Circuits and Systems
Mekelweg 4,
2628 CD Delft
The Netherlands
https://sps.ewi.tudelft.nl/

SPS-2024-5799996

M.Sc. Thesis

Dopple Group Chat

Long Xin B.Sc.

Abstract

In response to the echo and noise interference problems faced by multi-user full-duplex voice communication scenarios, this work proposes a complete solution from signal modeling to beamforming optimization. First, by converting the Bluetooth channel into a virtual acoustic channel, a comprehensive acoustic transmission model covering single-source and multi-source scenarios is established. On this basis, a variety of beamforming and filtering strategies are discussed, and finally, the Multichannel Wiener Filter (MWF) beamforming scheme is proposed. At the same time, for the time-varying spatial domain, the update of the covariance matrix combined with Voice Activity Detector (VAD) is discussed. The simulation shows that the MWF scheme has good robustness and adaptability under different Bluetooth channel latency and low Signal-to-Interference Ratio (SIR) environments; when the user end is equipped with a sufficient number of microphones or there are many bystanders, the system performance is further improved.

Dopple Group Chat

Dereverberation & Denoise in multi-channel full-duplex communication systems

---

Thesis

submitted in partial fulfillment of the
requirements for the degree of

Master of Science

in

Electrical Engineering

by

Long Xin B.Sc.
born in Lu'An, China

This work was performed in:

Circuits and Systems Group
Department of Microelectronics
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology

Delft University of Technology

Delft University of Technology
Department of
Microelectronics

The undersigned hereby certify that they have read and recommend to the Faculty of Electrical Engineering, Mathematics and Computer Science for acceptance a thesis entitled **"Dopple Group Chat"** by **Long Xin B.Sc.** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: January 9, 2025

Chairman: _____
prof.dr.ir. R.C. Hendriks

Advisor: _____
ir. G. Bologni

Committee Members: _____
prof.dr. J.A. Martinez Castaneda

_____
dr.ir. Jaap Haartsen

Abstract

In response to the echo and noise interference problems faced by multi-user full-duplex voice communication scenarios, this work proposes a complete solution from signal modeling to beamforming optimization. First, by converting the Bluetooth channel into a virtual acoustic channel, a comprehensive acoustic transmission model covering single-source and multi-source scenarios is established. On this basis, a variety of beamforming and filtering strategies are discussed, and finally, the Multichannel Wiener Filter (MWF) beamforming scheme is proposed. At the same time, for the time-varying spatial domain, the update of the covariance matrix combined with Voice Activity Detector (VAD) is discussed. The simulation shows that the MWF scheme has good robustness and adaptability under different Bluetooth channel latency and low Signal-to-Interference Ratio (SIR) environments; when the user end is equipped with a sufficient number of microphones or there are many bystanders, the system performance is further improved.

Acknowledgments

I write these words with excitement as they mark the end of my student life in Delft, and I am about to move on to a new phase of life.

Looking back at this point, I have experienced many long and difficult times, and I had to make many important choices. In the end, I have reaped great rewards. For this, I would like to thank those who supported and encouraged me.

First of all, I would like to thank my supervisor, Dr.ir. R.C. Hendriks, and my daily supervisor, PhD candidate G. Bologni, for their help. They provided a lot of guidance for my work. Without their assistance, I would have lost my direction when I was confused. In addition, I am grateful for their critical questions and valuable suggestions, which helped me to gain a deeper understanding of the problem and explore other possibilities. I would also like to thank Dopple and Dr.ir. Jaap C. Haartsen, provided me with an interesting topic for my master's thesis, allowing me to apply the knowledge I had learned in practice. They were always willing to answer my questions.

Secondly, I would like to thank my family, who are always willing to understand and support me. Their encouragement is my driving force. Without their financial and spiritual support, I would not be where I am today. I would also like to thank all my friends in Delft who have shared wonderful times with me. The time I spent with them is a memory I will always cherish. Studying, chatting, partying, traveling, and even adventuring with them made my time abroad colorful.

Finally, I wish them all good health and a happy life. I look forward to seeing them again one day.

Long Xin B.Sc.
Delft, The Netherlands

Nomenclature

Linear Algebra

$(\cdot)^{-1}$	Inverse
$(\cdot)^{\dagger}$	Pseudoinverse
$(\cdot)^{\top}$	Transpose
$(\cdot)^{H}$	Hermitian transpose
$\mathbb{R}^n$	Real vector space of n-dimensional vectors
$\mathbb{R}^{n \times n}$	Real matrices space of n by n matrices
$\mathbf{1}$	All-ones vector
$\mathbf{I}$	Identity matrix
$\text{diag}(\mathbf{a})$	Diagonal matrix with $\mathbf{a}$ on the diagonal

Mathematical Objects

a	Scalar
$\mathbf{a}$	Vector
$\mathbf{A}$	Matrix

Other symbols

$\hat{(\cdot)}$	Estimated value

Acronyms

ATFs Acoustic Transfer Functions. 26, 27, 32, 33, 36, 49

BEM Boundary Element Method. 11

BER Bit Error Rate. 11, 12

DFT Discrete Fourier transform. 20

DGC Dopple Group Chat. 1, 2, 17, 25, 35, 40, 56, 57

DOA Direction of Arrival. 5, 6

FDFD Finite-Difference Time-Domain. 11

FEM Finite Element Method. 11

FFT Fast Fourier Transform. 20, 40

GEVD Generalized Eigenvalue Decomposition. 29, 30

GMMs Gaussian Mixture Models. 36

GSC Generalized Sidelobe Canceller. 5, 6

IETF Internet Engineering Task Force. 36

LCMV Linearly Constrained Minimum Variance. 5, 26, 30–32, 36, 37, 50, 56

MIC Microphone. x, 18

MMSE Minimum Mean Square Error. 6, 33, 34, 37

MSE Mean Square Error. 39

MVDR Minimum Variance Distortionless Response Filter. 5, 6, 26, 28, 30, 31, 34, 36, 37, 44, 45, 55, 56

MWF Multichannel Wiener Filter. i, iii, 6, 26, 33, 34, 37, 44–46, 50, 55–57

NMSE Normalized Mean Square Error. 39, 50

RIR Room Impulse Response. 8, 14

RTFs Relative Acoustic Transfer Functions. 26–30, 32, 33, 36, 56

SCSE Single-Channel Speech Enhancement. 4

SIR Signal-to-Interference Ratio. i, iii, 41, 42, 44, 46, 49, 52, 57

SNR Signal-to-Noise Ratio. 5, 6, 11, 12, 14, 16, 26, 27, 32, 38, 41

SPL sound pressure level. 12, 14, 46

STFT Short-Time Fourier Transform. 20, 24, 35, 40

STOI Short-Time Objective Intelligibility. 38, 54

VAD Voice Activity Detector. i, iii, 35–37, 53, 55–58

VPU Voice Processing Unit. x, 18

W3C World Wide Web Consortium. 36

WebRTC Web Real-Time Communication. 36

Contents

List of Figures

1 Introduction

Voice communication plays an integral role in our daily life and work communications, allowing people to exchange information with each other. During voice communication, various factors, such as interference from other people's speech, ambient noise, and internal noise from communication equipment, will inevitably affect the clarity of the voice signal. The combined influence of these interfering elements causes the received voice signal to move away from its original state and become noisy. This not only affects the performance of the voice processing system but also reduces the clarity of the voice and seriously hinders communication efficiency.

In addition to the negative impact of interference and noise on voice communication systems, echo is another key factor that degrades call quality in video conferencing, calls using hands-free devices, and other situations. When sound comes from the talker and travels directly or through multiple reflections, it is superimposed with the original dialogue signal. It enters the microphone, causing the speaker to hear back his or her previous words. This phenomenon creates the so-called voice reverberation. If this delay exceeds 150 milliseconds, it will seriously affect the quality of communication and reduce the smoothness of the conversation [2]. Therefore, it is important to take measures to reduce the negative impact of these factors in the communication system on speech signals, which will help improve the overall performance of the speech processing system.

Dopple has developed Dopple Group Chat (DGC), a new multi-person full-duplex communication protocol that runs on top of Bluetooth radios. DGC allows multiple users to communicate directly with each other over relatively short distances without the intervention of a smartphone, tablet, or laptop. Areas of application include sports (e.g.cycling, jogging, and fitness centers), people with hearing impairments, and people working in noisy environments. In addition, there may be a transparent mode protocol on DGC-based products. The transparent mode allows external sounds to enter the headphones, allowing users to hear the surrounding sounds, for those who do not know about communication protocols, full-duplex communication protocols mean that voice data can be transmitted in both directions at the same time. A common example is telephone communication, where users can speak and hear each other at the same time.

Because there are multiple users (multiple microphones and speakers), a full-duplex working mode is involved, and there is an acoustic propagation path between users at a relatively close distance in addition to the communication protocol. The acoustic coupling will occur between the sending end and the receiving end, causing an echo effect. Based on the above statement, we need to propose a reliable audio processing algorithm to solve the echo effect (dereverberation), improve the overall performance and clarity of the DGC system, and ensure high-quality communication.

1.1 Research Statement

This thesis project aims to find the answer to the following questions:

- How to model the acoustic structure in the network of DGC?
- How to do dereverberation and reduce the noise in the DGC system?
- When the system becomes more complex (e.g. the users increase), how does the system evolve? (The number of users considered is among 3 persons and 10 persons.)

1.2 Thesis Structure

This topic is explored progressively. We begin with a relatively simple scenario, gradually increasing the complexity of the situations as we try to solve the more difficult problems.

The thesis is organized in the following ways.

- chapter 2 - Background: First, we give a brief literature review about related work. Then we give all practical background information to the acoustic structure. First, we introduce the principle of room acoustics simulation showing how the acoustic transfer function is simulated. Then we will give the idea of Bluetooth channel transmission and Bluetooth channel transfer function approximation. In the end, we investigate the microphone's self-noise volume, the speech volume, and the speech volume attenuation in space.
- chapter 3 - Problem Formulation: we define the two problems that are going to be discussed gradually. In the single-speaker and multiple-speaker cases, we gradually extend the signal model from the scalar form to the vector form with multiple bystanders involved.
- chapter 4 - Proposed Methods: we solve the single speaker case by using the MVDR beamformer. An LCMV beamformer is used to generalize the beamformer to multiple speaker cases. However, due to the LCMV's limitations, the MWF beamformer is used. In addition, we discuss the noise covariance matrix estimation method.
- chapter 5 - Simulation and Result: we give simulation evaluation metrics and simulation setup at first, then we implement the solution mentioned before for two cases. The ideal VAD model was then embedded into the algorithm for the single-speaker case to test how much improvement it could bring.
- chapter 6 - Conclusion and Future Work: we conclude and propose the potential future work directions.

2 Background

In this chapter, we provide a brief literature review and an overview of room acoustics simulation, modeling of echo problems, microphone noise characteristics, and speech signal analysis. We compare the advantages and disadvantages of different acoustic modeling methods. Then, we explore the modeling of Bluetooth channels. The analysis of microphone noise studies the impact of different sources and their main contributors. Methods for quantizing noise are discussed. Finally, the volume and spatial attenuation of speech signals are studied. In summary, this chapter estimates the variables required in the experiment or provides a theoretical basis for their calculation.

2.1 Literature Review

This section will review the literature on related speech enhancement work. This section is mainly divided into spectral enhancement methods and spatial processing methods.

With the advancement of technology, people are increasingly interested in mobile voice devices. However, with increasingly complex usage scenarios, people's demand for robust signals is also growing. A mainstream approach is to suppress noise signals and improve voice quality, which involves single-microphone or multi-microphone noise reduction solutions. Although some special scenarios may only use a single microphone, in general, multi-microphone technology can provide better performance.

2.1.1 Spectral Enhancement

Spectral enhancement refers to modifying the short-time spectrum of the received speech signal to enhance the speech signal. Most spectral enhancements are used in techniques such as single-channel noise reduction. Single-channel noise reduction plays an important role in many topics, both as a standalone module and as a post-filter to improve the performance of a multi-microphone system. In [3], R.C. Hendriks provides a comprehensive overview of the single-channel noise reduction topic.

The Wiener filter is an optimal filter based on minimizing the mean square error [4]. Assuming that the speech and noise follow a normal distribution and are uncorrelated, the gain function of the Wiener filter can be represented by the power spectrum function of the clean speech and the power spectrum function of the noise as shown below,

$$H_{\text{wiener}}(\omega) = \frac{P_s(\omega)}{P_s(\omega) + P_d(\omega)}. \tag{2.1}$$

A disadvantage of the Wiener filter is that its gain is fixed at all frequencies.

Spectral subtraction is one of the earliest algorithms proposed in history to enhance single-channel speech. In this method, the noise spectrum is estimated during speech

pauses and subtracted from the noisy audio spectrum to estimate the interested speech. The disadvantage of this method is the presence of processing distortion, known as residual noise.

Some beneficial variants have been proposed [5], including a multi-band spectral subtraction method for real-world colored noise with uniform frequency spacing, which provides additional control for noise subtraction by customizing the attenuation degree of each band, and the results show excellent performance.
In [6], an iterative method was proposed, which uses the output of the enhanced speech as the input signal for the next iterative process. After the spectral subtraction process, the output signal is used as the input signal for the next iterative process, and the residual noise is re-estimated for the next iteration. Subsequent studies have shown that the performance of this method is affected by the number of iterations, and the greater the number of iterations, the better the effect [7, 8].
The perceptual characteristics of the human ear can also be taken into account. It is also a feasible method to improve the clarity of the speech signal by attenuating the noise that is inaudible due to the masking effect of the human ear, which is the so-called perceptual characteristic-based spectral subtraction. When the noise is below the perceptual threshold, the human ear can tolerate the additional noise. In [9], the algorithm adjusts the parameters according to the masking characteristics. [10] gives the modeling of the masking effect.

In recent years, in addition to traditional methods, machine learning-based techniques have also performed well in this field. Generally, Single-Channel Speech Enhancement (SCSE) algorithms can be divided into two categories: unsupervised algorithms and supervised algorithms. In unsupervised SCSE methods, statistical models are used to process noisy speech signals to estimate clean speech, requiring no prior knowledge of the noise source or target speech source. On the other hand, supervised SCSE algorithms are trained with clean speech and noisy speech data, and the model learns the mapping to recover clean speech from noisy signals. Since this part is not involved in the work, we only provide some related reviews. [11] gives a comprehensive and detailed overview of supervised SCSE algorithms from the two aspects of speech intelligibility and quality. Specifically, Krishnamoorthy and Prasanna gave a comprehensive review paper on supervised speech enhancement algorithms [12].

In addition to single-channel noise reduction tasks, there are also some studies on multi-channel spectral enhancement techniques that expect to constructively combine the correlated information between different channels to improve signal quality. As early as 1977, Allen et al. conducted relevant research [13]. They used sub-band technology to perform spectrum processing on a dual-microphone system. In each frequency band, they adjusted the phase between the signals to align them and eliminate the time delay in the "coherent part" of the two microphone signals. These phase-corrected signals are then added, a process known as in-phase superposition and band addition. Subsequently, the gain of each frequency band is adjusted based on the normalized cross-correlation function. The effect of this method is to reduce the energy in the low "coherence" frequency bands (these frequency bands contain mainly reverberation), while retaining or slightly enhancing the energy in the high "coherence" frequency bands (these frequency bands usually contain significant direct signal com-

ponents and early reflections). Several subsequent studies investigated [14] the effect of this technique on speech signals recorded in rooms with long reverberation times (1.3 seconds). However, Allen's spectral enhancement technique did not show very positive effects. Bloom improved on this technique by employing a narrower analysis band and generating a gain function based on an estimate of the time-varying amplitude-squared coherence function while smoothing based on critical bands in the frequency domain [15]. However, his research also showed that these modifications did not necessarily improve average recognition scores.

2.1.2 Spatial Processing

Multi-microphone-based spatial processing techniques can be used for speech enhancement tasks. Multi-microphone systems use spatial information to selectively enhance signals from specific directions. Most related studies focus on speech enhancement and robustness in noisy environments but usually do not address dereverberation capabilities, which makes comparisons difficult. A thorough overview can be found in Van Trees' Optimal Array Processing [16]. McCowan has authored an educational tutorial that concentrates on enhancing speech signals [17]. A prestigious article [18] also provides a good summary of this topic.

Based on the far-field assumption, acoustic plane waves entering a microphone array will typically arrive at slightly different times. Depending on the spacing and geometry of the microphone array, the component frequencies, and Direction of Arrival (DOA), the signals received by the sensors can be enhanced or canceled by constructive combination. Thus, by linearly combining the received signals, the array can produce a directional response that favors certain directions, a technique known as beamforming.

An easily implemented beamforming is delay-and-sum beamforming, which compensates for the delay of the signal in each channel to maximize the response in any desired direction. It is easy to implement and its directional response is stable under all environmental conditions, but its ability to distinguish signals from different directions is relatively low [19]. Adaptive beamforming techniques typically exploit statistical properties of the signal (e.g., second-order statistics) to optimize processing. However, in practice, these statistics usually need to be estimated from real-time observations. Therefore, the system needs to have adaptive capabilities to track and respond to these changes in real time. By dynamically adjusting the filter coefficients, adaptive beamforming maintains the response to the signal in a specific "observation" direction while suppressing the response of interfering noise sources. The Minimum Variance Distortionless Response Filter (MVDR), also known as the optimal beamformer, is the most well-known technique. It optimizes the output SNR ratio while ensuring a consistent gain for the desired direction. However, it is sensitive to Direction of Arrival (DOA) estimation errors and allows only one target signal to be acquired. The Linearly Constrained Minimum Variance (LCMV) beamformer adds a linear constraint to the MVDR beamforming to collect multiple target signals. To ensure the degrees of freedom of the LCMV beamformer, the number of microphones must exceed the number of imposed constraints. It is shown that the Generalized Sidelobe Canceller (GSC) proposed by Griffiths and Jim is essentially equivalent to the LCMV beamformer [20]. GSC provides an efficient way to implement LCMV and consists of two main parts: a

fixed beamformer (w_q) for generating non-adaptive outputs, and an adaptive module that focuses on sidelobe suppression.

In [21], a conventional uniform linear array of omnidirectional microphones is discussed. In [22], Zhang et al. proposed a noise suppression algorithm that can subtract background and music noise from speech samples by combining beamforming techniques and multi-band spectral subtraction based on microphone arrays. In [23], a novel two-channel spatial filtering method proves its value for speech enhancement.

2.1.3 Post-Filtering

In practice, beamformers often fail to achieve their theoretical optimal performance. This performance gap is usually due to inaccurate assumptions about environmental conditions or inaccurate steering vector estimation. It can be shown that the MVDR beamformer is optimal in the maximum likelihood sense and produces the best SNR for narrowband signals [24], but it cannot guarantee the best SNR for wideband signals such as speech signals. The Multichannel Wiener Filter (MWF) beamformer is identified as the optimal linear filter under the MMSE criterion and can be decomposed into an MVDR beamformer along with a single-channel post-Wiener filter [25]. In general, the multi-channel Wiener beamformer can produce better SNR than the MVDR beamformer, which is the motivation for discussing the post-filter in this section.

As introduced in the previous section, the design of the post-filter is based on the assumption that the noise signal is incoherent [26] [27]. Fischer et al. studied the combination of post-filtering and GSC to improve noise suppression in noise fields dominated by coherent sources [28]. Bitzer et al. studied the solution of combining superdirectional arrays with post-filtering [29].

In order to solve the problem that the assumption of an incoherent noise field is often not valid in reality, McCowan and Bourlard [30] and Lefkimmistis and Maragos proposed some new solutions by applying the diffuse noise coherence function to the post-filter design. They proposed a generalized post-filter design.

2.1.4 Brief Conclusion

Advances in noise reduction and voice quality enhancement for mobile voice devices have greatly benefited from spectral enhancement and spatial processing techniques. Spectral enhancement methods, ranging from traditional approaches such as Wiener filtering and spectral subtraction to modern algorithms based on machine learning, play a vital role in mitigating noise in both single-microphone and multi-microphone setups. While single-microphone solutions offer simplicity for specific scenarios, multi-microphone techniques generally achieve superior performance by exploiting the spatial diversity of sound sources. Spatial processing, particularly through various beamforming techniques such as delay-and-sum, MVDR, and LCMV beamformers, effectively isolate the desired signal by exploiting directional information, although challenges such as sensitivity to errors in Direction of Arrival (DOA) estimation remain. To bridge the performance gap between theory and practical implementations, post-filtering strategies have been employed that enhance noise suppression and ensure robustness to slight beamforming errors. Despite these advances, continued research is essential to address

limitations associated with reverberation handling and adaptive performance in dynamic environments.

2.2 Principle of Room Acoustics Simulation

In this section, we introduce the important knowledge about room acoustic modeling. Beginning with some assumptions, we introduce geometric methods and wave-based methods, among which we focus on geometric modeling.

2.2.1 Ray-Wave Interchange Assumption

Sound is a sequence of pressure waves propagating through compressible media such as air or water. The second-order partial differential equation shown below, known as acoustic wave equations, describes the temporal and spatial changes in its propagation through a lossless fluid.

$$\frac{\partial^2 p}{\partial x^2} - \frac{1}{c^2}\frac{\partial^2 p}{\partial t^2} = 0, \tag{2.2}$$

where p is the acoustic pressure, c is the speed of sound, x is the position and t is the time.

Enclosed space sound fields can be modeled mainly by two algorithms: geometric algorithms and wave-based algorithms [31]. The wave-based method finds the numerical solution to the wave equation to recover wave phenomena such as interference and diffraction without error, which has a great impact on low-frequency response but comes with a high computational cost. In contrast, geometric methods are less precise. It assumes here that the wave nature of sound is ignored to achieve some simplified modeling of the behavior of sound, making sound waves equivalent to rays or particles. This allows us to achieve faster simulations at the expense of lower accuracy. Figure 2.1 illustrates the connections between the most widely used techniques.

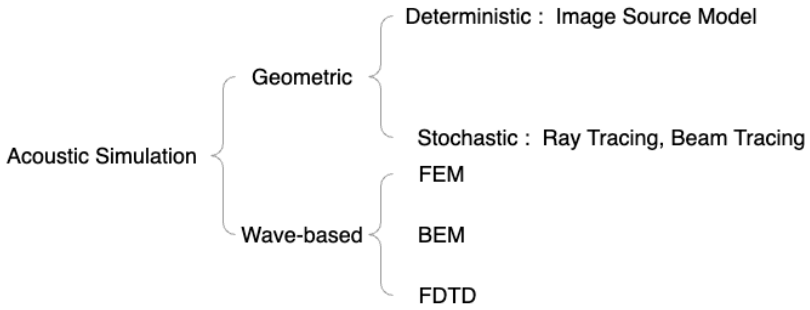


Figure 2.1: Summary of Various Acoustic Simulation Techniques Categorized by Type

As the field of computer graphics advances, modeling waves as particles have been proven successful. This technique is ideal for simulating relatively high-frequency waves (such as visible light). Since the wavelength of these waves is typically many times smaller than any surface in the scene, we can confidently say that wave phenomena can

be ignored and modeled entirely as rays.
Audible sound frequencies, spanning from 20 Hz to 20 kHz, correspond to wavelengths in air that vary from 17 m to 0.017 m. Comparing sound wavelengths to the most common surfaces, it is no longer suitable to ignore wave phenomena, which would otherwise lead to significant approximation errors [32].
But in many cases, these inaccuracies are acceptable and necessary trade-offs. Wave modeling is so computationally expensive that running large-scale simulations is out of the reach of ordinary people except for professionals with mainframe computers. This makes geometric methods the sole viable choice.

2.2.2 Geometric Methods

Geometric methods can be divided into two categories: stochastic methods and deterministic methods. Stochastic methods are usually based on statistical approximations. The purpose is to conduct random and continual sampling of the problem space, preserve samples that adhere to specific correctness criteria, and discard the non-conforming ones. By increasing the number of random samples, we can reduce the probability of false results and increase accuracy. A good number of samples should balance accuracy and operation speed.
The image source method is the main deterministic technique, searching exhaustively for all possible precise reflection paths connecting the source and receiver. It works well in shoebox-shaped rooms with completely rigid surfaces. However, this method is limited to modeling only specular reflections, neglecting the wave phenomenon. Therefore, its accuracy is reduced for non-standard shoebox rooms, and it suffers from deficiencies in calculating reverberation tails, which are mostly diffuse. Moreover, the method's cost escalates rapidly with higher reflection orders. Initially, the method mirrors the sound source against all scene surfaces, creating various image sources. Subsequently, each image source undergoes reflection against all surfaces, leading to a geometric increase in computational demands for higher reflection orders. Because of these challenges, the image-source method is best utilized for modeling early reflections. It is often combined with stochastic technique to estimate the late part of the Room Impulse Response (RIR) for a given scenario.
The wave-based method is accurate, but the computational cost is high at high frequencies. The image-based model is not very accurate at low frequencies, but it can greatly reduce the computational cost. Therefore, most geometric methods consider combining the two to construct a computationally affordable and accurate RIR.

The image-source model

The image-source techniques seek to identify all completely specular pathways between the source and the receiver. The image-source method assumes that sound travels only in straight lines and that all surfaces are purely reflective, like the reflection of light in a mirror, as shown in Figure 2.2. The sound traveling speed is fixed to the corresponding physical constant. The energy of each ray falls off as a function of $1/r^2$, which r is the distance the light travels.

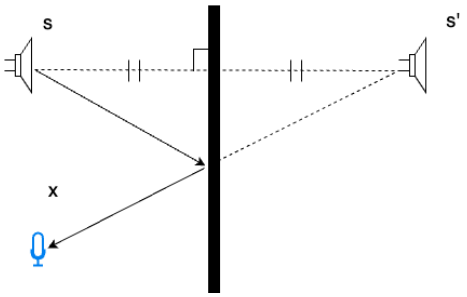


Figure 2.2: A sensor $\mathbf{x}$ and a speaker $\mathbf{s}$ are located next to a reflective wall. The image-source $\mathbf{s}'$ emulates the effects of the reflection of sound on the wall.

The ray creates a virtual source $\mathbf{s}'$ "behind" the bounding surface when it is reflected. Located on a line that intersects the wall perpendicularly, this source remains at the same distance from the source $\mathbf{s}$. Real sources and virtual sources (or mirror sources, image sources) emit the same sound at the same time. If the source undergoes reflection at only one boundary, it represents a first-order reflection. Acoustic rays can be reflected multiple times before reaching the receiver. As the number of reflections increases, the order of the virtual source generated by each reflection increases accordingly. Figure 2.3 shows a second-order image-source $\mathbf{s_3}$ model that is generated from a first-order virtual $\mathbf{s_2}$ source. It is worth mentioning that not all higher-order sources are valid. During the calculation, all high-level sources are listed in detail and their effectiveness is calculated based on *audibility test* [33, page 202].

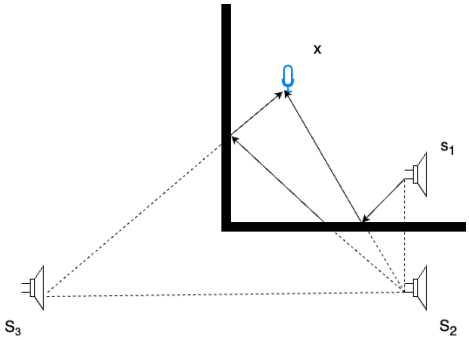


Figure 2.3: A sensor $\mathbf{x}$ and a speaker $\mathbf{s_1}$ are located next to a reflective wall. $\mathbf{s_2}$ is a first-order virtual sources, and $\mathbf{s_3}$ is a second-order virtual sources.

Given the system of a speaker $\mathbf{s}$, a sensor $\mathbf{x}$, and an environment providing a reflection path, the impulse response, the impulse response $h(n)$ is derived by summing the impulses emitted from all sources.

$$h(n) = \sum_{i=1}^{I} h_I(n) \tag{2.3}$$

where I is the number of mirror sources considered in the model and $h_I(n)$ is the impulse response between the i-th source and the receiver. The real source corresponds

to the index $i = 1$, so $h_1(n)$ models the direct path. As all sources emit the same sound simultaneously, the i-th source can be associated with several reflective surfaces at one time. $h_I(n)$ models the amplitude variation of the delay and attenuation of the source signal according to the distance between the source and receiver (i.e., the total distance of the specular reflections), and models the frequency variation of the signal according to the properties of the reflecting interface. In general, reflective surfaces attenuate high-frequency content more. If the parameter $0 < \gamma_I(n) \leq 1$ represents the combined effect of all the surfaces on the rays from the i-th source. Then,

$$h_I(n) = \gamma_I(n) * \frac{\delta(t - \Delta_i)}{\|\mathbf{x} - \mathbf{s}_i\|} = \frac{\gamma_i(t - \Delta_i)}{\|\mathbf{x} - \mathbf{s}_i\|} \tag{2.4}$$

$$\Delta_i = \frac{\|\mathbf{x} - \mathbf{s_i}\|}{c}. \tag{2.5}$$

Where Δ_i is the absolute time-of-arrival(TOA) of the i-th image source, $\|\mathbf{x} - \mathbf{s_i}\|$ is the distance between the receiver and the i-th source and $*$ denotes convolution. For real sources, $\mathbf{s}_1 = \mathbf{s}$ and $\gamma_1(n) = 1$.
However, employing a naive technique to locate all image sources in a scene is highly demanding in computational terms. Its computational complexity is $O(N^o)$, where N represents the number of reflection interfaces and o is the image source order. The image-source model is used to calculate direct sound and early reflection.

Ray Tracer

The image-source method is based on deterministic "ray" modeling, while ray tracers or ray tracing models sound energy propagation using stochastic "rays" modeling. It involves simultaneously sending uniformly random rays in various directions throughout a scene, moving at the speed of sound. Upon encountering a boundary, a ray loses part of its energy based on the characteristics of the boundary material. These rays bounce around, and only those that eventually hit the receiver are counted.
Ray tracing requires the receiver to have some volume, unlike the image-source method, which can treat the receiver as a point. This is because the chance of a random ray hitting exactly one point is extremely low but increases when the receiver occupies some space.
The accuracy of this method increases linearly with the number of rays used since more rays increase the probability of hitting the receiver. It is used as a beneficial complement to the Image-Source Model to find the reverberation tail.

2.2.3 Wave-Based Methods

Wave-based simulation methods effectively take into account wave effects such as interference and diffraction by numerically solving the wave equation [34] which leads to higher response accuracy in low frequency. Wave effects are great for correctly constructing the low-frequency response. Nevertheless, full spectrum wave-based simulations remain impractical, as the output frequency and the volume of the modeling space increase, the computational overhead grows rapidly. Wave-based methods can be

developed using Boundary Element Method (BEM), Finite Element Method (FEM), or Finite-Difference Time-Domain (FDFD) techniques. BEM and FEM are often collectively referred to as elemental methods.

2.3 Bluetooth Modeling

This section discusses the key concepts of Bluetooth channel estimation, focusing on the impact of latency and signal-to-noise ratio (SNR) on communication performance.

Bluetooth Channel Estimation

The standardized wireless protocol is implemented according to the Bluetooth® Low-Energy wireless standard.

Time latency

The microphone chain can be divided into three parts: the transmitter (abbreviated as TX), the radio processor, and the receiver (abbreviated as RX) displayed in Figure 2.4.

The TX consists of the microphone and TX audio processing process, which may consist of A/D conversion and encoding. The radio processor is through the Bluetooth protocol. The receiver consists of the speaker and RX audio processing process, which may consist of D/A conversion, decoding, combining the audio of multiple members in the group, etc.

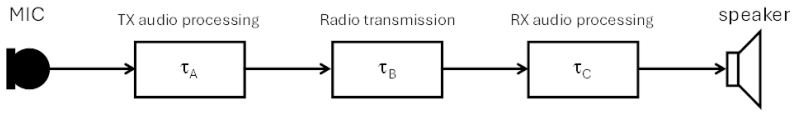


Figure 2.4: Microphone Chain

At the transmitter, buffering and audio encoding may cause extra delay τ_A, the radio transmission by the Bluetooth protocol may incur a delay of τ_B, and several audio functions, RX audio processing will introduce a delay of τ_C. For the TX audio processing and Radio transmission, we could assume $\tau_A + \tau_B \approx 15\,\text{ms}$, while the delay τ_C will mainly be determined by what delay the audio processing algorithm adds.

SNR Ratio

The Signal-to-Noise Ratio (SNR) is defined as the ratio of the desired signal power to the noise power and is usually expressed in decibels (dB). The calculation formula is shown as below,

$$\text{SNR} = 10\log_{10}\left(\frac{P_{\text{signal}}}{P_{\text{noise}}}\right) \tag{2.6}$$

When a device sends some information to another device through Bluetooth, a low received SNR will lead to difficulty in decoding the information contained within the signal without error. Bit Error Rate (BER) refers to the ratio of the number of error

bits to the total number of transmitted bits. High BER can cause communication failures.
According to Bluetooth Core Specification [35], a receiver must exhibit a BER of no more than 0.1 % at a signal strength of −70 dBm. The Minimum SNR($SNR_{out,min}$) for the receiver analog front end can be calculated from the BER. This depends on the different modulation schemes adopted by the standard and the assumed type of the digital demodulation algorithm [36]. According to Zhang [37], optimal demodulation requires an $SNR_{out,min}$ of 12 dB.
In conclusion, we can assume that the Bluetooth channel can be modeled as an impulse response with a delay of 15 ms.

2.4 Microphone Noise

We do not want to hear noise from the microphone. However, the microphone itself always generates some noise which is the so-called self-noise. Self-noise is a signal generated by the microphone itself, even when no sound source is present. Noise mainly originates from the current running in the circuit, typically including the "shot noise", thermal noise, and Brownian noise.

- Shot Noise
  "Shot noise" or Poisson noise is a type of noise that can be modeled by a Poisson process. In electronic circuits, electric current is the flow of discrete charges (electrons), and random fluctuations in current can cause shot noise [38]. In short, the discrete nature of electric current causes shot noise. Compared with other noise in circuits, such as thermal noise, discrete noise is generally insignificant and independent of temperature and frequency.

- Thermal Noise
  Thermal noise, also known as Johnson noise, is electronic noise produced by the thermal agitation of charge carriers within an electrical conductor in equilibrium and is present in all electrical circuits. The amount of thermal noise generated by any given circuit element depends on its impedance and temperature. The higher the impedance or the higher the temperature, the higher the thermal noise.

- Brownian Noise
  Brownian noise, also known as Brown noise or red noise, is generated by the movement of air molecules [39]. It occurs due to the air particles randomly hitting or rubbing against the diaphragm, and it is insignificant.

Quantifying noise

To make a connection to the physical world, we need to know how noise is quantized. This section mainly discusses the concepts of sound pressure level (SPL), perceptual weighting function, and equivalent noise level.

Sound Pressure Level

Because sound pressure amplitudes vary over large scales, microphone noise levels are often expressed in decibels (Lp) rather than pascals. The expression is given by,

$$Lp = 10\log_{10}(\frac{p}{p_0})^2 \tag{2.7}$$

where p is the sound pressure in pascals and p_0 is the reference sound pressure of 20 µPa. On the decibel scale, the range of audible sounds gradually increases from p_0 0 dB, which corresponds to the hearing threshold. It is considered noisy in the range of 60 dB to 90 dB and painful to the human ear at about 130 dB. Although by definition, doubling the sound pressure is equivalent to a 6 dB increase. However, if we want the sound to subjectively appear twice as loud, we need to increase it by about 10 dB [40, page 137]. Because the human ear has different sensitivities to different frequencies, with the most sensitive range being between 2 kHz and 5 kHz. To simulate the subjective human experience of sound, a weighting function is often applied. Figure 2.5 [1] shows the most common A weightings in the United States, the CCIR-468 (ITU-R 468) weightings in Europe, and the inverse ISO 226 equal loudness contours.

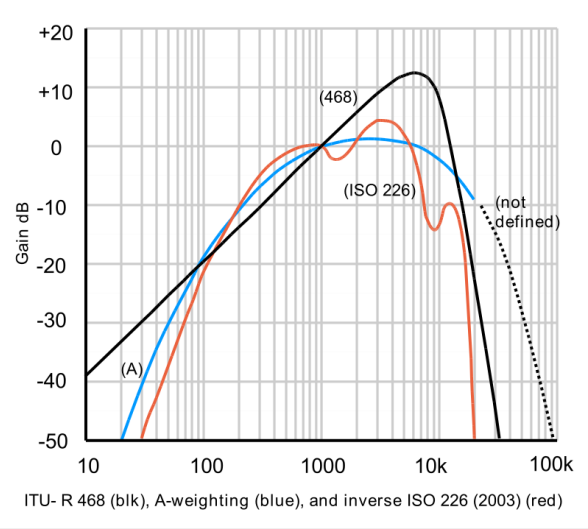


Figure 2.5: Three weighting curves: A, CCIR-468, and the inverse of ISO 226. Adapted from "A-weighting — Wikipedia, The Free Encyclopedia" [1].

Equivalent noise level

Different recording equipment has different performances. According to Neumann, one of the world's leading studio microphone manufacturers [41], overall recording equipment noise levels are below.

Below 10 dB-A Noise below 10 dB-A is extremely low, and even recordings in a very quiet environment may produce noise over 10 dB-A. Typically, extremely low self-noise figures are only found with modern large-diaphragm condenser microphones.

11-15 dB-A is still very good. It may be possible to discern some slight noise in key

places, but it's generally unlikely to be heard throughout the mix.
16-19 dB-A is good enough for most purposes. When recording relatively quiet sounds, some noise may be heard, but it's usually not noticeable.
20-23 dB-A is a very high self-noise coefficient for a studio microphone, and the performance of a microphone corresponding to this range cannot be called excellent. Within this range, each decibel increase is different. Because perceptible noise levels have been reached. This noise figure may be acceptable when recording loud sounds but will cause a noticeable impact when recording quieter sounds.
24 dB-A and above: Such self-noise figures are unworthy for recording.

The self-noise for dynamic microphones is not commonly specified because their noise performance largely depends on the microphone preamplifier used. From experience, dynamic microphones on ultra-low noise preamplifiers can achieve self-noise values of around 18 dB-A.

2.5 Speech Signal

In this section, we mainly introduce the knowledge related to speech signals in modeling, including empirical data obtained from well-known headphone manufacturers and point sound source attenuation rules.

2.5.1 Speech Volume

According to Shure, one of the leading manufacturers of audio electronics [42], typical dBA readings for speech are shown below,

- Maximum shout = 90 dBA at one meter (roughly 40 inches)
- Shout = 84 dBA at one meter
- Very Loud = 78 dBA at one meter
- Loud = 72 dBA at one meter
- Raised = 66 dBA at one meter
- Normal = 60 dBA at one meter
- Relaxed = 54 dBA at one meter

Other dBA readings can be estimated as follows: if the distance is divided by 2, the level increases by 6 dB. Example: Maximum Shout at 0.5 meter = 96 dB SPL.

2.5.2 Sound Attenuation from a Point Source

As introduced in section 2.4, the unit of actual microphone noise is the sound pressure level (SPL). However, to calculate the Room Impulse Response (RIR) in the Python library pyroomacoustics, we need to convert SPL to SNR. The SNR of the received

audio signal to the local microphone noise for each user can be calculated using the sound attenuation rule from a point source.

A sound source can be modeled as a point source if its dimension is small compared with the distance between the source and receiver. An omnidirectional point source uniformly radiates the sound to all directions in the space. Under free field conditions (i.e. ideal conditions without reflection and absorption), the sound attenuation from a point source can be estimated using the inverse square law, the formula of which is shown as

$$L_{P(2)} = L_{P(1)} - 20 \times \log_{10}\left(\frac{R_2}{R_1}\right), \tag{2.8}$$

where:

- $L_{P(2)}$ is the estimated sound pressure level at position 2 whose distance to the source is R_2.
- $L_{P(1)}$ is the measured sound pressure level at position 1 whose distance to the source is R_1.
- R_2 is the distance between position 2 to the source.
- R_1 is the distance between position 1 to the source.

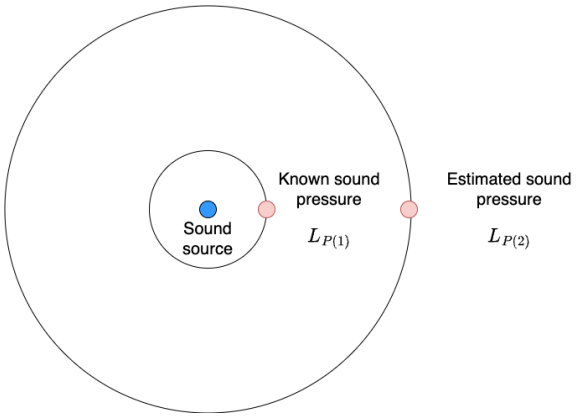


Figure 2.6: schematic diagram of sound attenuation of a point source

An example is given here. Combining the information provided before, the sound pressure level received at a distance of 3 meters from a person speaking at a normal volume is approximately 50.5 dBA.

$$\begin{aligned} L_{P(2)} &= L_{P(1)} - 20 \times \log_{10}\left(\frac{R_2}{R_1}\right) \\ &= 60 - 20 \times \log_{10} 3 \\ &= 50.5\,\text{dBA} \end{aligned} \tag{2.9}$$

Given the microphone self-noise info, the SNR at the given position is about 32.5 dB.

$$
\begin{aligned}
SNR_{dB} &= 20 \times log_{10}\left(\frac{p_s}{p_n}\right) \\
&= 20 \times log_{10}\left(\frac{\frac{p_s}{p_0}}{\frac{p_n}{p_0}}\right) \\
&= 20 \times \left(log_{10}\left(\frac{p_s}{p_0}\right) - log_{10}\left(\frac{p_n}{p_0}\right)\right) \\
&= L_{P(s)} - L_{P(n)} \\
&= 50.5 - 18 \\
&= 32.5\,\text{dBA}
\end{aligned}
\tag{2.10}
$$

3 Problem Formulation

In modern group communication systems (such as DGC), due to the complex acoustic structures involved, understanding how audio signals converge at the receiver and identifying the echo sources are often crucial for designing algorithms to solve the problem. This chapter aims to address the acoustic structure problem in a multi-user environment in a DGC system and develop a mathematical model to represent the audio signal at the receiver.
For a linearly uniformly distributed microphone array, we can effectively extract the target signal through beamforming. In the DGC scenario, we have a system that mixes Bluetooth channels and acoustic channels. For such a system where different types of communication media or protocols work together to transmit data in a communication system, we can call it a heterogeneous communication link. By abstracting the Bluetooth channel into a virtual acoustic channel, we transform the dereverberation and noise reduction task in the heterogeneous communication link into an equivalent microphone array problem and lay the foundation for the beamforming solution.

3.1 Single Speaker Case Signal Model

For simplicity, we assume only 3 users, A, B, and C, are involved here. Suppose A is speaking, B is listening, and C is a bystander. The signal emitted from user A is termed $s_A(n)$, and the signal received by user B is termed $x_B(n)$.

Let the following variables be defined as

- $h_{ij}^{acous.}(n)$ represents the impulse response of the air acoustic path from position i to microphone j. A time delay is possibly contained in the impulse response. It is a general form for $h_{AA}^{acous.}(n)$, $h_{AB}^{acous.}(n)$, $h_{AC}^{acous.}(n)$, $h_{IA}^{acous.}(n)$, $h_{IB}^{acous.}(n)$, $h_{IC}^{acous.}(n)$, the subscript A, B, C means user A, B, C, and I means interference source.
  It is worth mentioning that when $i = j$, it refers to the acoustic transfer function from the speaker's mouth to the speaker's microphone, such as $h_{AA}^{acous.}(n)$. It depends on the microphone geometry design and is assumed as an impulse response for the moment.
- $h^{BT}(n)$ is the digital transmission function that models the Bluetooth channel's effect, such as filtering, modulation, and demodulation with a time delay introduced by Bluetooth transmission.
- $h^{tr.}(n)$ is the impulse response of the transparency mode. Transparency mode is an optional mode that allows the user to actively pick up external ambient noise and play it inside the ear canal, helping the user to perceive possible dangers in the environment.

- $n_I(n)$ is the interference source.
- $n_A(n), n_B(n), n_C(n)$ are the microphone self-noise.

3.1.1 Received Signal at Listener B (No Leakage From Loudspeaker)

Voice leakage from the user's speaker to its own microphone is often related to the headset shape design and the microphone distribution. According to Dopple, we do not consider the speech leakage from the speaker to its own microphone in this work.

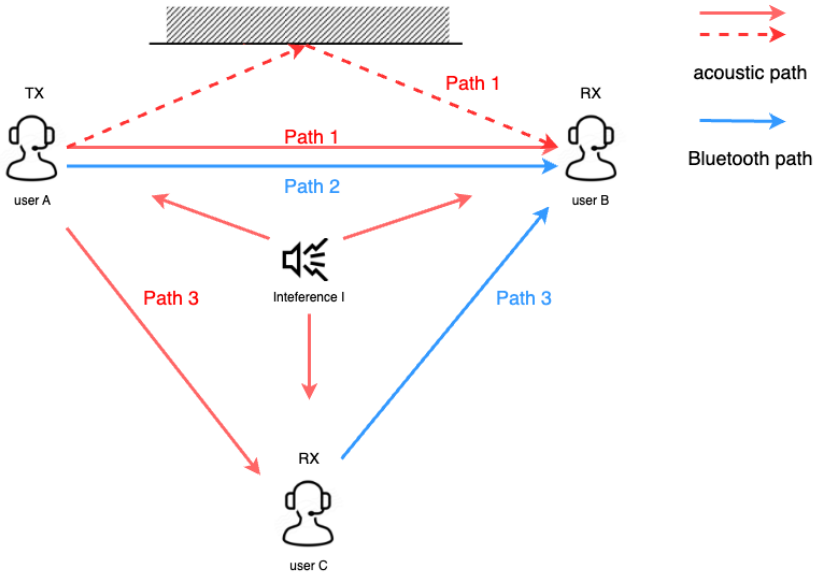


Figure 3.1: echoes at listener B with user A speaking, excluding leakage from loudspeaker to MIC, VPU

With each signal component representing a transmission path, the signal of interest picked up at listener B $x_B(n)$ can be represented as,

$$x_B(n) = x_B^1(n) + x_B^2(n) + x_B^3(n), \tag{3.1}$$

where $x_B^1(n)$, $x_B^2(n)$, $x_B^3(n)$ are the sub-signals defined below.

For path 1 (transparency mode), $x_B^1(n)$ represents the voice that reaches the recipient via sound waves through the air.

$$x_B^1(n) = (s_A(n) * h_{AB}^{acous.}(n) + n_I(n) * h_{IB}^{acous.}(n) + n_B(n)) * h^{tr.}(n) \tag{3.2}$$

For path 2, $x_B^2(n)$ represents the voice picked up by the speaker's microphone and sent digitally via Bluetooth to the recipient.

$$x_B^2(n) = (s_A(n) * h_{AA}^{acous.}(n) + n_I(n) * h_{IA}^{acous.}(n) + n_A(n)) * h^{BT}(n) \tag{3.3}$$

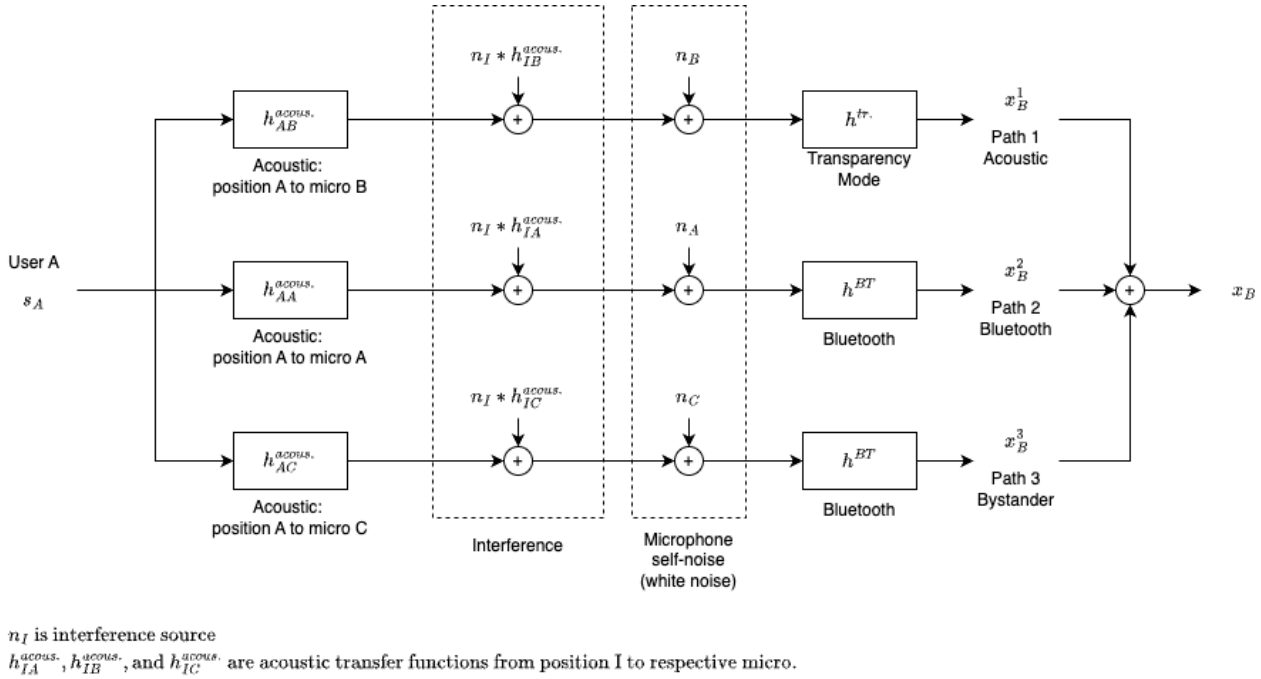


Figure 3.2: A schematic construction of signal x_B

For path 3, $x_B^3(n)$ represents the voice picked up by the bystander's microphone and sent digitally via Bluetooth to the recipient.

$$x_B^3(n) = (s_A(n) * h_{AC}^{acous.}(n) + n_I(n) * h_{IC}^{acous.}(n) + n_C(n)) * h^{BT}(n) \tag{3.4}$$

A schematic construction of signal x_B is given in Figure 3.2.

The listener hears the echo because the signal it receives is multichannel, and coupling between these signals through channels deforms them into deformed copies of each other. Therefore, in this scenario, the algorithm runs on the listener's hardware, and the signal we know is the signal received by the listener, or more precisely, we know every single signal transmitted through different channels. The signal we want to recover is the sound emitted by the speaker. This then leads to the problem of finding an estimate of the clean target signal $s_A(n)$, given $x_B(n)$.

3.1.2 Signal in Matrix Form

The above-described signals are separated and represented individually. In matrix form, the signals should be written as follows,

$$\begin{aligned}
\mathbf{x_B}(n) &= \begin{bmatrix} x_B^2(n) \\ x_B^1(n) \\ x_B^3(n) \end{bmatrix} \\
&= \begin{bmatrix} h_{AA}^{acous.}(n) * h^{BT}(n) \\ h_{AB}^{acous.}(n) * h^{tr.}(n) \\ h_{AC}^{acous.}(n) * h^{BT}(n) \end{bmatrix} * s_A(n) + \begin{bmatrix} h_{IA}^{acous.}(n) * h^{BT}(n) \\ h_{IB}^{acous.}(n) * h^{tr.}(n) \\ h_{IC}^{acous.}(n) * h^{BT}(n) \end{bmatrix} * n_I(n) + \begin{bmatrix} n_A(n) * h^{BT}(n) \\ n_B(n) * h^{tr.}(n) \\ n_C(n) * h^{BT}(n) \end{bmatrix} \\
&= \mathbf{a}(n) * s_A(n) + \mathbf{b}(n) * n_I(n) + \mathbf{v}(n) \\
&= \mathbf{a}(n) * s_A(n) + \mathbf{n}(n)
\end{aligned} \tag{3.5}$$

where $\mathbf{x_B}(n), \mathbf{a}(n), \mathbf{b}(n), \mathbf{v}(n), \mathbf{n}(n) \in \mathbb{R}^3$, $s_A(n), n_I(n) \in \mathbb{R}$. $\mathbf{a}$ is the transfer function from talker A to listener B, $\mathbf{b}(n)$ is the transfer function from the interference source to listener B, $\mathbf{v}$ is the microphone self-noise, and $\mathbf{n}$ is the overall noise which contains two parts, the environmental noise caused by external interference source and system noise caused by devices itself. s_A is the talker itself signal.

We want to transform the signal from the time domain to the frequency domain to facilitate the following processing. Since the signal we are dealing with is a speech signal, which is a non-stationary signal. However, in a short period (such as 20 ms to 30 ms), we can assume that the frequency of the signal will not change, which means that we need to perform a Discrete Fourier transform (DFT) on the short time frames of the signal, that is, Short-Time Fourier Transform (STFT).
Knowing Bluetooth channel $h^{BT}(n)$ can be modeled as a pure delay $h^{BT}(n) = \delta(n-n_1)$. Defining $\Delta^{BT}(k) = e^{-j2\pi k n_1/K}$, where K represents the FFT length in STFT, the signal model can be represented as,

$$\begin{aligned}
\mathbf{x_B}(k,l) &= \begin{bmatrix} X_B^2(k,l) \\ X_B^1(k,l) \\ X_B^3(k,l) \end{bmatrix} \\
&= \begin{bmatrix} H_{AA}^{acous.}(k,l)\Delta^{BT}(k) \\ H_{AB}^{acous.}(k,l)H^{tr.}(k,l) \\ H_{AC}^{acous.}(k,l)\Delta^{BT}(k) \end{bmatrix} S_A(k,l) + \begin{bmatrix} H_{IA}^{acous.}(k,l)\Delta^{BT}(k) \\ H_{IB}^{acous.}(k,l)H^{tr.}(k,l) \\ H_{IC}^{acous.}(k,l)\Delta^{BT}(k) \end{bmatrix} N_I(k,l) \\
&+ \begin{bmatrix} N_A(k,l)\Delta^{BT}(k) \\ N_B(k,l)H^{tr.}(k,l) \\ N_C(k,l)\Delta^{BT}(k) \end{bmatrix} \\
&= \mathbf{a}(k,l)S_A(k,l) + \mathbf{b}(k,l)N_I(k,l) + \mathbf{v}(k,l) \\
&= \mathbf{a}(k,l)S_A(k,l) + \mathbf{n}(k,l)
\end{aligned} \tag{3.6}$$

where k represents the frequency index, l represents the time index.

We note a simple fact that the received signal $\mathbf{x_B}(k,l)$ can be decomposed into a basic pattern as follows,

$$\begin{aligned}\mathbf{x_B}(k,l) &= \mathbf{a}(k,l)S_A(k,l) + \mathbf{b}(k,l)N_I(k,l) + \mathbf{v}(k,l) \\ &= \mathbf{a}(k,l)S_A(k,l) + \mathbf{n}(k,l),\end{aligned} \tag{3.7}$$

which is a combination of three components: target signal, interference signal, and self-noise. Based on this basic abstraction, we can easily extend the signal model as follows.

First, in our study, the main goal is echo cancellation within the system. One microphone per user is enough to form a three-channel system, forming a beamforming solution. However, it can be foreseen that as the number of microphones increases, the quality of beamforming can be improved because the available information increases. Assuming M microphones in use per user will cause our subchannels to change from $X_B^1(k,l), X_B^2(k,l), X_B^3(k,l) \in \mathbb{R}$ to $\mathbf{x_B^1}(k,l), \mathbf{x_B^2}(k,l), \mathbf{x_B^3}(k,l) \in \mathbb{R}^M$, making $\mathbf{x_B}(k,l), \mathbf{a}(k,l), \mathbf{b}(k,l), \mathbf{v}(k,l), \mathbf{n}(k,l) \in \mathbb{R}^{3M}$

Secondly, the analysis we conducted at the beginning is the simplest case, that is, there are only 3 people involved in the system, but the system is designed for group communication, and the number of people considered is not less than 3 and not more than 10. Therefore, if the system involves N people, we need to introduce $N-2$ bystanders. The increased number of bystanders should be accounted for in $\mathbf{x_B^3}(k,l)$, so the dimension of $\mathbf{x_B^3}(k,l)$ expands from $\mathbb{R}^M$ to $\mathbb{R}^{((N-2)M)}$.

$$\begin{aligned}\mathbf{x_B}(k,l) &= \begin{bmatrix} \mathbf{x_B^2}(k,l) \\ \mathbf{x_B^1}(k,l) \\ \mathbf{x_B^3}(k,l) \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{h_{AA}^{acous.}}(k,l)\Delta^{BT}(k) \\ \mathbf{h_{AB}^{acous.}}(k,l)H^{tr.}(k,l) \\ \mathbf{h_{AC}^{acous.}}(k,l)\Delta^{BT}(k) \end{bmatrix} S_A(k,l) + \begin{bmatrix} \mathbf{h_{IA}^{acous.}}(k,l)\Delta^{BT}(k) \\ \mathbf{h_{IB}^{acous.}}(k,l)H^{tr.}(k,l) \\ \mathbf{h_{IC}^{acous.}}(k,l)\Delta^{BT}(k) \end{bmatrix} N_I(k,l) \\ &+ \begin{bmatrix} \mathbf{n_A}(k,l)\Delta^{BT}(k) \\ \mathbf{n_B}(k,l)H^{tr.}(k,l) \\ \mathbf{n_C}(k,l)\Delta^{BT}(k) \end{bmatrix} \\ &= \mathbf{a}(k,l)S_A(k,l) + \mathbf{b}(k,l)N_I(k,l) + \mathbf{v}(k,l) \\ &= \mathbf{a}(k,l)S_A(k,l) + \mathbf{n}(k,l)\end{aligned} \tag{3.8}$$

where $\mathbf{x_B^1}(k,l), \mathbf{x_B^2}(k,l) \in \mathbb{R}^M$, $\mathbf{x_B^3}(k,l) \in \mathbb{R}^{((N-2)M)}$. We use $\mathbf{h_{example}}$ to represent the stacked microphone array transfer functions for $\mathbf{h_{AA}^{acous.}}, \mathbf{h_{AB}^{acous.}}, \mathbf{h_{AC}^{acous.}}, \mathbf{h_{IA}^{acous.}}, \mathbf{h_{IB}^{acous.}}, \mathbf{h_{IC}^{acous.}}$ and $\mathbf{n_{example}}$ to represent the stacked noises for $\mathbf{n_A}, \mathbf{n_B}, \mathbf{n_C}$ as following,

$$\mathbf{h_{example}} = \begin{bmatrix} H_1 \\ \vdots \\ H_M \end{bmatrix}, \quad \mathbf{n_{example}} = \begin{bmatrix} N_1 \\ \vdots \\ N_M \end{bmatrix} \tag{3.9}$$

Note that we have now reformulated the problem into the well-known form of $\mathbf{x} = \mathbf{a}s + \mathbf{n}$ with s noting the target and $\mathbf{x}$ noting the received signal.

3.2 Multiple Speakers Case Signal Model

The above analysis is based on a single sound source, but in reality, multiple speakers may be speaking simultaneously, which requires introducing more sound sources.

3.2.1 Received Signal at Listener B (No Leakage from the Loudspeaker)

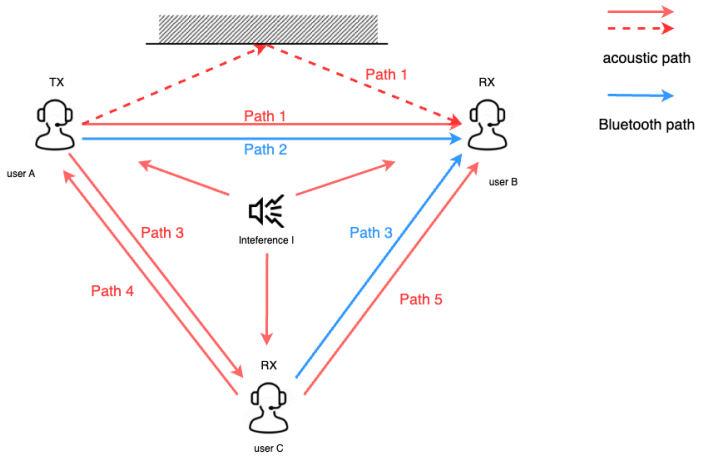


Figure 3.3: Echo at listener B with users A and C talking simultaneously, excluding leakage from speaker to MIC, VPU

There are 2 sources, $s_A(n)$ at talker A and $s_C(n)$ at speaker C. Compared with only one talker A, in Figure 3.3 there are additional components $x_B^4(n)$ and $x_B^5(n)$. Thus a few changes are made to the signal model. Again, $x_B(n)$ is the signal of interest picked up by listener B. With each signal component representing a transmission path, the signal received at user B is given by,

$$x_B(n) = x_B^1(n) + x_B^2(n) + x_B^3(n) + x_B^5(n). \tag{3.10}$$

For path 1 and path 5, these paths are both acoustic paths from different uses to listener B, thus can be combined. Speaker C's sound travels through the air to B's position and is actively picked up by transparency mode. Thus the

$$x_B^1(n) + x_B^5(n) = (s_A(n) * h_{AB}^{acous.}(n) + s_C(n) * h_{CB}^{acous.}(n) + n_I(n) * h_{IB}^{acous.}(n) + n_B(n)) * h^{tr.}(n) \tag{3.11}$$

For path 2, speaker C's sound travels through the air to A's microphone due to path 4 and is mixed with path 2 before being transmitted to B's speaker through the Bluetooth channel. Thus $x_B^2(n)$ can be reformatted as,

$$x_B^2(n) = (s_A(n) * h_{AA}^{acous.}(n) + s_C(n) * h_{CA}^{acous.}(n) + n_I(n) * h_{IA}^{acous.}(n) + n_A(n)) * h^{BT}(n). \tag{3.12}$$

For path 3, in addition to the original content, speaker C's sound is picked up and travels through the Bluetooth channel path 3 to listener B. Thus $x_B^3(n)$ can be reformatted as,

$$x_B^3(n) = (s_A(n) * h_{AC}^{acous.}(n) + s_C(n) * h_{CC}^{acous.}(n) + n_I(n) * h_{IC}^{acous.}(n) + n_C(n)) * h^{BT}(n). \tag{3.13}$$

A schematic construction of signal x_B is given in Figure 3.4.

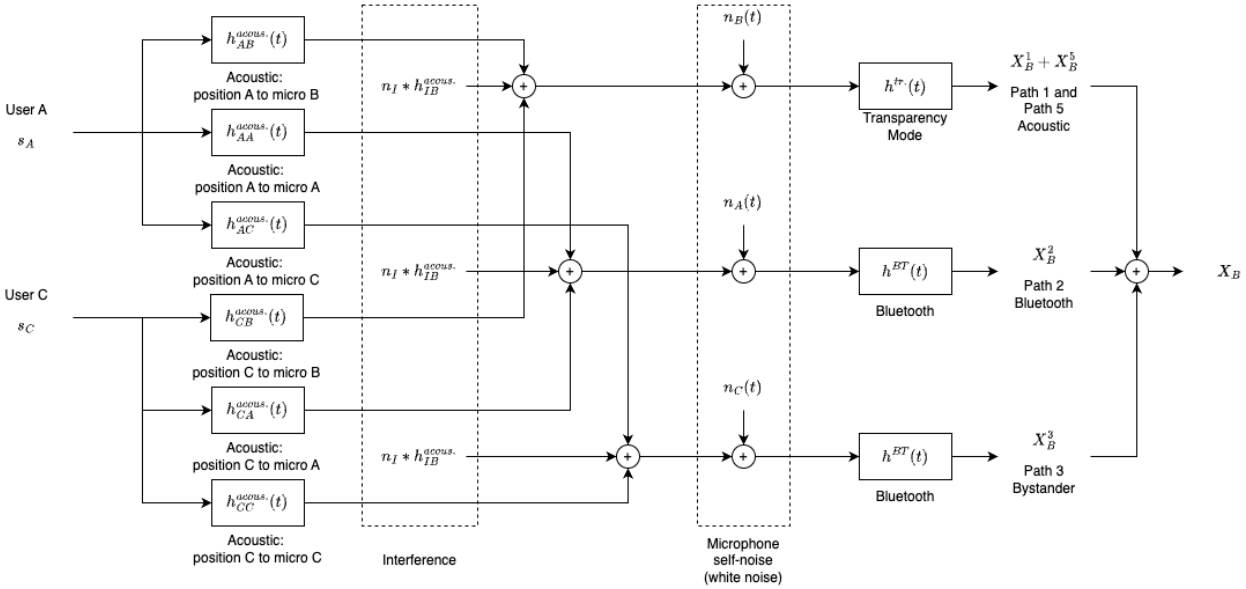


Figure 3.4: A schematic construction of signal x_B with additional speaker C

Compared with the previous scenario in section 3.1, the change here is that we have to restore an additional signal source $s_C(n)$. To be precise, we do not need to restore the separate copies of these two signals $s_A(n)$ and $s_C(n)$, but a combination of them, in the situation that the $x_B(n)$ is known.

3.2.2 Signal in Matrix Form

The above-described signals are separated and represented individually. Inheriting the previously extended multi-microphone and multi-spectator viewpoints, the signal in

matrix form should be written as follows,

$$
\begin{aligned}
\mathbf{x_B}(n) &= \begin{bmatrix} \mathbf{x_B^2}(n) \\ \mathbf{x_B^1}(n) + \mathbf{x_B^5}(n) \\ \mathbf{x_B^3}(n) \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{h_{AA}^{acous.}}(n) * h^{BT}(n) \\ \mathbf{h_{AB}^{acous.}}(n) * h^{tr.}(n) \\ \mathbf{h_{AC}^{acous.}}(n) * h^{BT}(n) \end{bmatrix} * s_A(n) + \begin{bmatrix} \mathbf{h_{CA}^{acous.}}(n) * h^{BT}(n) \\ \mathbf{h_{CB}^{acous.}}(n) * h^{tr.}(n) \\ \mathbf{h_{CC}^{acous.}}(n) * h^{BT}(n) \end{bmatrix} * s_C(n) \\
&+ \begin{bmatrix} \mathbf{h_{IA}^{acous.}}(n) * h^{BT}(n) \\ \mathbf{h_{IB}^{acous.}}(n) * h^{tr.}(n) \\ \mathbf{h_{IC}^{acous.}}(n) * h^{BT}(n) \end{bmatrix} * n_I(n) + \begin{bmatrix} \mathbf{n_A}(n) * h^{BT}(n) \\ \mathbf{n_B}(n) * h^{tr.}(n) \\ \mathbf{n_C}(n) * h^{BT}(n) \end{bmatrix} \\
&= \mathbf{a_A}(n) * s_A(n) + \mathbf{a_C}(n) * s_C(n) + \mathbf{b}(n) * n_I(n) + \mathbf{v}(n) \\
&= \mathbf{a_A}(n) * s_A(n) + \mathbf{a_C}(n) * s_C(n) + \mathbf{n}(n)
\end{aligned}
\tag{3.14}
$$

where $\mathbf{x_B^1}(n), \mathbf{x_B^2}(n), \mathbf{x_B^5}(n) \in \mathbb{R}^M$, $\mathbf{x_B^3}(n) \in \mathbb{R}^{((N-2)M)}$, $\mathbf{x_B}(n), \mathbf{a_A}(n), \mathbf{a_C}(n)$, $\mathbf{b}(n), \mathbf{v}(n), \mathbf{n}(n) \in \mathbb{R}^{NM}$.

Using STFT, the Equation 3.14 can be transformed from the time domain to the STFT domain, as shown below.

$$
\begin{aligned}
\mathbf{x_B}(k,l) &= \begin{bmatrix} \mathbf{x_B^2}(k,l) \\ \mathbf{x_B^1}(k,l) + \mathbf{x_B^5}(k,l) \\ \mathbf{x_B^3}(k,l) \end{bmatrix} \\
&= \begin{bmatrix} \mathbf{h_{AA}^{acous.}}(k,l)\Delta^{BT}(k) \\ \mathbf{h_{AB}^{acous.}}(k,l)H^{tr.}(k,l) \\ \mathbf{h_{AC}^{acous.}}(k,l)\Delta^{BT}(k) \end{bmatrix} S_A(k,l) + \begin{bmatrix} \mathbf{h_{CA}^{acous.}}(k,l)\Delta^{BT}(k) \\ \mathbf{h_{CB}^{acous.}}(k,l)H^{tr.}(k,l) \\ \mathbf{h_{CC}^{acous.}}(k,l)\Delta^{BT}(k) \end{bmatrix} S_C(k,l) \\
&+ \begin{bmatrix} \mathbf{h_{IA}^{acous.}}(k,l)\Delta^{BT}(k) \\ \mathbf{h_{IB}^{acous.}}(k,l)H^{tr.}(k,l) \\ \mathbf{h_{IC}^{acous.}}(k,l)\Delta^{BT}(k) \end{bmatrix} N_I(k,l) + \begin{bmatrix} \mathbf{n_A}(k,l)\Delta^{BT}(k) \\ \mathbf{n_B}(k,l)H^{tr.}(k,l) \\ \mathbf{n_C}(k,l)\Delta^{BT}(k) \end{bmatrix} \\
&= \mathbf{a_A}(k,l)S_A(k,l) + \mathbf{a_C}(k,l)S_C(k,l) + \mathbf{b}(k,l)N_I(k,l) + \mathbf{v}(k,l) \\
&= \mathbf{a_A}(k,l)S_A(k,l) + \mathbf{a_C}(k,l)S_C(k,l) + \mathbf{n}(k,l)
\end{aligned}
\tag{3.15}
$$

In section 3.1, we considered the extensions based on the microphone and bystander and noticed a basic signal composition pattern, where the received signal contains a combination of the target signal, the interference signal, and the self-noise.

$$
\begin{aligned}
\mathbf{x_B}(k,l) &= \mathbf{a}(k,l)S_A(k,l) + \mathbf{b}(k,l)N_I(k,l) + \mathbf{v}(k,l) \\
&= \mathbf{a}(k,l)S_A(k,l) + \mathbf{n}(k,l).
\end{aligned}
\tag{3.7}
$$

For scenarios with multiple signal sources, the expressions can be expanded accordingly. If the system concerns N people in total, the signal model of $\mathbf{x_B}(k,l)$ can be written as:

$$
\begin{aligned}
\mathbf{x_B}(k,l) &= \mathbf{A}(k,l)\mathbf{s}(k,l) + \mathbf{b}(k,l)N_I(k,l) + \mathbf{v}(k,l) \\
&= \mathbf{A}(k,l)\mathbf{s}(k,l) + \mathbf{n}(k,l).
\end{aligned}
\tag{3.7}
$$

where $\mathbf{A}(k,l)$ aggregates all the individual channel responses into a single matrix: $\mathbf{A}(k,l) = [\mathbf{a_A}(k,l), \mathbf{a_C}(k,l)]$. $\mathbf{s}(k,l)$ is the vector of all speech signals: $\mathbf{s}(k,l) = [S_A(k,l), S_C(k,l)]^T$

3.3 Chapter Summary

In this chapter, we first abstract the Bluetooth channel into a virtual acoustic channel, laying the foundation for the following methodology chapter. Next, we build a complete acoustic signal model for the DGC system in different scenarios and give problem formulations. For the three-user case(one speaker, one listener, and one bystander), the signal received by the listener is decomposed into contributions in vector form from multiple transmission paths, including the acoustic path (via Transparency mode) and the digital path (via Bluetooth). Noise and interference components are explicitly incorporated into the model.
For multiple active loudspeakers, the signal model is extended by introducing additional transmission paths. The task changes from isolating a single clean signal to estimating the combination of multiple signals perceived by the listener.

4 Proposed Methods

In the previous chapter, we systematically modeled the signal model in the entire system and formulated the problem. Based on the signal model, we will propose solutions for different scenarios in this chapter. The following mainly introduces the MVDR, LCMV, and MWF beamformers and the noise covariance matrix estimation.
Before proposing the methods, we need to make some assumptions here. In our work, we assume the target signals $s_A(n)$, $s_C(n)$ and noise signal $n(n) \in \mathbf{n}(n)$ are uncorrelated. The signal $\mathbf{v}(n)$ is zero-mean Gaussian noise. All signals considered in the work are broadband signals. Here, Acoustic Transfer Functions (ATFs) represents the direct acoustic path from the source to each microphone, while Relative Acoustic Transfer Functions (RTFs) is the normalized version of ATFs to the reference microphone.

4.1 Minimum Variance Distortionless Response Filter(MVDR)

The Minimum Variance Distortionless Response Filter (MVDR) beamformer, alternatively called the Capon beamformer [43], seeks to minimize the beamformer's output power under the condition of a single linear constraint on the array's response to the target signal. The MVDR beamformer can be generalized to the Linearly Constrained Minimum Variance (LCMV) beamformer developed by Er and Cantoni [44], which imposes multiple linear constraints. The MVDR is a commonly used technique for beamforming in array processing, allowing the microphone array to recover the signal coming from the direction of the target source. However, in our case, the beamformer tries to extract the signal of interest by taking the different channels as the input of the MVDR.

In theory, with the prior knowledge of the desired sources and ATFs, the MVDR beamformer is possible to achieve dereverberation and noise cancellation perfectly. With the MVDR, it is also possible to recover any component at the microphone $X_1(k,l), X_2(k,l), X_3(k,l) \dots X_N(k,l)$ when the corresponding channel is chosen as the reference channel. We want to recover the clean target signal $S_A(k,l)$ copy in one speaker source case and achieve noise reduction. Thus, we shall choose the channel with the highest SNR as the reference channel [45]. This leads to selecting the Bluetooth channel $X_1(k,l)$, which frames the problem as recovering $h_{AA}^{acous.}(k,l)h_{BT}(k,l)s_A(k,l)$, given $\mathbf{x_B}(k,l)$.

4.1.1 Filter Design

The objective of beamforming here is to construct a left-inverse $\mathbf{w}^H(k,l)$ of $\mathbf{a}(k,l)$ such that $\mathbf{w}^H(k,l)\mathbf{a}(k,l) = 1$ and hence $\mathbf{w}^H(k,l)\mathbf{x_B}(k,l) = S_A(k,l)$.

$$\begin{aligned}\hat{S}_A(k,l) &= \mathbf{w}^H(k,l)\mathbf{x_B}(k,l) \\ &= \mathbf{w}^H(k,l)\left(\mathbf{a}(k,l)S_A(k,l) + \mathbf{n}(k,l)\right) \\ &= \mathbf{w}^H(k,l)\mathbf{a}(k,l)S_A(k,l) + \mathbf{w}^H(k,l)\mathbf{n}(k,l)\end{aligned} \tag{4.1}$$

Here $\mathbf{a}(k,l)$ is called the acoustic transfer function. However, in many applications, it is not possible to know exactly the ATFs. Thus, we are interested in the relative acoustic transfer function, which is normalized with respect to a reference location, concerning one of the microphones. As demonstrated before, using the highest SNR channel as the reference channel will lead to taking $h_{AA}^{acous.}(k,l)e^{-j2\pi kn_1/K}S_A(k,l)$ as the reference signal, so the RTFs is given below,

$$\mathbf{a}'(k,l) = \left[1, A_2(k,l)/A_1(k,l), ..., A_M(k,l)/A_1(k,l)\right]^T \tag{4.2}$$

The estimated signal $\hat{s}(k,l)$ can be written as follows,

$$\begin{aligned}\hat{S}_A(k,l) &= \mathbf{w}^H(k,l)\left(\begin{bmatrix} 1 \\ h_{AB}^{acous.}(k,l)H^{tr.}(k,l)h_{AA}^{acous.-1}(k,l)\Delta^{BT}(k) \\ h_{AC}^{acous.}(k,l)h_{AA}^{acous.-1}(k,l)\end{bmatrix} S'_A(k,l) + \mathbf{n}(k,l)\right) \\ &= \mathbf{w}^H(k,l)\left(\mathbf{a}'(k,l)S'_A(k,l) + \mathbf{n}(k,l)\right) \\ &= \mathbf{w}^H(k,l)\mathbf{a}'(k,l)S'_A(k,l) + \mathbf{w}^H(k,l)\mathbf{n}(k,l)\end{aligned} \tag{4.3}$$

where $S'_A(k,l)$ is $S_A(k,l)h_{AA}^{acous.}(k,l)e^{-j2\pi kn_1/K}$

After applying the filter, one more noise residue term is left which is $\mathbf{w}^H(k,l)\mathbf{n}(k,l)$ as shown above. It automatically leads to the optimization problem below.

$$\begin{aligned}J &= \mathbb{E}\{|\mathbf{w}^H(k,l)\mathbf{x}(k,l)|^2\} \\ &= \mathbf{w}^H(k,l)\mathbf{R_x}(k,l)\mathbf{w}(k,l)\end{aligned} \tag{4.4}$$

The optimization problem is given by minimizing the output signal power and keeping the gain in the desired direction at 1.

$$\begin{aligned}&\min_{\mathbf{w}(k,l)} \; J(\mathbf{w}(k,l)) \\ &\text{s.t. } \mathbf{w}(k,l)^H\mathbf{a}'(k,l) = 1\end{aligned} \tag{4.5}$$

To find the optimization problem solution, we should construct the Lagrange function and set its derivative to 0, that is,

$$\begin{aligned}\frac{d}{d\mathbf{w}^H(k,l)}\left\{J(\mathbf{w}(k,l)) + \lambda\left(\mathbf{w}^H(k,l)\mathbf{a}'(k,l) - 1\right)\right\} &= \mathbf{R_x}(k,l)\mathbf{w}(k,l) + \lambda\mathbf{a}'(k,l) = 0 \\ \mathbf{w}(k,l) &= -\mathbf{R_x}^{-1}(k,l)\lambda\mathbf{a}'(k,l).\end{aligned} \tag{4.6}$$

Now we can use the constraint: $\mathbf{w}^H(k,l)\mathbf{a}'(k,l) = 1$, we can have $\mathbf{a}'^H(k,l)\mathbf{w}(k,l) = -\mathbf{a}'(k,l)^H\mathbf{R}_\mathbf{x}^{-1}(k,l)\lambda\mathbf{a}'(k,l) = 1$. Thus, we can represent λ as,

$$\begin{aligned} \lambda &= -\frac{1}{\mathbf{a}'^H(k,l)\mathbf{R}_\mathbf{x}^{-1}(k,l)\mathbf{a}'(k,l)} \\ \Rightarrow \mathbf{w}(k,l) &= \frac{\mathbf{R}_\mathbf{x}^{-1}(k,l)\mathbf{a}'(k,l)}{\mathbf{a}'^H(k,l)\mathbf{R}_\mathbf{x}^{-1}(k,l)\mathbf{a}'(k,l)} \end{aligned} \tag{4.7}$$

Given the relation between the matrix $\mathbf{R}_\mathbf{x}(k,l)$ and $\mathbf{R}_\mathbf{n}(k,l)$ that $\mathbf{R}_\mathbf{x}(k,l) = \mathbf{R}_\mathbf{n}(k,l) + \mathbf{a}'(k,l)\mathbf{a}'^H(k,l)\sigma_s^2(k,l)$, using the matrix inversion lemma, the equation can be expressed as

$$\begin{aligned} \mathbf{w}(k,l) &= \frac{\mathbf{R}_\mathbf{n}(k,l)^{-1}\mathbf{a}'(k,l)\left(1 - \frac{\mathbf{a}'^H(k,l)\mathbf{R}_\mathbf{n}(k,l)^{-1}\mathbf{a}'(k,l)\sigma_s^2(k,l)}{1+\mathbf{a}'^H(k,l)\mathbf{R}_\mathbf{n}(k,l)^{-1}\mathbf{a}'(k,l)\sigma_s^2(k,l)}\right)}{\mathbf{a}'^H(k,l)\mathbf{R}_\mathbf{n}(k,l)^{-1}\mathbf{a}'(k,l)\left(1 - \frac{\mathbf{a}'^H(k,l)\mathbf{R}_\mathbf{n}(k,l)^{-1}\mathbf{a}'(k,l)\sigma_s^2(k,l)}{1+\mathbf{a}'^H(k,l)\mathbf{R}_\mathbf{n}(k,l)^{-1}\mathbf{a}'(k,l)\sigma_s^2(k,l)}\right)} \\ &= \frac{\mathbf{R}_\mathbf{n}(k,l)^{-1}\mathbf{a}'(k,l)}{\mathbf{a}'^H(k,l)\mathbf{R}_\mathbf{n}(k,l)^{-1}\mathbf{a}'(k,l)} \end{aligned} \tag{4.8}$$

Equation 4.8 shows that MVDR performance does not necessarily depend on the noisy covariance matrix $\mathbf{R}_\mathbf{x}(\mathbf{k},\mathbf{l})$, and the noise matrix $\mathbf{R}_\mathbf{n}(\mathbf{k},\mathbf{l})$ will serve as an alternative method to calculate the result. It can be shown that the Equation 4.7 is actually equivalent to the Equation 4.8 in case of uncorrelated noise with respect to the target signal and precise estimation on steering vector [46]. However, if the steering vector estimation is imprecise, the Equation 4.7 leads to performance degradation and target cancellation [47]. The usage of the noise covariance matrix can provide more robustness, but it is harder to estimate in practice [48].

Thus,

$$\begin{aligned} \mathbf{w}(k,l) &= \frac{\mathbf{R}_\mathbf{x}^{-1}(k,l)\mathbf{a}'(k,l)}{\mathbf{a}'^H(k,l)\mathbf{R}_\mathbf{x}^{-1}(k,l)\mathbf{a}'(k,l)} \\ &= \frac{\mathbf{R}_\mathbf{n}(k,l)^{-1}\mathbf{a}'(k,l)}{\mathbf{a}'^H(k,l)\mathbf{R}_\mathbf{n}(k,l)^{-1}\mathbf{a}'(k,l)} \end{aligned} \tag{4.9}$$

4.1.2 Estimation of RTFs

A precise estimation of the RTFs is crucial to the MVDR's performance, as it helps steer the beamformer to the source direction and preserve spatial information. A poor estimation on $\mathbf{a}(k,l)$ will normally lead to the severe performance degradation of the beamformer.

Before we jump into RTFs, the autocorrelation matrix of microphone signals should be elaborated on. The autocorrelation matrix of the microphone signals in "speech and noise periods" and "noise only period" is necessary to construct the beamformer.

The autocorrelation matrix of microphone signals in "speech and noise periods", the desired speech component, and the noise component can be written as,

$$\begin{aligned}\mathbf{R_{x_B}} &= \mathbb{E}\{\mathbf{x_B}\mathbf{x_B}^H\} \\ \mathbf{R_s} &= \mathbb{E}\{\mathbf{x_S}\mathbf{x_S}^H\} \\ \mathbf{R_n} &= \mathbb{E}\{\mathbf{x_n}\mathbf{x_n}^H\},\end{aligned} \tag{4.10}$$

where $\mathbf{x_S}$ is the target signal component of the received signal, and $\mathbf{x_n}$ is the noise residual component.

- **Subspace-based RTFs Estimation:** A subspace-based RTFs estimator is based on the Generalized Eigenvalue Decomposition (GEVD) method for matrix pair $(\mathbf{R_{x_B}}, \mathbf{R_n})$. That is,

$$\mathbf{R_n^{-1}}\mathbf{R_{x_B}}\mathbf{U} = \mathbf{U}\mathbf{\Lambda} \tag{4.11}$$

  where $\mathbf{\Lambda}$ and $\mathbf{U}$ are the eigenvalue and eigenvector matrix pair of the covariance matrix $\mathbf{R_n^{-1}}\mathbf{R_{x_B}}$. In practice, the noise presented is not necessarily white noise. However, by performing GEVD on $(\mathbf{R_{x_B}}, \mathbf{R_n})$, pre-whitening is achieved.

  Under the assumption that signal and noise are uncorrelated, the covariance matrix $\mathbf{R_{x_B}}$ of the received microphone is given by,

$$\begin{aligned}\mathbf{R_{x_B}} &= \mathbf{U^{-H}}\mathbf{\Lambda}\mathbf{U^{-1}} \\ &= \mathbf{Q}\mathbf{\Lambda}\mathbf{Q^H} \\ &= \mathbf{Q}(\mathbf{\Lambda_s} + \mathbf{I_M})\mathbf{Q^H} \\ &= \mathbf{Q}\mathbf{\Lambda_s}\mathbf{Q^H} + \mathbf{Q}\mathbf{Q^H} \\ &= \mathbf{R_s} + \mathbf{R_n}\end{aligned} \tag{4.12}$$

  where $\mathbf{Q} = \mathbf{U^{-H}} = (\mathbf{q_1}, \ldots, \mathbf{q_M})$, $\mathbf{q_i} \in \mathbb{C}^{\mathbb{M}}$ is the left eigenvector matrix. Defining $\sigma_s^2 = \mathrm{E}[||s||^2]$ as the signal power, $\sigma_n^2 = \mathrm{E}[||n||^2]$ as the noise power, by definition, we have the noisy covariance matrix after pre-whitening as,

$$\begin{aligned}\mathbf{R_{x_B}} &= \mathbf{R_s} + \mathbf{R_n} \\ &= \sigma_s^2\mathbf{a'}\mathbf{a'^H} + \sigma_n^2\mathbf{I}\end{aligned} \tag{4.13}$$

  Based on the single source assumption, the matrix $\mathbf{R_s}$ is rank-1. By taking the eigenvector corresponding to the largest eigenvalue of the matrix $\mathbf{R_n^{-1}}\mathbf{R_{x_B}}$ as the RTFs vector, the RTFs vector can be estimated as,

$$\mathbf{a'} = \frac{\mathbf{q_1}}{\mathbf{e_1^T}\mathbf{q_1}} \text{with } \mathbf{e}_1 = [1, 0, \ldots, 0]^T \tag{4.14}$$

  Where $\mathbf{q_1}$ is the principal eigenvector corresponding to the largest eigenvalue of $\mathbf{\Lambda}$. In practice, the covariance matrix is estimated by taking the average of the

time frames for each frequency bin. Thus, the rank-1 assumption does not necessarily hold even for the single source case. Nevertheless, selecting the principal eigenvector corresponding to the largest eigenvalue of the matrix $\mathbf{R_n^{-1}R_x}$ is still a good estimation of the source RTFs vector, which points more to the signal component than the noise component [49].

- **Minimum Distortion-Based RTFs Estimation:** The minimum distortion-based RTFs estimation is based on minimizing the distortion between the estimated signal and the desired signal [50]. The following optimization problem induces a so-called spatial prediction vector [51],

$$\arg\min_{\mathbf{a'}} \mathrm{E}[(\mathbf{x_B} - \mathbf{a'x_B^2})^H(\mathbf{x_B} - \mathbf{a'x_B^2})]. \tag{4.15}$$

Which is

$$\mathbf{a'} = \frac{\mathbf{R_s e_1}}{\mathbf{e_1^T R_s e_1}} \tag{4.16}$$

To compute Equation 4.16, the signal covariance matrix $\mathbf{R_s}$ is required. Assuming we have an estimation of the noise covariance matrix $\mathbf{R_n}$, given the relationship in Equation 4.12, the signal covariance matrix can be estimated by the GEVD-based subspace method under the rank-1 assumption.

$$\mathbf{R_s} = \mathbf{Q}(\mathrm{diag}(\max(\lambda_1 - 1, 0), 0, \ldots, 0)\mathbf{Q^H} \tag{4.17}$$

where $\mathbf{Q} = \mathbf{U^{-H}} = (\mathbf{q_1}, \ldots, \mathbf{q_M}), \mathbf{q_i} \in \mathbb{C}^{\mathbb{M}}$ is the left eigenvector.

- **SCFA:** A few new methods such as simultaneous confirmatory factor analysis (SCFA) [52] could jointly estimate the RTFs and PSDs of sources. However, the problem formulation is not convex, and the computational complexity is high. Thus, it is not suitable for use in real-time applications.

In the experiment, we use the subspace-based RTFs estimation method, as the explicit use of $\mathbf{R_s(k, l)}$ in the minimum distortion-based RTFs estimation method can decrease the beamformer performance.

4.1.3 Limitation of MVDR

MVDR beamformer is a good solution to the single-speaker scenario, but it gets stuck in a two-speaker scene. As the MVDR beamformer can only preserve one target signal from the reference microphone. That is why we investigate the use of the LCMV beamformer for a two-speaker solution.

4.2 Linear-Constraint Minimum-Variance(LCMV)

The well-known MVDR beamformer imposes the single constraint to the desired signal. The intention of the MVDR beamformer is to ensure that the signal passes without distortion in a specified direction while minimizing the output power, thereby suppressing

interference from other directions. However, the MVDR filter has only one constraint, which is to maintain a constant gain in the specified direction. Therefore, it may have limited performance in the presence of other interference or if we want to preserve more signals of interest from different directions. The Linearly Constrained Minimum Variance (LCMV) beamformer extends the concept of MVDR by adding multiple linear constraints to ensure the filter maintains specific responses in various directions.

4.2.1 Filter Design

Let us first review Equation 3.15,

$$\begin{aligned}\mathbf{x_B}(k,l) &= \begin{bmatrix} \mathbf{x_B^2}(k,l) \\ \mathbf{x_B^1}(k,l) + \mathbf{x_B^5}(k,l) \\ \mathbf{x_B^3}(k,l) \end{bmatrix} \\ &= \mathbf{a_A}(k,l)S_A(k,l) + \mathbf{a_C}(k,l)S_C(k,l) + \mathbf{b}(k,l)N_I(k,l) + \mathbf{v}(k,l) \\ &= \mathbf{a_A}(k,l)S_A(k,l) + \mathbf{a_C}(k,l)S_C(k,l) + \mathbf{n}(k,l)\end{aligned} \tag{3.15}$$

A signal $\mathbf{x_B}(\mathbf{k},\mathbf{l})$ consists of two desired signal sources S_A, S_C, an interference source N_I, and some noise. The target signals are distributed in different channels; thus, we cannot preserve target signals simultaneously with the MVDR beamformer.

Assuming we have d sources received by M' microphone, where $M' = (2 + \#bystanders)M$. The objective of the beamformer is to recover a mixture of $K_d < d$ desired sources and the number of undesired sources is $K_u = d - K_d$. To give a generalized illustration, the Equation 3.15 should be extended as follows,

$$\begin{aligned}\mathbf{x_B}(k,l) &= \mathbf{A}(k,l)\mathbf{s}(k,l) + \mathbf{n}(k,l) \\ &= \mathbf{z}(k,l) + \mathbf{n}(k,l)\end{aligned} \tag{4.18}$$

where

$$\begin{aligned}\mathbf{x_B}(k,l) &= [X_{B1}(k,l) \;\; X_{B2}(k,l) \;\; \dots \;\; X_{BM'}(k,l)]^T \\ \mathbf{A}(k,l) &= [\mathbf{a_1}(k,l) \;\; \mathbf{a_2}(k,l) \;\; \dots \;\; \mathbf{a_{M'}}(\mathbf{k},\mathbf{l})]^T \\ \mathbf{a_{m'}}(k,l) &= [A_{1,m'}(k,l) \;\; A_{2,m'}(k,l) \;\; \dots \;\; A_{d,m'}(k,l)]^T \\ \mathbf{s}(k,l) &= [S_1(k,l) \;\; S_2(k,l) \;\; \dots \;\; S_d(k,l)]^T \\ \mathbf{n}(k,l) &= [N_1(k,l) \;\; N_2(k,l) \;\; \dots \;\; N_d(k,l)]^T \\ \mathbf{z}(k,l) &= [Z_1(k,l) \;\; Z_2(k,l) \;\; \dots \;\; Z_{M'}(k,l)]^T \\ Z_{m'}(k,l) &= \mathbf{a_{m'}^T}(k,l)\mathbf{s}(k,l)\end{aligned}$$

and $A_{d,m'}(k,l)$ is the acoustic transfer function from source d to the microphone m'.

The steering vector constraint condition can be constructed by a column vector $\mathbf{f}(k,l)$ of length d to preserve the signal from the directions of interest, while the rest degrees of freedom to minimize the influence of interference source and the additive noise to the array output as follows,

$$\begin{aligned}&\min_{\mathbf{w}(k,l)} \; \mathbf{w^H}(k,l)\mathbf{R_n}(k,l)\mathbf{w}(k,l) \\ &\text{s.t. } \mathbf{w^H}(k,l)\mathbf{A}(k,l) = \mathbf{f}^H(k,l)\end{aligned} \tag{4.19}$$

Where $\mathbf{A}(k,l) \in \mathbb{C}^{M' \times d}$, and $\mathbf{f}(\mathbf{k},\mathbf{l}) = [A^*_{1,1} \quad \dots \quad A^*_{K_d,1}(k,l) \quad 0 \quad \dots \quad 0]^T$. Thus, the spatial filtering is expanded from signal direction to multiple directions. In case $d < M'$, Equation 4.19 has a closed-form solution:

$$\mathbf{w}(k,l) = \mathbf{R_n^{-1}}(k,l)\mathbf{A}(k,l)\left(\mathbf{A}(k,l)\mathbf{R_n^{-1}}(k,l)\mathbf{A}(k,l)\right)^{-1}\mathbf{f}(k,l). \tag{4.20}$$

As demonstrated in Section 4.1, it is attractive to use RTFs to replace ATFs. Thus each microphone's steering vector $\mathbf{a_{m'}}(\mathbf{k},\mathbf{l})$ can be divided by the reference channel, which makes the d constraint condition $\mathbf{w^H}(k,l)\mathbf{A}(k,l) = \mathbf{f}^H(k,l)$ the following format,

$$\mathbf{w^H}(k,l)\mathbf{A'}(k,l) = \mathbf{f'}^H(k,l) \tag{4.21}$$

where

$$\mathbf{A'}(k,l) = \begin{bmatrix} 1 & \cdots & 1 \\ \frac{A_{1,2}(k,l)}{A_{1,1}(k,l)} & \cdots & \frac{A_{d,2}(k,l)}{A_{d,1}(k,l)} \\ \vdots & \ddots & \vdots \\ \frac{A_{1,M'}(k,l)}{A_{1,1}(k,l)} & \cdots & \frac{A_{d,M'}(k,l)}{A_{d,1}(k,l)} \end{bmatrix} \tag{4.22}$$

and vector $\mathbf{f'}(k,l)$ indicates desired and undesired signals,

$$\mathbf{f'} = [\underbrace{1 \dots 1}_{K_\mathrm{d}} \underbrace{0 \dots 0}_{K_\mathrm{u}}]^T. \tag{4.23}$$

Note that Equation 4.22 allows us to perform an executable RTFs estimate, whereas accurate ATFs estimation is generally challenging.

4.2.2 Limitations of LCMV

The improvement from ATFs to RTFs makes LCMV beamformer an applicable technique in practice. There are various techniques we can use in practice to identify RTFs [53, 18, 54]. All techniques require a reference microphone, which is usually chosen as the one with the highest input SNR.

However, in the case of multiple target speakers, for distributed microphone array systems or large-aperture microphone systems, we may need to select different reference microphones according to different speakers. More specifically, in the Equation 3.14 the highest energy of different speakers is marked in the block.

$$
\begin{aligned}
\mathbf{x_B}(k,l) &= \begin{bmatrix} \mathbf{x_B^2}(k,l) \\ \mathbf{x_B^1}(k,l) + \mathbf{x_B^5}(k,l) \\ \mathbf{x_B^3}(k,l) \end{bmatrix} \\
&= \begin{bmatrix} \boxed{\mathbf{h}_{\mathbf{AA}}^{\mathbf{acous.}}(k,l)\Delta^{BT}(k)} \\ \mathbf{h}_{\mathbf{AB}}^{\mathbf{acous.}}(k,l)H^{tr.}(k,l) \\ \mathbf{h}_{\mathbf{AC}}^{\mathbf{acous.}}(k,l)\Delta^{BT}(k) \end{bmatrix} S_A(k,l) + \begin{bmatrix} \mathbf{h}_{\mathbf{CA}}^{\mathbf{acous.}}(k,l)\Delta^{BT}(k) \\ \mathbf{h}_{\mathbf{CB}}^{\mathbf{acous.}}(k,l)H^{tr.}(k,l) \\ \boxed{\mathbf{h}_{\mathbf{CC}}^{\mathbf{acous.}}(k,l)\Delta^{BT}(k)} \end{bmatrix} S_C(k,l) \\
&+ \begin{bmatrix} \mathbf{h}_{\mathbf{IA}}^{\mathbf{acous.}}(k,l)\Delta^{BT}(k) \\ \mathbf{h}_{\mathbf{IB}}^{\mathbf{acous.}}(k,l)H^{tr.}(k,l) \\ \mathbf{h}_{\mathbf{IC}}^{\mathbf{acous.}}(k,l)\Delta^{BT}(k) \end{bmatrix} N_I(k,l) + \begin{bmatrix} \mathbf{n_A}(k,l)\Delta^{BT}(k) \\ \mathbf{n_B}(k,l)H^{tr.}(k,l) \\ \mathbf{n_C}(k,l)\Delta^{BT}(k) \end{bmatrix} \\
&= \mathbf{a_A}(k,l)S_A(k,l) + \mathbf{a_C}(k,l)S_C(k,l) + \mathbf{b}(k,l)N_I(k,l) + \mathbf{v}(k,l) \\
&= \mathbf{a_A}(k,l)S_A(k,l) + \mathbf{a_C}(k,l)S_C(k,l) + \mathbf{n}(k,l) \\
&= \mathbf{A}(k,l)\mathbf{s}(k,l) + \mathbf{n}(k,l)
\end{aligned}
\tag{3.15}
$$

The relative transfer function $\mathbf{A}'(k,l)$ should be then defined as follows,

$$
\begin{aligned}
\mathbf{A}'(k,l) &= [\mathbf{a}'_{\mathbf{A}}(k,l), \mathbf{a}'_{\mathbf{C}}(k,l)] \\
\mathbf{a}'_{\mathbf{A}}(k,l) &= \left[1, \frac{A_{12}(k,l)}{A_{11}(k,l)}, \frac{A_{13}(k,l)}{A_{11}(k,l)}\right]^T \\
\mathbf{a}'_{\mathbf{C}}(k,l) &= \left[\frac{A_{21}(k,l)}{A_{23}(k,l)}, \frac{A_{22}(k,l)}{A_{23}(k,l)}, 1\right]^T
\end{aligned}
\tag{4.24}
$$

However, in practice, estimating the multiple references RTFs is challenging. To distinguish the different reference microphones we need to estimate the true ATFs for each microphone, which is a known-to-all difficult question.

4.3 Multiple Channel Wiener Filter (MWF)

To overcome the dilemma of multiple speakers, Multichannel Wiener Filter (MWF) is considered. As is known, the single-channel Wiener filter can be considered as one of the most basic methods for single-channel noise reduction. The Wiener filter can produce the Minimum Mean Square Error (MMSE) estimate of the target speech by constructively combining the signal received from the microphone array. Doclo and Moonen [55] considered the MMSE estimation in multichannel systems, namely Multichannel Wiener Filter (MWF).

4.3.1 Filter Design

Similar to what is introduced before, the objective of the MWF beamformer is to construct a left-inverse $\mathbf{w}^H$ such that a MMSE criterion is minimized. As demonstrated

in the Section 4.1, the desired signal for the MWF is chosen from the Bluetooth channel signal $\mathbf{x_B^2}$, for example, the first component $\mathbf{e_1 x_B^2}$. This can be written as,

$$D_{MWF} = \mathbf{e_1^H x_B}, \text{ with } \mathbf{e_1} = [1, 0, 0, \ldots, 0]^T \tag{4.25}$$

Thus, the MWF minimization objective of the squared distance between the filtered microphone output and the desired signal (4.25) can be written as,

$$J_{\text{MWF}} = \mathbb{E}\{|\mathbf{w^H x_B} - \mathbf{e_1^H x_S}|^2\} \tag{4.26}$$

This expression is equivalent to the one below:

$$J_{\text{MWF}} = \mathbf{w}^H \mathbf{R_s w} - \mathbf{w}^H \mathbf{R_s e}_1 - \mathbf{e}_1^H \mathbf{R_s w} + \mathbf{e}_1^H \mathbf{R_s e}_1 + \mathbf{w}^H \mathbf{R_n w} \tag{4.27}$$

and the solution is given as:

$$\mathbf{w}_{\text{MWF}} = (\mathbf{R_s} + \mathbf{R_n})^{-1} \mathbf{R_s e}_1 \tag{4.28}$$

4.3.2 Decomposition

It is well known that MWF can be decomposed as a combination of MVDR beamformer and a post single channel Wiener filter [56]. Thus, MWF beamformer can be viewed as dual optimization in the spatial and time-frequency domains.

MVDR beamformer are pure spatial filters in the traditional sense. It uses directional information to construct weight vectors, suppress the signal in the interference direction in the spatial domain, and ensure that the signal in the target direction passes without distortion. However, since MVDR only suppresses signals from directions other than the target direction, significant residual noise will remain when the noise power is high.

In comparison, the multi-channel Wiener filter (MWF) can be viewed as further introducing the statistical characteristics of the time-frequency domain based on MVDR spatial filtering. Starting from the global optimization goal of the Minimum Mean Square Error (MMSE), the target signal is selected in the spatial direction, and the second-order statistical characteristics of the signal and noise are further used to perform additional spectral post-filtering optimization on the output result. Specifically, while implementing spatial filtering, MWF also achieves frequency domain smoothing noise reduction similar to a single-channel Wiener filter, thereby further reducing residual noise while ensuring signal fidelity. In this way, MWF can be decomposed into a "MVDR spatial beamforming + single-channel Wiener post-filtering" structure, thereby suppressing residual noise on the spectrum through post-filtering while ensuring distortion-free gain in the direction of the target signal.

4.4 Noise Covariance Matrix Estimation

As shown above, the noise covariance matrix $\mathbf{R_n}$ and the noisy covariance matrix $\mathbf{R_{x_B}}$ are required to construct the beamformer. In a practical real-time process stream, at

step 0, we can use the noise covariance matrix $\mathbf{R_n}$ as initialization for $\mathbf{R_{x_B}}$, with newly received data, we can continuously update the noisy covariance matrix $\mathbf{R_{x_B}}$ as shown below:

$$\begin{aligned} \text{At step 0 do calibration:} \quad & \hat{\mathbf{R}}_{\mathbf{x_{B_0}}} \leftarrow \hat{\mathbf{R}}_{\mathbf{n_0}}, \\ \text{At step } n \text{ do update:} \quad & \hat{\mathbf{R}}_{\mathbf{x_{B_n}}} \leftarrow \alpha \hat{\mathbf{R}}_{\mathbf{x_{B_{n-1}}}} + (1-\alpha)\mathbf{x_{B_n}}\mathbf{x_{B_n}}^H. \end{aligned} \tag{4.29}$$

where $\mathbf{x_{B_n}} \in \mathbb{C}^M$ is the latest data vector in the STFT domain, $\mathbf{R_{x_B}} \in \mathbb{C}^{M \times M}$ is the covariance matrix, and $\alpha \in \mathbb{R}$ is a scalar indicating the influence of new data on old data, usually taking a value between 0 and 1.

Now we need to find a good way to estimate the noise covariance matrix $\mathbf{R_n}$.

4.4.1 Noise Calibration

Assuming a spatially stationary environment, a simple idea for estimating the noise is that at the beginning of the algorithmic process, we can collect a few seconds of data to calculate the noise covariance and use it as a rough estimate of the noise signal, which can be represented as the following,

$$\widehat{\mathbf{R}}_{n_0} \leftarrow \frac{1}{L}\Sigma_{i=1}^{L}\mathbf{n}_i\mathbf{n}_i^H \tag{4.30}$$

where $\mathbf{n} \in \mathbb{C}^M$ is the noise vector in the STFT domain, L is the time frame length of calibration segments in the STFT domain, and $\hat{\mathbf{R}}_\mathbf{n} \in \mathbb{C}^{M \times M}$ is the estimated noise covariance matrix.

4.4.2 Voice Activity Detector (VAD)

Voice Activity Detector (VAD) is a crucial technique in audio signal processing that distinguishes between target speech periods with additive noise and noise-only periods. Based on the detection theory, VAD can be modeled as a two-hypothesis testing problem.

$$\begin{aligned} H_0&: \quad x(n) = n(n) \qquad\quad n = 0, 1, \ldots, N-1 \\ H_1&: \quad x(n) = n(n) + s(n) \quad n = 0, 1, \ldots, N-1 \end{aligned} \tag{4.31}$$

where H_0 represents the current frame only containing noise $\mathbf{N}$. H_1 represents the current frame containing noise $\mathbf{N}$ and speech $\mathbf{S}$.

In practice, the covariance matrices in DGC system may need to be updated in real-time with a Voice Activity Detector (VAD) for two reasons:

- The spatial stationarity is not guaranteed. Or specifically, target source moves.
- During the algorithm process, there are unexpected new noises that have not been calibrated.

$\mathbf{R_n}$ can be estimated in the "noise only period" and $\mathbf{R_{x_B}}$ can be estimated in the "speech and noise periods". If the VAD detects both speech and noise present, it

estimates the noisy covariance matrix as shown in Equation 4.29. If the VAD detects only noise, it estimates the noise covariance matrix as shown below,

$$\begin{aligned}\hat{\mathbf{R}}_{\mathbf{n_n}} &\leftarrow \alpha\hat{\mathbf{R}}_{\mathbf{n_{n-1}}} + (1-\alpha)\mathbf{n_n}\mathbf{n_n}^H, \text{ without target signal detected} \\ \hat{\mathbf{R}}_{\mathbf{x_{B_n}}} &\leftarrow \alpha\hat{\mathbf{R}}_{\mathbf{x_{B_{n-1}}}} + (1-\alpha)\mathbf{x_{B_n}}\mathbf{x_{B_n}}^H, \text{ with target signal detected}\end{aligned} \tag{4.32}$$

VAD system development has been an active field for the past decades. The implementation ideas of VAD mainly include threshold-based methods [57][58][59], statistical probability-based methods [60][61], and deep learning-based methods [62][63]. In earlier times, frame-based Voice Activity Detector (VAD) systems usually used two Gaussian Mixture Models (GMMs), trained on speech frames and non-speech frames, respectively, to estimate the probability of each frame belonging to speech. To ensure the temporal continuity of the detection results, this method introduced a hidden Markov model (HMM) to optimize the prediction by limiting the frequent switching between speech and non-speech states [60].

In this study, we only considered the performance improvement of the ideal VAD. Web Real-Time Communication (WebRTC) VAD [64] was developed for the WebRTC project, a set of open source projects and protocol stacks defined and standardized by the Internet Engineering Task Force (IETF) and World Wide Web Consortium (W3C) that enable real-time transmission of voice, video, and data over the Internet, and is a good component for embedding practical implementations into algorithms. WebRTC VAD is an example of a GMM-based VAD model whose input features are the logarithmic energies of six frequency bands between 80 Hz and 4000 Hz. By using fixed-point arithmetic, WebRTC VAD is optimized for real-time use over Internet transmissions.

4.5 Chapter Summary

In this chapter, we propose several signal processing methods for extracting the desired signal from multi-microphone systems in various scenarios, based on the previously constructed signal models. We first outline the key assumptions, including the uncorrelated nature of the target and noise signals, the Gaussian nature of the microphone self-noise, and the broadband nature of all considered signals.

We first introduce the Minimum Variance Distortionless Response Filter (MVDR) beamformer, also known as the Capon beamformer. The MVDR beamformer minimizes the output power of the array while maintaining an undistorted response to the desired signal. We present the filter design in detail and discuss the RTFs estimation method. We also discuss the limitations of MVDR, in particular, its inability to effectively handle scenarios with multiple desired loudspeakers, as it can only retain one target signal from the reference microphone.

To address multi-speaker scenarios, we propose the Linearly Constrained Minimum Variance (LCMV) beamformer. LCMV extends MVDR by incorporating multiple linear constraints, enabling it to preserve multiple signals of interest from different directions while minimizing interference and noise. We also address the challenges of estimating ATFs in systems with large apertures or distributed microphone arrays, making it a less than perfect method.

Recognizing the limitations of MVDR and LCMV in complex environments, we introduce Multichannel Wiener Filter (MWF). MWF aims to produce Minimum Mean Square Error (MMSE) estimates of the desired speech components by jointly optimizing noise reduction and signal distortion. We explore filter designs for MWF and discuss its advantages in balancing noise suppression and preserving multiple desired signals.

We also discuss the key task of noise covariance matrix estimation, which is critical to the performance of the aforementioned beamforming techniques. We explore methods for estimating the noise covariance matrix, including initial noise calibration and continuous updating using Voice Activity Detector (VAD). VAD helps distinguish between speech-plus-noise periods and noise-only periods, enabling real-time updating of the covariance matrix to accommodate nonstationary environments and unexpected noise.

In summary, this chapter presents a comprehensive set of beamforming methods applicable to different scenarios involving single or multiple speakers. By analyzing the theoretical foundations, practical considerations, and limitations of the MVDR, LCMV, and MWF beamformers, we discuss the development of robust signal processing algorithms that can operate effectively in challenging acoustic environments.

5 Simulation and Result

The previous chapter described the proposed beamforming strategy for single or multiple sound sources. In this chapter, we will numerically simulate the proposed method.

5.1 Evaluation Metrics

For one speaker, we use the clean Bluetooth channel signal as the comparison reference, which is $s_{ref}(n) = h_{AA}^{acous.}(n) * h^{BT}(n) * s_A(n)$, where $s_{ref}(n) \in \mathbb{R}$. For two speakers, we use the combination of clean speech signals $s_{ref}(n) = h_{AA}^{acous.}(n) * h^{BT}(n) * s_A(n) + h_{CC}^{acous.}(n) * h^{BT}(n) * s_C(n)$ as the reference signal.

Short-Time Objective Intelligibility (STOI)

Proposed by Cees H. Taal [65], Short-Time Objective Intelligibility (STOI) is a speech intelligibility metric comparing the processed speech signal with a clean reference signal. Based on the description in his paper, STOI is based on the correlation between the temporal envelopes of clean and degraded speech in a short segment (382 milliseconds). Through extensive comparison, STOI has advantages over other intelligibility comparison methods. The code implementation came from GitHub pystoi [66]. However, it should be noted that STOI is not designed to be used for mixing multiple reference signals.

STOI has a score ranging from 0 to 1, with 0 meaning poor intelligibility and 1 meaning high intelligibility. The STOI between the reference signal $s_{ref} \in \mathbb{R}^N$ and the processed signal $s_{result} \in \mathbb{R}^N$ is computed as:

$$\text{STOI} = stoi(s_{ref}(n), s_{result}(n)) \tag{5.1}$$

Frequency-weighted Segmental SNR

The Segmental SNR is an objective measure of speech enhancement based on the geometric mean of the SNR of all frames in a speech segment, which can be evaluated either in the time or frequency domain.

The silence segment may greatly impact the criteria performance due to its low energy as its logarithm result leads to negative infinity. To avoid the influence of the silence segment on the criteria performance, a common method is to use threshold comparison to exclude the silence segments. Another way is to apply perceptual weighted filters to constrain the output range. By perceptually weighting the filter coefficients, we can compute the segmental SNR based on the outputs of the filters after passing the clean and processed signals through them.

Richard [67] proposed a weighted strategy as follows:

$$\text{SNRseg} = \frac{10}{M}\sum_{m=0}^{M-1}\log_{10}\left(1+\frac{\sum_{n=Nm}^{Nm+N-1}x^2(n)}{\sum_{n=Nm}^{Nm+N-1}(x(n)-\hat{x}(n))^2}\right), \tag{5.2}$$

which limits the minimum value of the output to 0 dB rather than negative infinity. Its frequency extension is shown as,

$$fwSNRseg = \frac{10}{M}\sum_{m=0}^{M-1}\frac{\sum_{j=1}^{K}W_j\log_{10}\left[X^2(j,m)/\left(X(j,m)-\hat{X}(j,m)\right)^2\right]}{\sum_{j=1}^{K}W_j} \tag{5.3}$$

where:

W_j represents the weight assigned to the j-th frequency band,

K denotes the total number of frequency bands,

M refers to the total number of frames in the signal,

$X(j,m)$ is the filter-bank magnitude of the clean signal at the j-th frequency band and m-th frame,

$\hat{X}(j,m)$ is the filter-bank magnitude of the enhanced signal in the same frequency band.

Normalized Mean Square Error (NMSE)

The Normalized Mean Square Error (NMSE) quantifies the difference between the estimated signal and the reference signal. Normalizing the Mean Square Error (MSE) by the entire energy of the reference signal, NMSE gives a dimensionless measure of error. In this case, we use the clean Bluetooth channel signal, s_{ref}, as the reference.

The NMSE between the reference signal $s_{ref} \in \mathbb{R}^N$ and the processed signal $s_{result} \in \mathbb{R}^N$ is computed as:

$$\text{NMSE} = \frac{\sum_{n=1}^{N}|s_{ref}(n)-s_{result}(n)|^2}{\sum_{n=1}^{N}|s_{ref}(n)|^2} \tag{5.4}$$

5.2 Simulation Set Up

The proposed methods are evaluated in the context of dereverberation and noise reduction. The geometry of the acoustic setting is shown in Figure 5.1. The acoustic geometry is a group of users, each equipped with a maximum of four microphones within a closed shoe-box size room of size 10 meter $\times$ 10 meter $\times$ 2.4 meter. The sources used were speech signals from the TCD Timit audio-visual speech database [68], representing the words spoken by the speaker, and an impact drill interference source downloaded from the open-source website FreeSound [69], both with a duration of 12 s. Each speech signal undergoes convolution with the room impulse responses in the time domain. The room impulse responses are generated using the image source model and ray tracing principle described in section 2.2.2 with the Python library pyroomacoustics [70]. The room reflection coefficient is set to 0.95.

In this experiment, the following parameters are used: the sampling rate f_s =16 kHz, and the sampled noisy microphone signals are processed by STFT for all sub-time frames with a 32 ms hamming window function and 50 % overlapping. The FFT length is 512 points. The microphone self-noises are assumed to be zero-mean uncorrelated Gaussian process with a variance σ_v^2, calculated as described in section 2.5.1 for each user.

According to Dopple, the Bluetooth channel latency in the system is stable. To avoid the randomness of the results, we assume that the Bluetooth channel transfer function in the system has slight delay fluctuations in the millisecond range between channels.

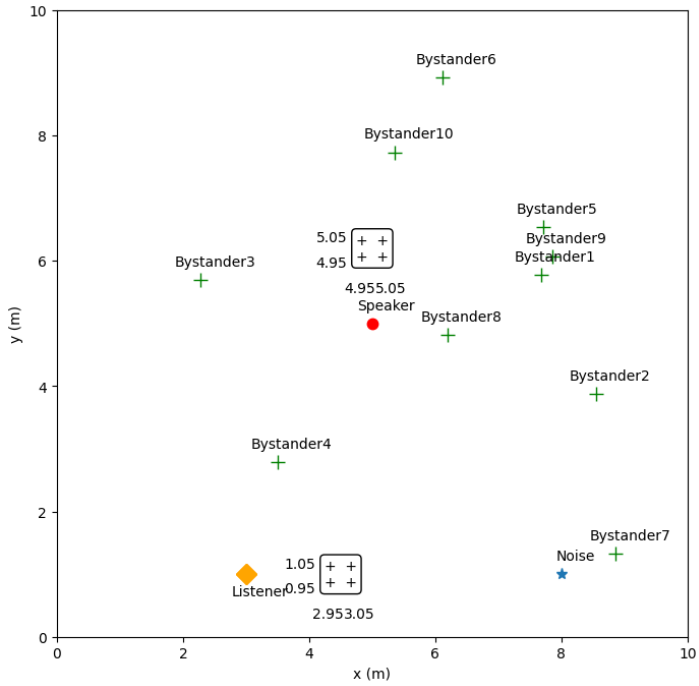


Figure 5.1: Top view of the acoustic scene with 10 bystanders.

The room geometry is shown in Figure 5.1. The DGC system is designed to involve people between 3 people to 10 people. The user number and the speaker number are assumed to be known in the system. The four microphones are designed to be distributed on both sides of the human ear, 2 on each side. The square block around the speaker and listener shows the general form of microphone array distribution for users. For example, the speaker is at position (5,5), the listener is at position (3,1), and the plot shows the corresponding speaker's and listener's microphone array positions. For simplicity, the orientation of people is not considered here.

5.3 Simulation Result

5.3.1 Single Speaker Case

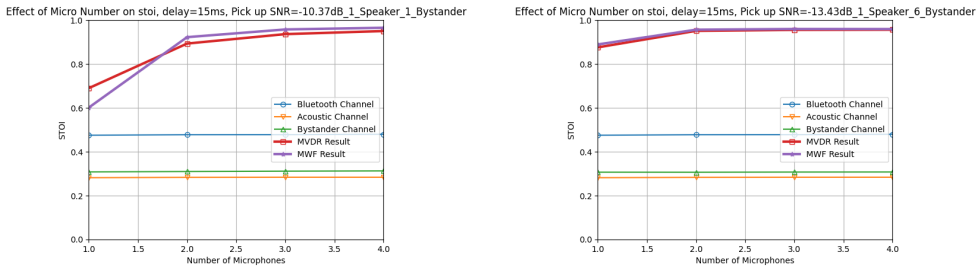
Regarding the system performance, we investigated the parameters involved in the signal models which are the Bluetooth channel latency, the Signal-to-Interference Ratio (SIR) of the sources. As the number of sensors available increases, we can obtain more information and get better results. Therefore, we also need to investigate the relationship between the total number of microphones and the results of the algorithm.

The impact of the number of microphones

As the number of microphones enabled per user increases, the number of signals received by the listener on the same channel will also increase accordingly, and the microphone signals received on the same channel are copies of each other with phase shift and attenuation in case of no interference source and noise. According to the beamforming theory, as we receive more signals, we get more information and we should be able to get better performance. During the design phase, we considered more than one microphone per user. We need to find out how many microphones are necessary.

In reality, more microphones bring higher performance at the cost of higher computational complexity.

The following plot is drawn in the setting of a typical Bluetooth channel latency of 15 ms, SIR of -20 dB.



(a) 1 Bystander (b) 6 Bystanders

Figure 5.2: relationship of enabled microphones and STOI

Experiments show that the more microphones there are, the better the speech intelligibility we get. The improvement is significant when the number of microphones increases from 1 to 2, but it is not significant after that. This is consistent with our expectation of the gain of the array, that is, increasing the number of antennas can increase the average SNR at the receiving end through a coherent combination of multi-path signals. The array gain is strongly correlated with the logarithm $lg(M)$ of the number of sensors M, i.e., by doubling the number of microphones, the SNR at the output increases by only 3 dB [19].

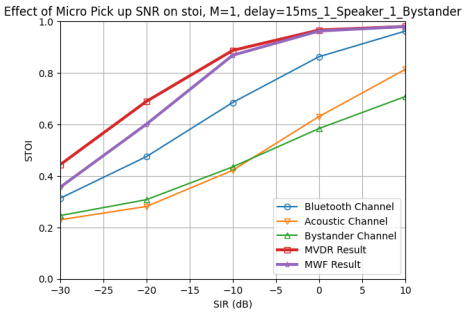
In addition, we observed that when there are more bystanders, speech intelligibility improves with the same number of microphones. This is a reasonable result because intuitively it would seem that increasing the number of microphones or the number

of bystanders would result in more knowledge and beamforming could achieve better results. Mathematically, with more channels, the signal available to the beamforming increases, and beamforming can achieve a more refined combination to recover the signal. This also suggests that we need to ensure a sufficient number of signals to obtain stable performance during the operation of the algorithm. More specifically, when the number of users in the system is small, we need to enable additional microphones to ensure the performance.

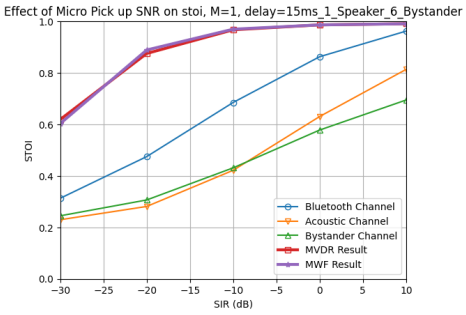
The impact of SIR

Due to the bad acoustic environment that can be encountered, the user will be in an extremely noisy environment. As SIR continues to decrease, we can foresee that the quality of the voice picked up by the microphone will decrease, which will significantly impact the performance of the algorithm. We hope to understand under what circumstances the algorithm will perform poorly, which leads us to understand the relationship between SIR and algorithm performance.

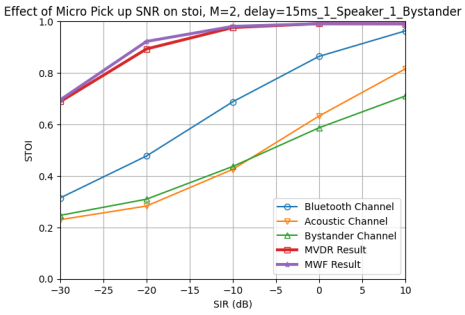
Here we consider using the global SIR, which is the logarithmic ratio of the speaker's speaking volume to the interference source volume. Due to the foreseeable noisy usage environment, the following figure is drawn when global SIR is [-30, -20, -10, 0, 10] dB with a typical Bluetooth channel latency 15 ms.



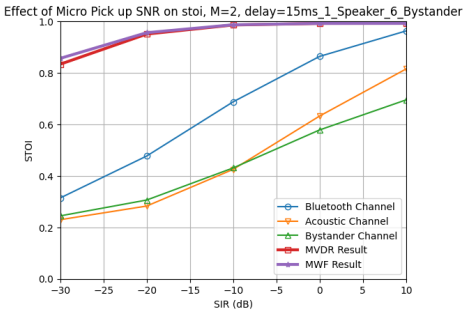
(a) 1 Bystander with 1 mic each user



(b) 6 Bystanders with 1 mic each user

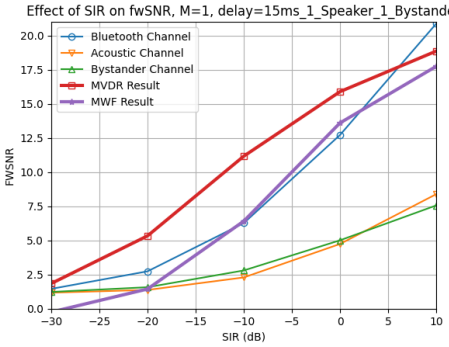


(c) 1 Bystander with 2 mic each user

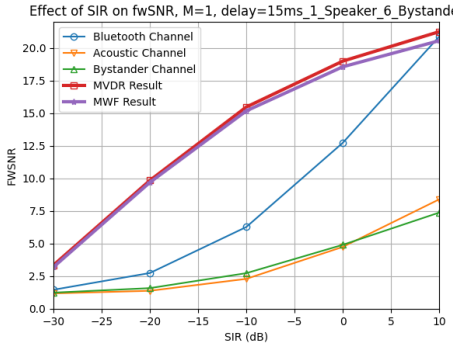


(d) 6 Bystanders with 2 mic each user

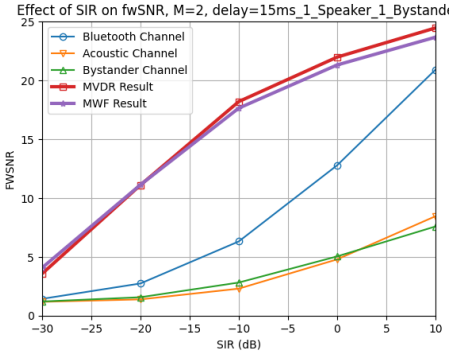
Figure 5.3: Relationship of SNR and STOI



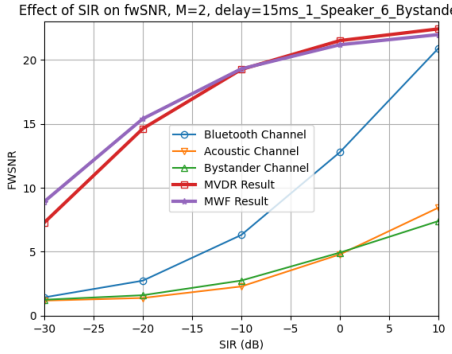
(a) 1 Bystander with 1 mic each user



(b) 6 Bystanders with 1 mic each user

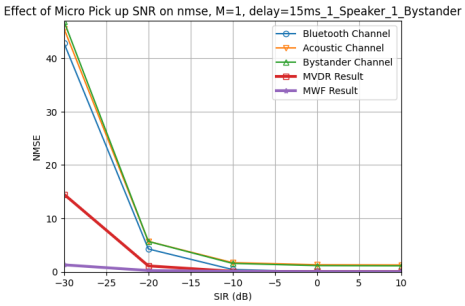


(c) 1 Bystander with 2 mic each user

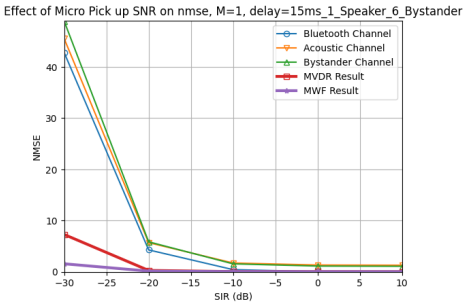


(d) 6 Bystanders with 2 mic each user

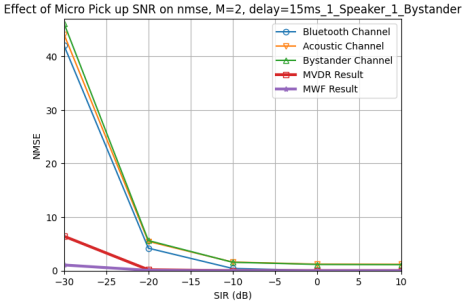
Figure 5.4: Relationship of SNR and fwSNR



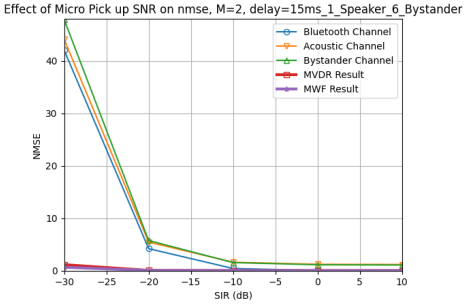
(a) 1 Bystander with 1 mic each user



(b) 6 Bystanders with 1 mic each user



(c) 1 Bystander with 2 mic each user

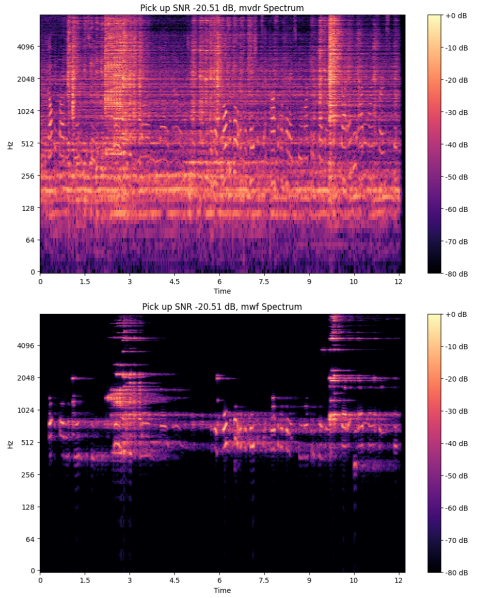


(d) 6 Bystanders with 2 mic each user

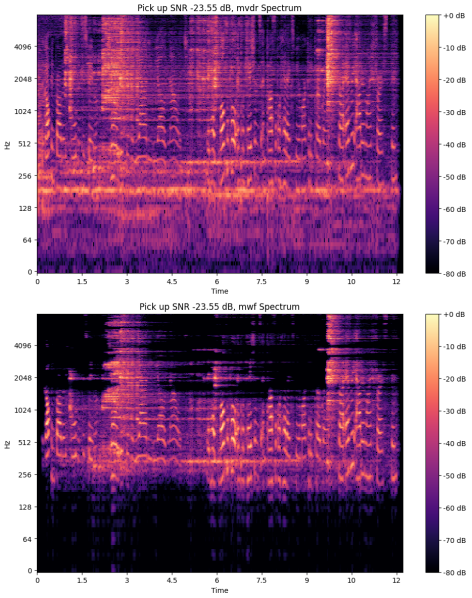
Figure 5.5: Relationship of SNR and NMSE

Naturally, we see that as SIR increases, the performance of the system improves. We observe that MWF performs slightly better than MVDR in bad acoustic environments, except for the case of insufficient information (i.e., when there are few bystanders and insufficient number of microphones). When SIR is good, MWF performs slightly worse than MVDR. It is due to the fact that MWF filter can be decomposed as a spatial filtering and a post-filtering dealing with the residual noise. The post-filtering is based on the covariance matrix which undergoes smoothing, so the estimate is not accurate, which will damage the results under high SIR conditions.

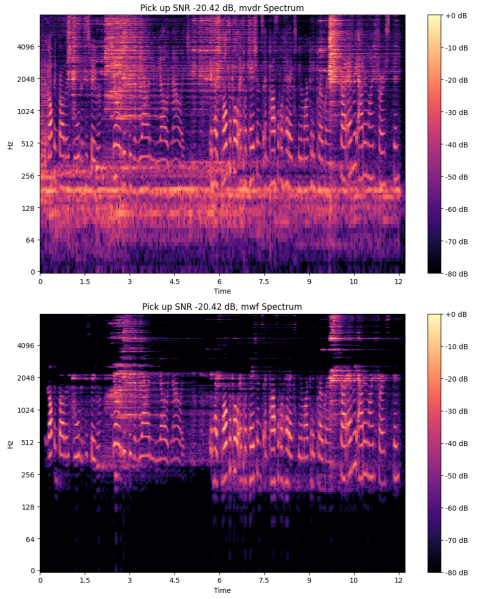
It is worth noting that when there is only one speaker and one bystander (insufficient information), the MWF beamformer has a smaller nmse error, but poorer speech intelligibility and fwSNR. We believe that the reason is that the MWF beamformer incorrectly suppresses part of the speech signal. The suppressed portion of the spectral content disappears. This result can also be verified from the spectrogram.



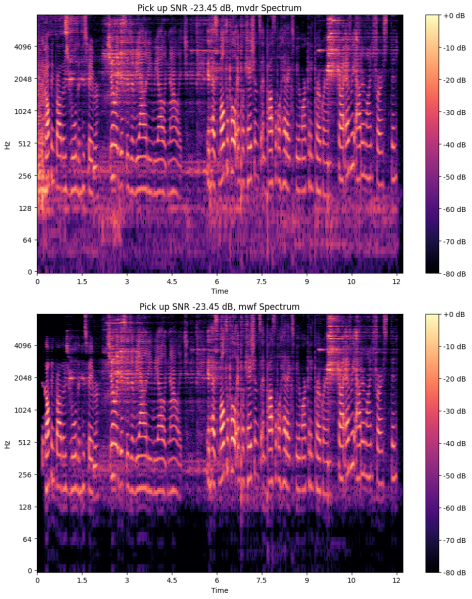
(a) Spectrum of 1 Bystander with 1 mic each user with SIR = -30 dB



(b) Spectrum of 6 Bystanders with 1 mic each user with SIR = -30 dB



(c) Spectrum of 1 Bystander with 2 mic each user with SIR = -30 dB



(d) Spectrum of 6 Bystanders with 2 mic each user with SIR = -30 dB

Figure 5.6: Enhanced spectrum plots in case of 1 speaker when SIR = -30 dB

The figure above shows the enhanced spectrum with the first one the MVDR spectrum and the second the MWF spectrum at each subplot. These plots explain well that the MWF beamformer can be decomposed into a combination of the MVDR beamformer and a single-channel post-Wiener filter, which is accordingly a combination of spatial filtering and time-frequency filtering. In the Figure 5.6, we can see that MWF beamformer eliminated part of the noise from MVDR beamformer result. In

Figure 5.6a, the reason why insufficient information leads to intelligibility degradation is MWF incorrectly suppresses the speech part.

The impact of Bluetooth channel latency

In Equation 3.15, we can see two delays, one is the delay included in the Bluetooth channel, and the other is the delay included in the acoustic channel. We did not consider the delay in the acoustic channel because the propagation attenuation of sound conforms to the following formula:

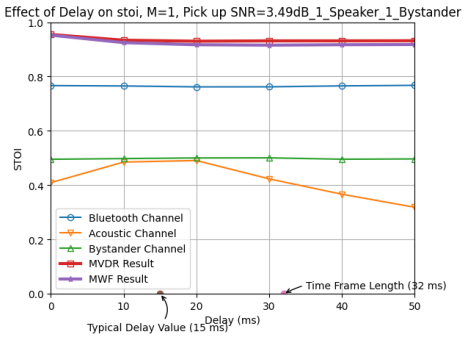
$$L \propto \frac{1}{r^2}, \tag{5.5}$$

where L is the SPL, and r is the distance. If the distance is far, the acoustic transmission channel will be negligible, while the delay difference can be ignored when the distance is close. However, the delay of the Bluetooth channel may vary relatively significantly in a short period of time, so it is worth studying the robustness of the algorithm to the Bluetooth channel latency.

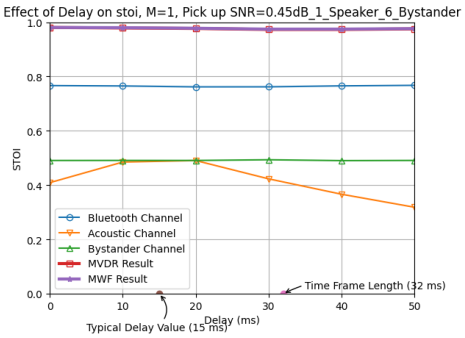
According to Dopple, the Bluetooth channel latency among different channels can be assumed to be the same thus in Equation 3.15, the Bluetooth channel transfer function $\Delta^{BT}(k)$ was assumed to be the same. However, in the simulation, assuming that the delays between different Bluetooth channels are the same would reduce the credibility of the results. Therefore, millisecond-level latency differences were created between different Bluetooth channels.

The following plot is drawn in the setting of SIR $= -6\,\mathrm{dB}$. A typical Bluetooth channel latency of 15 ms and a time frame length of 32 ms are marked on the horizontal axis.

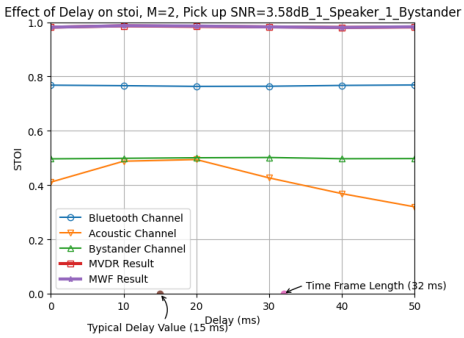
$$\text{time frame length} = \frac{512}{Fs} = \frac{512}{16000} = 32(\text{ms})$$



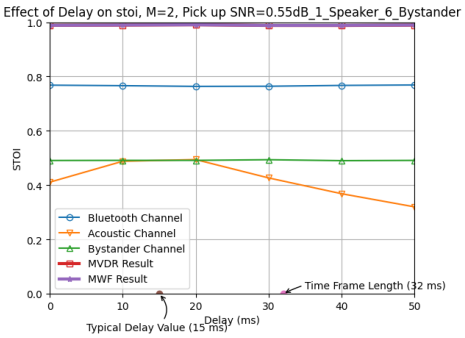
(a) 1 Bystander with 1 mic each user



(b) 6 Bystanders with 1 mic each user

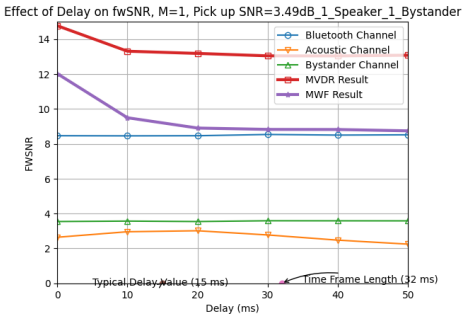


(c) 1 Bystander with 2 mic each user

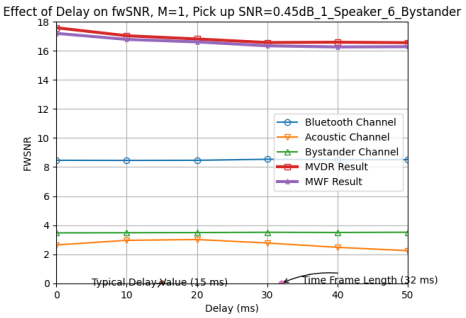


(d) 6 Bystanders with 2 mic each user

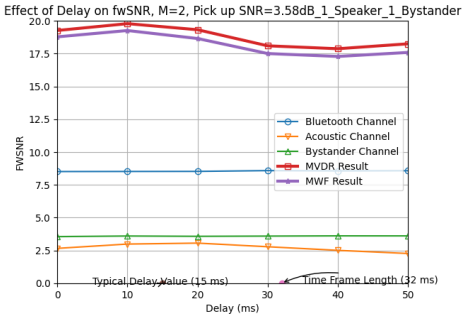
Figure 5.7: Relationship of Bluetooth Channel latency and STOI



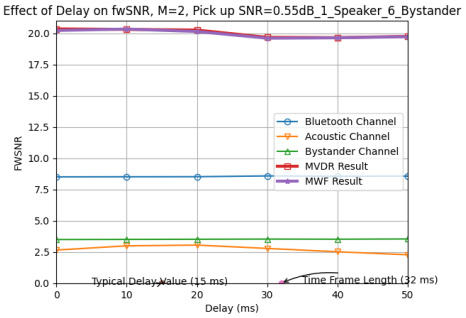
(a) 1 Bystander with 1 mic each user



(b) 6 Bystanders with 1 mic each user

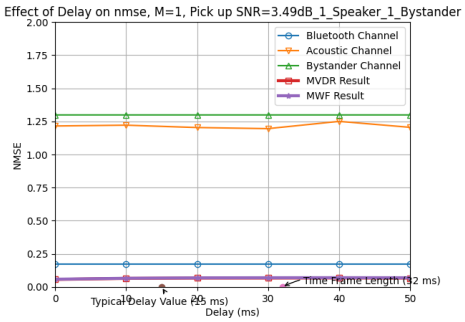


(c) 1 Bystander with 2 mic each user

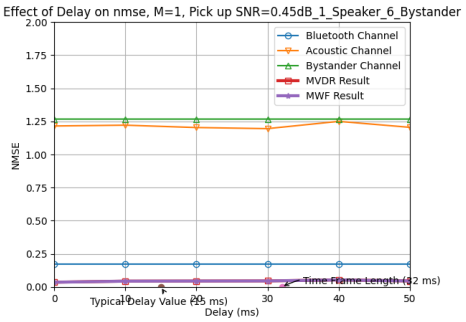


(d) 6 Bystanders with 2 mic each user

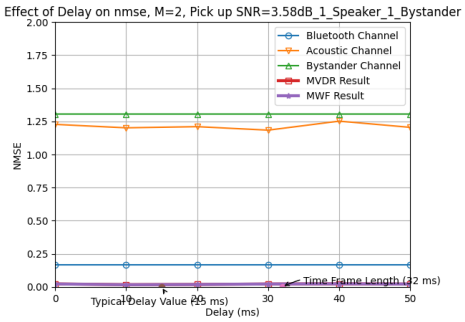
Figure 5.8: Relationship of Bluetooth Channel latency and fwSNR



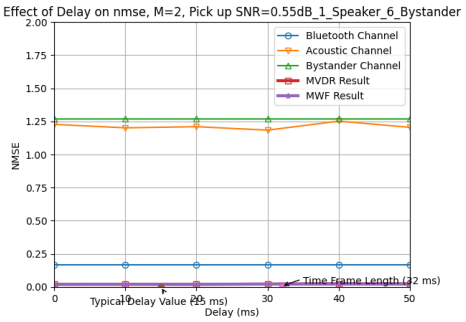
(a) 1 Bystander with 1 mic each user



(b) 6 Bystanders with 1 mic each user



(c) 1 Bystander with 2 mic each user



(d) 6 Bystanders with 2 mic each user

Figure 5.9: Relationship of Bluetooth Channel latency and NMSE

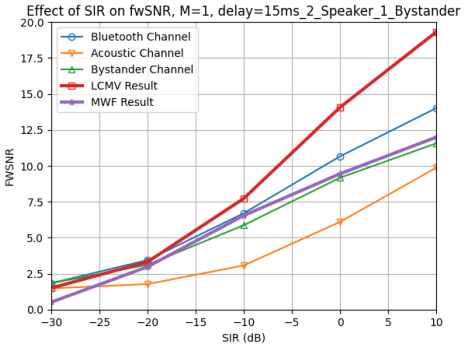
We can see that beamforming is effective and shows stable performance under different Bluetooth channel latency, and is quite robust to latency. From the experiment, we can conclude the beamforming method is robust against Bluetooth channel latency.

5.3.2 Multiple Speakers Case

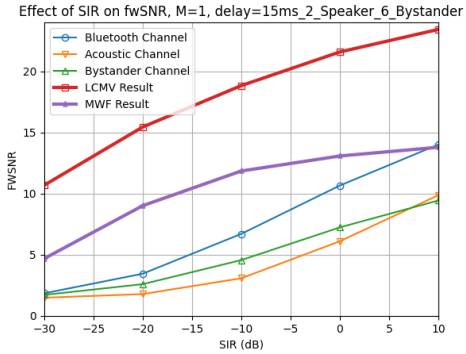
In this section, we present simulations under multiple sound sources. It should be noted that, as discussed before, estimating ATFs for each sound source is a challenging task. The LCMV plot here is based on the results of the true ATFs.

The impact of SIR

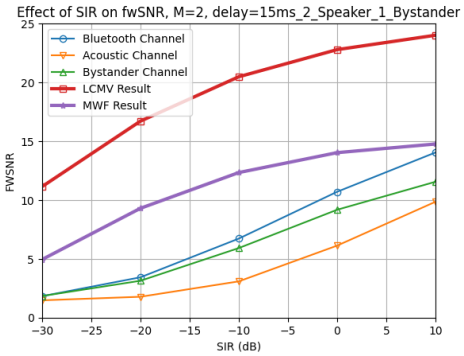
For the same reasons as before, we need to study the impact of SIR in the multiplayer case, so we use the same settings as the single speaker, with global SIR [-30, -20, -10, 0, 10] dB and a typical Bluetooth channel latency 15 ms.



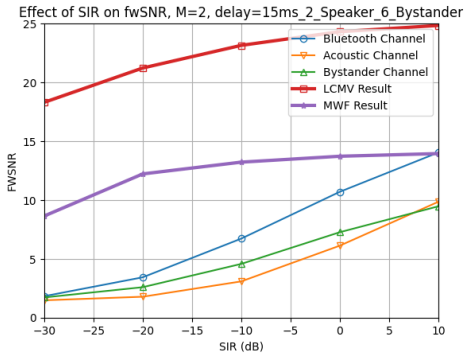
(a) 1 Bystander with 1 mic each user in case of 2 speakers



(b) 6 Bystanders with 1 mic each user in case of 2 speakers

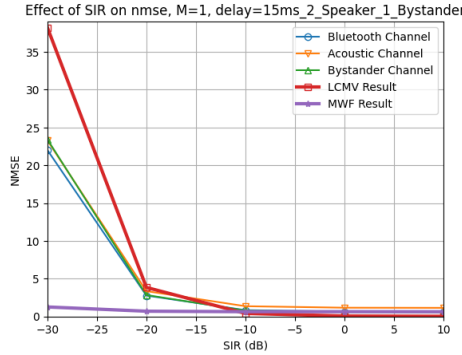


(c) 1 Bystander with 2 mic each user in case of 2 speakers

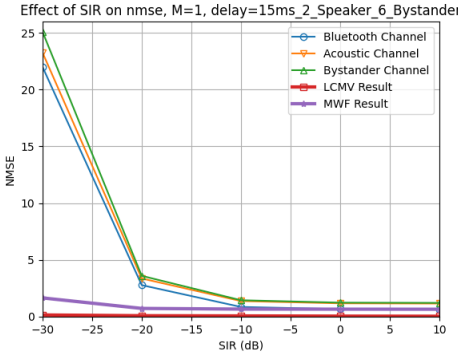


(d) 6 Bystanders with 2 mic each user in case of 2 speakers

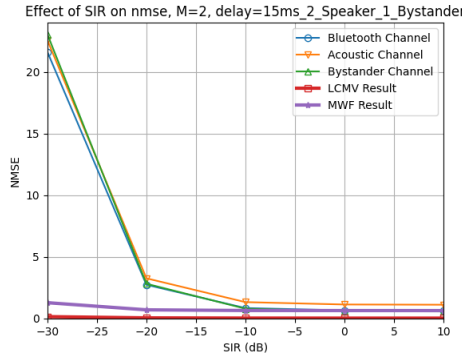
Figure 5.10: Relationship of SNR and fwSNR in case of 2 speakers



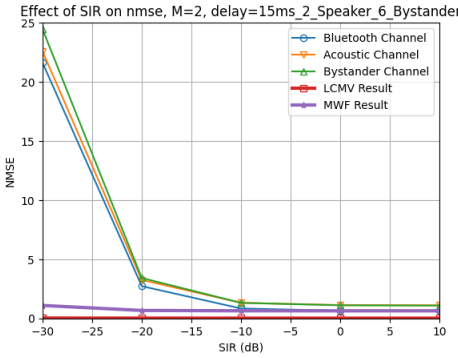
(a) 1 Bystander with 1 mic each user in case of 2 speakers



(b) 6 Bystanders with 1 mic each user in case of 2 speakers



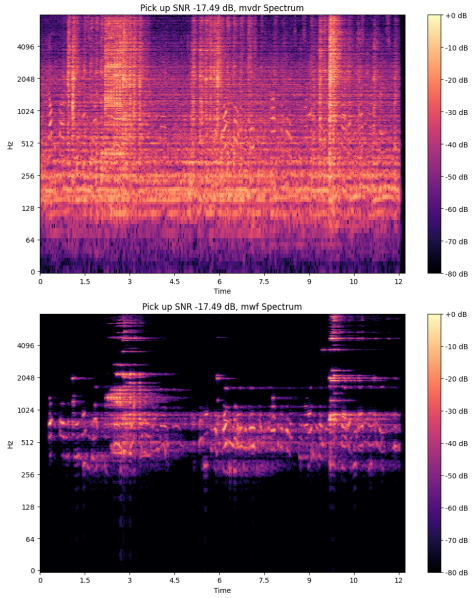
(c) 1 Bystander with 2 mic each user in case of 2 speakers



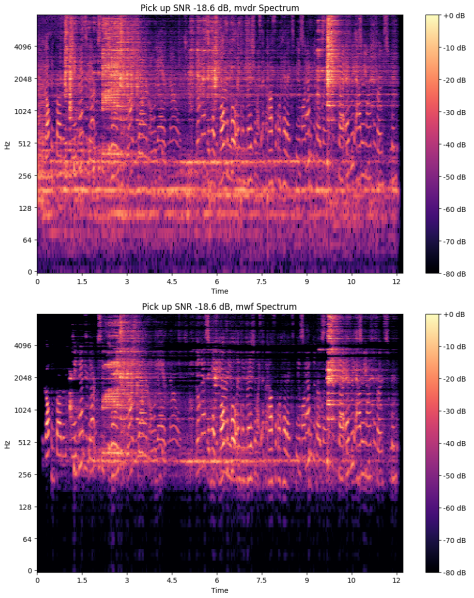
(d) 6 Bystanders with 2 mic each user in case of 2 speakers

Figure 5.11: Relationship of SNR and STOI in case of 2 speakers

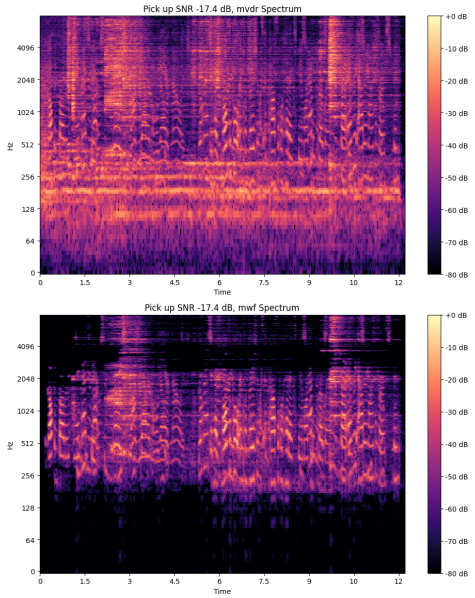
Based on the fwSNR results, we can see that MWF performs well except when there is insufficient information (again, a small number of bystanders and a small number of microphones). Comparing the results of MWF and theoretical LCMV, we can see that the theoretically achievable performance is higher. A noteworthy phenomenon is that when information is insufficient, the fwSNR of theoretical LCMV is better than that of MWF, but theoretical LCMV NMSE is higher. The reason is the same as before. The MWF suppresses most of the time-frequency segment due to inaccurate post-filtering, only a few segments are retained. Since most of the spectrum is empty, its NMSE error is small. It can be verified in the following spectrum plot.



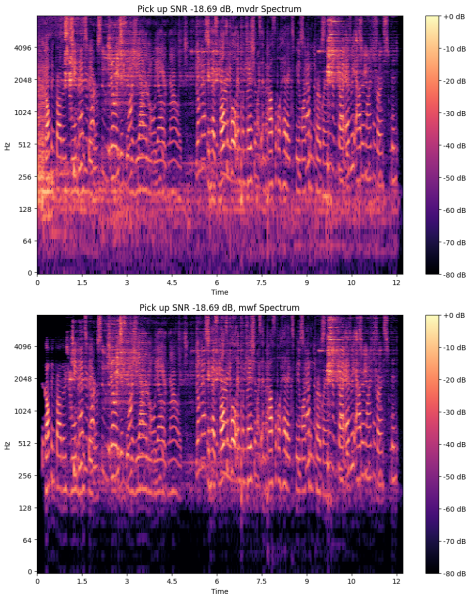
(a) Spectrum of 1 Bystander with 1 mic each user, SIR = -30 dB in case of 2 speakers



(b) Spectrum of 6 Bystanders with 1 mic each user, SIR = -30 dB in case of 2 speakers



(c) Spectrum of 1 Bystander with 2 mic each user, SIR = -30 dB in case of 2 speakers



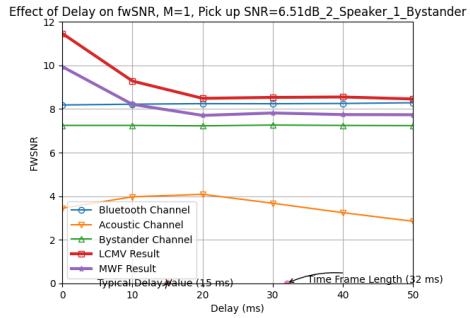
(d) Spectrum of 6 Bystanders with 2 mic each user, SIR = -30 dB in case of 2 speakers

Figure 5.12: A few spectrum plots in case of 1 speaker when SIR = -30 dB in case of 2 speakers

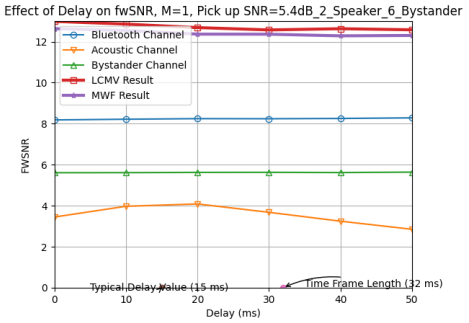
We can draw the same conclusion as before: with more channel information (enabled microphones per user or bystanders), the system performance improves.

The impact of Bluetooth channel latency

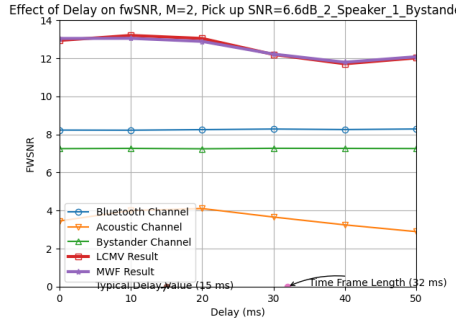
The plot setting is the same as a single speaker case, with SIR = −6 dB and 1 bystander.



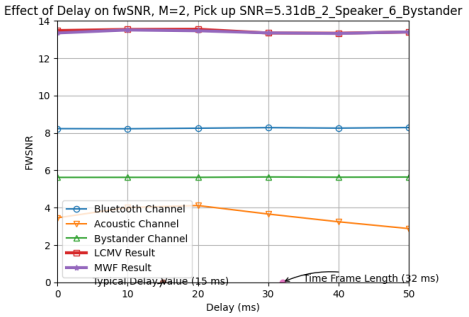
(a) 1 Bystander with 1 mic each user in case of 2 speakers



(b) 6 Bystanders with 1 mic each user in case of 2 speakers

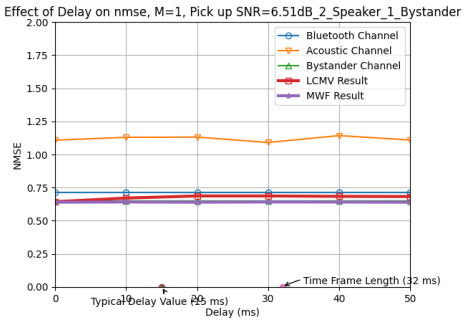


(c) 1 Bystander with 2 mic each user in case of 2 speakers

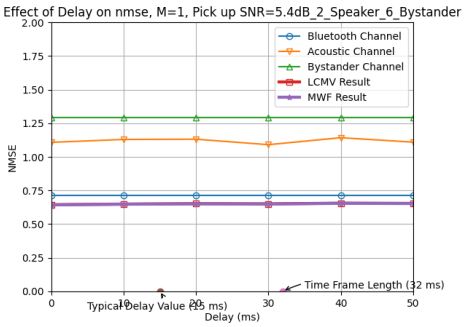


(d) 6 Bystanders with 2 mic each user in case of 2 speakers

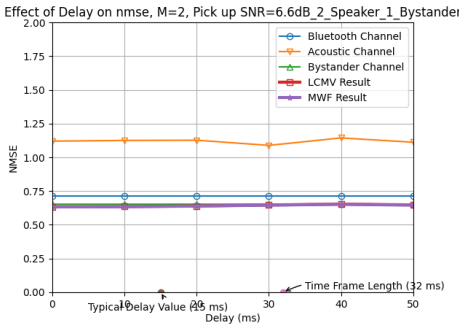
Figure 5.13: Relationship of Bluetooth Channel latency and fwSNR in case of 2 speakers



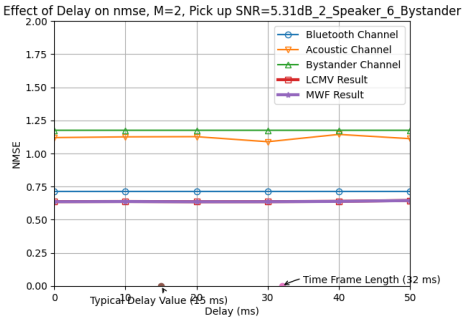
(a) 1 Bystander with 1 mic each user in case of 2 speakers



(b) 6 Bystanders with 1 mic each user in case of 2 speakers



(c) 1 Bystander with 2 mic each user in case of 2 speakers



(d) 6 Bystanders with 2 mic each user in case of 2 speakers

Figure 5.14: Relationship of Bluetooth Channel latency and STOI in case of 2 speakers

The performance remains the same as with a single speaker, i.e. the beamforming automatically takes care of the delay between channels. Therefore, by comparing the results of a single sound source, we can conclude that the Bluetooth channel latency has no influence on the performance regardless of the number of microphones.

5.4 Ideal VAD

As discussed before, VAD allows us to update the covariance matrix in case of moving target and uncalibrated noise, corresponding to the inaccurate acoustic transfer function and inaccurate noise covariance matrix. It is not easy to simulate with a moving target source, so we simulate with uncalibrated noise.

The noise source is divided into 2 different segments with their spectrum shown below. The first noise source is called “calibration noise” with a length of 5 s, followed by the second noise called “not recorded noise”. The calibration period is the first 2 s when the algorithm runs. Thus, the noise covariance matrix is calculated according to the “calibration noise” spectrum.

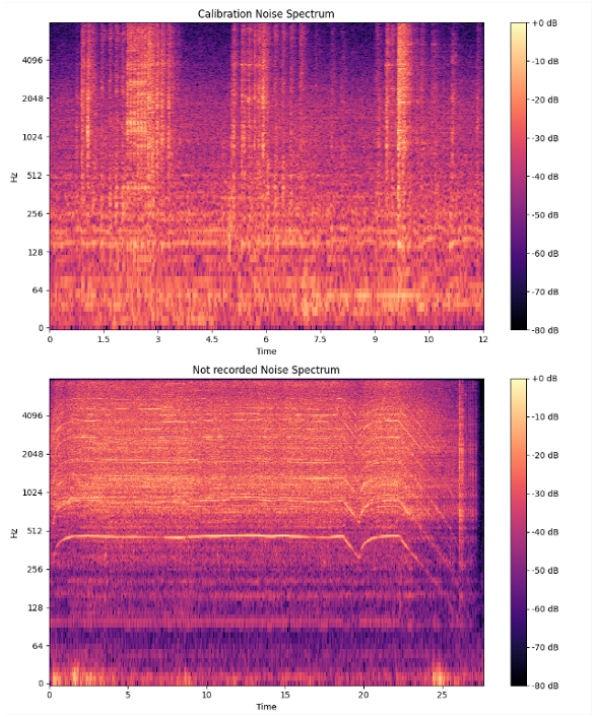


Figure 5.15: Spectrum of calibration noise segment and the one not recorded

The reference VAD result is obtained by applying *silero-vad*[71] to the clean speech audio. Figure 5.16 shows the plot of the speech occurrence time period detected by *silero-vad* and the clean sound waveform.

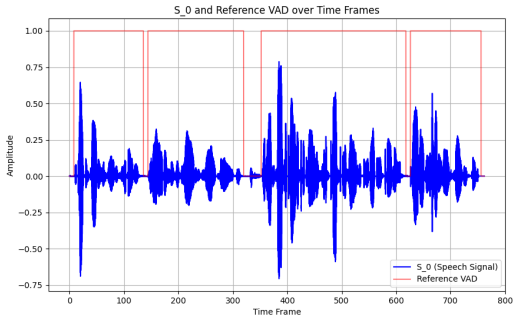


Figure 5.16: S_0 and Reference VAD over Time Frames

The simulation is done at the single-speaker scenario with a typical 15 ms Bluetooth channel delay and a global SIR $= -6$ dB, the criteria used is STOI. The result is shown below.

(a) 1 Bystander

(b) 6 Bystanders

Figure 5.17: comparison between ideal VAD and calibration only in case of uncalibrated noise

The results show that the ideal VAD brings an inconspicuous improvement independent of the number of bystanders. The algorithm does not crash in the presence of uncalibrated noise sources because even if there is an error in the estimation of the noise covariance matrix, the orientation of the noise in the signal space remains the same. That is, the ATF remains unchanged, and the beamforming direction is still correct, so the performance is not greatly affected. Since the source spatial information remains the same, the MVDR beamformer performance is basically not affected. For MWF beamformer, given the decomposition of "MVDR spatial beamforming + single-channel Wiener post-filtering", we can look at the problem in two parts. The performance of spatial filtering is comparable to that of MVDR, but the performance of post-filtering is much affected, meaning the noise residual is likely wrongly suppressed. That is all the reason for bigger improvement brought ideal VAD to MWF than MVDR.

Simulation of moving targets is of interest and is being actively explored.

5.5 Chapter Summary

In this chapter, we numerically evaluate the proposed beamforming strategy in various scenarios and conditions. Simulations in complex acoustic environments confirm that both single-speaker and multi-speaker algorithms are robust to Bluetooth channel latency and can effectively handle challenging conditions with low signal-to-interference ratios. Increasing the number of microphones and adding bystanders can provide more spatial information, further improving the performance. Notably, while the MVDR-based approach acts as a reliable spatial filter, the MWF-based approach (which can be interpreted as MVDR plus a Wiener postfilter) provides additional gains in noise suppression. Voice activity detection (VAD) brings only marginal improvements even in ideal situations, indicating that the spatial characteristics of the scene determine the performance of the proposed solution. In summary, simulations show that the proposed beamforming approach is versatile, robust, and scalable for distributed group communication systems operating in complex acoustic environments.

6 Conclusion and Future Work

In this chapter, we are going to conclude the work and give some possible directions for future work.

6.1 Conclusion

The DGC system is a new multi-person full-duplex communication protocol designed for Bluetooth radios. Echo effects and possible interference sources caused by acoustic coupling in close-range scenarios pose significant challenges to achieving high-quality communications. We addressed these issues through step-by-step problem formulation, modeling, algorithm development, and implementing beamforming techniques for the DGC system. Our work helps ensure that complex communication systems achieve effective and reliable communications in complex acoustic environments.

In Chapter 3, we focus on the problem formulation in the multi-user DGC scenario. We first abstract the Bluetooth channel as a virtual acoustic channel and build a detailed mathematical model to represent the received signal at each user end. The model reveals how the coupling and mixing of signals lead to echo. With increasing complexity, we extend the analysis to multiple active speakers. By introducing the corresponding acoustic and digital paths for each source, a comprehensive model of the multi-user environment is formed, laying the foundation for beamforming and signal separation strategies.

In Chapter 4, we propose several methods to extract the target signal in a multi-microphone system. First, we clarify several assumptions, then introduce the MVDR beamformer and discuss the estimation method of Relative Acoustic Transfer Functions (RTFs). Although MVDR performs well in a single-speaker environment, it faces limitations in multi-speaker scenarios. To address the problem of multi-speaker scenarios, we introduce the LCMV beamformer, but for distributed or large-scale microphone arrays, it is necessary to estimate RTFs at different reference points, which is challenging in practical applications. Subsequently, a multi-channel Wiener filter (MWF) can be used as the next step. MWF estimates the desired signal through the minimum mean square error criterion in multi-channel scenarios, achieving noise reduction while ensuring signal fidelity. Finally, we discuss the estimation method of the noise covariance matrix. In practical applications, preliminary noise statistics can be obtained through the calibration phase, and the voice activity detection (VAD) method can be used to update the received signal in real-time to adapt to the changes in the non-stationary acoustic environment and burst noise. In summary, this chapter proposes various solutions from single-speaker to multi-speaker scenarios based on the multi-source multi-channel signal model, including MVDR, LCMV and MWF.

In Chapter 5, we simulated and analyzed the results of the proposed beamforming

method in different scenarios. In the experiments, we focused on the impact of key system parameters on the performance, including Bluetooth channel delay, Signal-to-Interference Ratio (SIR), and the number of enabled microphones and bystanders. The results show that the beamforming scheme is insensitive to Bluetooth channel delay. In addition, under low SIR conditions, the algorithm can still robustly recover the target speech. Regarding the impact of the number of microphones, we found that when a single user is configured with 2 microphones, the performance is significantly improved. As the number of bystanders in the system increases, the additional observation channels are more beneficial to the beamforming algorithm. If there are fewer users in the system, appropriately increasing the number of microphones can make up for the lack of data and ensure the final processing effect. In general, the simulation results verify that the proposed MWF beamforming algorithm has good adaptability and robustness in multi-user distributed communication scenarios.

In summary, by abstracting the Bluetooth channel, a comprehensive signal model suitable for different scenarios is constructed. For different application scenarios, we discuss a variety of beamforming and filtering schemes, theoretically gradually extending from single target extraction to simultaneous recovery of multi-source speech, balancing the signal fidelity and noise suppression requirements. Additionally, by introducing ideal voice activity detection (VAD) and implementing dynamic methods for updating the noise covariance matrix, robustness can be maintained in time-varying acoustic environments. The results of the simulation support the theoretical hypothesis.

This work systematically analyzes and studies the speech processing problem in the Dopple Group Chat (DGC) system, and proposes a complete set of methods from signal modeling to beamforming strategy. Given the speech coupling echo and noise interference problems in multi-user full-duplex communication under the DGC architecture, an effective signal model and beamforming algorithm are proposed and verified, which provides a strong technical reference and direction guidance for the application of Dopple Group Chat (DGC) system in practical scenarios.

6.2 Future Work

Two main aspects can be explored in the future.

1. To distinguish between single-speaker and multi-speaker cases, we can perform detection by using VAD for each channel. However, we can explore more details. Multi-speaker speech signals collected by spatially separated microphones will have time delays with each other. Some research shows that it is possible to determine the number of speakers by determining the time delay [72][73] based on the direct components of the speech signals from the two microphones remaining unchanged. However, these approaches fail if the direct component is masked by high levels of ambient noise and reverberation. It would be interesting to investigate improving the performance of VAD or latency difference-based algorithms under robust conditions.

2. Speech information obtained under adverse acoustic conditions with high levels of non-stationary background noise will degrade speech intelligibility. The effectiveness of speech enhancement depends heavily on the accuracy of the noise covariance matrix. To cope with time-varying environments, we need to continuously update the noisy covariance matrix and the noise covariance matrix. Using VAD is a simple solution.
   Researchers have been continuously studying techniques for single-channel and multi-channel noise reduction, such as maximum likelihood (ML) [74][75][76], and maximum a posteriori (MAP) [74][77]. In addition, some popular deep neural network techniques have also performed well. The deep neural network makes a probability judgment on each time-frequency point based on the mask rule, thereby updating the covariance matrix. Its performance has been proven whether for the single-channel enhancement [78][79], or for the multi-channel enhancement [80][81]. It is also an interesting topic to investigate methods for providing a better noise covariance matrix.

Bibliography

[1] Wikipedia contributors. A-weighting — wikipedia, the free encyclopedia, 18/01/2024. [Online; accessed 18/04/2024].

[2] Halabi Hasbullah, Abas Md Said, and Kashif Nisar. The effect of echo delay on voice quality in voip network. In *Proceedings of the IASTED International Conference*, volume 1, page 200, 2009.

[3] Richard C Hendriks, Timo Gerkmann, and Jesper Jensen. *DFT-domain based single-microphone noise reduction for speech enhancement*. Springer Nature, 2022.

[4] Jae Soo Lim and Alan V Oppenheim. Enhancement and bandwidth compression of noisy speech. *Proceedings of the IEEE*, 67(12):1586–1604, 1979.

[5] Sunil Kamath, Philipos Loizou, et al. A multi-band spectral subtraction method for enhancing speech corrupted by colored noise. In *ICASSP*, volume 4, pages 44164–44164. Citeseer, 2002.

[6] S Ogata and Tetsuya Shimamura. Reinforced spectral subtraction method to enhance speech signal. In *Proceedings of IEEE Region 10 International Conference on Electrical and Electronic Technology. TENCON 2001 (Cat. No. 01CH37239)*, volume 1, pages 242–245. IEEE, 2001.

[7] Kohei Yamashita, Shin'Ya Ogata, and Tetsuya Shimamura. Improved spectral subtraction utilizing iterative processing. *Electronics and Communications in Japan (Part III: Fundamental Electronic Science)*, 90(4):39–51, 2007.

[8] Sheng Li, Jian-Qi Wang, Ming Niu, Xi-Jing Jing, Tian Liu, et al. Iterative spectral subtraction method for millimeter-wave conducted speech enhancement. *Journal of Biomedical Science and Engineering*, 3(02):187, 2010.

[9] Nathalie Virag. Single channel speech enhancement based on masking properties of the human auditory system. *IEEE Transactions on speech and audio processing*, 7(2):126–137, 1999.

[10] James D Johnston. Transform coding of audio signals using perceptual noise criteria. *IEEE Journal on selected areas in communications*, 6(2):314–323, 1988.

[11] Nasir Saleem and Muhammad Irfan Khattak. A review of supervised learning algorithms for single channel speech enhancement. *International Journal of Speech Technology*, 22(4):1051–1075, 2019.

[12] P Krishnamoorthy and SR Mahadeva Prasanna. Temporal and spectral processing methods for processing of degraded speech: a review. *IETE Technical Review*, 26(2):137–148, 2009.

[13] JB Allen, DA Berkley, and J Blauert. Multimicrophone signal-processing technique to remove room reverberation from speech signals. *The Journal of the Acoustical Society of America*, 62(4):912–915, 1977.

[14] P Bloom. Evaluation of a dereverberation process by normal and impaired listeners. In *ICASSP'80. IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 5, pages 500–503. IEEE, 1980.

[15] P Bloom and G Cain. Evaluation of two-input speech dereverberation techniques. In *ICASSP'82. IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 7, pages 164–167. IEEE, 1982.

[16] Harry L Van Trees. *Optimum array processing: Part IV of detection, estimation, and modulation theory*. John Wiley & Sons, 2002.

[17] Iain McCowan. Microphone arrays: A tutorial. *Queensland University, Australia*, pages 1–38, 2001.

[18] Sharon Gannot, Emmanuel Vincent, Shmulik Markovich-Golan, and Alexey Ozerov. A consolidated perspective on multimicrophone speech enhancement and source separation. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(4):692–730, 2017.

[19] S Unnikrishna Pillai. *Array signal processing*. Springer Science & Business Media, 2012.

[20] B.R. Breed and J. Strauss. A short proof of the equivalence of lcmv and gsc beamforming. *IEEE Signal Processing Letters*, 9(6):168–169, 2002.

[21] M. Shujau, C. H. Ritz, and I. S. Burnett. Speech enhancement via separation of sources from co-located microphone recordings. In *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 137–140, 2010.

[22] Qiquan Zhang, Mingjiang Wang, and Lu Zhang. A robust speech enhancement method based on microphone array. In *2017 IEEE 17th International Conference on Communication Technology (ICCT)*, pages 1673–1678, 2017.

[23] Jounghoon Beh, R.H. Baran, and Hanseok Ko. Dual channel based speech enhancement using novelty filter for robust speech recognition in automobile environment. *IEEE Transactions on Consumer Electronics*, 52(2):583–589, 2006.

[24] Robert A Monzingo and Thomas W Miller. *Introduction to adaptive arrays*. Scitech publishing, 2004.

[25] Huajun Yu. *Post-filter optimization for multichannel automotive speech enhancement*. Shaker, 2013.

[26] Rainer Zelinski. A microphone array with adaptive post-filtering for noise reduction in reverberant rooms. In *ICASSP-88., International Conference on Acoustics, Speech, and Signal Processing*, pages 2578–2579. IEEE Computer Society, 1988.

[27] KU Simmer and A Wasiljeff. Adaptive microphone arrays for noise suppression in the frequency domain. In *Second Cost 229 Workshop on Adaptive Algorithms in Communications*, pages 185–194, 1992.

[28] Klaus Uwe Simmer, Sven Fischer, and Alexander Wasiljeff. Suppression of coherent and incoherent noise using a microphone array. *Annals of telecommunications*, 49(7-8):439–446, 1994.

[29] Joerg Bitzer, Klaus Uwe Simmer, and Karl-Dirk Kammeyer. Multi-microphone noise reduction by post-filter and superdirective beamformer. In *Proc. IWAENC*, volume 99, pages 100–103, 1999.

[30] Iain A McCowan and Hervé Bourlard. Microphone array post-filter based on noise field coherence. *IEEE Transactions on Speech and Audio Processing*, 11(6):709–716, 2003.

[31] Alexander Southern, Samuel Siltanen, and Lauri Savioja. Spatial room impulse responses with a hybrid modeling method. In *Audio Engineering Society Convention 130*. Audio Engineering Society, 2011.

[32] Lauri Savioja and U Peter Svensson. Overview of geometrical room acoustic modeling techniques. *The Journal of the Acoustical Society of America*, 138(2):708–730, 2015.

[33] Jason E Summers. Auralization: Fundamentals of acoustics, modelling, simulation, algorithms, and acoustic virtual reality, 2008.

[34] Simon Benedict Shelley. *Diffuse boundary modelling in the digital waveguide mesh*. PhD thesis, University of York, 2007.

[35] Martin Woolley. Understanding reliability in bluetooth technology. *Bluetooth Special Interest Group (ISG)*, 2020.

[36] Bo Xia, Chunyu Xin, Wenjun Sheng, Ari Yakov Valero-Lopez, and Edgar Sánchez-Sinencio. A gfsk demodulator for low-if bluetooth receiver. *IEEE Journal of solid-state circuits*, 38(8):1397–1400, 2003.

[37] Ye Zhang, Aytac Atac, Lei Liao, and Stefan Heinen. A low-power high-efficiency demodulator in bluetooth low energy receiver. In *PRIME 2012; 8th Conference on Ph. D. Research in Microelectronics & Electronics*, pages 1–4. VDE, 2012.

[38] Prof. Cristofolini. Thermal and shot noise. Appendix C. Retrieved from class notes, University of Parma, 2018. Archived on Wayback Machine.

[39] DPA Microphones. The basics about noise in mics, 2018. Accessed Date: 18/04/2024.

[40] Brian CJ Moore. *An introduction to the psychology of hearing*. Brill, 2012.

[41] Neumann. What is self noise (or equivalent noise level)?, 2024. Accessed: 2024-04-22.

[42] Shure. Typical sound pressure levels of speech, 2024. Accessed: 2024-06-24.

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

[44] Meng Er and Antonio Cantoni. Derivative constraints for broad-band element space antenna array processors. *IEEE transactions on acoustics, speech, and signal processing*, 31(6):1378–1393, 1983.

[45] J Benesty. *Microphone array signal processing*. Springer Verlag, 2008.

[46] Harry L Van Trees. *Detection, estimation, and modulation theory, part I: detection, estimation, and linear modulation theory*. John Wiley & Sons, 2004.

[47] Livnat Ehrenberg, Sharon Gannot, Amir Leshem, and Ephraim Zehavi. Sensitivity analysis of mvdr and mpdr beamformers. In *2010 IEEE 26-th Convention of Electrical and Electronics Engineers in israel*, pages 000416–000420. IEEE, 2010.

[48] Alastair H. Moore, Sina Hafezi, Rebecca R. Vos, Patrick A. Naylor, and Mike Brookes. A compact noise covariance matrix model for mvdr beamforming. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:2049–2061, 2022.

[49] Romain Serizel, Marc Moonen, Bas Van Dijk, and Jan Wouters. Low-rank approximation based multichannel wiener filter algorithms for noise reduction with application in cochlear implants. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 22(4):785–799, 2014.

[50] Maja Taseska and Emanuel A. P. Habets. Relative transfer function estimation exploiting instantaneous signals and the signal subspace. In *2015 23rd European Signal Processing Conference (EUSIPCO)*, pages 404–408, 2015.

[51] Jingdong Chen, Jacob Benesty, and Yiteng Huang. A minimum distortion noise reduction algorithm with multiple microphones. *IEEE transactions on audio, speech, and language processing*, 16(3):481–493, 2008.

[52] Andreas I Koutrouvelis, Richard C Hendriks, Richard Heusdens, and Jesper Jensen. Robust joint estimation of multimicrophone signal model parameters. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 27(7):1136–1150, 2019.

[53] Shmulik Markovich, Sharon Gannot, and Israel Cohen. Multichannel eigenspace beamforming in a reverberant noisy environment with multiple interfering speech signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 17(6):1071–1086, 2009.

[54] Giovanni Bologni, Richard C Hendriks, and Richard Heusdens. Wideband relative transfer function (rtf) estimation exploiting frequency correlations. *arXiv preprint arXiv:2407.14152*, 2024.

[55] Simon Doclo and Marc Moonen. Gsvd-based optimal filtering for single and multimicrophone speech enhancement. *IEEE Transactions on signal processing*, 50(9):2230–2244, 2002.

[56] K. Uwe Simmer, Joerg Bitzer, and Claude Marro. *Post-Filtering Techniques*, pages 39–60. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001.

[57] Alan Davis, Sven Nordholm, and Roberto Togneri. Statistical voice activity detection using low-variance spectrum estimation and an adaptive threshold. *IEEE transactions on audio, speech, and language processing*, 14(2):412–424, 2006.

[58] Mohammad Hossein Moattar and Mohammad Mehdi Homayounpour. A simple but efficient real-time voice activity detection algorithm. In *2009 17th European signal processing conference*, pages 2549–2553. IEEE, 2009.

[59] S Gökhun Tanyer and Hamza Ozer. Voice activity detection in nonstationary noise. *IEEE Transactions on speech and audio processing*, 8(4):478–482, 2000.

[60] Jongseo Sohn, Nam Soo Kim, and Wonyong Sung. A statistical model-based voice activity detection. *IEEE signal processing letters*, 6(1):1–3, 1999.

[61] Youngjoo Suh and Hoirin Kim. Multiple acoustic model-based discriminative likelihood ratio weighting for voice activity detection. *IEEE Signal Processing Letters*, 19(8):507–510, 2012.

[62] Carlos E Galván-Tejada, Jorge I Galván-Tejada, José M Celaya-Padilla, J Rubén Delgado-Contreras, Rafael Magallanes-Quintanar, Margarita L Martinez-Fierro, Idalia Garza-Veloz, Yamilé López-Hernández, and Hamurabi Gamboa-Rosales. An analysis of audio features to develop a human activity recognition model using genetic algorithms, random forests, and neural networks. *Mobile Information Systems*, 2016(1):1784101, 2016.

[63] Neville Ryant, Mark Liberman, and Jiahong Yuan. Speech activity detection on youtube using deep neural networks. In *INTERSPEECH*, pages 728–731. Lyon, France, 2013.

[64] WebRTC Project. Webrtc voice activity detector (vad). https://webrtc.org/, 2024. Accessed: 2024-11-30.

[65] Cees H Taal, Richard C Hendriks, Richard Heusdens, and Jesper Jensen. An algorithm for intelligibility prediction of time–frequency weighted noisy speech. *IEEE Transactions on audio, speech, and language processing*, 19(7):2125–2136, 2011.

[66] Manuel Pariente. pystoi, 2024. GitHub repository.

[67] DL Richards. Speech-transmission performance of pcm systems. *Electronics Letters*, 1(2):40–41, 1965.

[68] Naomi Harte and Eoin Gillen. Tcd-timit: An audio-visual corpus of continuous speech. *IEEE Transactions on Multimedia*, 17(5):603–615, 2015.

[69] ondrosik. workers2. https://freesound.org/people/ondrosik/sounds/736752/, 2024. Accessed: 2024-05-22.

[70] Robin Scheibler, Eric Bezzam, and Ivan Dokmanić. Pyroomacoustics: A python package for audio room simulation and array processing algorithms. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 351–355. IEEE, 2018.

[71] Snakers4. Silero VAD: A set of VAD (Voice Activity Detection) models. https://github.com/snakers4/silero-vad. Accessed: Nov. 7, 2024.

[72] R Kumara Swamy, K Sri Rama Murty, and Bayya Yegnanarayana. Determining number of speakers from multispeaker speech signals using excitation source information. *IEEE Signal Processing Letters*, 14(7):481–484, 2007.

[73] Erich Zwyssig, Steve Renals, and Mike Lincoln. Determining the number of speakers in a meeting using microphone array features. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4765–4768. IEEE, 2012.

[74] Takuya Higuchi, Nobutaka Ito, Shoko Araki, Takuya Yoshioka, Marc Delcroix, and Tomohiro Nakatani. Online mvdr beamformer based on complex gaussian mixture model with spatial prior for noise robust asr. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 25(4):780–793, 2017.

[75] Takuya Higuchi, Nobutaka Ito, Takuya Yoshioka, and Tomohiro Nakatani. Robust mvdr beamforming using time-frequency masks for online/offline asr in noise. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5210–5214. IEEE, 2016.

[76] Mehrez Souden, Marc Delcroix, Keisuke Kinoshita, Takuya Yoshioka, and Tomohiro Nakatani. Noise power spectral density tracking: A maximum likelihood perspective. *IEEE Signal Processing Letters*, 19(8):495–498, 2012.

[77] Aleksej Chinaev, Alexander Krueger, Dang Hai Tran Vu, and Reinhold Haeb-Umbach. Improved noise power spectral density tracking by a map-based postprocessor. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4041–4044. IEEE, 2012.

[78] Hakan Erdogan, John R Hershey, Shinji Watanabe, Michael I Mandel, and Jonathan Le Roux. Improved mvdr beamforming using single-channel mask prediction networks. In *Interspeech*, pages 1981–1985, 2016.

[79] Aleksej Chinaev, Jahn Heymann, Lukas Drude, and Reinhold Haeb-Umbach. Noise-presence-probability-based noise psd estimation by using dnns. In *Speech Communication; 12. ITG Symposium*, pages 1–5. VDE, 2016.

[80] Jahn Heymann, Lukas Drude, and Reinhold Haeb-Umbach. Neural network based spectral mask estimation for acoustic beamforming. In *2016 IEEE International*

Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 196–200, 2016.

[81] Yutaro Matsui, Tomohiro Nakatani, Marc Delcroix, Keisuke Kinoshita, Nobutaka Ito, Shoko Araki, and Shoji Makino. Online integration of dnn-based and spatial clustering-based mask estimation for robust mvdr beamforming. In *2018 16th International Workshop on Acoustic Signal Enhancement (IWAENC)*, pages 71–75. IEEE, 2018.