Improving Signal Clarity through Interference Suppression and Emergent Signal Detection

Elizabeth A. Hoppe

Dissertation submitted to the faculty of the Virginia Polytechnic Institute and State University in partial fulfillment of the requirements for the degree of

Doctor in Philosophy

in

Mechanical Engineering

Michael Roan, Chairman
Tamal Bose,
Martin Johnson,
Alexander Leonessa,
Alessandro Toso

03 September 2009
Blacksburg, VA

Key words: signal clarity, interference suppression, emergent signal detection, array processing

Copyright 2009, Elizabeth A. Hoppe
Improving Signal Clarity through Interference Suppression and Emergent Signal Detection

Elizabeth A. Hoppe

ABSTRACT

Microphone arrays have seen wide usage in a variety of fields; especially in sonar, acoustic source monitoring and localization, telecommunications, and diagnostic medicine. The goal of most of these applications is to detect or extract a signal of interest. This task is complicated by the presence of interferers and noise, which corrupts the recorded array signals. This dissertation explores two new techniques that increase signal clarity: interferer suppression and emergent signal detection.

Spatial processing is often used to suppress interferers that are spatially distinct from the signal of interest. If the signal of interest and the interferer are statistically independent, blind source separation can be used to statistically extract the signal of interest. The first new method to improve signal clarity presented in this work combines spatial processing with blind source separation to suppress interferers. This technique allows for the separation of independent sources that are not necessarily simultaneously mixed or spatially distinct. Simulations and experiments are used to show the capability of the new algorithm for a variety of conditions. The major contributions in this dissertation under this topic are to use independent component analysis to extract the signal of interest from a set of array signals, and to improve existing independent component analysis algorithms to allow for time delayed mixing.

This dissertation presents a novel method of improving signal clarity through emergent signal detection. By determining which time frames contain the signal of interest, frames that contain only interferers and noise can be eliminated. When a new signal of interest emerges in a measurement of a mixed set of sources, the principal component subspace is altered. By examining the change in the subspace, the emergent signal can be robustly detected. This technique is highly effective for signals that have a near constant sample variance, but is successful at detecting a wide variety of signals, including voice signals. To improve performance, the algorithm uses a feed-forward processing technique. This is helpful for the VAD application because voice does not have a constant sample variance. Experiments and simulations are used to demonstrate the performance of the new technique.
# Table of Contents

List of Figures .............................................................................................................................. vi

Acknowledgements .................................................................................................................... xiii

Chapter 1: Introduction ................................................................................................................. 1

1.1 Research Motivation ............................................................................................................. 1
1.2 Problem Statement ............................................................................................................... 2
  1.2.1 Interference Suppression ............................................................................................... 3
  1.2.2 Emergent Signal Detection ......................................................................................... 4
1.3 Relevant Research ................................................................................................................. 4
1.4 Original Contributions ......................................................................................................... 6
1.5 Organization of Dissertation ............................................................................................... 7

Chapter 2: Background .................................................................................................................. 9

2.1 Introduction to Array Processing .......................................................................................... 9
2.2 Beamforming ......................................................................................................................... 9
  2.2.1 Narrowband Delay and Sum ....................................................................................... 11
  2.2.2 Broadband Delay and Sum ......................................................................................... 13
  2.2.3 Minimum Variance Distortionless Response (MVDR) ................................................ 16
2.3 Blind Source Separation ....................................................................................................... 18
  2.3.1 Principal Component Analysis ..................................................................................... 18
  2.3.2 Independent Component Analysis .............................................................................. 20
  2.3.3 Two ICA Algorithms of Interest ................................................................................. 23
    2.3.3.1 Infomax ................................................................................................................. 23
    2.3.3.2 FastICA ............................................................................................................... 25
2.4 Emergent Signal Detection .................................................................................................. 26
  2.4.1 Hypothesis Testing ........................................................................................................ 27
  2.4.2 Log Likelihood Test ...................................................................................................... 27
2.5 Voice Activity Detection ....................................................................................................... 29
  2.5.1 Higher Order Statistics Approach to VAD ................................................................. 30
  2.5.2 Power Spectral Density Content Approach to VAD .................................................. 32
2.6 Performance Measures - Receiver Operator Curves ............................................................... 34
2.7 Conclusions .......................................................................................................................... 36
Chapter 3: Blind Source Separation and Spatial Processing

3.1 Introduction

3.2 Proposed Algorithm

3.2.1 Strategy and Motivation

3.2.2 Choosing Possible Source Locations

3.2.3 Correcting for Delays

3.2.4 Applying the ICA Algorithm and Signal Selection

3.2.5 Implementing the Wiener Filter

3.3 Simulations and Results

3.3.1 One Signal of Interest with One Interferer

3.3.1.1 Minimum Variance Distortionless Response Processing

3.3.1.2 Processing with the New Algorithm

3.3.2 Effect of Angular Separation of Sources

3.3.3 Varying Signal to Interferer Ratio

3.3.4 Varying Signal to Noise Ratio

3.3.5 Used as a Pre-Processor for Voice Activity Detection

3.3.6 One Loud Interferer with Three Quieter Signals of Interest

3.4 Experiments and Results

3.4.1 One Signal of Interest with One Interferer

3.4.2 Varying Signal to Interferer Ratio

3.5 Conclusions

Chapter 4: Emergent Signal Detection

4.1 Introduction

4.2 Emergent Signal Detection using Principal Component Analysis

4.2.1 Specific Case: Voice Activity Detection

4.3 Emergent Signal Detection for One Signal and One Interferer

4.3.1 Simultaneous Mixing Model

4.3.2 Mixing Model Including TDOA

4.4 One Signal, One Interferer with Uncorrelated Background Noise

4.5 Frame to frame processing

4.6 Simulations and Results – Tonal Case

4.6.1 One Signal, One Interferer – Simultaneous Mixing

4.6.2 One Signal, One Interferer – Delayed Mixing

4.6.2.1 Varying Signal to Interferer Ratio

4.6.2.2 Varying Overhang Length

4.6.2.3 Varying Source/Receiver Geometry

4.6.2.4 Varying the Type of Signal of Interest
List of Figures

Figure 2.1: A typical beamformer. Weights are applied to each array sensor, and then summed together to produce the output. The value of the weights varies depending on the type of beamformer used. ........................................................................................................ 11

Figure 2.2: The effect of frequency on the narrowband DS beamformer. (b) The response of a 10 element linear array with equal spacing of one inch for varying frequency. A 3500Hz tonal signal is present at 20 degrees. (a) The output of the narrowband DS beamformer when the weights are calculated for 20 degrees and 3500Hz. (c) The output of the narrowband DS beamformer when the weights are calculated for 20 degrees and 4000Hz. ............................................................ 13

Figure 2.3: [56] Broadband beamformer block diagram based on narrowband decomposition. ........................................................................................................................................ 14

Figure 2.4: The response of the narrowband DS beamformer versus the response of the broadband beamformer for a bandpassed signal that is 1000Hz wide, centered at 3500Hz, and located at 20 degrees. For the series of narrowband DS beamformers examined, only the 3200Hz case has maximum power at 20 degrees. .................................................................................. 15

Figure 2.5: The kurtosis of the extracted signal varies as the weight vector is rotated around the origin. The amplitude of the first signal mixture versus the amplitude of the second mixture is plotted with blue dots. The kurtosis of the recovered signal, \( y(t) = w^T x(t) \), is plotted as a function of the weight vector orientation in black. For each orientation direction, kurtosis is plotted as the distance from the origin. The two red curves show the directions of maximum kurtosis. When the orientation of the weight vector is in the same direction as these red curves, the source signals are recovered exactly. ............................................................................................................... 22

Figure 2.6: (a) The joint pdf of two signals with uniform distributions. (b) The joint pdf of the mixed signals. The signals are linearly mixed according to the model in Equation 2.19. (c) The joint pdf of the unmixed signals when PCA is used to separate the signals. While the signals have been decorrelated, they are not statistically independent. (d) The joint pdf of the unmixed signals when ICA is used to separate them. The two signals are statistically independent. ........................................................................................................ 23

Figure 2.7: Decision process for the higher order statistic VAD. The conditions outlined are the conditions that must be met to switch from either a speech to noise state or a noise to speech state. ..................................................................................................................... 32

Figure 2.8: (a) The pdf for the significant statistic when the signal of interest is present. The P_D is the area of the blue shaded region. (b) The pdf for the significant statistic when the signal of interest is absent. The P_FA is the area of the green shaded region. ...................................... 35
Figure 2.9: A series of ROC curves. Increasing performance is indicated by the arrows. The pdfs of the signal present (blue) and signal absent (green) that generated each ROC curve are shown as insets, with the outline box color corresponding to the ROC curve it generates. As the overlap between the two pdfs increases, the performance decreases.

Figure 3.1: Processing strategy for the combination of beamforming and ICA processing to suppress interferers.

Figure 3.2: The geometry of the sensors and sources assumed for algorithm development is shown. The array is a linear microphone array and all sources are in front of the array. The time delays introduced in Equations 3.1 and 3.2 are due to the corresponding distances indicated with the red dashed and purple dotted line segments.

Figure 3.3: The coherence between the unmixed signal and the separated signal is shown for four cases: linear mixing, convolutive mixing, corrected convolutive mixing with only beamformed signals, and corrected convolutive mixing where the first and mth unaltered array signals are included.

Figure 3.4: BTR plot using a broadband beamformer for two signals having equal power. One signal is located at +5 degrees, while the other signal moves from -20 to +20 degrees at a rate of one degree per second.

Figure 3.5: Bearing track recorder plot using MVDR beamforming. The undesired, stationary source at +5 degrees has been nulled in all time steps, but numerous sidelobes in the beampattern obscure the bearing track recorder display.

Figure 3.6: BTR plot using a broadband beamformer after application of the proposed algorithm. The undesired, stationary source at +5 degrees has been suppressed in all time steps without significant suppression of the moving source.

Figure 3.7: The algorithm’s interference suppression performance as a function of the angular spacing between the interferer and the signal of interest is examined. (a) The coherence between the algorithm input data and the signal of interest. (b) The coherence between the algorithm output and the signal of interest.

Figure 3.8: The effect of SIR on the performance of the new algorithm. (a) The beampattern of the algorithm input using a broadband DS beamformer. This signal of interest is positioned at +5 degrees and the interferer is at -10 degrees. (b) The beampattern of the algorithm output using a broadband DS beamformer. The interferer has been suppressed by 20dB in all cases, and the signal of interest remains visible until SIR drops below -15dB.
Figure 3.9: The effect of SNR on the algorithm. (a) The beampattern of the algorithm input using a broadband DS beamformer. This signal of interest is positioned at +5 degrees and the interferer is at -10 degrees. (b) The beampattern of the algorithm output using a broadband DS beamformer. The interferer is suppressed in all cases and the signal of interest remains visible until SNR drops below -18dB.

Figure 3.10: Voice activity detection on data containing one voice signal of interest and one white Gaussian interferer. The three rows show the results at varying SNR values. The active voice frames are enclosed with red boxes. Column (a) shows the VAD results before the application of the new algorithm, column (b) shows the VAD results after MVDR, column (c) shows the VAD results after FastICA, and column (d) shows the VAD results after the new algorithm application.

Figure 3.11: BTR plot using a broadband beamformer for one loud interferer and three signals of interest. The signals of interest are all at -10dB SIR with the interferer. (a) The BTR of the mixed data before application of the new algorithm. Only the interferer is visible. (b) The BTR of the data after it has been processed using the new algorithm. The loud interferer has been completely removed, revealing the three signals of interest previously hidden.

Figure 3.12: BTR plot using a broadband beamformer for one loud interferer and three signals of interest. (a) The BTR of the mixed data before application of the new algorithm. Only the interferer is visible. (b) The BTR of the data after it has been processed the first time using the new algorithm. (c) The BTR of the data after it has been processed twice.

Figure 3.13: BTR using a broadband DS beamformer for the anechoic chamber testing. One signal is located at zero degrees, while the other signal moves from -20 to +20 degrees at a rate of one degree per second.

Figure 3.14: BTR using a broadband DS beamformer after the anechoic chamber data has been processed using the new algorithm. The moving interferer has been removed, and the signal of interest at 0 degrees is visible.

Figure 3.15: The effect of SIR on the performance of the new algorithm using data recorded in an anechoic chamber. (a) The beampattern of the algorithm input using a broadband beamformer. This signal of interest is positioned at -11 degrees and the interferer is at +19 degrees. (b) The beampattern of the algorithm output using the broadband beamformer.
Figure 4.1: A visualization of the joint pdf for two cases is presented: only one signal present and both signals present. A PCA analysis is performed on the data, and the directions of the PCs are plotted in red. (a) The joint pdf for the case where only the interferer is present. (b) The joint pdf for the case where both the interferer and the signal of interest are present.

Figure 4.2: (a) The signal of interest used for all simulations. (b) The mixed signal, X, containing the signal of interest and the white, Gaussian interferer mixed simultaneously at 0dB SIR according to Equation 4.1. (c) The significant statistic generated using PCA. Using this significant statistic and a user chosen threshold, each frame is categorized as signal present or signal absent. For the simultaneous mixing scheme, when the signal of interest is absent, the significant statistic drops to zero.

Figure 4.3: (a) The positions of the signal of interest, the interferer, and the two microphones are shown. The TDOAs calculated based on the source/receiver geometry presented are used as the time delays in Equation 4.7. (b) The ROC curves generated in the delayed mixing simulation corresponding to the geometry in (a). The solid curve shows the results using the new PCA technique, while the dashed curve shows the results using the log likelihood statistical test introduced in Section 2.4.2.

Figure 4.4: The effect of changing SIR on the algorithm’s performance. Each plot represents a different reference P_{FA} value. From left to right, the reference values are 1%, 5% and 10% P_{FA}. The signals are arranged in the geometry indicated in Figure 4.3(a). For all cases, the new technique has a better performance than the likelihood test.

Figure 4.5: Algorithm performance as a function of the number of overhang points. For the reference P_{FA} values of 0.5%, 1%, 5%, and 10%, the P_D as a function of overhang points is shown. For all reference P_{FA}, the P_D decreases with increasing overhang points.

Figure 4.6: The effect of changing the location of the signal of interest on the performance of the algorithm. The interferer is held stationary at (20,90) while the position of the voice signal is changed. The microphones remain stationary at (30,0) and (0,50). This is the geometry of Figure 4.3(a). The P_D at 5% P_{FA} is recorded for each (x,y) position of the voice.

Figure 4.7: The effect of changing the location of the interferer on the performance of the algorithm. The signal of interest is held stationary at (65,75) while the position of the interferer is changed. The microphones remain stationary at (30,0) and (0,50). This is the geometry of Figure 4.3(a). The P_D at 5% P_{FA} is recorded for each (x,y) position of the voice.
Figure 4.8: The results of changing the form of the signal of interest on the algorithm’s performance. The ROC curve for the -7dB SIR is shown in the right plot. For the reference $P_{FA}$ values of 5%, the $P_D$ as a function of SIR is also shown. For the tone, random signal, and uniform pdf cases, the performance curves lie on top of one another, indicating identical performance.

Figure 4.9: The ROC curves generated in the delayed mixing simulation with added uncorrelated noise. The solid curve shows the results using the new PCA technique, while the dashed curve shows the results using the log-likelihood algorithm.

Figure 4.10: The algorithm’s performance as a function of varying SNR. Each plot represents a different reference $P_{FA}$ value. From left to right, the reference values are 1%, 5% and 10% $P_{FA}$. The signals are arranged in the geometry indicated in Figure 4.3(a). For all cases, the new technique has a better performance than the likelihood test.

Figure 4.11: The co-effect of changing SIR and SNR. For each pair of SIR and SNR values, the $P_D$ at 5% $P_{FA}$ was calculated. The highest performance is at the highest SIR and SNR. Depending on the desired performance, an acceptable SIR and SNR limit can be determined.

Figure 4.12: (a) The fft of the white, Gaussian noise signal used to test the frequency response of the speakers. It has nearly constant power at all frequencies. (b) The frequency response of the speakers. After 5500Hz, the response drops off.

Figure 4.13: The effect of changing SIR on the algorithm’s performance. Each plot represents a different reference $P_{FA}$ value. From left to right, the reference values are 1%, 5% and 10% $P_{FA}$. The signals are arranged in the geometry indicated in Figure 4.3(a). For all cases, the performance when using experimental data closely mimics the performance when using filtered simulated data.

Figure 4.14: The percentage error between the filtered simulated and experimental performances. For all SIR above -5.5dB, the percentage error is below 6%.

Figure 4.15: The results of changing the form of the signal of interest on the performance of the algorithm. The ROC curve for the anechoic chamber data at -5dB SIR is shown.

Figure 4.16: The performance of the algorithm with respect to varying SIR. A comparison is made between the performances in an anechoic environment (blue curve) versus a reverberant environment (red curve).

Figure 4.17: The performance of the algorithm as a function of SNR. A hybrid experiment is performed where artificial noise is added to data recorded in an anechoic chamber.
Figure 4.18: (a) The original voice signal used for all simulations. (b) The mixed signal, X, containing the voice signal and the white, Gaussian interferer mixed simultaneously according to Equation 4.1. (c) The significant statistic generated using PCA. Using this significant statistic and a user chosen threshold, each frame is categorized as speech or non-speech. For the simultaneous mixing scheme, when speech is not present, the significant statistic drops to zero.

Figure 4.19: (a) The positioning of the signals and microphones is shown. The time delays introduced into the mixing scheme are due to the corresponding TDOA between the two sensors. (b) The ROC curves generated in the delayed mixing simulation corresponding to the geometry in (a). The solid curve shows the results using the new PCA technique, while the dashed curve shows the results using the statistical VAD algorithm introduced in Section 2.5.2.

Figure 4.20: The algorithm’s performance as a function of varying SIR. Each plot represents a different reference $P_{FA}$ value. From left to right, the reference values are 1%, 5% and 10% $P_{FA}$. The signals are arranged in the geometry indicated in Figure 4.19(a). For all cases, the new technique has a better performance than the statistical VAD algorithm.

Figure 4.21: The change in the distributions of the significant statistic for the $H_0$ and $H_1$ hypothesis for varying SIR. Column (a) shows the change in $P_0$ and $P_1$ for the tonal case, while column (b) shows the change for the voice case.

Figure 4.22: The results of changing the number of overhanging points on the algorithm’s performance. (a) For the reference $P_{FA}$ values of 0.5%, 1%, 5%, and 10%, the $P_D$ as a function of overhang points is shown. For all reference $P_{FA}$, the $P_D$ increases with increasing overhang points until it peaks at 17 points. After that point, the $P_D$ drops for all reference $P_{FA}$ values. (b) The ROC curves generated at three reference overhang point values. The 15 points curve shows an increase in performance over the five points curve. The 25 points curve shows a drop in performance, especially in the lower $P_{FA}$ region.

Figure 4.23: The effect of increasing overhang length on the significant statistic. The blue curves represent the significant statistic as calculated with a one point overhang value, while the red curves are calculated using a 10 point overhang value. Plot (a) is for the tonal case, while plot (b) is for the VAD case.

Figure 4.24: The effect of changing the location of the voice signal on the performance of the algorithm. The interferer is held stationary at (20,90) while the position of the voice signal is changed. The microphones remain stationary at (30,0) and (0,50). This is the geometry of Figure 4.19(a). The $P_D$ at 5% $P_{FA}$ is recorded for each (x,y) position of the voice.
Figure 4.25: The effect of changing the location of the interferer on the performance of the algorithm. The voice signal is held stationary at (65,75) while the position of the interferer is varied. The microphones remain stationary at (30,0) and (0,50). This is the geometry of Figure 4.19(a). The $P_D$ at 5% $P_{FA}$ is recorded for each (x,y) position of the interferer.

Figure 4.26: The ROC curves generated in the delayed mixing simulation with added uncorrelated noise. The solid curve shows the results using the new PCA technique, while the dashed curve shows the results using the statistical VAD algorithm.

Figure 4.27: The algorithm’s performance as a function of varying SNR. Each plot represents a different reference $P_{FA}$ value. From left to right, the reference values are 1%, 5% and 10% $P_{FA}$. The signals are arranged in the geometry indicated in Figure 4.19(a). For all cases, the new technique has a better performance than the statistical VAD algorithm.

Figure 4.28: The co-effect of changing SIR and SNR. For each pair of SIR and SNR values, the $P_D$ at 5% $P_{FA}$ was calculated. The highest performance is at the highest SIR and SNR. Depending on the desired performance, an acceptable SIR and SNR limit can be determined.

Figure 4.29: The effect of changing SIR on the algorithm’s performance. Each plot represents a different reference $P_{FA}$ value. The performance of the algorithm is reduced when the signal of interest and interferer are low pass filtered at 5500Hz. For all cases, the performance of the algorithm for the experimental data mimics the performance for the filtered simulated data.

Figure 4.30: The percentage error between the performance of the algorithm for the filtered simulated data and the experimental data. For all SIRs, the percentage error is below 6%.

Figure 4.31: The performance of the algorithm with respect to varying SIR. A comparison is made between the performance in an anechoic environment (blue curve) versus a reverberant environment (red curve).

Figure 4.32: The performance of the algorithm as a function of SNR. A hybrid experiment is performed where artificial noise is added to data recorded in an anechoic chamber.
Acknowledgements

I would like to my research advisor, Michael Roan for the guidance and support he provided throughout the course of my time here at Virginia Tech. Thanks for spending countless hours reviewing papers, going over presentations, and helping through all the glitches and rough spots of my research. I would like to extend my thanks to Dr. Tamal Bose, Dr. Marty Johnson, Dr. Alex Leonessa, and Dr. Ale Toso for helping me to get through my research, especially at the end, while writing this dissertation. My PhD would not have been possible without funding from ONR.

Secondly, I want to thank all the members of the VAL, both current and past. Thanks for all the laughs during the good times, and the support in the not so good times. A special thanks to all the former grad students who have been with me since the beginning and helped make coming into the lab everyday an enjoyable experience, especially Philip Gillett, Brent Gold, Caroline Hutchison, and Sean Egger. Thank you Gail for making sure all my paperwork was kept straight and feeding my Skittle addiction.

Last, but not least, I want to thank my family for all the support they have provided me over the years. Without them, I would not have made it this far. Mom and Dad, thanks for all the free lunches, the tailgate parties, moving help, and being constantly encouraging. To my brother, Michael, thanks for always being willing to go see really awful superhero/action movies with me. Finally, thanks to my wonderful fiancé, Steven, who never has to hear the phrase “just six more months” again. I love you all.

Thank you Craigslist - you rock!
Chapter 1: 

Introduction

1.1 Research Motivation

Biological systems, such as the human brain, have evolved to very effectively process sound. Using a complex biological neural network and a lifetime of training data, the brain has evolved to processes a wide spectrum of audio data in numerous complex and difficult environmental conditions. With only two sensors (ears), the brain can perform multiple complex tasks, such as tracking multiple sources, detecting emergent sources, and focusing on a single source in a mixture of sources. Despite the difficulties of background noise, interferers, reverberation, and changing environments, the human brain is still able to provide exceptional audio processing over a wide range of situations. The goal of acoustic signal processing is to design techniques that replicate (and in some cases exceed) human auditory processing capability. The range of topics of interest in acoustic/audio signal processing is vast. This dissertation focuses on two main approaches to improving signal clarity.

The goal of this research is to capture a particular signal of interest with the highest possible clarity. The signal of interest exists in an environment that also contains interferers and uncorrelated background noise. An array of sensors collects audio data from this environment, and post-processing is performed on this data. There are three points in this stream where signal clarity can be improved: manipulating the environment (as in passive quieting applications), altering the sensors (i.e. improving sensitivity), and applying various algorithms in post-processing (the focus of this dissertation). Signal clarity can be greatly improved if the environment in which the signal of interest exists can be manipulated. By physically removing the interfering sources and insulating the environment against echoes, signal clarity will increase. If the interfering signals cannot be removed, the source/receiver geometry can be manipulated to improve signal clarity. For this dissertation, it is assumed that the environment cannot be manipulated.
The second point where signal clarity can be improved is at the sensors. The type of microphones used can have a large impact on the clarity of the recorded signals. Microphones with higher sensitivity and a constant frequency response can more effectively capture the signal of interest. Aliasing is a problem that exists whenever signals are sampled, either spatially or temporally. Since infinite sample rates are impossible to achieve, higher frequency or wavenumber components will always be lost or aliased. These higher frequencies must be filtered from the data, which removes some information about the signal of interest, and can decrease intelligibility. In array processing, the spacing between elements can cause spatial aliasing. The signal clarity can be improved by choosing the best set of microphones, array element spacing, and sampling frequency. Other techniques, such as adaptive or non-uniform sampling, can be used to suppress interferers in the recording of the environment. In this dissertation, only uniform, Nyquist sampling is used, and the array is linear with uniform spacing between elements.

The third area where signal clarity can be improved is in post processing the recorded data. In this dissertation, it is assumed that the environment and the sensors cannot be manipulated. The recorded signal of interest will be corrupted by the interferers and background noise present in the environment. The main goal of this dissertation is to develop new algorithms that will improve the signal clarity of the recorded signal of interest. Several techniques will be used to improve signal clarity, including spatial processing, source separation, filtering, and signal detection.

1.2 Problem Statement

For the topics under consideration in this dissertation, the signal of interest is corrupted by interferers and uncorrelated background noise. For the models and simulations, the environment is assumed to be anechoic. Increasing signal clarity is accomplished in two ways in this dissertation: interferer suppression and emergent signal detection. By emergent signal detection, it is meant that an attempt is made to detect signals not previously present in a mixture of sources. Examples of this are a person talking in a previously quiet room, or a radio emitter turning on. The first method to improve signal clarity explored in this dissertation is to suppress
interfering sources. This is accomplished using spatial signal processing techniques, combined with blind source separation (BSS) techniques. The second method is to detect the presence of a signal of interest that emerges in a mixture of sources. By determining which time frames contain the signal of interest, frames that are irrelevant (do not contain the signal of interest) can be eliminated. Successfully detecting the signal of interest has two benefits: it reduces communications bandwidth requirements and reduces the transmission of irrelevant time frames, thus increasing intelligibility.

1.2.1 Interference Suppression

When the signal of interest and the interferer exist in distinct locations, beamforming is often used to suppress the interferer. In the simplest beamformer, the array elements are weighted and summed such that the signal of interest is summed coherently. At the same time, these weights cause destructive interference in signals arriving from all other directions. In addition, the signal to noise ratio (SNR) between the signal of interest and uncorrelated background noise increases by coherent summation of multiple channels. A wide variety of beamformers exist that spatially suppress interferers. If the signal of interest and the interferer are statistically independent, BSS techniques can be used to statistically separate them. By projecting the data along the direction of maximum statistical independence, individual source signals can be extracted from the mixed data. This process completely removes the interferer, and provides a clean sample of the signal of interest.

Both spatial processing and BSS are effective tools for interferer suppression. However, it is rare that both beamforming and BSS techniques are used together to improve the signal clarity. Where spatial processing has been used is in attempts to solve the BSS permutation ambiguity problem. Beamformers require spatially separated sources to suppress interferers, while BSS requires independent sources with simultaneous mixing. This dissertation presents a method of source separation and interference suppression based on the combination of beamforming and BSS techniques. This allows for separation of independent sources that are not necessarily simultaneously mixed.
1.2.2 Emergent Signal Detection

Several BSS algorithms separate independent non-Gaussian sources by projecting the mixed data along a direction that maximizes nongaussianity (Section 2.3.2 will explain the specifics for this technique). This projection, and the corresponding weight vector, changes depending on the number of sources present because each new independent source lies along a different direction. This dissertation uses this property of BSS to detect emergent signals. When a signal (for this dissertation, the signal of interest) emerges in the mixed data signals, the BSS subspace is altered. By examining this change in the subspace, a new method of emergent signal detection is developed. Existing methods of emergent signal detection typically rely on detecting changes in the statistics of the mixed data, especially the mean and variance. By using BSS, and specifically principal component analysis (PCA), the variance along the specific projection directions corresponding to the source signals can be examined. This leads to a higher probability of detecting the signal of interest.

1.3 Relevant Research

Research on improving signal clarity spans a wide variety of application areas. This section examines some of the relevant areas of research.

1. Passive Sonar – The underwater environment is highly reverberant, and numerous reflections can dramatically decrease the clarity of the signal of interest [1-4]. In addition, sound travels much farther underwater than in air. As a result, an interferer that is far from the receiver can still contribute to the decrease in signal clarity. Beamforming is widely used in passive sonar to determine the direction of sources and electronically steer the towed array to focus on particular sources of interest [5-8].

2. Telecommunications – In cellular telephones, signal clarity is a top priority for most users. Voice activity detection (VAD) algorithms are often employed in cell phone signal processing algorithms [9-11]. These VAD algorithms provide three benefits to the users and carriers: by detecting and transmitting only frames with active speech, the clarity of
the signal is improved by removing frames that only contain background noise and interferers; the battery life of the cell phone is extended by reducing the time of active transmission; and lastly, overall bandwidth usage per user is reduced because fewer active frames must be transmitted [12-13]. This allows the carrier to add more users per cell tower while maintaining the quality of service.

a. Voice over IP – A subset of telecommunications is voice over IP. This technology allows for the transmission of voice signals over the internet. It has many of the same requirements as cell phone technologies. Especially the goals of removing interfering sources and transmitting only active voice frames [14-18].

3. Acoustic Monitoring – Wireless sensor networks can be deployed to monitor specific environments [19-20]. These teams of sensors often collaborate to observe their environment and report detections of signals of interest, such as dismounts and vehicles [21]. Often, the sensors are monitoring the environment for speech signals, so they employ VAD algorithms to detect those voice sources [22]. Other times, the goal is to localize a signal of interest. The sensors will use beamforming to make estimates of the source bearing, and then collaborate with the other sensors to triangulate the source position [23].

a. Wildlife Monitoring – Acoustic arrays can be used to monitor wildlife, especially marine animals [24-26]. Acoustic localization and spatial processing is used to detect and track the movements of schools of fish [27-28], dolphins [29], and whales [30]. Also, emergent signal detectors are used to detect the calls of whales [31].

b. Mechanical Monitoring – Automated monitoring of mechanical systems allows for repairs to be made to systems before a mechanical failure causes catastrophic damage to the system [32]. If emergent signal detection is used to detect weakening gear teeth, the gear can be replaced before there is total failure [33].
Acoustic localization can be used to detect internal fouling in tube and ducts before there is complete blockage [34].

4. *Bio-medical Engineering* – The use of acoustic sensors is widespread in the medical community, especially for ultrasound imaging [35-37] and heart sound monitoring [38-40]. BSS has been used to extract human epileptic spikes from interfering signals [41], remove undesired contamination from EEG signals [42], and separate jaw sounds [43]. Suppressing interferers in medical signals is important for reducing false alarms [44] and improving the signal clarity for diagnostic purposes [45]. In some instances, suppressing one signal of interest improves the clarity of another. Often times, the sounds of the heart need to be suppressed to improve the clarity of the respiratory sounds [46-48].

1.4 Original Contributions

The objective of this dissertation is to develop two methods to increase signal clarity. The first uses independent component analysis (ICA) to suppress interferers, and the second uses PCA to detect the emergent signal of interest.

The original contributions of this dissertation are:

1. To use ICA to extract the signal(s) of interest from a set of mixed signals. This improves interference suppression as compared to minimum variance distortionless response (MVDR) spatial processing. It allows for interferers to be removed, even if they lie along the same direction as the signal of interest (i.e. the signals are spatially indistinct).

2. To improve all existing ICA algorithms to allow for delayed mixing. Current ICA algorithms require instantaneous mixing to be effective. By processing the data using traditional spatial beamforming, the delays within the mixed signals can be removed. This allows for the application of all existing ICA algorithms.

3. To use PCA as an emergent signal detector. By taking advantage of the ability of PCA to find projection directions corresponding to independent sources, emergent signals can be
detected. This technique is highly effective for signals that have a near constant sample variance, but is successful at detecting a wide variety of signals.

4. To make specific the general emergent signal detector to the case of VAD. While containing no voice specific attributes, in certain instances, this method is effective at detection speech mixed with an interferer. In order to improve performance, the algorithm uses a feed-forward processing technique. This is helpful for the VAD application because voice does not have a constant sample variance.

5. To perform simulations and experiments to support the previous tasks and demonstrate the usefulness of the developed techniques.

1.5 Organization of Dissertation

Including this introduction chapter, this dissertation is organized into five chapters and one appendix. The second chapter discusses technical background relevant to the remaining sections. A review of spatial processing, BSS, emergent signal detection, VAD, and receiver operating characteristic (ROC) curves is included. For each subject, a brief overview of the topic is given, and any specific algorithms that will be used in this dissertation are discussed. The algorithms discussed will be used in one of two ways: as a building block for the new techniques or as a comparison for the new techniques. The section on spatial processing discusses basic array processing and describes the broadband beamformer that is used throughout the dissertation. In the next section two traditional methods of BSS (ICA and PCA) are discussed. Two ICA algorithms, Infomax and FastICA, are described. FastICA will be used throughout the dissertation to perform BSS. The log-likelihood ratio method of emergent signal detection is discussed in Section 2.4.2, and will provide a comparison for the new, PCA based, emergent signal detection algorithm. There are two VAD algorithms presented: one based on higher order statistics and one based on spectral content. These two algorithms will be used as comparisons for Chapters 3 and 4. Finally, the ROC curves are discussed. These curves allow the performance of several emergent signal detection algorithms (and by extension VAD algorithms) to be compared.
Chapter 3 addresses points 1 and 2 of the original contributions list. That chapter focuses on improving signal clarity through suppressing interferers. A unique approach is presented that combines spatial processing and BSS techniques. Existing BSS techniques require that the source signals be simultaneously mixed. The new method corrects for time delays inherent in the mixed signals due to source/receiver geometry by using spatial processing in the form of beamforming. Using simulations, the performance of the new method is assessed under varying conditions. These conditions include: source/receiver geometry, signal to interferer ratio, number of interferers or signals of interest, and signal to noise ratio. MVDR is used as a comparison to demonstrate the advantage of the new technique over null beamforming techniques. Experiments are also used to support the simulated results.

Improving signal clarity through emergent signal detection is the focus of Chapter 4. That chapter describes points 3 and 4 of the original contributions list. PCA will be used as an emergent signal detection algorithm. Several models are developed and the principal components for each model are derived. These models encompass the conditions of simultaneous mixing, delayed mixing, and mixing in the presence of uncorrelated background noise. A feed-forward processing method is also discussed. The significant statistic calculated at previous frames is used to influence the current frame’s result using a uniform weighting system (averaging). The PCA technique is also applied to the specific subset of emergent signal detection known as VAD. Simulations and experiments assess the performance of the new technique under a number of conditions, in both the general emergent signal detection framework and the VAD specific case.

Chapter 5 discusses the summary and conclusions of this dissertation. It includes a section on suggestions for future work. The Appendix contains the calculations from Chapter 4 that were not included in the body of the dissertation for brevity. This includes the eigenvalue derivation for a 2x2 matrix, and the full PCA for all of the models discussed in Chapter 4.
Chapter 2:

Background

2.1 Introduction to Array Processing

A sensor array is used to measure wavefields [49]. The measured wavefields can be acoustic, vibrational/seismic, or electromagnetic depending on the type of sensors used in the array. For this dissertation, the focus is on acoustic wavefields captured using microphone arrays. There are numerous applications of microphone arrays, ranging from sonar to acoustic surveillance, telecommunications, and medical diagnostics [50-53]. An array adds to the measurement domain, by collecting spatial information about the signals present in the environment. This can help mitigate some of the difficulties encountered in an acoustic environment. For example, by placing nulls in particular directions, interferers can be suppressed in recordings.

This chapter provides background information on the various techniques and algorithms that form the building blocks for the original research in this dissertation. Since the two main focus areas of this dissertation are blind source separation with spatial processing and emergent signal detection, most of this chapter covers previous algorithms that exist in those three areas. The emergent signal detection algorithms mentioned in this chapter, including the important subset of voice activity detection algorithms, do not rely on arrays. However, new work in this dissertation has developed new array processing techniques that have a higher performance than the existing algorithms.

2.2 Beamforming

Beamforming is a widely used array processing technique [54-55]. A beamformer is a spatial filter that focuses the array towards a specific angle. By choosing the proper channel
weightings, a desired directivity pattern can be formed. A beamformer can enhance a signal from a specific direction of arrival (DOA), while simultaneously attenuating sources whose wavefronts arrive from other angles. The microphones in an array should be positioned in order to best capture the spatial information of the signal of interest. In an optimal position, the array can be used to extract the signal of interest [56]. Beamformers can range from the simplest, data-independent delay and sum beamformer, to complex, data driven, adaptive techniques [57-59]. The type of beamformer to be used depends on a variety of factors, including the complexity, the desired interference suppression performance, and the information on the geometry and signals of interest available a priori.

For this dissertation, the sources are assumed to be point sources located in the far-field that arrive at the array as plane waves. The polar coordinate system is used, and the sources’ positions are (r,θ). The range to the source is mostly disregarded for Chapter 3 and the focus is on the angle of arrival, θ. Beamforming is used in two ways in this dissertation: first to scan through all possible angles and determine the angular location of sources, and second, to focus the array toward a particular source. The beamformer applies a vector of weights to the array elements and then sums the results to provide a single output per hypothesized focusing angle. A block diagram of a typical beamformer is shown in Figure 2.1 [56]. The signal model for these beamformers is only a function of attenuation and delay. If the weights are chosen properly, the signal of interest will be coherently summed, and increase in gain. Meanwhile, all sources in other locations will have non-aligned phase, and will not be summed in the output. The net result is a gain in signal to noise ratio and signal to interferer ratio.
Figure 2.1: A typical beamformer. Weights are applied to each array sensor, and then summed together to produce the output. The value of the weights varies depending on the type of beamformer used.

2.2.1 Narrowband Delay and Sum

The narrowband delay and sum (DS) beamformer is the simplest beamformer. A time delay is applied to each element based on the DOA of the source, and the shifted signals are summed [60]. The input to the array is a plane wave propagating in the direction \( \mathbf{a} \) with frequency \( \omega \). The direction \( \mathbf{a} \) is defined in polar coordinates as

\[
\mathbf{a} = \begin{bmatrix} -\sin \theta \\ -\cos \theta \end{bmatrix}
\]

(2.1)

where \( \theta \) is the DOA of the source [60]. As previously mentioned, this dissertation only considers the two-dimensional problem. Also a linear array is used and front back ambiguity is not considered. The results are easily extended into the third-dimension (and any array element geometry) by writing the direction \( \mathbf{a} \) in spherical coordinates to include an azimuth angle, and using a planar array. The array sensors lie along the \( z \) axis, and the \( n \)th array element is positioned at

\[
p_{x,n} = 0 \\
p_{y,n} = 0 \\
p_{z,n} = \left( n - \frac{N - 1}{2} \right) d
\]

for \( n = 0,1,\ldots,N-1 \)

(2.2)
where \( N \) the number of elements in the array and \( d \) is the distance between array elements (a constant value) [60]. If \( s(t) \) is the source signal received at the origin, then the array sensors receive this signal as

\[
x(t, p) = \begin{bmatrix}
s(t - \tau_0) \\
s(t - \tau_1) \\ \\
\vdots \\
s(t - \tau_{N-1})
\end{bmatrix}
\]  \( (2.3) \)

where

\[
\tau_n = \frac{a^T p_n}{c}
\]  \( (2.4) \)

and \( c \) is the speed of sound. For the linear array case, the time delay constant reduces to

\[
\tau_n = \frac{p_{z,n} \cos \theta}{c}.
\]  \( (2.5) \)

By delaying each sensor’s output by \( \tau_n \) and summing, the source signal is aligned in time and summed coherently.

In the time domain, the array output signals would have to be convolved to implement this beamformer [60]. However, convolution in the time domain is equivalent to multiplication in the frequency domain. The Fourier transform of the \( n \)th array element signal is

\[
X_n(\omega) = \exp(-j \omega \tau_n) \mathcal{S}(\omega)
\]  \( (2.6) \)

where \( \mathcal{S}(\omega) \) is the Fourier transform of the source signal. To remove the time delay, the \( n \)th array signal is multiplied by the weight

\[
v_n(\omega) = \exp \left( -j \omega \frac{p_{z,n} \cos \theta}{c} \right).
\]  \( (2.7) \)

Each array element’s Fourier transformed output is multiplied by the corresponding weight, then inverse Fourier transformed, and summed. This process is less computationally intensive than the time domain implementation. Unfortunately, the dependence on \( \omega \) in Equation 2.7 causes the beamformer to be narrowband. Unlike in the time domain implementation, the directivity pattern of the frequency domain implementation is frequency dependent.
2.2.2 Broadband Delay and Sum

One of the challenges of acoustic array processing is that many of the signals of interest are broadband (such as speech). The narrowband DS beamformer calculates its weights based on a single frequency of interest. Therefore, its response is highly dependent on the selected frequency. Figure 2.2(b) shows how the directivity of the array varies with respect to frequency when the array is steered towards 20 degrees. If the beamformer weights are calculated at the incorrect frequency, there can be serious degradation in the signal. As an example, suppose there is a 3500Hz tonal signal located at 20 degrees. If the array is steered using a narrowband DS beamformer with weights calculated for 3500Hz, the output is shown in Figure 2.2(a). The result is the unaltered 3500Hz tone. If however, the beamformer weights are calculated at 4000Hz, the result is shown in Figure 2.2(c). The amplitude of the signal is considerably less than the signal in Figure 2.2(a), as well as being out of phase with the signal in Figure 2.2(a). This means that if a narrowband DS beamformer is used for a broadband signal, the frequency content that does not correspond to the frequency at which the weights are calculated will be distorted in the output. A broadband beamformer is one where the power remains nearly constant at a certain look direction over a broad range of frequencies.

![Figure 2.2](image)

**Figure 2.2**: The effect of frequency on the narrowband DS beamformer. (b) The response of a 10 element linear array with equal spacing of one inch for varying frequency. A 3500Hz tonal signal is present at 20 degrees. (a) The output of the narrowband DS beamformer when the weights are calculated for 20 degrees and 3500Hz. (c) The output of the narrowband DS beamformer when the weights are calculated for 20 degrees and 4000Hz.
Some broadband beamformers use nested sub-arrays to achieve constant power in the look direction for a band of frequencies [56]. For different frequencies, different combinations of array elements are used for beamforming. While this method is effective, it usually requires a high number of microphone elements to achieve the correct sub-array geometries. For this dissertation, a narrowband decomposition process is used to achieve broadband performance. At each frequency, a narrowband DS beamformer is applied and the results from each beamformer are summed together [56]. If this beamformer is transformed into its time-domain counterpart, it is more commonly known as the filter-and-sum beamformer, first developed by Frost [61].

Figure 2.3 shows the block diagram for the frequency domain version of this beamformer. Figure 2.4 shows the response of the broadband beamformer as compared to the response with a narrowband beamformer. A white, Gaussian noise source is bandpass filtered between 3000 and 4000Hz and located at 20 degrees. A series of narrowband DS beamformers, with the weights calculated for 2750, 3200, 3500, 3800, and 4250Hz, were applied to the array data. The maximum beamformer power only appears at 20 degrees for the 3200Hz case. The black curve in Figure 2.4 shows the response of the broadband beamformer when the narrowband decomposition frequencies are between 3000 and 4000Hz. The maximum power occurs at 20 degrees for the broadband beamformer.

**Figure 2.3:** [56] Broadband beamformer block diagram based on narrowband decomposition.
Figure 2.4: The response of the narrowband DS beamformer versus the response of the broadband beamformer for a bandpassed signal that is 1000Hz wide, centered at 3500Hz, and located at 20 degrees. For the series of narrowband DS beamformers examined, only the 3200Hz case has maximum power at 20 degrees.

This technique has many of the same properties as the narrowband DS beamformer. Especially useful is the ability to calculate all the weights in advance, which can drastically reduce processing time. The weights for all frequencies up to the Nyquist frequency are calculated in advance and stored in a table. From that table, any group of frequencies can be selected. Therefore, the frequencies of interest can be changed whenever desired. The shortcoming of this method is that the frequencies of interest must be carefully selected. This is to prevent leakage into the beamformer output from frequencies outside the range of interest, especially low frequency noise. For example, let the signal of interest be a voice signal with most of its power occurring between 500 and 5000Hz. If no other noise sources are present, the range of frequencies of interest could be chosen between 10Hz and 10kHz without signal degradation. If however, an interferer is in the room producing 100Hz noise, choosing the range between 10Hz and 10kHz will cause leakage from the unwanted interferer into the signal of interest.
2.2.3 Minimum Variance Distortionless Response (MVDR)

The two beamformers discussed above are data independent beamformers. This section reviews an adaptive, data-dependent beamformer, known as the Capon (or MVDR) beamformer. The MVDR strategy is to minimize the variance of noise in the look direction, without distorting the signal of interest coming from the look direction [62]. The weights are adaptively calculated, using knowledge of the received data, to coherently sum the signal of interest, while simultaneously minimizing the noise power at the output. The derivation begins with a set of spatially compact acoustic sources in a free field with amplitudes \( s(k) = [s_1(k), s_2(k), \ldots, s_n(k)]^T \) that exist at directions \( \theta_1, \theta_2, \ldots, \theta_n \). The sources are assumed to be narrowband and Gaussian distributed. The signals \( x(k) \) are

\[
x(k) = a(\theta)s(k) + n(k)
\]

(2.8)

where \( x(k) \) is the output of the array, \( a(\theta) = [a(\theta_1), a(\theta_2), \ldots, a(\theta_n)] \) is the array manifold vector corresponding to the directions of arrival of the sources, and \( n(k) \) is a vector of noise that represents the undesired signals including background noise and interferers. The array manifold vector is a function of the array geometry, the signal frequency, and the look direction, \( \theta \), and is the same vector that is used in the narrowband DS beamformer to steer the array toward the look direction. The combined beamformer output of the MVDR process is denoted \( y(k) \) and is given by [63]

\[
y(k) = W^* x(k) = W^* a(\theta)s(k) + W^* n(k)
\]

(2.9)

where \( W \) is a vector of array weights and \( \{.\}^* \) is the Hermitian transpose. The goal is to force the signals of interest, \( s(k) \), to be undistorted and to minimize the effect of the noise, \( n(k) \). In other words, force \( W^* a(\theta) = 1 \) and \( W^* n(k) \) to be small. In doing this, \( y(k) \approx s(k) \), and \( s(k) \) is recovered from the array signals. The variance of the noise at the beamformer output when the beamformer is steered to \( \theta_s \) is given by

\[
E\{W^* n(n) W\} = W^* R_n W
\]

(2.10)

where the variance of the noise is represented by the noise covariance matrix, \( R_n = E\{nn^*\} \) and \( E\{.\} \) is the expectation operator. The requirement that \( W^* a(\theta) = 1 \) ensures that \( s(k) \) in Equation 2.9 is undistorted by the application of the beamforming weights. The weight vector, \( W \), is given as the optimal solution to the LaGrange problem.
\[
\min W^*R_nW \quad \text{subject to} \quad W^*a(\theta) = 1
\]  
(2.11)

In typical applications, the noise covariance is unknown, and the matrix \( R_n \) cannot be determined. The recently received array samples are used to calculate the sample covariance matrix \( R_y \), which is used as a substitute for the noise covariance matrix. Replacing \( R_n \) with \( R_y \) in Equation 2.10 and solving produces the analytical MVDR solution [63]

\[
W = \frac{R_y^{-1}a(\theta)}{a^*(\theta)R_y^{-1}a(\theta)}.
\]  
(2.12)

The MVDR equation shows that the weights depend on the covariance matrix and the presumed array manifold vector. As a consequence, the beamformer is sensitive to inaccuracies in the manifold vector and covariance matrix estimate. Array manifold errors are often a problem for towed array sonar where the shape of the array is unknown due to the motion of the submarine causing bending of the array [64]. In some of these instances, the performance of the MVDR beamformer can degrade below the performance of the standard non-adaptive techniques [65], especially if the inaccuracy is great enough that the beamformer treats the signal of interest as an interferer [66]. In addition, the MVDR filter reduces to a delay and sum beamformer when the noise signals at the array are mutually correlated and have the same power [56].

The advantage of the MVDR beamformer is that the locations of the interferers do not need to be known \textit{a priori}; the beamformer suppresses contributions from all sources not lying along the steering direction \( \theta_s \). The algorithm allows for maximum noise reduction by nulling out anything not in the look direction [56]. This technique provides much better resolution and interference rejection than the data independent methods [66]. Due to the numerous advantages of the MVDR beamformer, many techniques have been developed to improve its robustness [67-70]. Among these techniques are adding diagonal loading [71-72] and using a set of steering vectors to account for look direction uncertainty [73-74].

The previous sections discussed three of the numerous beamforming strategies that exist. As long as the sources are spatially distinct, they can be separated or extracted using any beamforming technique. This is because beamforming takes advantage of the spatial geometry of the sources to separate and extract the signals of interest. Another class of algorithms, known as blind source separation (BSS) relies on the statistics of the signals for separation and extraction.
Two methods of BSS, known as principal component analysis and independent component analysis, will be discussed in the following section.

2.3 Blind Source Separation

Consider the situation where several people are talking at the same time. The brain has the ability to focus on one particular speaker while ignoring other conversations and background noise. This is known as the cocktail party problem, and is widely studied in acoustic signal processing. The problem was originally studied to aid air traffic controllers, who had difficulty understanding individual pilots when several pilot's voices were played simultaneously over a single loudspeaker [75]. BSS algorithms use signal processing techniques to separate the mixed sources in order to extract one (or more) of the sources. This improves the intelligibility of the signal of interest by focusing on a single source from within a mixture.

BSS encompasses a wide class of algorithms whose goal is to recover a set of sources that have been linearly mixed without any prior knowledge of the signals or the mixing matrix. The sources, $s(t) = [s_1(t),...,s_M(t)]^T$, are mixed using a linear mixing matrix, which produces the mixed signals, $x(t) = [x_1(t),...,x_N(t)]^T$. The BSS algorithm only has access to the mixed signals, $x(t)$. For these algorithms to work, the pdf of the source signals needs to be known and the sources need to be statistically independent. Many algorithms separate the sources by removing any dependencies or correlation between the received signals [76-77]. Early efforts found that if all the source signals are Gaussian distributed, there could be no solution to the BSS problem [78]. Therefore, at most, one of the sources can have a Gaussian distribution, while all the others must be distinctly non-Gaussian.

2.3.1 Principal Component Analysis

PCA is a tool used for analyzing multivariate data that has seen use across a wide variety of application areas such as classification, pattern recognition, and noise reduction [79-82]. PCA transforms the mutually correlated variables into a set of uncorrelated variables that preserves the variation in the original data set. This allows PCA to separate sources that are uncorrelated. This correlation transformation is only dependent on the second order statistics of the signals and has
a simple, closed form solution [78]. The transformed variables are ordered so that the first few components contain most of the original variation information, and the last components contain little to no additional variance information. By examining the variances of each new component, PCA can identify patterns and trends in large, complex, mutually correlated data sets [85]. In addition, if the components with minimal variance contribution are eliminated, PCA can be used as a data compression tool [83-85].

Suppose that \( \mathbf{x} \) is a vector of \( p \) random variables that are mutually correlated. The first principal component (PC) is a linear combination, \( y_1 = \mathbf{w}_1^T \mathbf{x} \), such that the variance of \( y_1 \) is a maximum. In other words, the first PC is calculated by finding a weight vector, \( \mathbf{w}_1 \), that is the optimal solution to the LaGrange problem

\[
\max \{ \text{var}(y_1) \} \text{ subject to } \|\mathbf{w}_1\|=1
\]  

(2.13)

where the norm of \( \mathbf{w}_1 \) is the Euclidean norm. The norm of \( \mathbf{w}_1 \) is constrained to be equal to one to prevent the variance from growing without limits. The variance of the PC can be rewritten as

\[
\text{var}(y_1) = \mathbb{E}\{y_1^2\} = \mathbb{E}\{\mathbf{w}_1^T \mathbf{x} \mathbf{w}_1\} = \mathbf{w}_1^T \mathbb{E}\{\mathbf{x} \mathbf{x}^T\} \mathbf{w}_1 = \mathbf{w}_1^T \mathbf{C}_\mathbf{x} \mathbf{w}_1
\]  

(2.14)

where \( \mathbb{E}\{\cdot\} \) is the expectation and \( \mathbf{C}_\mathbf{x} \) is the covariance matrix of \( \mathbf{x} \). Substituting Equation 2.14 into Equation 2.13 and differentiating the LaGrange function produces the equation

\[
\mathbf{C}_\mathbf{x} \mathbf{w}_1 - \lambda \mathbf{w}_1 = (\mathbf{C}_\mathbf{x} - \lambda) \mathbf{w}_1 = 0,
\]  

(2.15)

which shows that the weight vector \( \mathbf{w}_1 \) is the unit length eigenvector of the covariance matrix.

The kth PC of \( \mathbf{x} \) is \( y_k = \mathbf{w}_k^T \mathbf{x} \), and has a variance given by

\[
\text{var}(y_k) = \mathbf{w}_k^T \mathbf{C}_\mathbf{x} \mathbf{w}_k = \mathbf{w}_k^T \lambda_k \mathbf{w}_k = \lambda_k \mathbf{w}_k^T \mathbf{w}_k = \lambda_k
\]  

(2.16)

where \( \lambda_k \) is the kth largest eigenvalue of \( \mathbf{C}_\mathbf{x} \). The second PC is another linear combination, \( y_2 = \mathbf{w}_2^T \mathbf{x} \), such that \( \mathbf{w}_2 \) is orthogonal to \( \mathbf{w}_1 \) (uncorrelated with the first PC). Since eigenvectors are always orthogonal, the weight vectors, \( \mathbf{w}_i \), are the set of eigenvectors of the data covariance matrix [85].

For this dissertation, PCA will be used in an application where only two channels of data are measured. Therefore, the data set \( \mathbf{x} \) is a 2xn matrix (also assumed zero mean). This allows for a closed form derivation of the PCs in order to test the hypothesis that PCA can be used as a simple and robust emergent signal detection technique. The sample covariance matrix is calculated element by element according to
\[ c_s(i, j) = \text{E}\{x_i^T x_j\} = \frac{1}{n} \sum_{k=1}^{n} x_i(k) x_j(k) \]  

For the data set \( x \), the covariance matrix will be a 2x2 matrix, and have two eigenvalues. The closed form solution to the eigenvalue problem is straightforward to derive, and is

\[ \lambda_{1,2} = 0.5[c_s(1,1) + c_s(2,2)] \pm 0.5\sqrt{[c_s(1,1)^2 - c_s(2,2)^2 + 4c_s(1,2)^2]} \]  

The derivation of the eigenvalues of the covariance matrix is given in Appendix A.

PCA separates uncorrelated sources by determining directions that have maximum variance. Uncorrelated sources are recovered by projecting the mixed signals along these directions. The addition of emerging signals alters the eigenvalue subspace, and therefore, the projection directions. By examining the changes in the individual PCs, the change in the subspace can be uncovered, and PCA can be used to detect emergent signals. The advantage of the PCA method is that it is non-adaptive and has an easily derived closed form solution. However, the condition that the sources are uncorrelated is not as strong as the condition that the source signals are statistically independent. In some instances, separation based on correlation leads to incomplete separation. The following section will demonstrate this phenomenon and discuss another separation technique known as independent component analysis (ICA).

### 2.3.2 Independent Component Analysis

The main difference between PCA and ICA is that ICA separates sources by removing the statistical dependence between the mixed signals rather than removing the correlation between the mixed signals. Let \( s(t) = [s_1(t), s_2(t), \ldots, s_M(t)]^T \) be a set of statistically independent sources. These sources are mixed according to the model

\[ x(t) = As(t) \]  

where \( A \) is a full rank, scalar matrix. ICA algorithms seek to find a linear weight matrix, \( W \), that forms a set of signals \( y(t) \) according to

\[ y(t) = Wx(t) \]  

The goal of ICA is to form this weight matrix such that the separated signals, \( y(t) \), are all statistically independent [78]. In this way, the original signals, \( s(t) \), are recovered in \( y(t) \). Two assumptions for ICA are that no (or only very low) additive noise is present and that the number of sensors must be greater than or equal to the number of source [78].
A mixture of two or more sub- or super-Gaussian signals is more Gaussian than either of its source signals [86]. If the source signals are all non-Gaussian, one BSS method is to find the direction where the unmixed signals are least Gaussian [86]. This method is known as projection pursuit, and it provides a good visualization of how BSS works. For projection pursuit, the kurtosis provides a measure of the Gaussianity of a signal [86]. The kurtosis of the extracted signal, \( y(t) = w^T x(t) \), is a function of the fourth order moment defined as

\[
K = \frac{\frac{1}{N} \sum_{t=1}^{N} (y - y')^4}{\left(\frac{1}{N} \sum_{t=1}^{N} (y - y')^2\right)^2} - 3, \tag{2.21}
\]

which has a value of zero for a Gaussian distribution and a non-zero value for sub- and super-Gaussian distributions. Figure 2.5 shows how the kurtosis of the extracted signal varies as the weight vector is rotated around the origin. The two sources are recovered exactly when the weight vector is aligned with the direction of maximum kurtosis. These are indicated using red kurtosis in Figure 2.5. ICA is a multivariate parallel version of projection pursuit [86]. Instead of searching out the directions that are the most non-Gaussian individually, ICA finds all the directions at once using various projection measures. Usually, this search is implemented using a gradient based method [86].
Figure 2.5: The kurtosis of the extracted signal varies as the weight vector is rotated around the origin. The amplitude of the first signal mixture versus the amplitude of the second mixture is plotted with blue dots. The kurtosis of the recovered signal, $y(t) = w^T x(t)$, is plotted as a function of the weight vector orientation in black. For each orientation direction, kurtosis is plotted as the distance from the origin. The two red lines show the directions of maximum kurtosis. When the orientation of the weight vector is in the same direction as these red lines, the source signals are recovered exactly.

Whereas PCA decorrelates the mixed signals, which involves only the second order statistics, ICA reduces the higher order statistical dependencies [78]. The ICA assumption that all the sources are statistically independent is more strict than the PCA assumption that the sources are only uncorrelated. Figure 2.6 shows the difference between ICA and PCA by showing the joint pdfs of the original sources, the mixed sources, and the unmixed sources using both PCA and ICA [78]. The two sources, whose independence is demonstrated in Figure 2.6(a), are linearly mixed using the model of Equation 2.19. Figure 2.6(b) shows the joint pdf of the mixed signals, which are dependent [78]. The PCA solution decorrelates the mixed signals, but as Figure 2.6(c) shows, the recovered signals are still dependent. The two independent sources are recovered using ICA, as shown in Figure 2.6(d). Another difference from PCA is that ICA requires that no more than one of the sources be normally distributed.
2.3.3 Two ICA Algorithms of Interest

Two specific ICA algorithms are discussed in this section. These algorithms are Infomax and FastICA. These particular algorithms are included because they are widely used for separation problems. They are widely published and often used as benchmarks for other source separation techniques. Both algorithms accomplish source separation based on minimizing mutual information between the mixed signals. Another similarity is that both algorithms require an estimate of the input signals’ pdfs.

2.3.3.1 Infomax

Bell and Sejnowski [87] developed a method for BSS based on minimizing the mutual information between array sensor outputs using arguments based on entropy. By reducing the mutual information, the redundancy between the signals is removed. This causes each output to contain information about only one source signal, and therefore separates the sources [78]. A set of sources, $s(t)$, are linearly mixed by a scalar mixing matrix, $A$, which produces a set of mixed
signals $x(t)$. This is the standard ICA model given in Equation 2.19. Consider an input vector $x(t)$, a weight matrix $W$, and a transformed output vector, $y = g(Wx)$ [87]. Maximizing the joint entropy of the outputs of the neural processor minimizes (approximately) the mutual information between the output signals, $y(t)$ [78]. The entropy of the outputs is given by

$$H(y) = -E[\ln p_y(y)]$$

(2.22)

where $p_y(y)$ is the pdf of the sensor output $y(t)$, and $E\{\cdot\}$ is the expectation operator. The function $g(x)$ has the form of the assumed cumulative distribution function (cdf) of the source signals, which bounds the pdf of the output signal. This maximizes the entropy in a constrained fashion, rather than simply increasing the variance (and therefore the entropy) to infinity. If $g(x)$ is a monotonically increasing function, the pdf of the array output can be written as a function of the pdf of the source signals $x$, according to

$$p_y(y) = \frac{p_x(x)}{|J|}$$

(2.23)

where $|J|$ is the absolute value of the Jacobian of $g(x)$. Substituting Equation 2.23 into Equation 2.22 gives the final entropy

$$H(y) = E[\ln |J|] - E[\ln p_x(x)].$$

(2.24)

Since the second term in Equation 2.24 is unaffected by changes in $W$, only the first term needs to be maximized in order to maximize the entropy. An online stochastic gradient ascent learning rule is used to determine the optimal weight set, $W$, to separate the sources. The learning rule for Infomax is

$$\Delta W \propto \frac{\partial H(y)}{\partial W}$$

$$\frac{\partial H(y)}{\partial W} = \frac{\partial}{\partial W} \left[ \ln \det \left( \frac{\partial y}{\partial x} \right) \right].$$

(2.25)

$$= \left[ W^T \right]^{-1} - 2yx^T$$

The stochastic gradient ascent learning rule contains an inverse, which can lead to instability if the weight vector $W$ is poorly conditioned. The natural gradient is used to rescale the entropy gradient by post multiplying by $W^T W$ [88]. This changes the form of the learning rule to
\[
\frac{\partial H(y)}{\partial W} = [W^T]^T W^T W - 2y^T W^T W
\]
\[
= W - 2y(Wx)^T W
\]
\[
= \left[ I - 2y(Wx)^T \right] W
\]

The advantage of this learning rule is that the inverse has been removed and the algorithm is more stable.

### 2.3.3.2 FastICA

Instead of using the joint entropy of the mixed signals, Hyvärinen [89] used the negentropy as the measure of nongaussianity. Negentropy is a measure of differential entropy, which is defined as

\[
J(y) = H(y_{gauss}) - H(y)
\]

where \( J \) is the negentropy, \( H(.) \) is the entropy, and \( y_{gauss} \) is a Gaussian random vector of the same covariance matrix as \( y \). This provides a measure of nongaussianity that is always non-negative and only zero for a signal with Gaussian distribution. The FastICA algorithm seeks to find a direction, \( w \), such that the projection, \( w^T x \), maximizes the negentropy. Using Equation 2.27 to find the negentropy is computationally very difficult [89], so it is approximated as

\[
J(w^T x) \approx \left[ E[g(w^T x)] - E[g(v)] \right]^2
\]

where \( g(.) \) is any non-quadratic function, and \( v \) is a Gaussian variable of zero mean and unit variance. The variance of \( w^T x \) must be constrained to unity, which is equivalent to constraining the norm of \( w \) to be unity for whitened data [89]. The FastICA algorithm chooses \( w \) as the solution to the optimization problem

\[
\max \left[ E[g(w^T x)] - E[g(v)] \right]^2 \text{ subject to } E\left( w^T x \right)^2 = \| w \|^2 = 1
\]

Similar to the Infomax method, only the first term in Equation 2.29 is dependent on updated values of \( w \). Using Newton’s methods and algebraic simplification, the stabilized fixed point algorithm can be solved as

\[
w^+ = E[xg(w^T x) - E[g'(w^T x)]]
\]

An initial value of the weight vector, \( w \), is chosen and \( w^+ \) is calculated. This updated value of the weight vector is normalized to unit length and compared to the previous weight vector. If the two
vectors have a dot product of zero, they define the same direction and the algorithm has converged. A typical choice of nonlinear function is the hyperbolic tangent, which is ideal for separating super-Gaussian sources, such as voice signals. There are several advantages to using the FastICA algorithm instead of the Infomax algorithm for source separation [89]. The convergence for the FastICA algorithm is at least quadratic (and often cubic), which is much faster than the linear convergence of the Infomax method. FastICA is a neural algorithm, and has most of the advantages of neural methods. It is parallel, distributed, computationally simple, and requires very small amounts of memory space [89]. While a non-linearity function needs to be specified in advance, the algorithm will find independent components using any non-linearity. In contrast, for the Infomax method, the pdf of the source signals must be known in advance so the proper non-linearity function can be selected.

Up to this point, the background has focused on algorithms designed to increase the clarity of the signal of interest by reducing/removing interferers. Another side of signal clarity is the ability to both detect signals of interest and determine the occurrence of emergent signals in the measurements. Emergent signal detection can be used to eliminate segments of time that do not contain the signal of interest. This eliminates portions of the signal that contains only interferes. For segments when the signal of interest is present, BSS and beamforming can be used to remove the interferer and improve the signal of interest’s clarity. The following sections discuss emergent signal detection algorithms and the specific subset of voice activity detection.

### 2.4 Emergent Signal Detection

Emergent signal detection is a widely studied problem with a variety of applications [90-94]. The goal is to detect the presence of a desired signal in a noisy observation. A signal of interest, whether it is a radar signal or an earthquake seismograph signal, is distorted or corrupted in some unknown manner. For example, the radar signal may be distorted by clutter, reverberation, or background noise. For this section, the signal of interest is assumed to be corrupted by an interferer according to the model

\[ x(t) = s(t) + v(t) \]  

(2.31)
where \( x(t) \) is the recorded signal, \( s(t) \) is the signal of interest, and \( v(t) \) is the interferer. For the purpose of this dissertation, emergent signal detection will use the framework of hypothesis testing to determine if the signal of interest is present.

### 2.4.1 Hypothesis Testing

Statistical hypothesis testing is one of the primary tools used to detect the presence of an emergent signal in a set of measurements. For this dissertation, it is assumed that there are only two hypotheses: the signal of interest is present or the signal of interest is absent. In hypothesis testing, a sufficient statistic is calculated from a small sample of data and used to select one of the hypotheses [95]. The two hypotheses are labeled as \( H_0 \) and \( H_1 \), for signal present and signal absent respectively. The sufficient statistic associated with the \( H_0 \) hypothesis has an amplitude probability distribution \( P_0 \), while the sufficient statistic associated with the \( H_1 \) hypothesis has the amplitude probability distribution \( P_1 \) [96]. The goal in hypothesis testing is to develop a decision rule that partitions the space of all possible sufficient statistic values into two sets, with each set corresponding to one of the hypotheses. In other words, the decision rule seeks to determine if a particular sufficient statistic belongs to distribution \( P_0 \) or \( P_1 \). When a sample of data is taken and its sufficient statistic is calculated, the decision rule determines which pdf the sample falls under. By associating the sample with a particular pdf, the sample can be classified as belonging to either the \( H_0 \) or \( H_1 \) hypothesis. The log-likelihood ratio test [97] is one general test used to determine a decision rule for hypothesis testing.

### 2.4.2 Log Likelihood Test

This test for emergent signal detection is a test between two simple hypotheses. The \( H_0 \) hypothesis states that both the signal of interest and the interferer are present, while \( H_1 \) states only the interfering source is present. For this derivation, the signal of interest and interferer are additively mixed. The interferer is assumed to be white, Gaussian noise, with a mean of one and a variance of \( \sigma_0^2 \). The signal of interest has an unknown distribution. However, regardless of the signal distribution, signal mixtures tend to have Gaussian pdfs [86]. Therefore, the pdf of the \( H_1 \) hypothesis is assumed to be a normal distribution with zero mean and a variance of \( \sigma_1^2 \). A frame of data, \( y = (y_1, \ldots, y_N) \), is collected and the two hypotheses’ pdfs are
\[ H_0 : p_0(y_1, ..., y_N) = \frac{1}{\sqrt{2\pi} \sqrt{\sigma_0^2}} \exp \left( -\frac{1}{2} \sum_{k=1}^{N} \frac{y_k^2}{\sigma_0^2} \right) \]
\[ H_1 : p_1(y_1, ..., y_N) = \frac{1}{\sqrt{2\pi} \sqrt{\sigma_1^2}} \exp \left( -\frac{1}{2} \sum_{k=1}^{N} \frac{y_k^2}{\sigma_1^2} \right) \] (2.32)

where \( N \) is the length of the signal \( y \) [97].

A frame of data is collected, which has a normal distribution with zero mean and \( \sigma_y^2 \) variance, given as
\[ p(y_1, ..., y_N) = \frac{1}{\sqrt{2\pi} \sqrt{\sigma_y^2}} \exp \left( -\frac{1}{2} \sum_{k=1}^{N} \frac{y_k^2}{\sigma_y^2} \right). \] (2.33)

The goal is to determine if this frame belongs to the \( H_0 \) or \( H_1 \) hypothesis. The likelihood ratio is a ratio between two pdfs; the likelihood ratio between the received data and the \( H_1 \) hypothesis is
\[ L(y_1, ..., y_N) = \frac{p_0(y_1, ..., y_N)}{p_1(y_1, ..., y_N)} = \frac{\sqrt{2\pi} \sqrt{\sigma_0^2} \exp \left( -\frac{1}{2} \sum_{k=1}^{N} \frac{y_k^2}{\sigma_0^2} \right)}{\sqrt{2\pi} \sqrt{\sigma_1^2} \exp \left( -\frac{1}{2} \sum_{k=1}^{N} \frac{y_k^2}{\sigma_1^2} \right)} \]
\[ = \sqrt{\frac{\sigma_0^2}{\sigma_1^2}} \exp \left( \frac{1}{2} \sum_{k=1}^{N} \frac{\left(\sigma_0^2 - \sigma_y^2\right)}{\sigma_0^2 \sigma_y^2} \right) \] (2.34)

It is convenient to consider the natural log of the likelihood ratio in order to remove the exponential from Equation 2.34, and form a test statistic that is linear with respect to the data [94]. The log-likelihood test statistic is
\[ \ln[L(y_1, ..., y_N)] = \ln \sqrt{\frac{\sigma_y^2}{\sigma_0^2}} + \frac{1}{2} \sum_{k=1}^{N} \frac{y_k^2}{\sigma_0^2 \sigma_y^2} \] (2.35)

The variance of the \( H_1 \) hypothesis is estimated using a number of initial frames that are assumed to contain only the interferer. This estimate can be updated whenever a future frame is declared to satisfy the \( H_1 \) hypothesis. For each incoming frame, the log-likelihood ratio is calculated, and used to determine if the frame belongs to the \( H_0 \) or the \( H_1 \) hypothesis. When the signal variance matches the \( H_1 \) variance, the test statistic reduces to
\[ \ln[L(y_1, ..., y_N)] = \ln \sqrt{(1)^2} + \frac{1}{2} \sum_{k=1}^{N} \frac{y_k^2(0)}{(\sigma_0^2)^2} \]
\[ = \ln 1 + 0 = 0 \] (2.36)
On the other hand, if the signal variance matches the $H_0$ variance, the test statistic becomes

$$\ln[L(y_1, \ldots, y_N)] = \ln \left( \frac{\sigma_1^2}{\sigma_0^2} \right)^N + \frac{1}{2} \sum_{i=1}^{N} y_i^2 \left( \frac{\sigma_0^2 - \sigma_1^2}{\sigma_0^2 \sigma_1^2} \right).$$

(2.37)

The presence of the signal of interest introduces a change in the variance of the received signal. This change in variance causes a change in the test statistic. Therefore, a threshold can be assigned that distinguishes between the two hypotheses.

The advantage of the log-likelihood detection test is that it gives a general framework for the detection procedure [94]. As long as the variance of the $H_1$ hypothesis is different than the variance of the $H_0$ hypothesis, the log-likelihood can be used to detect the emergent signal. Since the likelihood ratio is the ratio between the sampled distribution and the $H_1$ distribution, the test statistic distinguishes between the two hypotheses without knowledge of the $H_1$ variance. While only the normal distribution was used for this derivation, the procedure can accommodate a variety of distributions. By substituting the assumed pdf into Equation 2.34, the log-likelihood ratio can be adapted to other distributions.

A specific application of emergent signal detection is voice activity detection (VAD). Whereas the log-likelihood test can accommodate a variety of pdf forms and detects numerous types of signals, VAD is specifically formulated to detect the presence of speech in a noisy measurement. Since the signal of interest is defined to be speech, emergent signal detection algorithms can be tailored based on the known statistical properties of speech. The following section discusses the specific class of emergent signal detectors known as VAD.

### 2.5 Voice Activity Detection

The goal of a VAD algorithm is to determine whether a voice signal is present in a measured signal. Mobile telephones use VAD algorithms to decide when to transmit in order to avoid transmitting noise only blocks of data that occur between active voice frames. Only transmitting during active voice frames helps to conserve power and increase comprehension [12, 18]. There are a large set of VAD algorithms in use [98-102]. One class of algorithms uses higher order statistics, such as kurtosis and skewness, to determine if a voice signal is present.
Others rely on energy content in voice specific frequency bands [104], while still others use pattern recognition to identify syllables of speech [105].

Similar to emergent signal detection, a VAD algorithm divides the data into frames and for each frame calculates a significant statistic. The significant statistic is used to determine if the current frame contains a voice signal. (Sometimes, like in the higher order statistic based method presented in Section 2.5.1, the VAD algorithm’s decision is based on multiple significant statistics.) One challenge that VAD algorithms face is an issue referred to as “clipping” in the VAD literature [106]. Many algorithms have difficulty determining when speech begins and ends, and as a result can cut off the beginnings and endings of words or sentences. This occurs because the SNR is lowest at the onset and conclusion of speech, making it particular difficult to detect [106]. Some algorithms use an overhang period to overcome the clipping at the end of words/sentences. This means that when the VAD algorithm detects a drop off in speech amplitude, it waits a fixed number of frames before it stops declaring the frames as speech [106]. This is important because clipping the end of sentences or words can significantly decrease the intelligibility [107]. In real time VAD systems, the same technique cannot be used to buffer the beginning of sentences. Fortunately, front end clipping goes mostly unnoticed by users [106]. In this dissertation, this overhang processing is not included, since intelligibility is not the main focus; however, it would be easily added to the algorithm should it be desired later.

2.5.1 Higher Order Statistical Approach to VAD

One commonly used approach to VAD uses the higher order statistics of the signal to distinguish between active voice frames and noise frames [103]. The received signal is divided into frames, and several higher order statistic metrics are calculated. Using these metrics, the frame is declared as either speech or noise. Every 10ms, the second, third, and fourth-order moments are calculated using

$$M_{k,t} = \frac{1}{N} \sum_{n=0}^{N-1} [x(n)]^k$$  \hspace{1cm} (2.38)

where $x(n)$ is the received signal and $k$ is the moment order. Using these moments, the unbiased normalized skewness (SK) and the unbiased kurtosis (KU) are calculated using
where $v_g$ is the noise energy. The measure for SK and KU are normalized by the signal energy to give

$$
\gamma_3 = \frac{SK}{M_{2,x}^{1.5}},
\gamma_4 = \frac{KU}{M_{2,x}^{2}}.
$$

(2.40)

The noise power is estimated from frames that are declared as non-speech. For initialization, the first three frames are declared as non-speech and used to initialize the noise energy. Whenever a frame is declared as non-speech, its energy is used to update the estimate for the noise energy according to an autoregressive averaging

$$
v_g(k) = (1 - \beta)v_g(k - 1) + \beta M_{2,x}.
$$

(2.41)

where $k$ is the iteration index, and $\beta$ is $0.1 \times \text{Prob}[\text{Noise}]$. The probability of the frame being noise is based on the value of $KU_b$, which is the unit-variance version of the kurtosis defined as

$$
KU_b = \frac{KU}{\sqrt{3v_g^2 \left( \frac{104}{N} + \frac{452}{N^2} + \frac{596}{N^2} \right)}}.
$$

(2.42)

where $N$ is the number of samples in the frame. Using this value and the value of $SK$ from Equation 2.39, the probability of the frame being noise is

$$
\text{Prob}[\text{Noise}] = \frac{\text{erfc}(SK) + \text{erfc}(KU_b)}{2}
$$

(2.43)

The final parameter calculated is the SNR. The SNR of the frame is

$$
\text{SNR} = \text{Pos}\left[ \frac{M_{2,x} - 1}{v_g} \right]
$$

(2.44)

where $\text{Pos}[x] = x$ for $x > 0$ and 0 otherwise. The values of $\gamma_3$, $\gamma_4$, $\text{Prob}[\text{Noise}]$, and SNR are used with thresholds to determine the state of the current frame. The VAD algorithm is a two-state...
machine, as shown in the Figure 2.7. If the previous frame was declared as noise, and either the Prob[Noise] is below its threshold value or the SNR is greater than its threshold value, a transition is triggered and the frame is declared as a speech state. A transition from a speech state to a noise state is triggered by the Prob[Noise] being greater than its threshold, and $\gamma_3$ and $\gamma_4$ being above their respective thresholds. For extensive details on the VAD algorithm see [103].

![Decision process for the higher order statistic VAD. The conditions outlined are the conditions that must be met to switch from either a speech to noise state or a noise to speech state.](image)

**Figure 2.7:** Decision process for the higher order statistic VAD. The conditions outlined are the conditions that must be met to switch from either a speech to noise state or a noise to speech state.

### 2.5.2 Power Spectral Density Content Approach to VAD

The previously discussed VAD algorithm required the calculation of several metrics and a complicated decision rule. Several thresholds must be jointly selected, and must be altered based on the characteristics of the noise and environment. In addition, the decision on the state of the current frame is affected by several previous decisions. The second VAD algorithm used in this work is based on a hypothesis test operating on a single value – the low-variance spectrum estimate [108]. The kth received speech signal frame, $s_k(n)$, is corrupted by a stationary additive noise signal, $v_k(n)$. The measured signal is $x_k(n) = s_k(n) + v_k(n)$. This method determines the
presence of speech based on the SNR, which is calculated using the power spectral density (PSD) of the signal. The SNR is defined as [108]

$$\psi_k(f_l) = \frac{P_{ss,k}(f_l)}{P_{vv}(f_l)} - 1.$$  \hspace{1cm} (2.45)

$P_{vv}(f_l)$ is the estimated value of the noise PSD and $P_{ss,k}(f_l)$ is the PSD of the current frame (both at frequency $f_l$). The estimated value of the noise PSD is calculated as

$$P_{vv}(f_l) = \frac{1}{K} \sum_{k=0}^{K-1} P_{xx,k}(f_l)$$  \hspace{1cm} (2.46)

assuming an initial $K$ frames that contain no speech. In this algorithm, there are two hypotheses: $H_1$ that represents the case where only noise is present and $H_0$ where both speech and noise are present. These are given by

$$H_0 : \psi_k(f_l) = \frac{P_{ss,k}(f_l) + P_{vv,k}(f_l)}{P_{vv}(f_l)} - 1$$

$$H_1 : \psi_k(f_l) = \frac{P_{vv,k}(f_l)}{P_{vv}(f_l)} - 1$$  \hspace{1cm} (2.47)

where $P_{vv,k}(f_l)$ and $P_{ss,k}(f_l)$ represent the actual PSD of the noise and speech for the kth frame at the frequency $f_l$. When only noise is present, the test statistic is zero mean and Gaussian distributed. The presence of voice introduces a significant shift in the mean of the test statistic. Therefore, a threshold can be assigned that distinguishes between the two hypotheses. In [108], an appropriate threshold was derived based on the user’s desired false alarm probability. The threshold is

$$\eta_k(f_l) = \sqrt{2\sigma_v^2(f_l)} \cdot \text{erfc}^{-1}(2P_{fa})$$  \hspace{1cm} (2.48)

where $P_{fa}$ is the probability of false alarm, \text{erfc}(.) is the complementary error function, and $\sigma_v^2$ is the variance of the test statistic during periods of non-speech activity in the $f_l$ frequency bin.

While the previous sections only discussed one emergent signal detector and two VAD algorithms, a variety of other techniques exist. As previously mentioned, these techniques vary greatly in the method used to detect emergent signals. A procedure is needed to measure the performance of these techniques, and provide a comparison between several techniques. This procedure should be algorithm independent so that the performance of many algorithms can be compared with the same metric. Typically, the performance of an emergent signal detection
algorithm is assessed using a receiver operating characteristic (ROC) curve. The following section discusses the ROC curve.

2.6 Performance Measures - Receiver Operating Characteristic Curves

For emergent signal detection and VAD algorithms, a frame by frame decision is made as to whether the signal (or voice) is present or not. ROC curves are used to compare methods and assess their performance. These operating characteristics are based on the percentages of properly and improperly classified frames. There are two cases of correct classification. The first is when the signal of interest is present, and the decision rule declares that it is present, and the second is when the signal of interest is absent and the decision rule declares that it is absent. The first type of correct classification is known as a detection, and the probably of it occurring is called the probability of detection ($P_D$).

There are also two cases of misclassification. In statistics, these are known as Type I and Type II errors. The Type I error, also known as the false negative, occurs in situations when the signal of interest is present, but the decision rule has declared it is absent. On the other hand, the Type II error, occurs when the signal of interest is absent, but the decision rule has declared it is present. The probability of a Type II error occurring is known as the probability of false alarm ($P_{FA}$) [109]. These probabilities and the choice of threshold are visualized in Figure 2.8. The pdf of the test statistic for the signal of interest present (called $P_0$ in Section 2.4.1) is shown in Figure 2.8(a), while the pdf of the test statistic for the signal of interest absent ($P_1$) is shown in Figure 2.8(b). A threshold, $\lambda$, is used to differentiate between the two hypotheses. The probability of detection is the area shaded blue in Figure 2.8(a), and the probability of false alarm is the area shaded green in Figure 2.8(b).
Figure 2.8: (a) The pdf for the significant statistic when the signal of interest is present. The $P_D$ is the area of the blue shaded region. (b) The pdf for the significant statistic when the signal of interest is absent. The $P_{FA}$ is the area of the green shaded region.

Figure 2.9 shows that the $P_D$ and $P_{FA}$ are dependent on the threshold chosen. As the threshold varies, so do the values for the probabilities. Typically, the performance of a particular hypothesis test is expressed in the form of a ROC curve. The threshold value is swept over a range of values, and the corresponding probability of detection and false alarm pairs are plotted. Figure 2.9 shows an example of several ROC curves. A perfect classifier would have a $P_D$ of one for all possible $P_{FA}$. In Figure 2.9 increasing performance is indicated by the arrows. By examining the ROC curves of two or more hypothesis tests, the performance is easily visualized and compared. The inset plots show the pdfs for the signal present (blue) and signal absent (green) that generated each ROC curve. As the overlap between the pdfs of the two hypotheses increase, the performance decreases.
In this chapter, the basics of array signal processing were introduced. This chapter focused on techniques used to extract and detect a signal of interest. Two approaches to signal extraction were discussed: spatial extraction using beamforming and statistical extraction using blind source separation. Chapter 3 explores a method that combines these techniques to suppress interferers in order to mitigate their individual weaknesses and exploit their strengths. This chapter also introduced emergent signal detection, and its specific application of voice activity detection. Chapter 4 investigates a method for emergent signal detection (and subsequently VAD) by using PCA. The ROC curve introduced will be used to assess the performance of PCA as an emergent signal detector.
3.1 Introduction

Interference suppression is an important and widely studied problem in acoustics. There are numerous techniques used to suppress undesired interfering signals and enhance signals of interest. One body of work uses spatial processing to focus an array of sensors on a specific bearing angle, while suppressing sidelobe contributions. In cases where the bearing angle to an interferer is known, a fixed null beamformer can be used to suppress contributions to the beamformer output from that specific direction. In most applications however, the locations of interfering sources are not known a priori, and the interference suppression algorithms mostly focus on adaptively placing nulls in the directions of interferers or minimizing sidelobe height.

Another large body of work uses time or frequency domain BSS to separate interferers from signals of interest. These techniques do not typically exploit spatial information about the signals. Where spatial information has been used is in attempts to solve the BSS permutation ambiguity problem [110-113]. This chapter presents a unique approach to interference suppression that combines methods from BSS and phased array processing. This method suppresses contributions to the spatial processor output from non-Gaussian interferers using a combination of beamforming and ICA techniques.

3.2 Proposed Algorithm

The FastICA algorithm, along with many other ICA algorithms, assumes instantaneous mixing and the linear mixing arrangement presented in Section 2.3.2. When multiple sources are spatially distributed, the signal mixing involves delays due to the time difference of arrival.
(TDOA) at each sensor. To successfully implement an ICA algorithm based on instantaneous mixing, the signals recorded from an array need to be altered before they can be separated. The spatial processing portion of the new algorithm aligns the signals in time, which allows application of FastICA (or other ICA techniques for linear/simultaneous mixing) because the mixing is no longer convolutive. Spatial knowledge is also used to resolve the permutation ambiguity problem inherent in the ICA algorithm.

The high-level operation of the new algorithm is as follows: beamforming is used to correct for time delays caused by the propagation of wavefronts arising from the spatial distribution of sources. The FastICA algorithm is applied to these delay-corrected signals. Individual signals that are retrieved from the ICA algorithm are then filtered from the mixed signals in all channels and the remaining signals are processed spatially for display. A block diagram of the proposed new algorithm is provided in Figure 3.1. An array containing $m$ sensors is used to collect the input signals for the algorithm. The $m$ signals are beamformed at $j$ angles, which are selected based on different criteria, as will be described in Section 3.2.2. These $j$ beamformed signals are then passed through the FastICA algorithm along with two unaltered array signals. The FastICA outputs are sorted to determine which signals will be removed. Finally, the selected signals are passed through a Wiener filter and removed from the original $m$ array signals. The remainder of this section discusses the details of each step of the new algorithm.
3.2.1 Strategy and Motivation

Assuming that there are $m$ sensors and $n$ sources, the goal of the algorithm is to remove $j$ sources from each array channel. In this way, signals such as loud interferers can be removed, leaving behind the $(n - j)$ signals of interest. The inputs to the algorithm are the measured microphone array signals, and the output is the set of signals of interest. The $m$ array signals are first beamformed, which is used to estimate the DOA of the $j$ signals. These angles are used to steer the array and extract signals from the $j$ directions of interest. These signals, along with two unaltered array sensor signals, are passed to the FastICA algorithm, which separates the signals as explained in Section 2.3.3.2. A Wiener filter is used to cancel the interfering signals from the mixture of signals in each channel.

One advantage of using this new method is that interferer rejection is not spatially based like in null beamformers such as MVDR. As an interferer moves closer to the signal of interest, beamformers cannot distinguish the signal of interest from the interferer because they are spatially indistinct. This is especially true for compact arrays (i.e. low $m$), where the main beam lobe is very wide. In the new method, the higher order statistics of the signal are used for separation in addition to the spatial location and interfering signals can still be suppressed in
spatially indistinct mixtures. While the algorithm’s performance is degraded in this region, interferer suppression is still possible.

3.2.2 Choosing Possible Source Locations

Choosing the locations of the interfering sources for this algorithm can be done either manually or automatically. For some situations, the angles of interest are input manually into the algorithm. For example, in a sonar application, an operator would manually choose a source to be removed in order to help him/her visualize what other sources are present in the environment. In other instances, the angles of the interferers are generated automatically as a processing step in the algorithm. Both steps require a method to scan the sources present in the environment.

This scanning is accomplished using a beamformer. In this dissertation, the beamforming is accomplished using the broadband beamformer described in Section 2.2.2. This method is based on the traditional narrowband DS beamformer. Other more sophisticated techniques could be used in this step, but this beamformer is used for convenience due to its ease of implementation and processing speed. The weights for this beamformer are data independent and can be calculated in advance. Using the broadband beamformer to scan the environment produces a measure of the power (over the frequencies of interest) at all angles between -90 and +90 degrees. Given this plot, an operator could manually choose which directions contain interfering sources, or an automatic peak picker could be used to select the interferer directions.

There are two ways that peak picking is implemented in this dissertation. In the first way, the user selects the number of sources to be removed ($j$). The peak picker selects the $j$ largest peaks in the beamformer scan. The angles at which these peaks occur correspond to the $j$ signals that will be removed. In this method, the user must guess the number of interfering sources that are present in the environment. The second method uses a threshold to select the peaks. Instead of selecting the number of source to be removed, the user inputs a threshold. The ratio between the magnitude of the power at each angle and the magnitude of the maximum power is calculated. Any peak with a ratio greater than the selected threshold is identified as a source to be removed. Once again, the angles at which these peaks occur correspond to the $j$ signals that will be removed. These techniques are used when all the peaks in the beamformer scan are assumed to be interferers. For the case where the signal of interest is visible in the beamformer scan
(higher signal to interferer ratio) and the goal is to remove all other sources in the environment, the operator must manually input which peak corresponds to the signal of interest.

### 3.2.3 Correcting for Delays

For simplicity of derivation, a linear array with \( m \) microphone elements is used. The \( n \) sound sources are all assumed to be in front of the array, with the source DOAs ranging from -90 to +90 degrees. Figure 3.2 provides the scenario geometry and a definition of the delays appearing in Equations 3.1 and 3.2. Each array sensor sees a mixture of delayed versions of each of the \( n \) sources according to the model

\[
x_i(t) = \sum_{k=1}^{n} A_{ki} s_k(t - \tau_{ki}) + n_i(t), \quad \tau_{ki} \equiv \frac{d_k}{c}, \quad \text{for } i=1,2,\ldots,m
\]

where \( x_i \) is the response of the \( i \)th sensor of the array, \( A_{ki} \) is the linear mixing component, \( s_k \) is the \( k \)th source, \( \tau_{ki} \) is the delay from the \( k \)th source to the \( i \)th sensor based on the distance between the two, and \( n_i \) is the noise associated with the \( i \)th sensor. The time delays caused by the sensor and source geometry must be removed before FastICA can be applied. Electronic steering of the array is used to compensate for the delays.
Figure 3.2: The geometry of the sensors and sources assumed for algorithm development is shown. The array is a linear microphone array and all sources are in front of the array. The time delays introduced in Equations 3.1 and 3.2 are due to the corresponding distances indicated with the red dashed and purple dotted line segments.

The array is beamformed at the $j$ DOAs corresponding to the $j$ signals to be removed. The ideal beamformer output is given by

$$b_l(t) = \sum_{i=1}^{m} x_i(t - \tau_{il}) = \sum_{i=1}^{m} \sum_{k=1}^{n} A_{ik} s_k \left(t - \tau_{ki} - \tau_{il}\right) + n_i \left(t - \tau_{il}\right), \text{ for } l = 1, 2, ..., j$$

$$\tau_{il} = \frac{d_c}{c} (i-1) \sin \theta_k,$$

where $b_l$ is the $l$th beamformed signal, $\tau_{il}$ is the delay from the delay-and-sum beamformer, and $\theta_k$ is the DOA of the $k$th source. When the array is steered towards the $k$th source, the spatial delay represented by $\tau_{ki}$ is exactly canceled by the beamformer delay given by $\tau_{il}$. The result of beamforming at the $l$th source is

$$x'_l(t) = m \left(\sum_{i=1}^{m} A_{il}\right) s_i(t) + \sum_{i=1}^{m} \sum_{k=1 \atop k \neq l}^{n} A_{ik} s_k \left(t - \tau_{ki} - \tau_{il}\right) + n_i \left(t - \tau_{il}\right),$$

where the $l$th signal is no longer delayed. Once corrected using beamforming, the group of $j$ signals are approximately linearly mixed and can be processed by the FastICA algorithm.
3.2.4 Applying the ICA Algorithm and Signal Selection

While beamforming focuses the array towards a specific steering angle, which eliminates the time delays of the signal located at that direction, it distorts signals located away from the steering angle. When the array is steered towards the kth source, the \((j-1)\) other sources become distorted. Since the delays are dependent on the steering angle, the distortion introduced into all non-look direction signals varies as a result of the changing look direction. If the FastICA algorithm is applied to a set of beamformed signals, the separation results are poor because the mutual information between the beamformed signals is low. The beamformed outputs, combined with the unaltered first and mth array signal form the set of mixed signals for the ICA algorithm. Although not a requirement of FastICA, the first and mth array signals are included to introduce additional unaltered information about the mixed signals into the FastICA algorithm to aid in the separation process.

To quantify the performance of the new algorithm, a simulation was done to compare unmixing performance under four conditions. The conditions examined are: linear mixing, convolutive mixing as described in Equation 3.1, corrected convolutive mixing where only the beamformed signals are included, and corrected convolutive mixing where the first and mth unaltered array signals are included. The signal of interest is located at +5 degrees, while the interferer is located at -10 degrees. A 35 channel array is used with an element spacing of 0.035 meters. For the convolutive mixing case, the first and mth array signal are passed directly to the FastICA algorithm. The two corrected convolutive mixing cases use the beamforming technique to correct the time delays as described above. Figure 3.3 shows the coherence between the original and unmixed signal for the four cases. While the linear mixing case has a coherence of nearly one for all frequencies, the other three cases show degradation of the signal after separation. However, the new algorithm shows much better performance for most of the frequencies considered. In addition, the inclusion of the first and mth array signals increases the separation performance across most of the frequencies.
The coherence between the unmixed signal and the separated signal is shown for four cases: linear mixing, convolutive mixing, corrected convolutive mixing with only beamformed signals, and corrected convolutive mixing where the first and mth unaltered array signals are included.

The mixed signal set generated by the beamforming process contains \((j + 2)\) signals. The FastICA algorithm will also produce \((j + 2)\) outputs, \(j\) of which will contain unmixed signals. The other two outputs are a residual mixture of all the signals and Gaussian noise. In addition, the FastICA algorithm has a permutation ambiguity in the outputs, so additional post processing is necessary to identify the separated sources. The unmixed signals must be classified as either interferers or residual mixtures. This task is accomplished using the coherence between the beamformed signals and the FastICA outputs. The interferer signals are chosen as the FastICA output channels with the highest coherence with the beamformer output at the DOA of the interferers. The selected signals are then passed onto the Wiener filter.

### 3.2.5 Implementing the Wiener Filter

The output of the FastICA algorithm provides a sample of the signal that will be removed. Since a sample of the “noise” signal is available, the Wiener filter can be used to remove the interfering signal. For this step, the frequency domain Wiener filter is implemented.
Let \( y(k) \) be a clean, zero mean signal of interest, which is contaminated by additive zero mean noise \( v(k) \) according to the model
\[
x(k) = y(k) + v(k)
\] (3.4)
where \( x(k) \) is the observed signal. For our problem, \( y(k) \) is the signal of interest, \( v(k) \) is the output signal of the FastICA algorithm that will be removed, and \( x(k) \) is the array data. The goal of the Wiener filter is to find an optimal estimate of \( y(k) \) [114]. In deriving the Wiener filter, it is assumed that a sample of the noise signal exists; the noise is uncorrelated with the signal of interest; and both the signal of interest and the noise are stationary [56]. Let \( h = [h_0, h_1, \ldots, h_{L-1}]^T \) be the FIR Wiener filter of length \( L \) that will estimate \( y(k) \) when applied to \( x(k) \). The error signal between the clean signal \( y(k) \) and its estimate is defined as
\[
e(k) = y(k) - h^T x(k). \] (3.5)

The performance criterion for the Wiener filter is the minimum mean-square error, which is written as
\[
J(h) = E\left\{ e^2(k) \right\} = h^T R_{xx} h - 2r_{xy} h + \sigma_y^2 \] (3.6)
where \( E\{.\} \) is the expectation. \( R_{xx} \) is the correlation matrix of the observed signal, \( r_{xy} \) is the cross-correlation vector between the signal of interest and the observed signal, and \( \sigma_y^2 \) is the variance of the signal of interest, given by
\[
R_{xx} = E\{x(k)x^T(k)\}, \quad r_{xy} = E\{x(k)y(k)\}, \quad \sigma_y^2 = E\{y^2(k)\}. \] (3.7)

Using Equation 3.6, the optimal Wiener filter [114] is obtained as
\[
h = \arg\min_h J(h) = R_{xx}^{-1} r_{xy}. \] (3.8)
This solution requires knowledge of \( y(k) \), which is unobservable in this case. An estimate of \( r_{xy} \) can be determined by solving for \( y(k) \) in Equation 3.4 and substituting into Equation 3.8. The result is
\[
r_{xy} = E\{x(k)y(k)\} = E\{x(k)(x(k) - v(k))\}
= E\{x(k)x(k) - x(k)v(k)\} = E\{x(k)x(k) - [y(k) + v(k)]v(k)\},
= r_{xx} - r_{vy} \] (3.9)
which depends on the correlation vectors for the observed signal and the noise signal. Both of these signals are observable.

Other, more sophisticated filters, such as the LMS filter could be substituted at this stage. The Wiener filter was chosen because it is simple to implement, requires no additional inputs, and always converges. The Wiener filter is a causal filter, so no information about the future is necessary. The noise reduction factor for the Wiener filter is always greater than one, so noise is never increased with the application of the Wiener filter, and with the optimal Wiener filter, the output SNR is always greater than or equal to the input SNR [56]. However, the cost of the reduction of the noise is a distortion of the signal of interest [115]. There will always be some distortion of the signal of interest, and for low SNRs, this distortion effect can be high [56]. Using this new approach, a technique can be implemented where, starting with the loudest, interferers can be cancelled from displays such as bearing time recorder (BTR) displays leaving behind successively more quiet sources. Examples of this technique are provided via both simulation and experiment in the following sections.

3.3 Simulations and Results

For the simulations presented in Sections 3.3.1 through 3.3.4, the data contains two voice signals: one signal of interest and one interferer signal. The sources are digitally recorded files of voices, which are artificially placed at various locations in the environment with respect to the array. The data in the simulation in Section 3.3.6 contains four voice signals: three signals of interest and one interferer, and the simulation in Section 3.3.5 contains one voice signal and one broadband interferer signal. In order to simulate the TDOA, the source signals are up-sampled to five times their sampling frequencies then shifted by the appropriate number of samples corresponding to the DOA delay. This method is not frequency dependent and creates minimal distortion in the signal.

All of the simulations assume an anechoic environment, where the speed of sound is 343 m/s. A linear array is used in all cases, where the element spacing is constant. The number of elements and the elemental spacing varies depending on the simulation. For each case, the sampling frequency is 11025 Hz, which is a standard sampling frequency used in wav audio
files. The frequency band of interest used for simulations is between 100 and 3500Hz, which was chosen based on the frequency content of the simulated signals. Unless stated otherwise, the two signals are at equal power, equidistance from the array, and there is no additional uncorrelated background noise.

### 3.3.1 One Signal of Interest with One Interferer

A simulation with two voice signals (one stationary, one moving) was done in order to examine the algorithm’s performance with respect to a variety of spatial situations. The interferer remains stationary at +5 degrees, while the signal of interest is moved from -20 degrees to +20 degrees at a rate of one degree per second. Both sources are assumed to be in the far-field and have equal power. The data is simulated for a 35 channel linear array with an inter-microphone spacing of 0.035 meters.

Figure 3.4 shows the BTR plot using the broadband DS beamformer from Section 2.2.2. Each horizontal slice in Figure 3.4 is the magnitude of the beamformer output calculated using one second of data. The two sources are clearly seen. The interferer appears as the vertical signature appearing at +5 degrees. The signal of interest is the signature that crosses diagonally from -20 to +20 degrees. In this case, an operator would have to specify which signal is the interferer. Using a peak picking algorithm, both the signal of interest and the interferer would be identified; the peak picking algorithm is based only on beamformer power and has no way to distinguish between the signal of interest and interferer in this instance.
Figure 3.4: BTR plot using a broadband beamformer for two signals having equal power. One signal is located at +5 degrees, while the other signal moves from -20 to +20 degrees at a rate of one degree per second.

3.3.1.1 Minimum Variance Distortionless Response Processing

MVDR beamforming is used in this dissertation to provide a comparison for the new algorithm with spatial null forming techniques. The result of MVDR processing can be seen in the BTR plot of Figure 3.5. Over the entire duration of the simulation, the MVDR beamformer has placed a null in the direction of the interferer at +5 degrees. While the MVDR algorithm has good performance with regard to the interferer suppression at +5 degrees, as the signal of interest passes near the interferer, both signals are suppressed by the null at +5 degrees. This is one of the shortcomings of null based interference suppression, such as MVDR. The beampattern has numerous sidelobes that distort the bearing track of the signal of interest.
Figure 3.5: Bearing track recorder plot using MVDR beamforming. The undesired, stationary source at +5 degrees has been nulled in all time steps, but numerous sidelobes in the beampattern obscure the bearing track recorder display.

3.3.1.2 Processing with the New Algorithm

Figure 3.6 shows the results of applying the new algorithm to the simulated data. The interferer at +5 degrees is suppressed in all time steps. Whereas before, the signal and interferer had equal power in the BTR, now the interferer’s power is around 20dB below the signal of interest’s power in all time steps. The signal of interest is clearly visible in all the time steps, even when the two signals are co-located. Using a peak picking process at this point would result in the selection of the signal of interest for all time steps.
3.3.2 Effect of Angular Separation of Sources

The beamformer output shows the reduction of power from the interferer direction with the application of the new algorithm. In Figure 3.7, the algorithm’s effect on coherence between the original signal of interest and the output of the algorithm is shown. For each second of data, instead of plotting the beamformer output as in Figure 3.6, the coherence is plotted. The previous section demonstrated the algorithm’s ability to suppress the interferer in a spatial display, and Figure 3.7 shows a measure of the improvement in clarity of the signal of interest. The frequencies of interest in this simulation are the same as the frequencies chosen for beamforming (100 to 3500Hz). When the angular separation is greater than three degrees, the coherence across those frequencies of interest is greater for the algorithm output than for the mixed array data. This is especially true for the frequencies between 1000 and 3500Hz, which includes the majority of frequency content of the voice signal of interest in this case. For angular separation less than three degrees, the coherence increases with application of the new algorithm, but not as significant as for greater angular spacing. In these cases, the sources are not spatially distinct, and the beamforming stage does not contribute new information to the algorithm. In these cases,
the mixing matrix tends to be ill conditioned and the FastICA stage cannot completely separate the sources. The exception is at zero degrees of separation because at this point, there is no time difference of arrival, and the signals are linearly mixed. In this instance, the original ICA model is applicable, and the coherence increases to nearly one across most of the frequencies.

Figure 3.7: The algorithm’s interference suppression performance as a function of the angular spacing between the interferer and the signal of interest is examined. (a) The coherence between the algorithm input data and the signal of interest. (b) The coherence between the algorithm output and the signal of interest.

3.3.3 Varying Signal to Interferer Ratio

The results presented previously are for the case where the signal of interest and the interferer have equal power. Next, in order to test the robustness of the new algorithm in terms of the signal to interferer ratio (SIR), cases were generated where the signal of interest and interferer powers are not equal. The SIR is given by

\[
SIR \triangleq 10 \log_{10} \frac{P_{signal}}{P_{int}}
\]  

where \( P_{signal} \) and \( P_{int} \) are the power of the signal of interest and interferer respectively. For all cases, the signal of interest was positioned at +5 degrees, while the interferer was at -10 degrees.
Again, a 35 channel array with element spacing of 0.035 meters is used. Figure 3.8 illustrates the impact of SIR on the new algorithm’s performance. Figure 3.8(a) shows the beampattern of the mixed data before processing. As SIR decreases, the power of the interfering signal at -10 degrees increases while the power of the signal of interest at +5 degrees remains the same. Around -15dB SIR, the beampattern becomes dominated by the interferer, and the signal of interest is no longer visible. The ridge that appears at +3 degrees beyond -15dB is a side-lobe of the interferer, not the signal of interest. Figure 3.8(b) shows the beampattern of the output of the algorithm. For all values of SIR greater than -15dB, the signal of interest is clearly visible at +5 degrees while the interferer at -10 degrees is suppressed. When the SIR values falls below -15dB, the signal of interest is completely masked by the interferer. However, even beyond -15dB, the interfering signal sees nearly 20dB of suppression as a result of the algorithm’s application.

Figure 3.8: The effect of SIR on the performance of the new algorithm. (a) The beampattern of the algorithm input using a broadband DS beamformer. This signal of interest is positioned at +5 degrees and the interferer is at -10 degrees. (b) The beampattern of the algorithm output using a broadband DS beamformer. The interferer has been suppressed by 20dB in all cases, and the signal of interest remains visible until SIR drops below -15dB.
3.3.4 Varying Signal to Noise Ratio

All of the cases previously presented contain no additional noise. To further test the performance of the new algorithm, cases were generated where the signal of interest and interferer are at equal power and additional, uncorrelated, white, Gaussian noise is added to the system. The SNR is given by

\[ \text{SNR} = 10 \log_{10} \frac{P_{\text{signal}}}{P_{\text{noise}}} \]  \hspace{1cm} (3.11)

where \(P_{\text{noise}}\) is the power of the uncorrelated background noise. Since the signal of interest and the interferer have the same power, the SNR could also be calculated using \(P_{\text{int}}\) instead of \(P_{\text{signal}}\) in Equation 3.11. For all cases, the signal of interest was positioned at +5 degrees, while the interferer was at -10 degrees. Figure 3.9 shows a plot of the effect of SNR on the new algorithm’s performance. Figure 3.9(a) shows the beampattern of the mixed data, while Figure 3.9(b) shows the results of applying the new algorithm.

Figure 3.9: The effect of SNR on the algorithm. (a) The beampattern of the algorithm input using a broadband DS beamformer. This signal of interest is positioned at +5 degrees and the interferer is at -10 degrees. (b) The beampattern of the algorithm output using a broadband DS beamformer. The interferer is suppressed in all cases and the signal of interest remains visible until SNR drops below -18dB.
For source localization and tracking, peak picking on the beamformer output is used to identify sources. Any peak that rises above a selected threshold on the beamformer output is considered a source. This threshold value is dependent on the data processed, especially on the background noise. For Figure 3.9(a), two distinct signal peaks are apparent in the beamformer output when the SNR value is above -18dB. Use of a peak picking algorithm would identify two sources in the environment up to -18dB, when it is unlikely that either source would be identified as a peak. Figure 3.9(b) shows that even in the presence of noise, the interferer at -10 degrees sees suppression after the application of the new algorithm. In addition, the signal of interest remains a distinct peak for all values of SNR greater than -18dB. When the SNR value falls below -18dB, the signal of interest is not distinguishable from the background noise, but there is still some suppression at -10 degrees. The breakdown of the algorithm beyond a SNR of -18dB is comparable to the performance of other spatial techniques.

3.3.5 Used as a Pre-Processor for Voice Activity Detection

A VAD algorithm was used to further quantify the performance of the new algorithm and its ability to separate signals. For this simulation, the signal of interest is a voice signal at +5 degrees, while the interferer is a white Gaussian noise source at -10 degrees. Three different SIR ratios were tested: 5dB, 0dB, and -6dB. A 35 channel array with element spacing of 0.035 meters is used. The VAD algorithm used in this simulation was based on higher-order statistics, and was discussed in Section 2.4.1. The VAD algorithm was applied to the data before and after it was passed through the proposed algorithm. In addition, to benchmark performance, the VAD algorithm was applied after MVDR processing and using the FastICA algorithm alone. Figure 3.10 shows the results of the application of VAD to the data. Each red box encloses a time segment that the VAD identified as an active voice frame. Column (a) shows the frames identified as active voice frames in the original simulated data for the three SIR cases. Column (b) shows the active voice frames after the data was processed using MVDR, column (c) shows the results when using FastICA alone, while column (d) shows the active voice frames after the data has been passed through the separation algorithm. As the SIR drops, the VAD algorithm is unable to correctly identify the active speech sections of the signal in the unprocessed data. Application of the MVDR algorithm increases the VAD algorithm’s ability to identify the active
voice frames, especially in the higher SIR cases. However, for the lower SIR cases, some of the active voice frames are not identified. Using FastICA alone encounters a similar problem due to the incomplete separation of the noise from the voice signal. After application of the new separation algorithm, the VAD algorithm is able to identify all the active voice frames. Even in the lowest SIR case, it is able to identify the active voice frames.

Figure 3.10: Voice activity detection on data containing one voice signal of interest and one white Gaussian interferer. The three rows show the results at varying SNR values. The active voice frames are enclosed with red boxes. Column (a) shows the VAD results before the application of the new algorithm, column (b) shows the VAD results after MVDR, column (c) shows the VAD results after FastICA, and column (d) shows the VAD results after the new algorithm application.

3.3.6 One Loud Interferer with Three Quieter Signals of Interest

The previous simulations have all examined the case of one interferer with one signal of interest. In most of the simulations, the signal of interest was visible in the BTR plot, and a choice had to be made to determine which of the signatures to remove. Another scenario where
this new algorithm has applications is in situations where one loud interferer is masking hidden signals of interest. For this simulation, there are three signals of interest that are each at -10dB SIR with the single interferer. Figure 3.11(a) shows the BTR for this mixed data over a 50 second time period. In this case, only the interferer is visible in the BTR plot. Unlike before, the objective is to remove the visible signal from the BTR to reveal the hidden paths of the signals of interest. In this case, peak picking could be employed to identify the DOA of the interfering signal. Figure 3.11(b) shows the BTR after the application of the algorithm to the mixed data. The loud interferer, which was the only signal visible in Figure 3.11(a), has been completely removed, revealing the three signals of interest.

![BTR plot using a broadband beamformer for one loud interferer and three signals of interest. The signals of interest all are at -10 dB SIR with the interferer. (a) The BTR of the mixed data before application of the new algorithm. Only the interferer is visible. (b) The BTR of the data after it has been processed using the new algorithm. The loud interferer has been completely removed, revealing the three signals of interest previously hidden.](image)

Figures 3.11(a) and 3.11(b) show how the new algorithm can be used to remove a single interferer to reveal masked signals of interest. For this simulation, there is one loud interferer, one signal of interest at -6dB SIR, and two signals of interest at -12dB SIR. Figure 3.12(a) shows the BTR of the mixed data before processing. It is similar to the previous simulation where only one signal is present. The data is processed using the new algorithm, just as before, and the BTR of the processed data is shown in Figure 3.12(b). The loud interferer that was present in Figure
3.12(a) has been removed, revealing one loud signal of interest and two fainter, possible signals of interest. The signal moving around +50 degrees is partially masked by the louder signal of interest at +15 degrees. The processed data is passed through the new algorithm a second time, this time assuming the signal at +15 degrees is the new “interferer.” Figure 3.12(c) shows the results after the second round of processing. The paths of the final two signals of interest are much clearer and can be selected by using a peak picking algorithm. For this case, the algorithm has been used multiple times to reveal signals previously masked in a BTR display.

![Figure 3.12: BTR plot using a broadband beamformer for one loud interferer and three signals of interest. (a) The BTR of the mixed data before application of the new algorithm. Only the interferer is visible. (b) The BTR of the data after it has been processed the first time using the new algorithm. (c) The BTR of the data after it has been processed twice.](image)

### 3.4 Experiments and Results

Two of the simulations presented in Section 3.3 are repeated in experiments to prove the algorithm’s performance in an anechoic environment. The case with one interferer and one signal of interest at equal power is repeated. In addition, case with varying SIR is repeated. A 24 element linear array was used to collect the data. The array had an inter-elemental spacing of 0.75 inches and was sampled at 50 kHz, which is dependent on the data acquisition equipment. The frequency band of interest is the same as used for the simulations (100 to 3500Hz), which was chosen based on the frequency content of the simulated signals.
3.4.1 One Signal of Interest with One Interferer

This section discusses the experiments performed with one signal of interest and one interferer at equal power. The signals were two voices. The signal of interest is stationary and located at zero degrees, while the interferer is moved from -20 to +20 degrees. This geometry replicates the scenario in Section 3.3.1, but the signal of interest and interferer are in opposite locations. Both sources were located 15 feet away from the center of the array at all time steps. This experiment was performed in an anechoic chamber. Figure 3.13 shows the BTR of the array data collected during in-chamber testing. The signal of interest at zero degrees is not visible in most instances, while the interferer is always easily observable. Using a peak picker at this stage would identify the interferer as the only source in the environment.

![Figure 3.13: BTR using a broadband DS beamformer for the anechoic chamber testing. One signal is located at zero degrees, while the other signal moves from -20 to +20 degrees at a rate of one degree per second.](image)

The new algorithm was used to process the experimental data. To illustrate the performance of the algorithm in removing a slowly moving interferer, the crossing signal was chosen to be the interferer (opposite to the simulation case). The result of the application of the new algorithm is shown in Figure 3.14. In almost every time step, the signal of interest is now
visible while the interferer is significantly suppressed when compared to Figure 3.13. This supports the results from the simulation.

![Figure 3.14](image)

**Figure 3.14**: BTR using a broadband DS beamformer after the anechoic chamber data has been processed using the new algorithm. The moving interferer has been removed, and the signal of interest at 0 degrees is visible.

### 3.4.2 Varying Signal to Interferer Ratio

The results presented previously are for the case where the signal of interest and the interferer have equal power. In order to test the robustness of the new algorithm in an anechoic environment, tests were performed where the signal of interest and interferer powers are not equal. The SIR was varied between 0.84 and -11.13dB. For all cases, the signal of interest was positioned at -11 degrees, while the interferer was at +19 degrees. Again, a 24 channel array with element spacing of 0.75 inches was used. Figure 3.15 illustrates the impact of SIR on the performance of the new algorithm. Figure 3.15(a) shows the beampattern of the mixed data before processing. As SIR decreases, the power of the interfering signal at +19 degrees increases while the power of the signal of interest at -11 degrees remains the same. Figure 3.15(b) shows the beampattern of the output of the algorithm. For all values of SIR, the interferer at +19
degrees is suppressed. In addition, the signal of interest at -11 degrees is not suppressed for any of the SIR cases tested. This confirms the result of the simulations for an anechoic environment.

![Figure 3.15](image)

**Figure 3.15:** The effect of SIR on the performance of the new algorithm using data recorded in an anechoic chamber. (a) The beampattern of the algorithm input using a broadband beamformer. The signal of interest is positioned at -11 degrees and the interferer is at +19 degrees. (b) The beampattern of the algorithm output using a broadband beamformer.

### 3.5 Conclusions

This chapter introduced a new method for interferer suppression based on a combination of spatial and ICA techniques. The new method has the capability to suppress non Gaussian interferers, ideal for telecommunication and speech processing applications. The new algorithm allows for the separation of signals that are not simultaneously mixed. In addition, the introduction of spatial processing allows the permutation ambiguity of the output of the FastICA algorithm to be solved. Using the location of a source, that source can be associated with an output of the FastICA algorithm using coherence.

Simulations showed the capabilities of the new algorithm in suppressing interferers in a variety of conditions. During the simulations, the interferer was suppressed while the signal of
interest saw little suppression for 0dB SIR. Even when the signals become spatially indistinct, the algorithm was still able to partially suppress the interferer. This is not possible using spatial nulling techniques, which remove all signals from a specific location. Anechoic experimental testing confirmed the results obtained in the simulations. Although the signal of interest was not visible in many of the time steps of the original BTR, the new algorithm was able to increase its visibility by removing the interferer. In simulations, the algorithm showed 20dB of suppression of broadband interferers, even when the SIR dropped to -30dB. In addition, the algorithm was shown to have good performance for SNR values above -18dB, which is comparable to other spatial techniques. The performance of the new algorithm as a front-end to a VAD processor was shown to be superior to MVDR in the cases studied; however, future work will include a statistical performance on this enhancement using many types of voice signals and many more geometries.

The following chapter presents a second method for increasing signal clarity; emergent signal detection. The same properties of ICA that were used to separate sources are used to identify emergent signals. ICA was used in this chapter to separate source because the independence assumption is stronger than the uncorrelated assumption used in PCA. As discussed in Section 2.3.2 using PCA does not always lead to total source separation. While PCA may not always lead to complete source separation, it is capable of locating directions corresponding to uncorrelated sources. Since it is a non-adaptive algorithm (and a closed form solution can be written), PCA instead of ICA will be used in the next chapter to detect emergent signals.
Chapter 4:  

Emergent Signal Detection

4.1 Introduction

The previous chapter focused on improving signal clarity by suppressing undesired interferers. Another important aspect of enhancing signal clarity is to provide the ability to detect an emergent signal of interest in a noisy or interference heavy environment. The passive sonar problem provides a good example here. The combined spatial and BSS processing in the previous chapter is used to eliminate the interferers, such as the submarine’s engine noise or surface ship noise. However, a method is still needed to detect if a signal of interest is present in the environment. For example, the goal could be to identify if another submarine’s engine signature is present as it enters a surveillance zone. This chapter develops an array processing method to identify the emergence of new signals in the environment.

There are three classes of emergent signal detection: detecting a known signal in noise, detecting a signal with unknown parameters in noise, and detecting random signals in noise [116]. The first case often applies to pattern recognition, and will not be examined in this dissertation. For both the second and third cases, where the exact form of the signal of interest is unknown, many signal detection algorithms use a statistical approach to detect the signal of interest. For the noise only case, a particular pdf is assumed, and the mean and variance for that case are estimated. The presence of an emergent signal causes a detectable shift in the mean or change in the variance of the pdf. When the statistics of the received signal do no match the assumed pdf of the noise only case, the emergent signal is declared to be present.

Existing detection algorithms use a sample signal recorded from a single sensor. This chapter presents a unique approach to emergent signal detection based on the signals recorded from an array of at least two sensors. While previous signal detection methods have required the knowledge of the pdfs of both the noise and the signal of interest, the new method does not require specific knowledge of their individual distributions. The mixed signals are all assumed to
be zero mean, and the new technique relies on detecting an increase in the variance of the mixed signals when the signal of interest is present.

4.2 Emergent Signal Detection using Principal Component Analysis

ICA theory states that the mixture of two or more sub- or super-Gaussian signals is more Gaussian than either of the individual unmixed source signals [86]. This property is used in the projection pursuit algorithm to separate mixed signals by finding the projection direction that is least Gaussian, as mentioned in Chapter 2. If the two signals are Gaussian, the mixed signal will have greater variance than either of the individual signals. This chapter exploits this property to develop an emergent signal detection technique. If the interferer only case has a certain distribution, the addition of the signal of interest to that interferer will cause an increase in the variance of that distribution. The advantage here is that the individual distributions do not need to be known in advance, and the signal and noise could have identical pdfs without affecting the method. If one sensor of data was available, the detection would be based on a change in the variance. Since this method involves two sensors of data, the detection is based on a change in the variance of the joint pdf.

Figure 4.1 shows a plot of the amplitude of the first sensor versus the amplitude of the second sensor. This is one way to visualize the joint pdf of the two sensors’ data. Figure 4.1(a) represents the case where only the interferer is present, while Figure 4.1(b) represents the case where both the signal of interest and the interferer are present. Section 2.3.1 explained how PCA finds directions of maximum variance and can be used to separate uncorrelated sources. The red lines in Figure 4.1 lie along the directions of the PCs. In Figure 4.1(a), one direction (and therefore one PC) contains most of the variance. This direction corresponds to the interferer, and since the interferer is the only signal present, there is little variance in any other direction. In Figure 4.1(b), the directions of the PCs have changed, and the variance along both PC directions has increased. This increase is especially pronounced along the second PC, which is denoted by the label ‘PC 2’ in Figure 4.1. This increase indicates the presence of one additional signal, and therefore, PCA can be used to detect the presence of the additional signal.
Figure 4.1: A visualization of the joint pdf for two cases is presented: only one signal present and both signals present. A PCA analysis is performed on the data, and the directions of the PCs are plotted in red. (a) The joint pdf for the case where only the interferer is present. (b) The joint pdf for the case where both the interferer and the signal of interest are present.

Chapter 2 discussed how PCA is used to decorrelate mixed signals by finding the direction of minimum correlation. The theory in Chapter 2 provides another explanation of why PCA can provide an effective tool for emergent signal detection. When only the interferer is present, there is a certain level of correlation between the two measured sensor signals. By adding the signal of interest to the interferer, additional correlation exists between the two sensor signals; the correlation between the signal of interest in each sensor, as well as between the interferer in each sensor. The addition of the signal of interest causes an increase in the level of correlation between the two sensors, both in the same direction as before and along an orthogonal direction. This causes an increase in the both PCs. For this dissertation, only the smallest PC will be considered. The smallest PC is chosen because of a unique attribute that occurs in the simultaneous mixing case discussed below.

4.2.1 Specific Case: Voice Activity Detection

A specific case of emergent signal detection is VAD; therefore, the PCA method discussed in this chapter can be applied to the specific problem of speech detection. As discussed
in Section 2.5, most VAD algorithms exploit some characteristic of speech (such as spectral content or higher order statistics) to determine if speech is present within a received signal. This PCA technique does not rely on knowing any specific characteristics of the signal of interest. Therefore, there is nothing inherent in the method that lends it to VAD specifically. The model developed in the following sections only allows for the presence of one signal of interest and one interferer. For the sections dedicated to VAD, it is assumed that the signal of interest is a speech signal. Any time an emergent signal is detected, it is assumed to be a voice detection. For the specific case of one signal of interest and one interferer, the PCA technique for emergent signal detection can be used as a VAD and can robustly determine if a voice signal is present.

4.3 Emergent Signal Detection for One Signal and One Interferer

The assumption underlying this technique is that there exists only one interferer and one signal of interest. This is an ideal case since there is no uncorrelated background noise included. The smallest PC is derived for the model where the signal of interest and the interferer are mixed additively. The interferer for all models is assumed to be a stationary signal. In the simulations, the interferer is a white, Gaussian noise source. In order to accommodate a variety of signal of interest types, the only assumption made about the signal of interest is that it has zero mean. Therefore, the mixed signals, for both the signal of interest present and absent case, are assumed to have zero mean.

There are two forms of mixing introduced in this section. The first is simultaneous mixing. It assumes there is no time delay between the signal received at the first microphone and at the second microphone. This is the instantaneous mixing case that is used in many of the ICA models. The second case reflects the more realistic case, and allows for a TDOA between the two sensors. While adding another sensor increases the amount of information available, it comes with the drawback that the signals are not time aligned between the two sensors. Following sections will show how the performance changes due to this time misalignment.
4.3.1 Simultaneous Mixing Model

In this section, a simultaneous mixing model is presented. Let \( s_1 \) be the signal of interest with variance \( \sigma_{s1} \), and \( s_2 \) be an interfering signal with variance \( \sigma_{s2} \). The goal is to detect the presence of the signal of interest \( s_1 \). \( H_0 \) represents the case where the signal of interest, \( s_1 \), and the interferer, \( s_2 \), are both present and \( H_1 \) represents the case where only the interferer is present. For the simultaneous mixing model, the two hypotheses are represented by the models

\[
H_1 : \quad x_1(k) = a s_1(k) + b s_2(k), \quad x_2(k) = c s_1(k) + d s_2(k)
\]

\[
H_0 : \quad x_1(k) = b s_2(k), \quad x_2(k) = d s_2(k)
\]

where \( k \) is the sample number, and \( a, b, c, \) and \( d \) are the linear mixing coefficients. Following the steps outlined in Section 2.3.1, the smallest eigenvalue of the covariance matrix is determined for each hypothesis. In each case, the covariance matrix elements are calculated using Equation 2.17. For brevity, only the derivation for the (1,2) element will be shown. The full derivation for this, and all subsequent models, can be found in Appendix A. The (1,2) element of the covariance matrix for the \( H_1 \) hypothesis is calculated as

\[
c_{x}(1,2) = \frac{1}{n} \sum_{k=1}^{n} x_1 x_2 = \frac{1}{n} \sum_{k=1}^{n} (a s_1 + b s_2)(c s_1 + d s_2)
\]

\[
= \frac{1}{n} \sum_{k=1}^{n} (a c s_1^2(k) + (a d + b c)s_1(k)s_2(k) + b d s_2^2(k))
\]

Using the properties of the summation, Equation 4.2 can be rewritten as

\[
c_{x}(1,2) = ac \left[ \frac{1}{n} \sum_{k=1}^{n} s_1^2(k) \right] + (a d + b c) \left[ \frac{1}{n} \sum_{k=1}^{n} s_1(k)s_2(k) \right] + b d \left[ \frac{1}{n} \sum_{k=1}^{n} s_2^2(k) \right]
\]

\[
= ac \sigma_{s1} + (a d + b c) \sigma(s_1s_2) + b d \sigma_{s2}
\]

where \( \sigma(s_1s_2) \) is the covariance between the signal of interest and the interferer. For the \( H_1 \) hypothesis, the covariance matrix is

\[
C_x = \begin{bmatrix}
    a^2 \sigma_{s1} + 2 a b \sigma(s_1s_2) + b^2 \sigma_{s2} & ac \sigma_{s1} + (a d + b c) \sigma(s_1s_2) + b d \sigma_{s2} \\
    ac \sigma_{s1} + (a d + b c) \sigma(s_1s_2) + b d \sigma_{s2} & c^2 \sigma_{s1} + 2 c d \sigma(s_1s_2) + d^2 \sigma_{s2}
\end{bmatrix}
\]

Substituting the values from Equation 4.4 into Equation 2.18 gives the smallest eigenvalue of the covariance matrix for the \( H_1 \) hypothesis as
\[ \lambda_i = 0.5[(a^2 + c^2)\sigma_{s_1} + 2(ab + cd)\sigma(s_1s_2) + (b^2 + d^2)\sigma_{s_2}] \]

\[ -0.5 \sqrt{[(a^2 - c^2)\sigma_{s_1} + 2(ab - cd)\sigma(s_1s_2) + (b^2 - d^2)\sigma_{s_2}]^2 + 4[ac\sigma_{s_1} + (ad + bc)\sigma(s_1s_2) + bd\sigma_{s_2}]^2} . \]  

Equations 4.5 and 4.6 show that the smallest eigenvalue can be used to detect the presence of the signal of interest when it is mixed with an interferer. When the signal of interest is present, the smallest eigenvalue is a function of the variances of the signal of interest and the interferer, the covariance between the two signals, and the elements of the linear mixing matrix. However, when the signal of interest is not present, the eigenvalue becomes zero. This theoretical result shows, interestingly, that for the simultaneous mixing case with one interferer, the signal of interest can be detected for any SIR because the elements of the linear mixing matrix do not influence the result for the H_0 hypothesis.

### 4.3.2 Mixing Model Including TDOA

The previous section derived the smallest eigenvalue in the case of simultaneous mixing. This model, while widely used for ICA derivations, is not suitable for most recorded array signals because time delays between array elements are introduced by the source/receiver geometry [117]. This section applies the new PCA technique to a model that includes these delays. Similar to the previous section, let \( s_1 \) be the signal of interest with variance \( \sigma_{s_1} \), and \( s_2 \) be an interfering signal with variance \( \sigma_{s_2} \). The two hypotheses are

\[ H_1 : x_1(k) = as_1(k) + bs_2(k), \quad x_2(k) = cs_1(k + \tau_1) + ds_2(k + \tau_2) \]

\[ H_0 : x_1(k) = bs_2(k), \quad x_2(k) = ds_2(k + \tau_2) \]  

where \( k, a, b, c, \) and \( d \) are the same as in the previous section, and \( \tau_1 \) and \( \tau_2 \) are the time delays for the signal of interest and interferer respectively. As before, the covariance matrix is constructed element by element, and the (1,2) element calculation for the \( H_1 \) hypothesis is given next as an example. The (1,2) element is calculated as
\[ c_x(1,2) = ac \left[ \frac{1}{n} \sum_{k=1}^{n} s_1(k) s_1(k + \tau_1) \right] + ad \left[ \frac{1}{n} \sum_{k=1}^{n} s_1(k) s_2(k + \tau_2) \right] + \\
bc \left[ \frac{1}{n} \sum_{k=1}^{n} s_1(k + \tau_1) s_2(k) \right] + bd \left[ \frac{1}{n} \sum_{k=1}^{n} s_2(k) s_2(k + \tau_2) \right] \]  

(4.8).

Equation 4.8 introduces the need for several new covariance terms. These covariance terms arise due to delays between the signals. A superscript \( \tau \) denotes a delayed signal. Equation 4.8 is rewritten as

\[ c_x(1,2) = ac \sigma(s_1,s_1^\tau) + ad \sigma(s_1,s_2^\tau) + bc \sigma(s_1^\tau,s_2) + bd \sigma(s_2,s_2^\tau) \]

(4.9)

where \( \sigma(.) \) is the covariance between the signals in parentheses. For the simulations and experiments in this dissertation, the interfering signal, \( s_2 \), is assumed to be white, Gaussian noise. Other interferers could be used, but white, Gaussian noise is chosen because it is short-term stationary. Over small intervals, the statistics do not change, and the variance of the delayed signal, \( s_2(k+\tau_2) \), can assumed to be equal to the variance of signal \( s_2 \). The full covariance matrix is given by

\[
C_x = \begin{bmatrix}
  a^2 \sigma_{11} + 2ab \sigma(s_1,s_2) + b^2 \sigma_{22} & ac \sigma(s_1,s_1^\tau) + ad \sigma(s_1,s_2^\tau) + bc \sigma(s_1^\tau,s_2) + bd \sigma(s_2,s_2^\tau) \\
  ac \sigma(s_1,s_1^\tau) + ad \sigma(s_1,s_2^\tau) + bc \sigma(s_1^\tau,s_2) + bd \sigma(s_2,s_2^\tau) & a^2 \sigma_{22} + 2ab \sigma(s_1,s_2) + b^2 \sigma_{11} + (b^2 + d^2) \sigma_{22}
\end{bmatrix}.
\]

(4.10)

The smallest eigenvalue for the \( H_1 \) hypothesis is

\[
\lambda_1 = 0.5 \left( a^2 \sigma_{11} + c^2 \sigma(s_1,s_1^\tau) + 2ab \sigma(s_1,s_2) + 2cd \sigma(s_1^\tau,s_2) + (b^2 + d^2) \sigma_{22} \right) \\
-0.5 \sqrt{\left( a^2 \sigma_{11} - c^2 \sigma(s_1,s_1^\tau) + 2ab \sigma(s_1,s_2) - 2cd \sigma(s_1^\tau,s_2) + (b^2 + d^2) \sigma_{22} \right)^2} + 4[ac \sigma(s_1,s_1^\tau) + ad \sigma(s_1,s_2^\tau) + bc \sigma(s_1^\tau,s_2) + bd \sigma(s_2,s_2^\tau)]^2.
\]

(4.11)

The smallest eigenvalue for the \( H_0 \) hypothesis is

\[
\lambda_1 = 0.5 (b^2 + d^2) \sigma_{22} - 0.5 \sqrt{(b^2 - d^2)^2 \sigma_{22}^2 + 4b^2d^2 \sigma(s_2,s_2^\tau)^2}.
\]

(4.12)

Unlike in the previous section, the eigenvalue for the \( H_0 \) hypothesis does not reduce to zero in the absence of the signal of interest. The covariance terms introduced by the delays in the model do not cancel with each other. However, switching from the \( H_0 \) to the \( H_1 \) hypothesis still causes an increase in the smallest eigenvalue, which allows for the detection of the signal of interest’s presence, but complicates threshold calculation.
4.4 One Signal, One Interferer with Uncorrelated Background Noise

The model in this section builds on the previous section by using the same delayed signal scheme, but adding uncorrelated noise to the two mixed signals. As before, $s_1$ is the signal of interest with variance $\sigma_{s_1}$, and $s_2$ is the interfering signal with variance $\sigma_{s_2}$. The two hypotheses are written as

$$H_1: \begin{align*}
x_1(k) &= as_1(k) + bs_2(k) + n_1(k) \\
x_2(k) &= cs_1(k + \tau) + ds_2(k + \tau) + n_2(k)
\end{align*}.$$ \hspace{1cm} (4.13)

$$H_0: \begin{align*}
x_1(k) &= bs_2(k) + n_1(k) \\
x_2(k) &= ds_2(k + \tau) + n_2(k)
\end{align*}$$

In this case, the noise signals, $n_1$ and $n_2$, represent a combination of diffuse background noise, sensor noise, and electrical noise, and are uncorrelated between sensors. The covariance matrix is constructed element by element as before. As before, the interferer signal, $s_2$, is assumed to be short-term stationary, and the variance of the delayed signal, $s_2(k+\tau_2)$, is assumed to be equal to the variance of signal $s_2$. For brevity, the covariance matrix has been omitted, and the smallest eigenvalue for the $H_1$ hypothesis is

$$\lambda_1 = 0.5 \left[ a^2 \sigma_{s_1} + c^2 \sigma(s_1^* s_1^*) + (b^2 + d^2) \sigma_{s_2} + \sigma_{n_1} + \sigma_{n_2} + 2ab \sigma(s_1 s_2) + 2cd \sigma(s_1^* s_2^*) \right]$$

$$- \frac{0.5}{\left[ a^2 \sigma_{s_1} - c^2 \sigma(s_1^* s_1^*) + (b^2 - d^2) \sigma_{s_2} + \sigma_{n_1} - \sigma_{n_2} + 2ab \sigma(s_1 s_2) - 2cd \sigma(s_1^* s_2^*) \right]^2} \left[ a^2 \sigma_{s_1} - c^2 \sigma(s_1^* s_1^*) + (b^2 - d^2) \sigma_{s_2} + \sigma_{n_1} - \sigma_{n_2} + 2ab \sigma(s_1 s_2) - 2cd \sigma(s_1^* s_2^*) \right]$$

$$+ 4 \left[ abc(s_1 s_2) + abd(s_1 s_2^*) + cde(s_1 s_2 s_2^*) + abd(s_1 s_2 s_2^*) + cde(s_1 s_2 s_2^*) \right]$$

$$+ b \sigma(s_2 n_2) + c \sigma(s_1 n_2) + d \sigma(s_2^* n_2) + \sigma(n_2)$$

$$\lambda_1 = 0.5 [b^2 + d^2] \sigma_{s_2} + \sigma_{n_1} + \sigma_{n_2} + 2bd \sigma(s_2 n_1) + 2d \sigma(s_2^* n_2)]$$

$$- 0.5 \sqrt{[b^2 + d^2] \sigma_{s_2} + \sigma_{n_1} - \sigma_{n_2} + 2bd \sigma(s_2 n_1) - 2d \sigma(s_2^* n_2)^2} \quad \text{.} \hspace{1cm} (4.14)$$

where $\sigma(.)$ is the covariance between the signals in parentheses, and as before, delayed signals are indicated with a superscript $\tau$. For the $H_0$ hypothesis, the smallest eigenvalue is

$$\lambda_1 = 0.5 [b^2 + d^2] \sigma_{s_2} + \sigma_{n_1} + \sigma_{n_2} + 2bd \sigma(s_2 n_1) + 2d \sigma(s_2^* n_2)]$$

$$- 0.5 \sqrt{[b^2 + d^2] \sigma_{s_2} + \sigma_{n_1} - \sigma_{n_2} + 2bd \sigma(s_2 n_1) - 2d \sigma(s_2^* n_2)^2} \quad \text{.} \hspace{1cm} (4.15)$$
As in the previous section, the eigenvalue for the $H_0$ hypothesis does not reduce to zero because the new covariance terms introduced by the delays in the model and the uncorrelated noise do not cancel. Fortunately, switching from the $H_0$ to the $H_1$ hypothesis causes an increase in the smallest eigenvalue that may be thresholded for detected.

4.5 Frame to Frame Processing

In most VAD and change detection applications, processing is done in sequential frames. Some process each frame independently; however, more sophisticated approaches consider a variety of frame to frame processing techniques [118-120]. For this dissertation, a feed-forward technique is developed to calculate the test statistic of the current frame using several previous frames. If both the signal of interest and the interferer are stationary, then the two hypotheses have distinct, constant values for their smallest PCs for all time segments. The challenge is that the voice signals are non stationary, so short frames must be used. However, the processing depends on the calculation of the sample covariance, which can only be accurately calculated using very long frames. In order to resolve the short frame/long frame conflict, the current calculated significant statistic is averaged with several previous significant statistics (this is referred to as the overhang length). The number of previous points included in the average is a chosen parameter. The appropriate value for the overhang length varies depending on the stationarity of the signal of interest. For a non stationary signal, an overhang length that is too low will not smooth the significant statistic fluctuations. On the other hand, with a nearly stationary signal, an overhang length that is too high will cause a high level of misclassification, particularly in the transition regions.

4.6 Simulations and Results – Tonal Case

Section 4.4 presented several models that demonstrated how PCA could be used to detect a signal in the presence of an interferer. This section presents simulations that examine the algorithm’s performance while changing a variety of parameters. All the simulations assume an anechoic in-air environment, with the speed of sound of 343 m/s, and in all cases, the interferer is
white, Gaussian noise. The signal of interest, unless otherwise stated, is a series of tone bursts with random frequency (up to the Nyquist of the sampling). The duration of these bursts is also a random value. The data signal is divided into 100 sample frames, and the significant statistic for each frame is calculated as the smallest PC. Unless otherwise stated, the overhang length is chosen to be one point. The sampling frequency is 44100Hz.

### 4.6.1 One Signal, One Interferer – Simultaneous Mixing

The first simulation uses the simultaneous mixing model from Section 4.3.1. This model is only applicable in geometries where the voice signal and the interferer are equidistance from both sensors. Despite its limited applicability, it is included for its ease of model derivation and threshold selection. The analytical solution given in Equation 4.6 showed that SIR has little effect. Therefore, for this simulation, the SIR was set to 0dB. The two signals are mixed using a linear mixing matrix as shown in Equation 4.1. Figure 4.2(a) shows the voice signal before mixing, Figure 4.2(b) shows the mixed signal, and Figure 4.2(c) shows the significant statistic. During periods of speech inactivity, the significant statistic drops to zero as derived in Equation 4.6. For this simulation, any threshold above zero can accurately differentiate between the active and inactive voice frames.
Figure 4.2: (a) The signal of interest used for all simulations. (b) The mixed signal, X, containing the signal of interest and the white, Gaussian interferer mixed simultaneously at 0dB SIR according to Equation 4.1. (c) The significant statistic generated using PCA. Using this significant statistic and a user chosen threshold, each frame is categorized as signal present or signal absent. For the simultaneous mixing scheme, when the signal of interest is absent, the significant statistic drops to zero.

4.6.2 One Signal, One Interferer – Delayed Mixing

While the previous simulation’s model is only applicable for specific geometries, the model in Section 4.3.2 has been expanded to be correct for any source/receiver geometry. For this simulation, the voice signal and the interferer are mixed according to the model of Equation 4.7 with the delays calculated based on the geometry shown in Figure 4.3(a). The voice signal and interferer are at -7dB SIR. Figure 4.3(b) shows the ROC curve for the PCA technique. As shown in the analysis in Section 4.3.2, the smallest PC does not always tend to zero during periods when the signal of interest is absent, so the $P_{FA}$ is no longer zero. However, there is still a detectable difference in the significant statistic between signal present and signal absent frames. For a 5% $P_{FA}$, this new technique has a 98.76% $P_{D}$. The log likelihood statistical test presented in Section 2.4.2 was implemented to provide a comparison with the new technique. The dashed curve in Figure 4.3(b) shows the ROC curve for the likelihood test. For a 5% $P_{FA}$, the likelihood test has a $P_{D}$ of 80.71%.
Figure 4.3: (a) The positions of the signal of interest, the interferer, and the two microphones are shown. The TDOAs calculated based on the source/receiver geometry presented are used as the time delays in Equation 4.7. (b) The ROC curves generated in the delayed mixing simulation corresponding to the geometry in (a). The solid curve shows the results using the new PCA technique, while the dashed curve shows the results using the log likelihood statistical test introduced in Section 2.4.2.

4.6.2.1 Varying Signal to Interferer Ratio

For the previous simulation, the two signals were at -7dB SIR. In order to evaluate the new technique’s robustness to low SIR, cases were generated where the SIR is varied between 0dB and -15dB. For all cases, the delays for both signals remain constant and are the same delays used in the previous simulation. The series of plots in Figure 4.4 illustrates the impact of SIR on the technique’s performance. For each SIR, the ROC curve is calculated and the $P_D$ at 1%, 5%, and 10% $P_{FA}$ is recorded. For all three reference $P_{FA}$ values, the $P_D$ decreases as SIR decreases. The solid curve gives the performance of the new PCA technique. For the 5% $P_{FA}$ reference point, the new algorithm has near perfect performance for SIRs above -8dB. After that point, the performance decreases as SIR decreases. The dashed curves in Figure 4.4 give the performance of the likelihood test. For all SIRs and all reference $P_{FA}$ values, the new technique outperforms the likelihood test. Both the new technique and the likelihood test show similar trends in their degradation with respect to SIR.
Figure 4.4: The effect of changing SIR on the algorithm’s performance. Each plot represents a different reference $P_{FA}$ value. From left to right, the reference values are 1%, 5% and 10% $P_{FA}$. The signals are arranged in the geometry indicated in Figure 4.3(a). For all cases, the new technique has a better performance than the likelihood test.

4.6.2.2 Varying Overhang Length

Section 4.5 explained the averaging technique used to improve the separation between the $H_0$ and $H_1$ hypotheses. The previous simulations used a one point overhang length. This section examines the effect of changing the number of overhang points on the algorithm’s performance. The delayed mixing model is used, with the delays calculated from the geometry in Figure 4.3(a). The signal of interest and the interferer are at equal power. Figure 4.5 shows the $P_D$ as a function of overhang length. Each curve represents a specific reference $P_{FA}$ value. As the number of overhang points increases beyond one, the $P_D$ decreases for all of the reference $P_{FA}$ values. This effect is especially pronounced for the 0.5% and 1% $P_{FA}$ reference values. The frame-to-frame processing is not helpful in this instance. Later, when the technique is used as a VAD, frame-to-frame processing greatly improves performance. The reasons for this will be discussed in Section 4.8.2.2.
Figure 4.5: Algorithm performance as a function of the number of overhang points. For the reference $P_{FA}$ values of 0.5%, 1%, 5%, and 10%, the $P_D$ as a function of overhang points is shown. For all reference $P_{FA}$, the $P_D$ decreases with increasing overhang points.

4.6.2.3 Varying Source/Receiver Geometry

In order to evaluate the technique’s robustness to a variety of geometries, a simulation was completed where the positions of the two microphones and the interferer were held constant in the locations indicated in Figure 4.3(a), while the position of the signal of interest varied. Since the SIR at the measurement location is a function of the distance from the signal of interest and interferer to the microphone, changing the source/receiver geometry also changes the SIR at the measurement location. Therefore, this simulation examines the combined effect of changing geometry and SIR. For each case, a ROC curve was calculated and the $P_D$ at a 5% $P_{FA}$ was recorded. Figure 4.6 shows the $P_D$ for each voice signal position for a range of x-positions of 5 to 70 inches and y-positions of 5 to 100 inches. For most (x,y) positions of the signal of interest, the performance of the algorithm is above 90%. There is a zone located along the y-axis between 15 and 80 inches where the performance is lower than in the other regions. In some cases, it is significantly lower than the surrounding area. This decrease in performance is a numerical artifact of the processing. At this point, the condition number of the covariance matrix is extremely high.
As a second test, the positions of the microphones and the signal of interest were held constant in the positions of Figure 4.3(a) while the interfering signal’s position was changed. Once again, the ranges of x-positions of 5 to 70 inches and y-positions of 5 to 100 inches were used. Figure 4.7 shows the PD at 5% PFA for the varying positions of the interferer. Unlike in the previous case, there is a tear-drop shaped zone of extremely poor performance. When the interferer is positioned in this zone, it lies directly between the two microphones. It is much closer to either microphone than the signal of interest, and as a result, the SIR is high. This decreases the performance of the technique, as was seen in Section 4.6.2.1. As the interferer moves farther away from the microphones, the performance greatly increases. The exception to this rule is the diagonal line where PD remains constant at 58%. This line marks the points at which the interferer is equidistance from both microphones, and therefore has zero TDOA. These positions are closest to the simultaneous mixing case, and as a result have a much higher PD value than the surrounding areas.
4.6.2.4 Varying the Type of Signal of Interest

In the previous simulations, the signal of interest was a pure tone at a random frequency with random duration. This simulation investigates the algorithm’s performance for a variety of signal of interest types. This is to determine if the algorithm can be used to detect a number of different source types. The five types of signal forms used in this section are referred to as tones, Gaussian burst, bandpass, uniform pdf, and impulse. The tones signal of interest is the previously described series of tonal bursts at random frequencies. The Gaussian burst is a series of burst of random, white, Gaussian noise. The bandpassed signal is a series of burst of 500Hz wide filtered white noise. The center frequencies for these bursts are the same as the random frequencies used for the tone signal. The uniform pdf case is a series of bursts of a uniformly distributed signal with zero mean. Finally, the impulse signal is a series of impulses. In order to replicate the ring-down characteristic of sensors, the impulse decays exponentially.
Figure 4.8 shows the performance of the new technique for each of the signal type trials. Figure 4.8(a) shows the ROC curves generated when the SIR is -7dB and no additional noise is added to the system. Figure 4.8(b) shows the performance for the various signal types as SIR decreases. For the varying SIR case, the ROC curve is calculated at each SIR and the $P_D$ at 5% is recorded. In both plots, the performance for the tone, Gaussian burst, and uniform pdf cases are identical. The bandpass case has a lower performance. At 0dB SIR case, the new technique’s performance for the bandpass case is nearly perfect, and then decreases for all decreasing SIR. The impulse case has the worst performance of all the signal types examined. The $P_D$ never achieves a value above 45% for all the SIRs examined. Its performance is only slightly better than a random guess. (A random guess would be a straight line where $P_D = P_{FA}$, on the ROC curve, or a horizontal line at 0.05 for the varying SIR plot). Impulses have very small variances; in this example the sample variance of one impulse is 0.0169, while the Gaussian interferer has a sample variance of 0.999. The change in variance of the joint pdf due to the addition of the impulse to the Gaussian interferer is negligible. Since this new method relies on detecting the increase in the variance of the joint pdf, it fails when detecting impulses.

![Figure 4.8](image)

**Figure 4.8:** The results of changing the form of the signal of interest on the algorithm’s performance. The ROC curve for the -7dB SIR is shown in the right plot. For the reference $P_{FA}$ values of 5%, the $P_D$ as a function of SIR is also shown. For the tone, random signal, and uniform pdf cases, the performance curves lie of top of one another, indicating identical performance.
4.6.3 One Signal, One Interferer with Noise

For this series of simulations, the signal of interest and the interferer are mixed according to the model of Equation 4.13, which includes the addition of uncorrelated noise. A simulation was completed where the two signals have equal power, the delays were the same as calculated in Section 4.6.2, and the SNR is -7dB. Figure 4.9 shows the ROC curve for the PCA technique. For a 5% \( P_{FA} \), this new technique has a 97.16% \( P_D \). As before, the log-likelihood test was implemented to provide a comparison with the new technique. The dashed curve in Figure 4.9 shows the ROC curve for the log-likelihood ratio test. For a 5% \( P_{FA} \), the likelihood test has a \( P_D \) of 63.67%. For the condition of added noise, the new PCA based technique outperforms the likelihood test by a large margin.

![ROC curve generated in the delayed mixing simulation with added uncorrelated noise.](image)

**Figure 4.9:** The ROC curves generated in the delayed mixing simulation with added uncorrelated noise. The solid curve shows the results using the new PCA technique, while the dashed curve shows the results using the log-likelihood algorithm.

### 4.6.3.1 Varying Signal to Noise Ratio

This simulation evaluates the new technique’s robustness to decreasing SNR. The signal of interest and interferer are maintained at 0dB SIR, and the delays for both signals remain constant (at the same delays used in the previous simulations). The series of plots in Figure 4.10
illustrates the impact of SNR on the technique’s performance. For each SNR, the ROC curve is calculated and the $P_D$ at 1%, 5%, and 10% $P_{FA}$ is recorded. For all three reference $P_{FA}$ values, the new algorithm maintains nearly perfect performance for high SNR, and then the performance decreases as SNR decreases. The trend seen in this simulation is similar to the trend seen in the varying SIR case. The dashed curves in the plots of Figure 4.10 show the performance of the likelihood test, which is worse for all reference $P_{FA}$ and SNR. For each reference $P_{FA}$ value, the PCA technique and the likelihood test show similar trends in the performance degradation as a function of SNR.

![Figure 4.10: The algorithm’s performance as a function of varying SNR. Each plot represents a different reference $P_{FA}$ value. From left to right, the reference values are 1%, 5% and 10% $P_{FA}$. The signals are arranged in the geometry indicated in Figure 4.3(a). For all cases, the new technique has a better performance than the likelihood test.](image)

### 4.6.4 Co-Effect of Varying SIR and SNR

The technique’s performance with respect to SIR and SNR has been investigated individually in the previous simulations. This section examines the interaction between SIR and SNR. The same signals, mixing matrix, and delays from the previous section were used and held constant. The SNR was varied between 5 and -15dB and the SIR was varied between 0 and -15dB. For each SIR/SNR pair, the $P_D$ at 5% $P_{FA}$ is calculated and collected in a matrix. That matrix is displayed as an image in Figure 4.11. As expected, the $P_D$ is highest for the highest SIR
and SNR pair, and decreases as both SIR and SNR decreases. Following the results seen in the varying SIR and SNR simulations, the performance is nearly perfect for the highest set of SIR/SNR pairs, and then drops off. Figure 4.11 can be used to determine the limits on SIR and SNR based on the individual application’s $P_D$ tolerances.

![Figure 4.11: The co-effect of changing SIR and SNR. For each pair of SIR and SNR values, the $P_D$ at 5% $P_{FA}$ was calculated. The highest performance is at the highest SIR and SNR. Depending on the desired performance, an acceptable SIR and SNR limit can be determined.](image)

### 4.7 Experiments and Results – Tonal Case

The simulations in the section above demonstrated the new PCA technique’s ability to detect the presence of a new signal in the ideal case. In this section, experiments were performed that correspond to the situations presented in the simulations. The simultaneous mixing model is omitted from the experiments because it is not applicable for a variety of geometries. The experiments were performed in an anechoic chamber using two microphones. The geometry for all the experiments is from Figure 4.3(a). The data has a sampling frequency of 50 kHz. In all cases, the interferer is white, Gaussian noise. The overhang length is chosen to be one point. The data signal is split into 113 sample frames, and the significant statistic for each frame is
calculated as the smallest PC. The 113 sample point frame length corresponds to 2.3 ms (100 samples) from the simulations.

There are two points of concern for these experiments: the DAQ system and the speakers. The DAQ system used has a very large 60Hz noise component. All data recorded using this system must be highpass filtered at 100Hz to remove this unwanted signal. Overall, this does not have a large impact on the data or the performance of the algorithm because the frequency content for the tonal signal of interest is all above 500Hz. The second concern is more critical. The output power of the speakers is not constant across all frequency bands. To test the frequency response, white, Gaussian noise is played through the speakers and recorded using the microphones. Figure 4.12(a) shows the fft of the white, Gaussian noise signal. Figure 4.12(b) shows the frequency response of the speakers. There is a drop in power for all frequencies above 5500Hz, which is similar to a lowpass filtering of the data at 5500Hz. The effect of this response will be examined in the experiments.

![Figure 4.12](image-url) **Figure 4.12:** (a) The fft of the white, Gaussian noise signal used to test the frequency response of the speakers. It has nearly constant power at all frequencies. (b) The frequency response of the speakers. After 5500Hz, the response drops off.
4.7.1 Varying Signal to Interferer Ratio

The first experiment performed illustrates the performance of the algorithm to varying SIR in an anechoic chamber. Within the anechoic chamber, the SNR is 15dB. The series of plots in Figure 4.13 illustrates the impact of SIR on the technique’s performance. For each SIR, the ROC curve is calculated and the $P_D$ at 1%, 5%, and 10% $P_{FA}$ is recorded. The blue curve represents the performance of the algorithm with simulated data from Section 4.6.2.1. The red curve represents the performance when the simulated signal and interferer have been low passed filtered at 5500 Hz. The loss of the higher frequency components has a negative impact on the performance. The green curve shows the performance of the experimental data. For all three reference $P_{FA}$ values, the $P_D$ decreases as SIR decreases. The trends exhibited by the filtered simulation and the experiment follow the same general trend as in the original simulation. The performance of the experimental case closely follows the filtered simulation. The percentage error between the filtered simulation and the experiment is shown in Figure 4.14. For all SIR above -5.5dB, the percentage error is below 6%.

![Figure 4.13: The effect of changing SIR on the algorithm’s performance. Each plot represents a different reference $P_{FA}$ value. From left to right, the reference values are 1%, 5% and 10% $P_{FA}$. The signals are arranged in the geometry indicated in Figure 4.3(a). For all cases, the performance when using experimental data closely mimics the performance when using filtered simulated data.](image-url)
Figure 4.14: The percentage error between the filtered simulated and experimental performances. For all SIR above -5.5dB, the percentage error is below 6%.

4.7.2 Varying the Type of Signal of Interest

This experiment investigates the performance of the new algorithm for a variety of signal of interest types in an anechoic environment. The five types of signal forms are the same as simulated in Section 4.6.1.4: tones, Gaussian burst, bandpass, uniform pdf, and impulse. Figure 4.15 shows the performance of the new technique for each of the signal trials. For each signal type, the SIR is -5dB. The bandpass signal has very similar performance as in the simulated case. In the simulation, using the new technique to detect the impulse was only slightly better than a random guess. The experiment confirmed this result. For the cases of the tones, the Gaussian burst, and the uniform pdf, the performance is lower than in the simulation. This is particularly true in both the Gaussian and uniform pdf case. This is due to the frequency response of the speakers, as discussed in a previous section. For the tones and bandpass cases, the cutoff frequency of the speakers only affects the burst with a (center) frequency greater than 5500Hz. However, the Gaussian and uniform pdf are broadband signals, so the cutoff frequency affects every burst of the emergent signal and the performance is significantly reduced.
4.7.3 Non Anechoic Environment

The previous experiments were performed in an anechoic chamber. This experiment measures the performance of the algorithm to varying SIR in a reverberant environment. The uncorrelated background noise has higher power in the reverberant environment than in the anechoic chamber, and thus, the data has a lower SNR in the reverberant environment. In the anechoic chamber, the SNR was approximately 15dB. For this experiment the SNR is approximately 2dB. The series of plots in Figure 4.16 illustrates the impact of SIR on the technique’s performance. The blue curve represents the performance of the algorithm with data recorded in the anechoic chamber. The red curve represents the performance with data recorded in the reverberant environment. The performance is lower for the reverberant data than the anechoic data, and the degradation in performance is especially pronounced for the 1% $P_{FA}$ reference value. Some performance degradation is due to the higher noise in the reverberant environment. The tonal case is particularly sensitive to the reverberant environment because each burst is composed of only one frequency. Any out-of-phase reflection causes interference with
the direct path, and in some instances could lead to complete cancellation of the signal of interest at the measurement point.

![Graph](image)

**Figure 4.16:** The performance of the algorithm with respect to varying SIR. A comparison is made between the performances in an anechoic environment (blue curve) versus a reverberant environment (red curve).

### 4.7.4 Varying Signal to Noise Ratio

The previous experiments occurred at a constant SNR that was determined by the environment in which the data was collected. This section describes a hybrid simulation/experiment that examines the performance of the new algorithm with respect to varying SNR. A simulated white, Gaussian signal is added to the data from the anechoic chamber recorder at 0.139dB SIR. By changing the power of the white, Gaussian signal, a hybrid experiment is created where the SNR varies. The results of this hybrid experiment are shown in Figure 4.17. The blue curve represents the simulations from Section 4.6.3.1 and the red curve represents the hybrid experiment. As in the previous sections, the simulated data is filtered at 5500Hz. The hybrid experiment closely matches the simulated results.
Figure 4.17: The performance of the algorithm as a function of SNR. A hybrid experiment is performed where artificial noise is added to data recorded in an anechoic chamber.

4.8 Simulations and Results – Voice Activity Detection

The previous two sections presented the general case of detecting an unknown signal in the presence of an interferer. The signal of interest was a series of random tone bursts (other signals were also investigated). This section, and the one following, addresses the specific case where the unknown signal is a voice signal. These simulations replicate the simulations seen in Section 4.6, and are designed to verify the technique’s performance under several conditions. The same assumptions and conditions from Section 4.6 apply here. Namely, all the simulations assume an anechoic in-air environment, with the speed of sound of 343 m/s, and in all cases, the interferer is white, Gaussian noise. The data signal is divided into 100 sample frames, and the smallest PC is calculated as the significant statistic for each frame. The 100 sample point frame length was chosen corresponding to 2.3 ms to ensure constant variance across the frame. Unless otherwise stated, the overhang length is chosen to be 10 points. The voice signal is a sample of an audio book, where the sampling frequency is 44100Hz.

4.8.1 One Voice Signal, One Interferer – Simultaneous Mixing

The first simulation uses the simultaneous mixing model. As mentioned before, this model has limited applicability, but is included for completeness. For this simulation, the SIR was set to -8dB. The model derivation proved that SIR has no effect on the results. Figure
4.18(a) shows the voice signal before mixing, Figure 4.18(b) shows the mixed signal, and Figure 4.18(c) shows the significant statistic. During periods of speech inactivity, the significant statistic drops to zero as derived in Equation 4.6. For this simulation, any threshold above zero can accurately differentiate between the active and inactive voice frames. This is an idealized case. Any uncorrelated background noise will complicate the threshold selection.

Figure 4.18: (a) The original voice signal used for all simulations. (b) The mixed signal, X, containing the voice signal and the white, Gaussian interferer mixed simultaneously according to Equation 4.1. (c) The significant statistic generated using PCA. Using this significant statistic and a user chosen threshold, each frame is categorized as speech or non-speech. For the simultaneous mixing scheme, when speech is not present, the significant statistic drops to zero.

One large difference to note between this case and the tonal case is the range of values for the significant statistic during active speech frames. Referring to Figure 4.18(c), when the tonal signal was present, the values for the significant statistic were centered at 0.8. While there was some fluctuation around this value, the significant statistic never dropped below 0.5 when the signal of interest was present. In other words, there was a clear distinction between the values attained when the signal was present versus when it was not present. This is due to the near constant sample variance of a tonal signal. On the other hand, voice is non-stationary. The sample variance of speech changes as a function of time, which causes the significant statistic to
vary in time over a wider range of values. The values for the significant statistic can be very close to zero. This is especially true in the transition regions where speech is either ramping up in amplitude or ringing down. This lowers the performance for the technique when it is used as a VAD algorithm below that of the tonal case.

### 4.8.2 One Signal, One Interferer – Delayed Mixing

For this simulation, the voice signal and the interferer are mixed according to the model of Equation 4.7 with the delays calculated based on the geometry shown in Figure 4.19(a). The voice signal and interferer have equal power. Figure 4.19(b) shows the ROC curve for the PCA technique. For a 5% $P_{FA}$, this new technique has a 93.0% $P_D$. The VAD algorithm presented in Section 2.5.2 was implemented to provide a comparison with the new technique. The frequency bins used for the statistical VAD algorithm are the bins between 500 and 3500 Hz, which were determined based on the frequency content of the voice signal. The dashed curve in Figure 4.19(b) shows the ROC curve for the statistical VAD algorithm. For a 5% $P_{FA}$, the statistical VAD algorithm has a $P_D$ of 75.3%. The performance of the statistical algorithm is based on using a fixed threshold, not the adaptive threshold used in [108]. The performance could be increased by using the adaptive method, however, to simplify the calculation of the ROC curve, only a fixed threshold is employed.
Figure 4.19: (a) The positioning of the signals and microphones is shown. The time delays introduced into the mixing scheme are due to the corresponding TDOA between the two sensors. (b) The ROC curves generated in the delayed mixing simulation corresponding to the geometry in (a). The solid curve shows the results using the new PCA technique, while the dashed curve shows the results using the statistical VAD algorithm introduced in Section 2.5.2.

4.8.2.1 Varying Signal to Interferer Ratio

This simulation examines the algorithm’s performance under varying SIR conditions. For all cases, the delays for both signals remain constant and are the same delays used in the previous simulation. The series of plots in Figure 4.20 illustrates the impact of SIR on the technique’s performance. The reference $P_{FA}$ values are the same as before: 1%, 5%, and 10% $P_{FA}$. As before, for all three reference $P_{FA}$ values, the $P_D$ decreases as SIR decreases. The dashed curves in Figure 4.20 give the performance of the statistical VAD algorithm. For all SIRs and all reference $P_{FA}$ values, the new technique outperforms the statistical VAD algorithm.
Figure 4.20: The algorithm’s performance as a function of varying SIR. Each plot represents a different reference $P_{FA}$ value. From left to right, the reference values are 1%, 5% and 10% $P_{FA}$. The signals are arranged in the geometry indicated in Figure 4.19(a). For all cases, the new technique has a better performance than the statistical VAD algorithm.

The performance of the technique as a VAD algorithm differs from the trend seen in Figure 4.4. For high SIR values, the VAD has lower $P_D$ than in the tonal detection case. However, the VAD performance does not roll off as steeply as in the tonal case. This is due to the non-stationary characteristics of speech mentioned in Section 4.8.1. The sample variance of the tone in the tonal case is nearly constant for all frames. On the other hand, the voice signal has a variance that is a function of time. As a result, the pdf of the significant statistic for the $H_1$ case (called $P_1$ in Section 2.4.1) has a different shape for the tonal case versus the voice case. For the tonal case, $P_1$ has a small variance but in the voice case, $P_1$ has a much larger variance. Figure 4.21 shows how the distributions $P_0$ and $P_1$ change as a function of SIR. Column (a) shows the tonal case while column (b) shows the voice case.
Figure 4.21: The change in the distributions of the significant statistic for the \( H_0 \) and \( H_1 \) hypothesis for varying SIR. Column (a) shows the change in \( P_0 \) and \( P_1 \) for the tonal case, while column (b) shows the change for the voice case.

Figure 2.9 in Section 2.6 showed how increasing overlap for the \( P_0 \) and \( P_1 \) distributions caused a decrease in the performance of the algorithm. This explains why the tonal case has a much higher performance for higher SIR, but the voice case has a more constant decrease in performance with decreasing SIR. For the -5dB SIR case, there is a negligible amount of overlap between the \( P_0 \) and \( P_1 \) case. The algorithm has nearly perfect classification because a threshold can be chosen that completely divides the \( P_0 \) and \( P_1 \) distributions. On the other hand, for the VAD case at -5dB SIR, there is a noticeable overlap between the \( P_0 \) and \( P_1 \) distributions. 18.3% of the area under the \( P_1 \) distribution overlaps the \( P_0 \) distribution. This accounts for the lower performance in the VAD case as compared to the tonal case for the -5dB SIR. For increasing
SIR, the overlap between the $P_1$ and $P_0$ increases for both the tonal case and the voice case. However, the percentage of area overlapped does not increase as much for the VAD case as for the tonal case. For the -10dB SIR case, the percentage of area of the $P_1$ distribution that overlaps the $P_0$ distribution in the tonal case is 50.03%, compared to only 31.04% for the voice case. Therefore, the tonal case has higher performance than the voice case for higher SIR values, but decreases in performance more rapidly for decreasing SIR.

### 4.8.2.2 Varying Overhang Length

This simulation examines the effect of changing the number of overhang points on the algorithm’s performance. The voice signal and the interferer are at equal power. Figure 4.22(a) shows the $P_D$ as a function of overhang length. Each curve represents a specific reference $P_{FA}$ point. As the number of overhang points increases from 0 to 9, the $P_D$ increases for all of the reference $P_{FA}$ values, and then begins to level off. Once the number of overhang points increases beyond 15, the $P_D$ begins decreasing for all $P_{FA}$ values. This effect is especially pronounced for the 0.5% and 1% $P_{FA}$ reference values, which indicates that a high $P_D$ can only be achieved with a tradeoff of higher false alarm rate. Figure 4.22(b) shows the ROC curve generated for the overhang lengths of 5, 15, and 25 points. It confirms that up to 15 points, an overall increase in performance can be achieved. Beyond that point however, the performance for low $P_{FA}$ values decreases significantly.
Figure 4.22: The results of changing the number of overhanging points on the algorithm’s performance. (a) For the reference $P_{FA}$ values of 0.5%, 1%, 5%, and 10%, the $P_D$ as a function of overhang points is shown. For all reference $P_{FA}$, the $P_D$ increases with increasing overhang points until it peaks at 17 points. After that point, the $P_D$ drops for all reference $P_{FA}$ values. (b) The ROC curves generated at three reference overhang point values. The 15 points curve shows an increase in performance over the five points curve. The 25 points curve shows a drop in performance, especially in the lower $P_{FA}$ region.

In the tonal case, the overhang value used was one. For the VAD sections, the simulations used a 10 point overhang length. While the feed-forward processing drastically decreased the performance in the tonal case for any value greater than one, it significantly improved the performance of the VAD. The fluctuations in the significant statistic, caused by changes in the sample variance of the speech signal, can be smoothed by applying the feed-forward averaging process. In the transition regions at the end of words and sentences, the small significant statistic values are weighted by the previous frames, and detections in those regions can be increased. Figure 4.23 shows the effect of the frame-to-frame processing on the tonal case as compared to the VAD case. The significant statistic calculated for a one point overhang length is shown in blue, and the significant statistic calculated for a 10 point overhang length is shown in red. Figure 4.23(a) shows the tonal case. As previously mentioned, the values for the significant statistic of the signal present and signal absent frames are distinct. Increasing the overhang length from one to 10 in the tonal cases caused leakage from the signal present frames into the signal absent frames, and vice versa. The algorithm did not declare frames as signal...
present until several frames after the source appeared, and continued to declare frames as signal present for several frames after the emergent source disappeared. Figure 4.23(b) shows the effect of increasing the overhang length for the voice case. The significant statistic for the one point overhang length (the blue curve) has lot of fluctuation, especially for frames 125 to 225. This causes a high rate of misclassification when using a fixed threshold. Increasing the overhang length to ten points (the red curve) has significantly smoothed this fluctuation. This has caused a greater distinction between the values for the speech active and inactive frames. The $P_D$ in this region is drastically improved.

![Figure 4.23](image)

**Figure 4.23:** The effect of increasing overhang length on the significant statistic. The blue curves represent the significant statistic as calculated with a one point overhang value, while the red curves are calculated using a 10 point overhang value. Plot (a) is for the tonal case, while plot (b) is for the VAD case.

### 4.8.2.3 Varying Source/Receiver Geometry

In order to evaluate the technique’s robustness to a variety of geometries, a simulation was completed where the positions of the two microphones and the interferer were held constant in the locations indicated in Figure 4.19(a), while the position of the voice signal moved. This illustrates the technique’s robustness to changes in the source/receiver geometry. As mentioned in Section 4.6.2.3, the SIR is also affected by the changing geometry, and this simulation examines the combined effect of changing geometry and SIR. For each case, a ROC curve was calculated and the $P_D$ at a 5% $P_{FA}$ was recorded. Figure 4.24 shows the $P_D$ for each voice signal.
position for a range of x-positions of 5 to 70 inches and y-positions of 5 to 100 inches. There are three zones in Figure 4.24. The first zone is the small semi-circle centered at an x-position of 30 inches. This decrease in performance is a numerical artifact of the processing. At this point, the condition number of the covariance matrix is extremely high. The second zone is the arc that covers most of the y-positions between 0 and 70 inches. In this zone, the technique has very good performance. In addition, changes in geometry in this region do not cause significant changes in the performance of the VAD. Lastly, in the arc above the y-position of 70 inches, the performance begins to degrade. At this point, the voice source is further from the microphones than the interferer, and the SIR is lowered. As demonstrated in Section 4.8.2.1, this causes degradation in the performance of the algorithm.

**Figure 4.24:** The effect of changing the location of the voice signal on the performance of the algorithm. The interferer is held stationary at (20,90) while the position of the voice signal is changed. The microphones remain stationary at (30,0) and (0,50). This is the geometry of Figure 4.19(a). The P_D at 5% P_FA is recorded for each (x,y) position of the voice.

As a second test, the positions of the microphones and the voice signal were held constant in the positions of Figure 4.19(a) while the interfering signal’s position was changed. Once again, the ranges of x-positions of 5 to 70 inches and y-positions of 5 to 100 inches were used. Figure 4.25 shows the P_D at 5% P_FA for the varying positions of the interferer. Unlike in the
previous case, there is a tear-drop shaped zone of extremely poor performance. When the interferer is positioned in this zone, it lies directly in between the two microphones. It is much closer to either microphone than the voice signal, and the SIR is low. As was seen in the tonal case, this causes a decrease in the performance of the algorithm. As the interferer moves further away from the microphones, thus increasing SIR, the performance greatly increases. The exception to this rule is the diagonal line where $P_D$ remains constant at 96%. This line marks the points at which the interferer is equidistance from both microphones, and therefore has zero TDOA. These positions are closest to the simultaneous mixing case, and as a result have a much higher $P_D$ value than the surrounding areas.

Figure 4.25: The effect of changing the location of the interferer on the performance of the algorithm. The voice signal is held stationary at (65,75) while the position of the interferer is varied. The microphones remain stationary at (30,0) and (0,50). This is the geometry of Figure 4.19(a). The $P_D$ at 5% $P_{FA}$ is recorded for each (x,y) position of the interferer.

4.8.3 One Signal, One Interferer with Noise

For this series of simulations, the voice signal and the interferer are mixed according to the model of Equation 4.13, which includes the addition of uncorrelated noise. A simulation was completed where the two signals have equal power, the delays were the same as calculated in
Section 4.8.2, and the SNR is 5dB. Figure 4.26 shows the ROC curve for the PCA technique. For a 5% P_{FA}, this new technique has a 91.4% P_{D}. As before, the statistical VAD algorithm was implemented to provide a comparison with the new technique using the same frequency bins previously mentioned. The dashed curve in Figure 4.26 shows the ROC curve for the statistical VAD algorithm. For a 5% P_{FA}, the statistical VAD algorithm has a P_{D} of 72.3%. For this high SNR, both the PCA technique and the statistical VAD algorithm have mostly maintained their performances.

**Figure 4.26:** The ROC curves generated in the delayed mixing simulation with added uncorrelated noise. The solid curve shows the results using the new PCA technique, while the dashed curve shows the results using the statistical VAD algorithm.

### 4.8.3.1 Varying Signal to Noise Ratio

In the previous simulation, it was shown that the technique did not suffer performance degradation with the addition of noise for a high SNR case. This simulation evaluates the new technique’s robustness to decreasing SNR. The voice signal and interferer are maintained at 0dB SIR, and the delays for both signals remain constant (at the same delays used in the previous simulations). The series of plots in Figure 4.27 illustrates the impact of SNR on the technique’s performance. For each SNR, the ROC curve is calculated and the P_{D} at 1%, 5%, and 10% P_{FA} is
recorded. For all three reference $P_{FA}$ values, the $P_D$ decreases as SNR decreases. The trend as SNR decrease is similar to the trend when SIR decreases. The performance gradually decreases until a certain point. After this “knee” in the curve, the performance decreases more dramatically. The dashed curves in the plots of Figure 4.27 show the performance of the statistical VAD algorithm, which is worse for all reference $P_{FA}$ values and SNRs. For each reference $P_{FA}$ value, the PCA technique and the statistical show similar trends in the performance degradation as a function of SNR.

![Figure 4.27: The algorithm’s performance as a function of varying SNR. Each plot represents a different reference $P_{FA}$ value. From left to right, the reference values are 1%, 5% and 10% $P_{FA}$. The signals are arranged in the geometry indicated in Figure 4.19(a). For all cases, the new technique has a better performance than the statistical VAD algorithm.](image)

### 4.8.4 Co-Effect of Varying SIR and SNR

This section examines the interaction between SIR and SNR. The same signals, mixing matrix, and delays from the previous section were used and held constant. The SNR was varied between 5 and -15dB and the SIR was varied between 0 and -15dB. For each SIR/SNR pair, the $P_D$ at 5% $P_{FA}$ is calculated and collected in a matrix. That matrix is displayed as an image in Figure 4.28. As expected, the $P_D$ is highest for the highest SIR and SNR pair, and decreases as both SIR and SNR decreases. Following the results seen in Sections 4.8.2.1 and 4.8.3.1, decreasing SIR causes a more steady reduction in $P_D$ than decreasing SNR. Figure 4.28 can be used to determine the limits on SIR and SNR based on the individual application’s $P_D$ tolerances.
Figure 4.28: The co-effect of changing SIR and SNR. For each pair of SIR and SNR values, the $P_D$ at 5% $P_{FA}$ was calculated. The highest performance is at the highest SIR and SNR. Depending on the desired performance, an acceptable SIR and SNR limit can be determined.

### 4.9 Experiments and Results – Voice Activity Detection

As in the tonal case, experiments were performed to correspond to the situations presented in the simulations to verify the performance of the PCA technique as a VAD algorithm. The experiments replicate the experiments shown in Section 4.7; omitting the varying signal of interest type. The experiments were performed in an anechoic chamber using two microphones, and the geometry shown in Figure 4.19(a). The data has a sampling frequency of 50000 Hz. In all cases, the interferer is white, Gaussian noise. The overhang length is chosen to be 10 points. The data signal is split into 113 sample frames, and the significant statistic for that segment is calculated as the smallest PC. The 113 sample point frame length was corresponds to 2.3 ms (100 samples) from the simulations.

#### 4.9.1 Varying Signal to Interferer Ratio

This experiment tested the algorithm’s performance to varying SIR for the VAD case in the anechoic chamber. The series of plots in Figure 4.29 shows the performance of the VAD for
the reference $P_{FA}$ values of 1%, 5%, and 10%. The blue curve represents the performance of the algorithm with simulated data from Section 4.8.2.1. The red curve represents the performance when the simulated signal and interferer have been low passed filtered at 5500 Hz. As seen in the tonal case, the loss of the higher frequency components has a negative impact on the performance. While most of the voice’s frequency content is between 500 and 3500Hz, there are several frames with higher frequency components. Lowpass filtering the data removes these higher frequency components and decreases the performance. However, since most of the frequency content is below 5500Hz, the decrease in performance is not as severe as in the tonal case. The green curve shows the performance using experimental data. The trends exhibited by the filtered simulation and the experimental data follow the same general trend as in the original simulation. The performance of the experimental case closely matches the filtered simulation. The percentage error between the filtered simulation and the experiment is shown in Figure 4.30. For all SIR, the percentage error is below 6%.

![Figure 4.29](image)

**Figure 4.29:** The effect of changing SIR on the algorithm’s performance. Each plot represents a different reference $P_{FA}$ value. The performance of the algorithm is reduced when the signal of interest and interferer are low pass filtered at 5500Hz. For all cases, the performance of the algorithm for the experimental data mimics the performance for the filtered simulated data.
4.9.2 Non Anechoic Environment

The previous experiments were performed in an anechoic chamber. This experiment measures the performance of the algorithm to varying SIR in a reverberant environment. Once again, the SNR is approximately 2dB. The series of plots in Figure 4.31 illustrates the impact of SIR on the technique’s performance. The performance with respect to SIR is shown for two types of data: the blue curve represents anechoic data, while the red curve represents reverberant data. While the performance has decreased when moving from an anechoic to a reverberant environment, the decrease is not as severe as in the tonal case. This is due to the wideband nature of speech. Unlike the tonal case, where each burst was composed of a single frequency, each speech frame encompasses a wide band of frequencies. If only a single frequency component exists, destructive interference at that single frequency will significantly degrade the performance. Since the voice signal spans a number of frequencies, the destructive interference at any one frequency will not cause a decline in performance as severe as in the tonal case.
Figure 4.31: The performance of the algorithm with respect to varying SIR. A comparison is made between the performance in an anechoic environment (blue curve) versus a reverberant environment (red curve).

4.9.3 Varying Signal to Noise Ratio

The previous experiments occurred at a constant SNR that was determined by the environment in which the data was collected. This section describes a hybrid simulation/experiment that examines the performance of the new algorithm with respect to varying SNR. A white, Gaussian signal at varying power is added to the anechoic chamber data recorded at -1.552dB SIR. The results of this hybrid experiment are shown in Figure 4.32. The blue curve represents the simulations from Section 4.8.3.1 and the red curve represents the hybrid experiment. As in the previous simulations, the simulation is filtered at 5500Hz. The hybrid experiment closely matches the simulated results.
Figure 4.32: The performance of the algorithm as a function of SNR. A hybrid experiment is performed where artificial noise is added to data recorded in an anechoic chamber.

4.10 Conclusions

This chapter introduced a new method for emergent signal detection based on PCA. The new method has the capability to detect tonal signals in the presence of a Gaussian interferer for a variety of conditions. It was also shown that a variety of other signals could be detected using this new technique, including Gaussian bursts, bandpass filtered signals, and signals with a uniform pdf. Existing algorithms detect the emergent signal by detecting changes in the pdf of the mixed signals. The new method uses PCA to detect the change in variance along the projection direction corresponding to an uncorrelated source. The new technique was also shown to be a robust VAD algorithm for the specific case of one voice signal corrupted by one interferer. The addition of a feed-forward processing technique further improved the performance as a VAD algorithm by using previous significant statistic values to weight the current value.

Simulations showed the capabilities of the new algorithm in detecting emergent signals and voice signals in a variety of conditions. Even in the presence of uncorrelated background noise, the new algorithm was able to detect the emergent signal of interest for both the tonal and voice case for 0dB SIR. For both varying SIR and SNR, the new algorithm outperformed the existing algorithms; the log-likelihood ratio test for the tonal signal, and the spectrum based VAD algorithm for the voice signal. Anechoic experimental results confirmed the results.
obtained in the simulations. The frequency response of the speakers hindered the performance of the algorithm, but the experiments matched the simulations when the simulated data was low pass filtered at 5500Hz. Experiments were also performed in a reverberant environment. The tonal case saw an extreme decrease in the reverberant situation because of the destructive interference cause by the echoes.

The final chapter in this dissertation presents the conclusions of this research. A short summary is provided of the contribution of each chapter. The significance of the research with respect to improving signal clarity is presented. Finally, future research directions are explored.
Chapter 5:

**Summary, Conclusions, and Future Research**

### 5.1 Summary and Conclusions

The goal of the work presented in this dissertation is to improve the clarity of a signal of interest through post processing of received acoustic array data. Signal clarity was improved in two ways: suppressing interferers and detecting the presence of emergent signals. Chapter 3 presented a new method of interference suppression based on a combination of beamforming and BSS techniques. The new method has the capability to suppress non Gaussian interferers, which makes it ideal for telecommunication and speech processing applications. The simulations performed showed the capabilities of the new algorithm in suppressing interferers in a variety of conditions. In comparison to spatial nulling techniques, such as MVDR, the new algorithm had a higher performance. This was quantified by the amount of suppression of the interferer in the BTR plots. Even when the signals were spatially indistinct, the algorithm was still able to partially suppress the interferer. This overcomes the major drawback of null beamformers. The algorithm showed 20dB of suppression of broadband interferers, even when the SIR dropped to -30dB. In addition, the algorithm was shown to have good performance for SNR values above -18dB, which is comparable to other spatial techniques.

Signal clarity was also improved by detecting emergent signals. A new emergent signal detection algorithm based on PCA was described in Chapter 4. Several models were developed to detect the emergent signal of interest in the presence of an interferer. These models took into account two types of mixing: simultaneous mixing and time delayed mixing. In addition, the effect of uncorrelated background noise was also examined. Simulations were performed using each of the models discussed where the parameters of SIR, SNR, and geometry were varied. When the signal of interest was a series of tonal bursts, the new technique was very effective at identifying the emergent signal in a variety of situations. Several other forms for the signal of
interest were used in the presence of a white, Gaussian interferer, and for any signal with nearly constant sample variance, the performance was mostly unchanged. For signals with very small variances, such as impulses, the performance was very poor. For all signal types tested, the new PCA algorithm outperformed the log-likelihood ratio test.

When the PCA technique was used as a VAD algorithm, the performance was dramatically different than in the tonal case. Due to the fluctuating sample variances of the voice signal, the performance of the new algorithm was reduced from that seen in the tonal case. However, the implementation of a feed-forward processing routine was able to recover some of the performance. Weighting the significant statistic of the current frame with values from previous frames smoothed some of the variation in the significant statistic and increased the probability of detecting the voice signal. Simulations and experiments demonstrated the performance of the new PCA technique as a VAD. The simulations showed that the algorithm was able to robustly detect the presence of speech for a variety of conditions, including varying SIR, varying SNR, and varying source/receiver geometry. The experiments confirmed the results seen in the simulations, and demonstrated the ability of the algorithm to detect a voice signal in an anechoic and a reverberant environment.

5.2 Significance of Research and Results

There are four main significant areas of contribution for this research: (1) introducing a new interference suppression method, (2) improving all existing ICA algorithms, (3) introducing a new emergent signal detection technique, and (4) introducing a new VAD technique. The first area of significance is to use ICA to extract the signal(s) of interest from a set of mixed signals. Previous research has focused on using ICA to separate simultaneous mixed source signals. This dissertation presents a method that takes input from an array of sensors and suppresses the interferer in all the received channels. If the input is an NxK matrix of data, where N is the number of array elements and K is the number of samples of data, the output of the new algorithm is an NxK matrix with the interferer(s) removed from each of the N channels of data. At this point, additional processing can be carried out depending on the specific need. For example, in passive sonar, additional spatial processing can be used to generate BTR plots and
track the signal of interest. This method improves interference suppression as compared to MVDR spatial processing and allows for interferers to be removed; even if they lie along the same direction as the signal of interest (i.e. the signals are spatially indistinct). The algorithm can be used repeatedly to continue to remove interferers and reveal previously undetectable signals.

The second area of significance is related to the first, and involves an improvement for all ICA algorithms. The FastICA algorithm, along with many other ICA algorithms, assumes instantaneous mixing for the source signals. This condition only applies to a few, specific source/receiver geometries. The new algorithm introduced in Chapter 3 used spatial processing to allow for the separation of signals that are not simultaneously mixed. Spatial processing, in the form of beamforming, was used to time align the signals. The array was electronically steered towards each source. This creates a series of signals, one for each source direction, that are all time aligned. If these signals are combined with the first and last unaltered array signals, a set of mixed signals is formed that is approximately instantaneously mixed. This set of signals can be used as the input for any ICA algorithm that assumes instantaneous mixing. The inclusion of the first and last unaltered array signals forms a set of M+2 mixed signals, where M is the number of independent sources. When FastICA (or most other ICA algorithms) is applied to this mixed signal set, M+2 outputs are produced; M separated source signals and two residual mixtures. Spatial processing is used to solve the ICA permutation ambiguity problem, as well as to determine which of the outputs contain the residual mixtures.

The ability of BSS to separate independent sources is a widely studied problem. The third significant contribution of this research is to use BSS to detect the presence of emergent sources. PCA separates uncorrelated sources by projecting the mixed data along the directions of maximum variance. The presence of an emergent signal alters the subspace, and by detecting the changes in the principal components, the emergent signal can be detected. Existing algorithms typically only have access to a single mixed signal, and detect emergent signals by detecting changes in the pdf of that signal. The new PCA method has access to two mixed signals. Instead of detecting the change in the mean or variance of the pdf of the mixed signal, the PCA algorithm can detect the change in the variance along the direction that is associated with the uncorrelated emergent signal. This new PCA based method of emergent signal detection is
robust to a variety of situations, and does not require previous knowledge of the pdfs of the source signals.

Finally, this research contributes to the subset of emergent signal detection known as VAD. While there is nothing inherent in the method that lends it specifically to speech detection, for the specific scenario of one voice signal of interest and one interferer, the PCA method of emergent signal detection is a robust VAD algorithm. A feed-forward processing technique is developed, which greatly improves the overall VAD performance by helping to smooth the effect of the fluctuating sample variance of the voice signal. The advantage of this new method is that the PCA method’s significant statistic is simple to calculate and frequency independent. In addition, the decision rule is based on applying a threshold to one significant statistic instead of a set of thresholds applied to a collection of metrics. While many VAD algorithms require an estimate of the significant statistic for the interferer only case, this new algorithm requires no previous knowledge of the interferer.

5.3 Suggestions for Future Work

There are limitless possibilities for continuing research on the subject of increasing signal clarity. The suggestions for future work specific to the research completed in this dissertation focuses on work that improves or expands on the two topics closely examined: interference suppression and emergent signal detection.

Within the framework of the existing interference suppression algorithm, there are a few areas that would benefit from further research. While this dissertation provided simulations for a variety of situations, a more comprehensive statistical study of the performance of the algorithm could be useful. Since FastICA is an adaptive algorithm, it has no closed form solutions. Therefore, a study on all the parameters that affect the algorithm could show areas where improved performance can be achieved for minimal effort. For example, the number of array elements, the inter-element spacing in the array, and the source/receiver geometry. Voice signals were used in this dissertation as the signal of interest and the interferer, and the performance of the algorithm should be investigated for a variety of signals.
The current algorithm has three main processing blocks: the spatial beamforming, the FastICA algorithm, and the Wiener filter. Each one operates independently and the outputs from one block are passed to subsequent processing blocks. One direction of future research is to determine if it is possible to merge the beamforming and FastICA processing. Instead of using the spatial processing to correct the delayed mixing to fit into the existing assumption of simultaneous mixing, it would be useful to include the spatial information as a part of the separation processing. Instead of the traditional ICA model where a linear matrix mixes the source signals, the model would be altered to have a series of transfer functions or filters convolve the source signals.

One of the assumptions for the interference suppression technique developed in Chapter 3 is that there are no echoes. For future research, it would be useful to modify the procedure to help eliminate echoes as well as interferers. This could be accomplished by incorporating a blind deconvolution algorithm into the existing algorithm. Instead of using the simple DS broadband beamformer implemented in this dissertation, different spatial processing techniques could be used. For example, MVDR could be used in place of the DS beamformer to help reduce the effect of echoes coming from directions other than the look direction.

Another interesting direction for future research is to try to train the new algorithm. Suppose the goal is to improve the clarity of the speech of a specific person. If a set of ICA basis vectors were matched to a specific person’s speech, the question is, could the new algorithm be trained to remove all signals except that specific voice pattern? One application of the new algorithm was to continually strip away interferers to reveal previously undetected signals of interest. Using this training data, the algorithm could be used to “mine” the data for a particular voice signal and to remove all other signals. Currently, the algorithm relies on the user to determine which signal to remove, training the algorithm to find a specific signal would allow the algorithm to adaptively decide which signals should be removed.

If the interference suppression technique could be trained to search for a particular signal based on some previous knowledge of that signal, then the new emergent signal detection algorithm could be trained to accomplish the same task. By training the emergent signal detector, the emergence of a specific signal could be detected instead of detecting any emergent signal. Even if there were multiple signals emerging in the data, the PCA method could be used to detect
only the signal for which it has been trained. For example, as a VAD, the trained algorithm could be used to detect the voice of a specific person from within a conversation involving several individuals.

The new PCA emergent signal detector is formulated for the existence of one signal of interest and one interferer. Since only two microphones are used, and therefore, only two mixed signals are available, PCA can only separate two source signals. An interesting area for future research would be to see if the addition of more microphones (and thus mixed signals) allows for the detection of additional emergent signals. If three microphones are used, could the new PCA method be used to detect the emergence of two signals in the presence of an interferer? If it was possible, further investigation would be needed to determine how the algorithm responded if only one or both of the emergent signals was present. Would the algorithm be able to determine which of the emergent signals was detected if only one was present?

The possibilities for future research are extensive. While there are many possible improvements and tangents that could be researched, the work of this dissertation provided two valuable algorithms to increase signal clarity. These algorithms provide a significant contribution to the overall topic of improving signal clarity and have pushed the boundaries of signal processing a little further into the realm of unknowns.
References


Appendix A:

PCA Model Equation Derivation

This appendix contains the full calculations for the eigenvalue problems in Chapter 4. For brevity only portions of the eigenvalue calculations were included in Chapter 4. The calculations of the covariance matrix, as well as the eigenvalues, for the three models developed in Chapter 4 are contained in full in the following sections. In addition, the formula used to calculate the eigenvalues for any symmetric $2 \times 2$ matrix is derived.

A.1 Eigenvalue Calculation

In this dissertation, the data is collected from two sensors. As a result, the covariance matrix is a $2 \times 2$ matrix. The solution to the eigenvalue problem for a $2 \times 2$ matrix is easy to derive. The covariance matrix is written as

$$\mathbf{C}_x = \begin{bmatrix} c_{x1}(1,1) & c_{x1}(1,2) \\ c_{x1}(1,2) & c_{x1}(2,2) \end{bmatrix}$$ \hspace{1cm} A.1.1

The eigenvalues are determined by solving the equation:

$$\det(\mathbf{C}_x - \lambda \mathbf{I}) = 0$$

$$\mathbf{C}_x - \lambda \mathbf{I} = \begin{bmatrix} c_{x1}(1,1) - \lambda & c_{x1}(1,2) \\ c_{x1}(1,2) & c_{x1}(2,2) - \lambda \end{bmatrix}$$

$$\det(\mathbf{C}_x - \lambda \mathbf{I}) = (c_{x1}(1,1) - \lambda)(c_{x1}(2,2) - \lambda) - c_{x1}(1,2)^2$$

$$0 = (c_{x1}(1,1) - \lambda)(c_{x1}(2,2) - \lambda) - c_{x1}(1,2)^2$$

$$0 = c_{x1}(1,1)c_{x1}(2,2) - c_{x1}(1,2)\lambda - c_{x1}(1,1)\lambda + \lambda^2 - c_{x1}(1,2)^2$$

$$0 = \lambda^2 - (c_{x1}(1,1) + c_{x1}(2,2))\lambda + c_{x1}(1,1)c_{x1}(2,2) - c_{x1}(1,2)^2$$

The eigenvalues are calculated using the quadratic equation:

$$\lambda_{1,2} = \frac{(c_{x1}(1,1) + c_{x1}(2,2)) \pm \sqrt{(-c_{x1}(1,1) - c_{x1}(2,2))^2 - 4c_{x1}(1,1)c_{x1}(2,2) + 4c_{x1}(1,2)^2}}{2}$$ \hspace{1cm} A.1.2
Algebraically reducing the expression under the radical gives:

\[
\begin{align*}
&(-c_s(1,1) - c_s(2,2))^2 - 4c_s(1,1)c_s(2,2) + 4c_s(1,2)^2 \\
&= c_s(2,2)^2 + 2c_s(1,1)c_s(2,2) + c_s(1,1)^2 - 4c_s(1,1)c_s(2,2) + 4c_s(1,2)^2 \\
&= c_s(2,2)^2 - 2c_s(1,1)c_s(2,2) + c_s(1,1)^2 + 4c_s(1,2)^2 \\
&= (c_s(1,1) - c_s(2,2))^2 + 4c_s(1,2)^2
\end{align*}
\]

The eigenvalues are:

\[
\lambda_{1,2} = \frac{(c_s(1,1) + c_s(2,2)) \pm \sqrt{(c_s(1,1) - c_s(2,2))^2 + 4c_s(1,2)^2}}{2} \quad \text{A.1.3}
\]

This dissertation focuses on the smallest eigenvalue, which is calculated using the negative form of Equation A.1.3.

### A.2 Simultaneous Mixing Case

The first model examined is for simultaneous mixing. It assumes there is no time delay between the signal received at the first microphone and at the second microphone. This is the instantaneous mixing case that is used in many of the ICA models. Let \(s_1\) be the signal of interest with variance \(\sigma_{s1}\), and \(s_2\) be an interfering signal with variance \(\sigma_{s2}\). \(H_1\) represents the case where the signal of interest, \(s_1\), and the interferer, \(s_2\), are both present and \(H_0\) represents the case where only the interferer is present. For the simultaneous mixing model, the two hypotheses are represented by the models

\[
\begin{align*}
H_0 : \quad x_1(k) &= as_1(k) + bs_2(k) \\
&\quad x_2(k) = cs_1(k) + ds_2(k) \\
H_1 : \quad x_1(k) &= bs_2(k) \\
&\quad x_2(k) = ds_2(k)
\end{align*}
\]

where \(k\) is the sample number, and \(a, b, c,\) and \(d\) are the linear mixing coefficients. The sample covariance matrix is calculated element by element according to
\[ c_{x}(i, j) = E\{x_i^* x_j\} = \frac{1}{n} \sum_{k=1}^{n} x_i(k)x_j(k). \]  

A.2.2

The covariance between the signal of interest and the interferer is denoted \( \sigma(s_1s_2) \).

For the \( H_0 \) hypothesis the covariance matrix elements are:

\[ c_{x}(1,1) = \frac{1}{n} \sum_{k=1}^{n} x_1 x_1 = \frac{1}{n} \sum_{k=1}^{n} (as_i + bs_2)(as_i + bs_2) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} (a^2 s_i^2(k) + 2abs_1(k)s_2(k) + b^2 s_2^2(k)) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} a^2 s_i^2(k) + \frac{1}{n} \sum_{k=1}^{n} 2abs_1(k)s_2(k) + \frac{1}{n} \sum_{k=1}^{n} b^2 s_2^2(k) \]
\[ = a^2 \sum_{k=1}^{n} s_i^2(k) + 2ab \sum_{k=1}^{n} \frac{s_1(k)s_2(k)}{n} + b^2 \sum_{k=1}^{n} \frac{s_2^2(k)}{n} \]
\[ c_{x}(1,1) = a^2 \sigma_{s1} + 2ab \sigma(s_1s_2) + b^2 \sigma_{s2} \]  

A.2.3

\[ c_{x}(1,2) = \frac{1}{n} \sum_{k=1}^{n} x_1 x_2 = \frac{1}{n} \sum_{k=1}^{n} (as_i + bs_2)(cs_i + ds_2) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} (acs_i^2(k) + (ad + bc)s_1(k)s_2(k) + bds_2^2(k)) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} acs_i^2(k) + \frac{1}{n} \sum_{k=1}^{n} (ad + bc)s_1(k)s_2(k) + \frac{1}{n} \sum_{k=1}^{n} bds_2^2(k) \]
\[ = ac \sum_{k=1}^{n} \frac{s_i^2(k)}{n} + (ad + bc) \sum_{k=1}^{n} \frac{s_1(k)s_2(k)}{n} + bd \sum_{k=1}^{n} \frac{s_2^2(k)}{n} \]
\[ c_{x}(1,2) = ac \sigma_{s1} + (ad + bc) \sigma(s_1s_2) + bd \sigma_{s2} \]  

A.2.4

\[ c_{x}(2,2) = \frac{1}{n} \sum_{k=1}^{n} x_2 x_2 = \frac{1}{n} \sum_{k=1}^{n} (cs_i + ds_2)(cs_i + ds_2) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} (c^2 s_i^2(k) + 2cDs_1(k)s_2(k) + d^2 s_2^2(k)) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} c^2 s_i^2(k) + \frac{1}{n} \sum_{k=1}^{n} 2cDs_1(k)s_2(k) + \frac{1}{n} \sum_{k=1}^{n} d^2 s_2^2(k) \]
\[ = c^2 \sum_{k=1}^{n} \frac{s_i^2(k)}{n} + 2cd \sum_{k=1}^{n} \frac{s_1(k)s_2(k)}{n} + d^2 \sum_{k=1}^{n} \frac{s_2^2(k)}{n} \]
\[ c_s(2,2) = c^2 \sigma_{s_1} + 2cd\sigma(s_{s_1} s_{s_2}) + d^2 \sigma_{s_2} \]  

The covariance matrix is:

\[
C_s = \begin{bmatrix}
  a^2 \sigma_{s_1} + 2ab\sigma(s_{s_1} s_{s_2}) + b^2 \sigma_{s_2} & ac\sigma_{s_1} + (ad + bc)\sigma(s_{s_1} s_{s_2}) + bd\sigma_{s_2} \\
  ac\sigma_{s_1} + (ad + bc)\sigma(s_{s_1} s_{s_2}) + bd\sigma_{s_2} & c^2 \sigma_{s_1} + 2cd\sigma(s_{s_1} s_{s_2}) + d^2 \sigma_{s_2}
\end{bmatrix}
\]

The smallest eigenvalue for the \( H_0 \) hypothesis is:

\[
\lambda_i = \frac{1}{2}\left[ (a^2 + c^2)\sigma_{s_1} + 2(ab + cd)\sigma(s_{s_1} s_{s_2}) + (b^2 + d^2)\sigma_{s_2} \right] \\
- \frac{1}{2} \sqrt{\left[ (a^2 - c^2)\sigma_{s_1} + 2(ab - cd)\sigma(s_{s_1} s_{s_2}) + (b^2 - d^2)\sigma_{s_2} \right]^2 + 4(ac\sigma_{s_1} + (ad + bc)\sigma(s_{s_1} s_{s_2}) + bd\sigma_{s_2})^2}
\]

For the \( H_0 \) hypothesis the covariance matrix elements are:

\[
c_s(1,1) = \frac{1}{n} \sum_{k=1}^{n} x_k x_1 = \frac{1}{n} \sum_{k=1}^{n} (bs_{s_2})(bs_{s_2}) \\
= \frac{1}{n} \sum_{k=1}^{n} b^2 s_{s_2}^2 (k) \\
= b^2 \frac{1}{n} \sum_{k=1}^{n} s_{s_2}^2 (k)
\]

\[ c_s(1,1) = b^2 \sigma_{s_2} \]  

\[
c_s(1,2) = \frac{1}{n} \sum_{k=1}^{n} x_k x_2 = \frac{1}{n} \sum_{k=1}^{n} (bs_{s_2})(ds_{s_2}) \\
= \frac{1}{n} \sum_{k=1}^{n} bds_{s_2}^2 (k) \\
= bd \frac{1}{n} \sum_{k=1}^{n} s_{s_2}^2 (k)
\]

\[ c_s(1,2) = bd \sigma_{s_2} \]  

\[ \text{Page 126} \]
The covariance matrix is:

\[
C_x = \begin{bmatrix}
b^2 \sigma_{s1} & bd \sigma_{s1} \\
bd \sigma_{s1} & d^2 \sigma_{s1}
\end{bmatrix}
\]  
A.2.11

The smallest eigenvalue for the H_0 hypothesis is:

\[
\lambda_i = 0.5(b^2 + d^2)\sigma_{s1} - 0.5\sqrt{[(b^2 + d^2)\sigma_{s1}]^2} = 0
\]  
A.2.12

A.3 Delayed Mixing Case

The second case reflects a more realistic case, and allows for a time difference of arrival between the two sensors. Similar to the previous model derivation, let \( s_1 \) be the signal of interest with variance \( \sigma_{s1} \), and \( s_2 \) be an interfering signal with variance \( \sigma_{s2} \). For the simulations and experiments in this dissertation, the interfering signal, \( s_2 \), is assumed to be white, Gaussian noise, which is short-term stationary. Over small intervals, the statistics do not change, and the variance of the delayed signal, \( s_2(k+\tau_2) \), can assumed to be equal to \( \sigma_{s2} \). A superscript \( \tau \) denotes a delayed signal.

A.3.1 One Signal and One Interferer

The first case examined for delayed mixing is the case with one interferer and one signal of interest. No additional, uncorrelated background noise is included in this model. The two hypotheses are
$H_0 : \begin{align*}
x_1(k) &= a s_1(k) + b s_2(k) \\
x_2(k) &= c s_1(k + \tau_1) + d s_2(k + \tau_2)
\end{align*} \hspace{1cm} A.3.1$

$H_1 : \begin{align*}
x_1(k) &= b s_2(k) \\
x_2(k) &= d s_2(k + \tau_2)
\end{align*}$

where $k$, $a$, $b$, $c$, and $d$ are the same as in the previous section, and $\tau_1$ and $\tau_2$ are the time delays for the signal of interest and interferer respectively. As before, the covariance matrix is constructed element by element using Equation A.2.2.

For the $H_0$ hypothesis the covariance matrix elements are:

$$c_s(1,1) = \frac{1}{n} \sum_{k=1}^{n} x_1 x_1 = \frac{1}{n} \sum_{k=1}^{n} (a s_1 + b s_2)(a s_1 + b s_2)$$

$$= \frac{1}{n} \sum_{k=1}^{n} (a^2 s_1^2 + 2 a b s_1 s_2 + b^2 s_2^2)$$

$$= \frac{1}{n} \sum_{k=1}^{n} a^2 s_1^2 + \frac{1}{n} \sum_{k=1}^{n} 2 a b s_1 s_2 + \frac{1}{n} \sum_{k=1}^{n} b^2 s_2^2$$

$$= a^2 \frac{1}{n} \sum_{k=1}^{n} s_1^2 + 2 a b \frac{1}{n} \sum_{k=1}^{n} s_1 s_2 + b^2 \frac{1}{n} \sum_{k=1}^{n} s_2^2$$

$$c_s(1,1) = a^2 \sigma_1 + 2 a b \sigma_1 \sigma_2 + b^2 \sigma_2 \hspace{1cm} A.3.2$$

$$c_s(1,2) = \frac{1}{n} \sum_{k=1}^{n} x_1 x_2 = \frac{1}{n} \sum_{k=1}^{n} (a s_1 + b s_2)(c s_1 + d s_2)$$

$$= \frac{1}{n} \sum_{k=1}^{n} (a c s_1 s_1 + a d s_1 s_2 + b c s_1 s_2 + b d s_2 s_2)$$

$$= \frac{1}{n} \sum_{k=1}^{n} a c s_1 s_1 + \frac{1}{n} \sum_{k=1}^{n} a d s_1 s_2 + \frac{1}{n} \sum_{k=1}^{n} b c s_1 s_2 + \frac{1}{n} \sum_{k=1}^{n} b d s_2 s_2$$

$$= a c \frac{1}{n} \sum_{k=1}^{n} s_1 s_1 + a d \frac{1}{n} \sum_{k=1}^{n} s_1 s_2 + b c \frac{1}{n} \sum_{k=1}^{n} s_1 s_2 + b d \frac{1}{n} \sum_{k=1}^{n} s_2 s_2$$

$$c_s(1,2) = a c \sigma_1 \sigma_1 + a d \sigma_1 \sigma_2 + b c \sigma_1 \sigma_2 + b d \sigma_2 \sigma_2 \hspace{1cm} A.3.3$$
\[ c_x(2,2) = \frac{1}{n} \sum_{k=1}^{n} x_k x_k = \frac{1}{n} \sum_{k=1}^{n} (cs_i^r + ds_i^z)(cs_i^r + ds_i^z) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} (c^2 s_i^r s_i^r + 2c d s_i^r s_i^z + d^2 s_i^z s_i^z) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} c^2 s_i^r s_i^r + \frac{1}{n} \sum_{k=1}^{n} 2 c d s_i^r s_i^z + \frac{1}{n} \sum_{k=1}^{n} d^2 s_i^z s_i^z \]
\[ = c^2 \frac{\sum_{k=1}^{n} s_i^r s_i^r}{n} + 2cd \frac{\sum_{k=1}^{n} s_i^r s_i^z}{n} + d^2 \frac{\sum_{k=1}^{n} s_i^z s_i^z}{n} \]
\[ c_x(2,2) = c^2 \sigma(s_i^r s_i^r) + 2cd \sigma(s_i^r s_i^z) + d^2 \sigma s_i^z \]

The smallest eigenvalue for the H_0 hypothesis is:
\[ \lambda_i = 0.5 \left[ a^2 \sigma_{ii} + c^2 \sigma(s_i^r s_i^r) + 2ab \sigma(s_i^r s_i^z) + 2cd \sigma(s_i^r s_i^z) + (b^2 + d^2) \sigma s_i^z \right] \]
\[ - 0.5 \sqrt{ \left[ a^2 \sigma_{ii} - c^2 \sigma(s_i^r s_i^r) + 2ab \sigma(s_i^r s_i^z) - 2cd \sigma(s_i^r s_i^z) + (b^2 - d^2) \sigma s_i^z \right]^2 + 4 \left[ ac \sigma(s_i^r s_i^z) + ad \sigma(s_i^r s_i^z) + bc \sigma(s_i^r s_i^z) + bd \sigma(s_i^r s_i^z) \right]^2} \]

For the H_0 hypothesis the covariance matrix elements are:
\[ c_x(1,1) = \frac{1}{n} \sum_{k=1}^{n} x_k x_k = \frac{1}{n} \sum_{k=1}^{n} (bs_i^r)(bs_i^z) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} b^2 s_i^z \]
\[ = b^2 \frac{\sum_{k=1}^{n} s_i^z}{n} \]
\[ c_x(1,1) = b^2 \sigma s_i^z \]

\[ c_x(1,2) = \frac{1}{n} \sum_{k=1}^{n} x_k x_k = \frac{1}{n} \sum_{k=1}^{n} (bs_i^z)(ds_i^z) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} bds_i^r s_i^z \]
\[ = bd \frac{\sum_{k=1}^{n} s_i^r s_i^z}{n} \]
\[ c_x(1,2) = bd\sigma(s_1s_2^\tau) \]  
\[ c_x(2,2) = \frac{1}{n} \sum_{i=1}^{n} x_2 x_2 = \frac{1}{n} \sum_{i=1}^{n} (ds_2^\tau)(ds_2^\tau) = \frac{1}{n} \sum_{i=1}^{n} d^2 s_2^\tau s_2^\tau = d^2 \sum_{i=1}^{n} \frac{s_2^\tau s_2^\tau}{n} \]  
\[ c_x(2,2) = d^2\sigma_{s_2} \]  

The smallest eigenvalue for the H\(_0\) hypothesis is:
\[ \lambda_1 = 0.5(b^2 + d^2)\sigma_{s_2} - 0.5\sqrt{(b^2 - d^2)^2 + 4b^2d^2\sigma(s_2s_2^\tau)^2} \]  

\section*{A.3.2 One Signal, One Interferer, and Noise}

The last model examined is the case of one interferer and one signal of interest in the presence of uncorrelated background noise. The uncorrelated noise signals are denoted \(n_1\) and \(n_2\) and have variances \(\sigma_{n_1}\) and \(\sigma_{n_2}\) respectively. The two hypotheses are

\[ H_0 : \begin{align*}
    x_1(k) &= as_1(k) + bs_2(k) + n_1(k) \\
    x_2(k) &= cs_1(k + \tau_1) + ds_2(k + \tau_2) + n_2(k)
\end{align*} \]  
\[ H_1 : \begin{align*}
    x_1(k) &= bs_2(k) + n_1(k) \\
    x_2(k) &= ds_2(k + \tau_2) + n_2(k)
\end{align*} \]  

where \(k, a, b, c, d\) are the same as in the previous section, and \(\tau_1\) and \(\tau_2\) are the time delays for the signal of interest and interferer respectively. As before, the covariance matrix is constructed element by element using Equation A.2.2.

For the \(H_0\) hypothesis the covariance matrix elements are:
\[ c_x(1,1) = \frac{1}{n} \sum_{k=1}^{n} x_1 x_1 = \frac{1}{n} \sum_{k=1}^{n} (as_1 + bs_2 + n_1)(as_1 + bs_2 + n_1) \]
\[ c_x(1,1) = \frac{1}{n} \sum_{k=1}^{n} (a^2 s_1^2 + b^2 s_2^2 + n_1^2 + 2ab s_1 s_2 + 2as n_1 + 2b s_2 n_1) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} a^2 s_1^2 + \frac{1}{n} \sum_{k=1}^{n} b^2 s_2^2 + \frac{1}{n} \sum_{k=1}^{n} 2ab s_1 s_2 + \frac{1}{n} \sum_{k=1}^{n} 2as n_1 + \frac{1}{n} \sum_{k=1}^{n} 2b s_2 n_1 \]
\[ = a^2 \sum_{k=1}^{n} \frac{s_1^2}{n} + b^2 \sum_{k=1}^{n} \frac{s_2^2}{n} + 2ab \sum_{k=1}^{n} \frac{s_1 s_2}{n} + 2a \sum_{k=1}^{n} \frac{s_1 n_1}{n} + 2b \sum_{k=1}^{n} \frac{s_2 n_1}{n} \]
\[ c_x(1,1) = a^2 \sigma_{s_1} + b^2 \sigma_{s_2} + \sigma_{n_1} + 2ab \sigma(s_1 s_2) + 2a \sigma(s_1 n_1) + 2b \sigma(s_2 n_1) \]  
\text{A.3.11}

\[ c_x(1,2) = \frac{1}{n} \sum_{k=1}^{n} x_1 x_2 = \frac{1}{n} \sum_{k=1}^{n} (as_1 + bs_2 + n_1)(cs_1^2 + ds_2^2 + n_2) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} \left[ acs_1 s_1^2 + ads_1 s_2^2 + as_1 n_1 + bc s_1^2 s_2 + bds_2 s_2^2 + bs_2 n_2 + cs_1^2 n_1 + ds_2^2 n_1 + n_1 n_2 \right] \]
\[ = ac \sum_{k=1}^{n} \frac{s_1^2 s_1^2}{n} + ad \sum_{k=1}^{n} \frac{s_1 s_2^2}{n} + a \sum_{k=1}^{n} \frac{s_1 n_1}{2} + bc \sum_{k=1}^{n} \frac{s_1^2 s_2}{2} + bd \sum_{k=1}^{n} \frac{s_2 s_2^2}{2} \]
\[ + b \sum_{k=1}^{n} \frac{s_1 n_2}{n} + c \sum_{k=1}^{n} \frac{s_1^2 n_1}{n} + d \sum_{k=1}^{n} \frac{s_1 s_2^2}{n} + \sum_{k=1}^{n} \frac{n_1 n_2}{n} \]
\[ c_x(1,2) = ac \sigma(s_1 s_1^2) + ad \sigma(s_1 s_2^2) + a \sigma(s_1 n_1) + bc \sigma(s_1^2 s_2) + bd \sigma(s_2 s_2^2) + b \sigma(s_2 n_2) + c \sigma(s_1^2 n_1) + d \sigma(s_1 s_2^2) + \sigma(n_1 n_2) \]  
\text{A.3.12}

\[ c_x(2,2) = \frac{1}{n} \sum_{k=1}^{n} x_2 x_2 = \frac{1}{n} \sum_{k=1}^{n} (cs_1^2 + ds_2^2 + n_1)(cs_1^2 + ds_2^2 + n_2) \]
\[ = \frac{1}{n} \sum_{k=1}^{n} \left[ c^2 s_1^2 s_1^2 + d^2 s_2^2 s_2^2 + n_2 n_2 + 2c d s_1^2 s_2^2 + 2c s_1^2 n_2 + 2d s_2^2 n_2 \right] \]
\[ = c^2 \sum_{k=1}^{n} \frac{s_1^2 s_1^2}{n} + d^2 \sum_{k=1}^{n} \frac{s_2^2 s_2^2}{n} + \sum_{k=1}^{n} \frac{n_2 n_2}{n} + 2cd \sum_{k=1}^{n} \frac{s_1^2 s_2^2}{n} + 2c \sum_{k=1}^{n} \frac{s_1^2 n_2}{n} + 2d \sum_{k=1}^{n} \frac{s_2^2 n_2}{n} \]
\[ c_x(2,2) = c^2 \sigma(s_1^2 s_1^2) + d^2 \sigma(s_2^2 s_2^2) + 2cd \sigma(s_1^2 s_2^2) + 2c \sigma(s_1^2 n_2) + 2d \sigma(s_2^2 n_2) \]  
\text{A.3.13}
The smallest eigenvalue for the $H_0$ hypothesis is:

$$\lambda_1 = 0.5 \left[ a^2 \sigma_{s_1}^2 + c^2 \sigma(s_i s_i) + (b^2 + d^2) \sigma_{s_2} + \sigma_{s_1} + \sigma_{s_2} + 2 n a \sigma(s_i n_i) + 2 b \sigma(s_i n_i) + 2 c \sigma(n_i n_i) + 2 d \sigma(s_i s_i) \right]$$

$$- 0.5 \left[ a^2 \sigma_{s_1}^2 - c^2 \sigma(s_i s_i) + (b^2 - d^2) \sigma_{s_2} - \sigma_{s_1} - \sigma_{s_2} + 2 a b \sigma(s_i s_i) - 2 c d \sigma(s_i s_i) \right]$$

$$+ 4 \left[ a c \sigma(s_i s_i) + a d \sigma(s_i n_i) + b c \sigma(n_i n_i) + b d \sigma(s_i s_i) + a \sigma(n_i n_i) + d \sigma(n_i n_i) \right]$$

A.3.14

For the $H_0$ hypothesis the covariance matrix elements are:

$$c_{x, (1,1)} = \frac{1}{n} \sum_{k=1}^{n} x_k x_k = \frac{1}{n} \sum_{k=1}^{n} (b s_2 + n_1) (b s_2 + n_1)$$

$$= \frac{1}{n} \sum_{k=1}^{n} b^2 s_2^2 + 2 b s_2 n_1 + n_1^2$$

$$= b^2 \sum_{k=1}^{n} \frac{s_2^2}{n} + 2 b \sum_{k=1}^{n} \frac{s_2 n_1}{n} + \sum_{k=1}^{n} \frac{n_1^2}{n}$$

$$c_{x, (1,1)} = b^2 \sigma_{s_2} + 2 b \sigma(s_2 n_1) + \sigma_{s_1}$$

A.3.15

$$c_{x, (1,2)} = \frac{1}{n} \sum_{k=1}^{n} x_k x_2 = \frac{1}{n} \sum_{k=1}^{n} (b s_2 + n_1) (d s_2 + n_2)$$

$$= \frac{1}{n} \sum_{k=1}^{n} b d s_2 s_2 + b s_2 n_2 + d s_2 + n_1 n_2$$

$$= b d \sum_{k=1}^{n} \frac{s_2 s_2}{n} + b \sum_{k=1}^{n} \frac{s_2 n_2}{n} + d \sum_{k=1}^{n} \frac{s_2 n_2}{n} + \sum_{k=1}^{n} \frac{n_1 n_2}{n}$$

$$c_{x, (1,2)} = b d \sigma(s_2 s_2) + b \sigma(s_2 n_2) + d \sigma(s_2 n_2) + \sigma(n_1 n_2)$$

A.3.16
\[ c_{x}(2,2) = \frac{1}{n} \sum_{i=1}^{n} x_{2i}x_{2i} = \frac{1}{n} \sum_{i=1}^{n} (ds_{2} + n_{2})(ds_{2} + n_{2}) \]
\[ = \frac{1}{n} \sum_{i=1}^{n} d^{2}s_{2}^{2}s_{2}^{2} + 2d_{2}s_{2}n_{2} + n_{2}^{2} \]
\[ = d^{2} \sum_{i=1}^{n} s_{2}^{2}s_{2}^{2} + 2d \sum_{i=1}^{n} s_{2}n_{2} + \sum_{i=1}^{n} n_{2}^{2} \]
\[ c_{x}(2,2) = d^{2}\sigma_{s_{2}} + 2d\sigma(s^{2}_{2}n_{2}) + \sigma_{n_{2}} \]

The smallest eigenvalue for the \( H_{0} \) hypothesis is:

\[ \lambda_{1} = 0.5 \left[ (b^{2} + d^{2})\sigma_{s_{2}} + \sigma_{n_{1}} + \sigma_{n_{2}} + 2b\sigma(s_{n_{1}}) + 2d\sigma(s^{2}_{n_{2}}) \right] \]
\[ - 0.5 \sqrt{ \left[ (b^{2} - d^{2})\sigma_{s_{2}} + \sigma_{n_{1}} - \sigma_{n_{2}} + 2b\sigma(s_{n_{1}}) - 2d\sigma(s^{2}_{n_{2}}) \right]^{2} \]
\[ + 4bd\sigma(s_{2}s_{2}) + b\sigma(s_{2}n_{2}) + d\sigma(s^{2}_{n_{2}}) + \sigma(n_{1})n_{1}^{2} \]