive Applications”. In: *2012 IEEE Radar Conference (RadarConf20)*. IEEE. 2020.

[17] Andrew Herschfelt et al. “Joint positioning-communications system design and experimental demonstration”. In: *2019 IEEE/AIAA 38th Digital Avionics Systems Conference (DASC)*. IEEE. 2019, pp. 1–6.

[18] Andrew Herschfelt et al. “Joint positioning-communications system design: Leveraging phase-accurate time-of-flight estimation and distributed coherence”. In: *2018 52nd Asilomar Conference on Signals, Systems, and Computers*. IEEE. 2018, pp. 433–437.

[19] Jacob Holtom et al. “Distributed Beamforming Techniques for Flexible Communications Networks”. In: *2021 55th Asilomar Conference on Signals, Systems, and Computers*. 2021, pp. 400–404. DOI: 10.1109/IEEECONF53345.2021.9723094.

[20] Jacob Holtom et al. “Rapid implementation and demonstration of radio applications using wiscanet”. In: *2021 IEEE 32nd Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*. IEEE. 2021, pp. 1500–1505.

[21] Jacob Holtom et al. “WISCANet: A Rapid Development Platform for Beyond 5G and 6G Radio System Prototyping”. In: *Signals* 3.4 (2022), pp. 682–707. URL: https://www.mdpi.com/2624-6120/3/4/41.

[22] National Instruments. *Software Defined Radios*. 2020. URL: https://www.ni.com/en-us/shop/hardware/software-defined-radios-category.html (visited on 11/07/2020).

[23] Leonardo Jimenez Rodriguez, Nghi Tran, and Tho Le-Ngoc. *Amplify-and-Forward Relaying in Wireless Communications*. June 2015. DOI: 10.1007/978-3-319-17981-0.

[24] S. Kraut, A. R. Margetts, and D. W. Bliss. "Reducing the fractional rank of interference with space-time-frequency adaptive beamforming". In: *IEEE Asilomar Conference on Signals, Systems and Computers*. 2013.

[25] Wenxing Li et al. "Adaptive Antenna Null Broadening Beamforming against Array Calibration Error Based on Adaptive Variable Diagonal Loading". In: *International Journal of Antennas and Propagation* 2017 (Oct. 2017), pp. 1–9. DOI: 10.1155/2017/3265236.

[26] Steven M. Kay. *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ: Prentice Hall, 1993.

[27] Joshua Mack et al. "Performant, Multi-Objective Scheduling of Highly Interleaved Task Graphs on Heterogeneous System on Chip Devices". In: *IEEE Transactions on Parallel and Distributed Systems* (2021).

[28] Xiaojun Mao et al. "Robust Adaptive Beamforming against Signal Steering Vector Mismatch and Jammer Motion". In: *International Journal of Antennas and Propagation* 2015 (July 2015). DOI: 10.1155/2015/780296.

[29] A. R. Margetts, E. G. Torkildson, and D. W. Bliss. "Loss characterization of distributed space-time transmit beamforming with embedded channel probing". In: *IEEE Statistical Signal Processing Workshop (SSP)*. 2012.

[30] Raghuraman Mudumbai et al. "Distributed transmit beamforming: challenges and recent progress". In: *IEEE Communications Magazine* 47.2 (2009), pp. 102–110. DOI: 10.1109/MCOM.2009.4785387.

[31] Raghu Mudumbai et al. "DSP-centric algorithms for distributed transmit beamforming". In: *2011 Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*. 2011, pp. 93–98. DOI: 10.1109/ACSSC.2011.6189962.

[32] Mark Johnson Omar M. Bakr. *Impact of phase and amplitude errors on array performance*. Tech. rep. Jan. 2009.

[33] Junhui Qian et al. “Null broadening adaptive beamforming based on covariance matrix reconstruction and similarity constraint”. In: *EURASIP Journal on Advances in Signal Processing* 2017 (Jan. 2017). DOI: 10.1186/s13634-016-0440-1.

[34] François Quitin et al. “Demonstrating distributed transmit beamforming with software-defined radios”. In: *2012 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. 2012, pp. 1–3. DOI: 10.1109/WoWMoM.2012.6263729.

[35] R. Mudumbai, G. Barriac, and U. Madhow. “On the Feasibility of Distributed Beamforming in Wireless Networks”. In: *IEEE Transactions on Wireless Communications* 6.5 (2007), pp. 1754–1763.

[36] Muhammad M. Rahman et al. “Fully wireless implementation of distributed beamforming on a software-defined radio platform”. In: *2012 ACM/IEEE 11th International Conference on Information Processing in Sensor Networks (IPSN)*. 2012, pp. 305–315. DOI: 10.1109/IPSN.2012.6920945.

[37] J. Riba, J. Goldberg, and G. Vazquez. “Robust beamforming for interference rejection in mobile communications”. In: *IEEE Transactions on Signal Processing* 45.1 (1997), pp. 271–275. DOI: 10.1109/78.552229.

[38] F. Rice. “Carrier-phase and frequency-estimation bounds for transmissions with embedded reference symbols”. In: *IEEE Transactions on Communications* 54.2 (2006), pp. 221–225. DOI: 10.1109/TCOMM.2005.863782.

[39] F. Rice et al. “Cramer-Rao lower bounds for QAM phase and frequency estimation”. In: *IEEE Transactions on Communications* 49.9 (2001), pp. 1582–1591. DOI: 10.1109/26.950345.

[40] Dzulkifli Scherber et al. “Coherent Distributed Techniques for Tactical Radio Networks: Enabling Long Range Communications with Reduced Size, Weight, Power and Cost”. In: *MILCOM 2013 - 2013 IEEE Military Communications Conference*. 2013, pp. 655–660. DOI: 10.1109/MILCOM.2013.117.

[41] Sharanya Srinivas, Andrew Herschfelt, and Daniel W Bliss. “Communications and High-Precision Positioning (CHP2): Joint Position and Orientation Tracking”. In: *2021 IEEE Aerospace Conference (50100)*. IEEE. 2021, pp. 1–6.

[42] Le Tian and Wang Jianbo. “An algorithm for carrier frequency offset estimation in BPSK burst transmission”. In: *The Fifth International Conference on the*

Applications of Digital Information and Web Technologies (ICADIWT 2014). 2014, pp. 50–53. DOI: 10.1109/ICADIWT.2014.6814692.

[43] *USRP Hardware Driver (UHD) - Ettus Research*. https://www.ettus.com/sdr-software/uhd-usrp-hardware-driver/. Accessed: 2021-01-21.

[44] Murat Uysal. *Cooperative Communications for Improved Wireless Network Transmission: Framework for Virtual Antenna Array Applications*. Jan. 2009. DOI: 10.4018/978-1-60566-665-5.

[45] Daniel W. Bliss and Siddhartan Govindasamy. *Adaptive Wireless Communications: MIMO Channels and Networks*. Cambridge: Cambridge University Press, 2013.

[46] Han Yan et al. "Software Defined Radio Implementation of Carrier and Timing Synchronization for Distributed Arrays". In: *2019 IEEE Aerospace Conference*. 2019, pp. 1–12. DOI: 10.1109/AERO.2019.8742232.

[47] Jian Yang et al. "Robust Null Broadening Beamforming Based on Covariance Matrix Reconstruction via Virtual Interference Sources". In: *Sensors* 20.7 (2020). DOI: 10.3390/s20071865. URL: https://www.mdpi.com/1424-8220/20/7/1865.

[48] H. Yu et al. "WISCA SDR network". In: *MILCOM 2017 - 2017 IEEE Military Communications Conference (MILCOM)*. 2017, pp. 750–755.

[49] Michael A. Zatman. "Production of adaptive array troughs by dispersion synthesis". In: *Electronics Letters* 31 (1995), pp. 2141–2142.