Poul Hoang

JUNE 2018 | MASTER'S THESIS

SIGNAL PROCESSING & COMPUTING

SPEECH ENHANCEMENT WITH DNN SUPPORTED ACOUSTIC BEAMFORMING

A HYBRID APPROACH

SUPERVISED BY
J. JENSEN & Z. H. TAN

This report is compiled in LaTeX, originally developed by Leslie Lamport, based on Donald Knuth's TeX. The main text is written in *Computer Modern* pt 11, designed by Donald Knuth. Flowcharts and diagrams are made using draw.io, Inkscape and Tikz, a TeXpackage for generating graphics.

AALBORG UNIVERSITY

STUDENT REPORT

Department of Electronic Systems
Fredrik Bajers Vej 7
DK-9220 Aalborg Ø
http://www.es.aau.dk/

Title:
A Hybrid Approach for Speech Enhancement with DNN Supported Acoustic Beamforming

Master's Programme:
Signal Processing and Computing

Project Period:
February 2018 - June 2018

Project Group:
18gr1070

Participants:
Poul Hoang

Supervisors:
Jesper Jensen
Zheng-Hua Tan

Number of Pages: 80

Date of Completion:
June 07, 2018

Abstract:

Modern hearing aids often have more than one microphone available for each device. It has been shown that substantial gains in speech intelligibility can to obtained by applying multichannel signal processing methods (e.g. beamformers) to noisy observations in noisy environments such as cocktail parties or restaurant-like environments. Model-based signal processing methods might, however, perform less well in acoustic environments where the SNR is low as the unknown parameters needed for the beamformers are harder to estimate. The motivation behind the work presented in this thesis, is thus to explore the possibility of applying a deep neural network (DNN) to support an acoustic beamformer as an alternative to the model-based methods. The DNN will in this thesis specifically estimate the direction-of-arrival (DOA) and the relative transfer function (RTF) vector needed for the examined beamformers.
We have proposed three types of DNN supported beamformers in this thesis: 1) A minimum power distortionless response (MPDR) beamformer supported by a DNN for DOA estimation, 2) an MPDR beamformer supported by a DNN estimating RTF-vectors, and 3) a Bayesian beamformer where the posterior probabilities are estimated by a DNN. The experimental results show that the DNN-supported beamformers are able to outperform a model-based Bayesian beamformer in acoustic scenes with isotropic babble noise in terms of ESTOI, PESQ, and segSNR scores.

This page intentionally left blank.

Preface

Four months of work between February and June 2018 culminated in the Master's thesis presented here. The Master's thesis is a 30-ETCS project and is written by Poul Hoang as a part of the required fulfillment to obtain a Master of Science in Signal Processing and Computing at Aalborg University.

The theme of the Master's thesis is acoustic signal processing, where the overall goal is to explore the possibility of using a deep neural network to support an acoustic beamformer for hearing aids.

I would like to thank my supervisors professor Jesper Jensen and professor Zheng-Hua Tan for their valuable feedback, critique, and discussions we have had related to my project, which most definitely helped me improve the quality of the work presented in this thesis. Furthermore, thanks to the PhD fellows Morten Kolbæk and Daniel Michelsanti from Aalborg University for their technical support of the GPU server made available for this thesis. Without the server and their help, this project would most likely not have been finished in the given time frame.

MATLAB files used to simulate the proposed beamformers are included in the zip-file "CD.zip". The MATLAB-file "demo.m" provides a demonstration of the proposed beamformers. In order to run the "demo.m", the minimum requirement is MATLAB 2018 with the additional support package "TensorFlow-Keras Models".

This page intentionally left blank.

Contents

Chapter 1

Introduction

Hearing loss is one of the most common sensory disorders [1, p. 3] and presents a great challenge for the hearing impaired, since hearing loss can have a negative impact on social interaction, well-being, and life quality in general. In particular for children with hearing loss, the consequences include reduced ability to learn spoken language which can influence later education, job, and social life, if not addressed [1, p. 48]. In 2012, WHO estimated that approximately 360 million individuals, of which 32 million are children, suffer from disabling hearing loss making it a worldwide issue [1, p. 48]. To accommodate individuals with hearing impairments, hearing aids are typically used to help restore normal hearing. Hearing aids have historically existed for centuries in form of horn shaped devices, but have since developed into advanced electronic devices with digital microcomputers, capable of applying digital signal processing on sampled sound signals with the overall objective of increasing speech intelligibility and sound quality [2, p. 18].

A typical modern hearing aid is shown in Figure 1.1 and overall consists of a microphone to pick up sound, a circuit board for electronics, a battery, a receiver for sound reproduction, and an antenna for wireless communication [3].

Figure 1.1: An Oticon Opn behind-the-ear with receiver-in-canal hearing aid [4].

Hearing aids face a large variety of different acoustic environments, and should ideally be able to adapt automatically to all types of environments with minimum feedback from the user, in order to provide the best user experience and listening comfort. Examples of acoustic scenes are reverberant rooms with a target speaker and a competing speaker as illustrated in Figure 1.2. For individuals with hearing loss, reverberation and interference, such as a competing speaker, can degrade speech intelligibility [1, p. 110]. As modern hearing aids usually have more than one microphone per device [1, p. 111], a common approach to accommodate this issue, is to combine the microphone signals to form a directional microphone, which to some extent is able to enhance the sound from the target speaker direction while attenuating noise, reverberation and interference impinging from other directions. Processing multichannel signals in order to enhance signals from a particular spatial direction is also called beamforming, and is a type of signal processing methods implemented in hearing aids, that has proven effective at increasing speech intelligibility [1, p. 9].

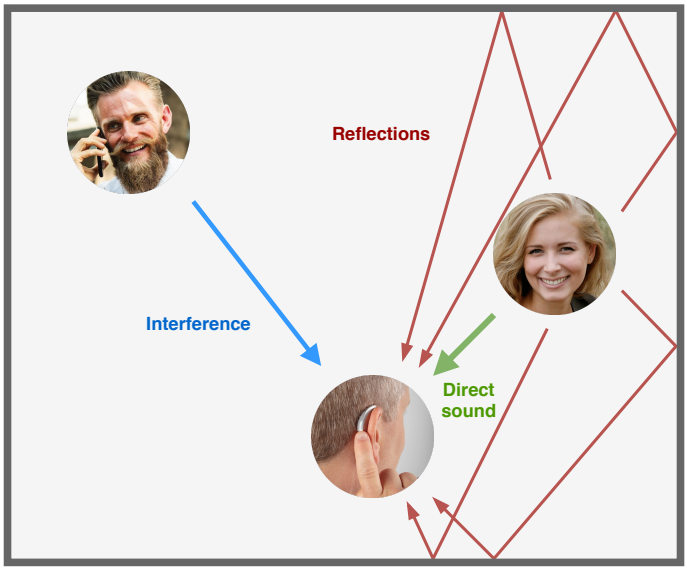


Figure 1.2: An example of an acoustic scene for hearing aid users. A hearing aid user is in a reverberant room with a target speaker and a competing speaker (interference). The hearing aid user is mostly interested in the direct sound from the target (green) and desires attenuation of noise from reverberation (red) and interference (blue).

Many beamformers, such as the minimum pariance distortionless response (MPDR) beamformer studied in chapter 3, require knowledge of the direction of the target speaker i.e. the direction-of-arrival (DOA). Model-based signal processing methods can estimate the DOA, but may fail when the signal-to-noise ratio (SNR) is too low. Unfortunately, acoustic scenes with low SNR are situations where the hearing aid user needs the beamformer the most. Alternatively, a data driven approach might be a possibility which, in contrast to the model-based approach, usually does not require making model assumptions. Deep learning methods, such as deep neural networks (DNNs), have gained much attention in recent years due to increased amount of training data, computational resources (GPUs), accessibility in terms of high-level implementation environments (e.g.

TensorFlow and Keras), and have proven effective at solving complicated tasks that otherwise would be difficult for model-based methods [5, p. 11]. The motivation behind this thesis is therefore to explore, if a DNN supported beamformer potentially can outperform a traditional model-based one in situations, where the SNR is low. We therefore in this thesis seeks to answer:

> How can a DNN be applied to support an acoustic beamformer and can it potentially outperform a model-based acoustic beamformer in terms of speech intelligibility and sound quality in acoustic scenes with low SNR?

A technical conceptual block diagram of the envisioned DNN supported beamforming system for hearing aids, is illustrated in Figure 1.3. Sound signals from the acoustic environment are picked up by a front and rear microphone, converted into electrical signals, and sampled into discrete sequences x_1 and x_2. Some beamformers are processed in the frequency domain by applying a beamformer to each frequency subband [1, p. 111]. To improve robustness and performance of the beamformers in low SNR, estimating the beamformer coefficients will in the proposed system be supported by a DNN. The outputs of the beamformers are transformed back into the time-domain $\hat{s}$ and a receiver (i.e. a loudspeaker) reproduces the sound in the ear canal of the user.

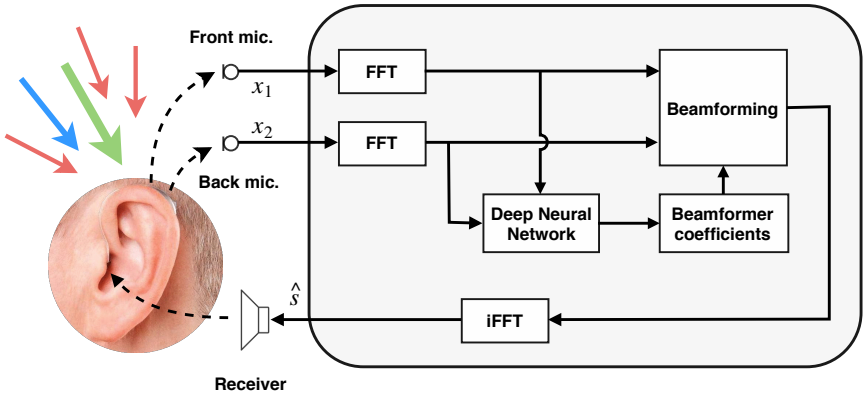


Figure 1.3: A simplified block diagram of the envisioned beamforming system for hearing aids supported by a deep neural network.

1.1 Report Structure

In chapter 2, a presentation of basic acoustic theory, the head-related transfer function, and the signal model will be provided. The chapter serves as a foundation for simulating acoustic scenes and later generating training data for the proposed DNN. Given the signal model, the thesis moves on to cover well-known model-based narrow-band beamforming methods in chapter 3. Chapter 4 will give an introduction to well-known model-based DOA estimation algorithms, followed by chapter 5 where the proposed DNNs will be presented. A model-based version of Figure 1.3 will be compared with the proposed

DNN-based system in chapter 6. The comparison will be in terms of confusion matrices for DOA estimation and segmental SNR improvement, short-time objective intelligibility (STOI) and perceptual evaluation of speech quality (PESQ) for beamforming performance. Finally in chapter 7 and chapter 8, a discussion and conclusion of the work presented in this thesis will be provided.

Chapter 2

Signal Model

The microphones of a modern hearing aid (HA) are typically placed either in-the-ear, completely in-the-ear canal, or behind-the-ear [1, p. 4] to convert the sound pressure into electrical signals, which can be sampled into discrete sequences and processed on an embedded device. As already mentioned, one of the objectives found in state-of-the-art HA devices, is to process the microphone signals, in order to enhance a target signal while suppressing noise signals from the environment to improve speech intelligibility, sound quality, and reduce listening effort. In a typical situation, the target signal is a speech signal, that originates from a person the HA user is listening to, and may be masked with noise. The sound waves radiated from sources in the acoustic scene, however have to obey fundamental properties of wave propagation in space. The purpose of this chapter is therefore to derive a signal model that sufficiently describes the behavior of spatio-temporal acoustic signals i.e. signals that are a function of space and time, which impinge on the microphone array of the HA.

2.1 Overview of the Acoustic Scene

The acoustic environment, a HA user may experience, comes at a large variety and might resemble cocktail parties where the sound waves approximately impinge from all directions, or a quiet listening environment with one target source and one noise source (e.g. a competing speaker). The noise sources may furthermore not necessary be speech signals, but can also be other noise types such as ambient noise. In addition, the target source and noise source are often placed in a reverberant room with several other objects that can reflect the sound wave such as tables and chairs etc. creating an approximately isotropic and diffuse sound field. For simplicity, we choose in this thesis to limit the scope to only include acoustic environments, where the sound waves propagated from a source may only be affected by the head and torso of the HA user, but otherwise propagates in free-field as illustrated in Figure 2.1, where v_q for all $q = 1, 2, ..., Q$ are noise sources or interference, s is the target source and x_1 and x_2 are the front and rear microphones of a

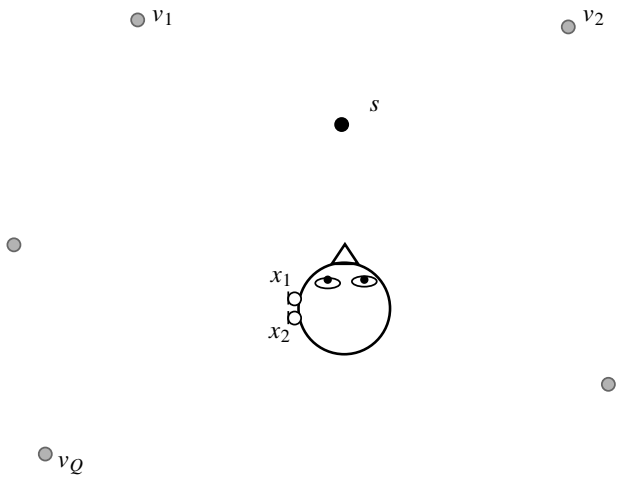


Figure 2.1: An example of the acoustic scene with microphones x_1 and x_2 placed of a left BTE HA, target (s) and noise sources v_q. It is assumed that each sound source can be modeled as a point source. The only object affecting the sound before reaching the microphones are the head and torso.

left behind-the-ear HA.

For convenience, it is assumed the target and noise sources can be modeled as point sources meaning that the sound sources generate spherical waves that propagate omni-directionally in space.

2.2 Radiation of Sound

Sound waves are spatio-temporal signals i.e. signals that are functions of space and time. It appears that in many fields of science and engineering such as acoustics, electromagnetism, seismology and physics to mention a few, energy can be transfered through a medium in terms of waves. The wave equation provides a mathematical expression that quantifies if a function is a wave. The solution to the wave equation plays an important role in the simulation and multi-microphone HA signal processing as it provides the mathematical foundation of how sound waves propagates in space over time.

2.2.1 Plane waves

The human ear is able to perceive sound which are changes in the air pressure and can be mathematically described as wave functions. Generally, wave functions have many different forms and shapes and can range from for example simple harmonic waves to speech signals. The wave functions are functions of space and time $x(\mathbf{r}, t)$ with $\mathbf{r} \in \mathbb{R}^3$ being a spatial position in the three-dimensional space where each element of $\mathbf{r}$ specifies a Cartesian coordinate and $t \in \mathbb{R}$ is the temporal variable. Under assumption that the speed of sound c is constant, then all wave functions have to obey the wave equation which

for the homogeneous case is given as [6, p. 8].

$$\nabla^2 x(\mathbf{r}, t) - \frac{1}{c^2} \frac{\partial^2 x(\mathbf{r}, t)}{\partial t^2} = 0, \tag{2.1}$$

which is a second-order partial differential equation where ∇^2 denotes the Laplacian operator. The solution to the homogeneous wave equation is the plane wave which is a wave whose wavefront is of equal phase along the hyperplane normal to the wavefront. For the one-dimensional case, one solution is [6, p. 11] [1]

$$x(r, t) = s\left(t - c^{-1} r\right), \quad r \in \mathbb{R}, \tag{2.2}$$

where $s\left(t - c^{-1} r\right)$ is an arbitrary wave function or a spatio-temporal signal. In context with the acoustic scene, the wave function would be a speech signal and we can interpret the signal $x(r, t)$ as a delayed version of the speech signal $s(t)$ with propagation delay $c^{-1} r$.

By applying the Fourier transform on the wave equation, one will obtain the Helmholtz equation, which can be interpreted as the wave equation in the frequency domain. Let $k \in \mathbb{R}$ be the wavenumber, $\tilde{x}(r, \omega)$ the Fourier transform of $x(r, t)$ with respect to time, then the one-dimensional case is given as [7, p. 71-72]

$$\nabla^2 \tilde{x}(r, \omega) - k^2 \tilde{x}(r, \omega) = \left(\nabla^2 - k^2\right) \tilde{x}(r, \omega) = 0, \tag{2.3}$$

It is easy to see that a solution to the one-dimensional Helmholtz equation is given as $\tilde{x}(r, \omega) = \tilde{s}(\omega) \mathrm{e}^{-j\omega c^{-1} r}$ since the Fourier transform of (2.2) is given by

$$\begin{aligned} \tilde{x}(r, \omega) &= \int_{-\infty}^{\infty} s\left(t - c^{-1} r\right) \mathrm{e}^{-j\omega t} dt = \int_{-\infty}^{\infty} s(t) * \delta\left(t - r c^{-1}\right) \mathrm{e}^{-j\omega t} dt \\ &= \tilde{s}(\omega) \mathrm{e}^{-j\omega c^{-1} r}, \end{aligned} \tag{2.4}$$

where $*$ denotes the convolution, $\delta(\cdot)$ is the Dirac-delta function, and $\delta(t - r c^{-1})$ is the free-field acoustic impulse response and its Fourier transform $G(\mathbf{r}, \omega) = \mathrm{e}^{-j\omega c^{-1} r}$ is the acoustic transfer function (ATF) between the source and microphone at $\mathbf{r}$. The ATF is of particular interest in array signal processing, since it is the free-field ATF for plane waves. It describes how the phase of a complex baseband signal $\tilde{x}(r, \omega)$ changes linearly as a function of frequency ω and spatial position r. For example, if r is kept fixed, then the phase changes linearly as a function of frequency ω, meaning that the group delay of the free-field ATF is constant. In time-domain this will be observed as an equal time-delay for all frequencies i.e. the propagation delay does not depend on the frequency. The same is true when keeping ω fixed while varying the spatial position r where the phase now changes linearly as a function of r.

2.2.2 Point sources

Modeling the impinging wave as a plane wave can offer simplicity later, when modeling the time or phase differences between the microphones of the impinging wave. However,

[1] Note that Equation 2.2 is not the general solution and is only valid for the one-dimensional case, but it shows the principle behind wave propagation.

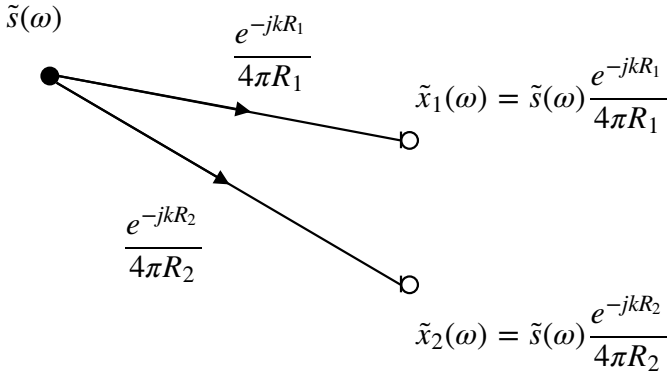


Figure 2.2: The free field ATF between a source $s(\omega)$ and two microphones x_1 and x_2.

when generating the acoustic scenes, it is desirable to adjust the power received at the microphones according to the distance between the microphones and the source i.e. the inverse-square law. It is seen that the current ATF $G(\mathbf{r}, \omega) = \mathrm{e}^{-j\omega c^{-1} r}$ does not scale the power as a function of position r, and it is desired to include a term to the ATF that scales the power as a function of distance r. This can be done by explicitly modeling the source as a point sources with coordinate $\mathbf{r}$ and a microphone position at $\mathbf{r}_0$. In contrast to a plane wave which is a solution to the homogeneous wave equation, introducing a point source will make the sound field inhomogeneous. Modifying the homogeneous Helmholtz equation into the inhomogeneous Helmholtz equation with a point source as source function, leads to Green's function given as [6, p. 160]

$$G_0(R) = \frac{\mathrm{e}^{-j\omega c^{-1} R}}{4\pi R} \text{ and } R = \|\mathbf{r}_0 - \mathbf{r}\|_2 \,, \tag{2.5}$$

and the received wave at the microphone is therefore

$$\tilde{x}(R, \omega) = \tilde{s}(\omega)\frac{\mathrm{e}^{-j\omega c^{-1} R}}{4\pi R}, \tag{2.6}$$

of which we simplify $\tilde{x}(\omega) = \tilde{x}(R, \omega)$. For a two microphone setup placed in free-field without any object affecting the sound wave, the source signal $\tilde{s}(\omega)$ is radiated from a point source and propagates through spaced and received at $\tilde{x}_1(\omega)$ and $\tilde{x}_2(\omega)$ as illustrated in Figure 2.2.

2.3 Head-Related Transfer Function

The ATF discussed so far assumes free-field conditions with no objects reflecting sound waves. This is of course not true for HA applications where the head and body of the user are objects that influence the sound field. The alteration is significant for HA applications since the HA devices are placed at the user's ear, meaning that reflections and diffraction caused by the head, pinnae, and torso [8, p. 20] will affect the performance of array signal processing algorithms if not accounted for. These influences can be modeled as being linear time-invariant (LTI) and can be considered as a finite impulse response (FIR) system and is referred to as the head-related transfer function (HRTF) [8, p. 20].

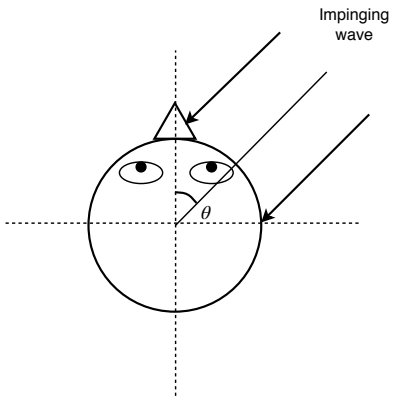


Figure 2.3: Impinging wave on head from angle θ.

When the sound wave radiated from a source impinge on the head as illustrated in Figure 2.3, the sound will be reflected and diffracted [8, p. 20] changing the free field the ATF discussed in previous section. The reflection and diffraction caused by the head is depend on several factors such as shape, the distance between the head and source, the angle of incident, and position of the microphones. We start by considering the case where the shape of the head is generic, and the angle of incident is only from the azimuth angle θ. The HRTF is then defined as the ratio between the sound pressure, $P_\nu^{(m)}(R_{\text{meas}}, \theta, \omega)$, measured at the ν'th ear and m'th microphone with distance R_{meas}, and the sound pressure, $P_0(R_{\text{meas}}, \omega)$, measured at the center of the head at the absence of the head. $P_0(R_{\text{meas}}, \omega)$ is essentially the free field ATF between the loudspeaker used to measure the HRIRs and the center of the head. The HRTF is thus given as [8, p. 20]

$$G_\nu^{(m)}(R_{\text{meas}}, \theta, \omega) = \frac{P_\nu^{(m)}(R_{\text{meas}}, \theta, \omega)}{P_0(R_{\text{meas}}, \omega)}, \quad \nu \in \{\text{Left,Right}\}, \tag{2.7}$$

and to simplify, we omit ω from the function for now. A model-based HRTF can be obtained by simulating the reflections from a spherical head model, but can otherwise also be obtained through acoustic measurement of the head-related impulse response (HRIR) on either an artificial head-and-torso simulator (HATS) or on human subjects. In this thesis, the HRTF are obtained from acoustic measurements of a HATS.

2.3.1 HRTF database description

The HRIR measurements are obtained from an open-source database provided by [9]. The measurement setup in [9] makes it especially suitable for simulation of behind-the-ear HAs, since HRIRs are available at various positions on a BTE HA. In this thesis we refer the HRTF as the acoustic. A HA dummy is placed on a head-and-torso simulator (HATS) by Brüel & Kjær as seen in Figure 2.4. The HA dummy only contains microphones and is without other miscellaneous electronics [9].

The HRIR are available for both the left and right ear to allow development and evaluation of binaural algorithms. For each side, one microphone placed in-the-ear and a 3-microphone array is behind-the-ear. Furthermore, the HRIRs are measured in various

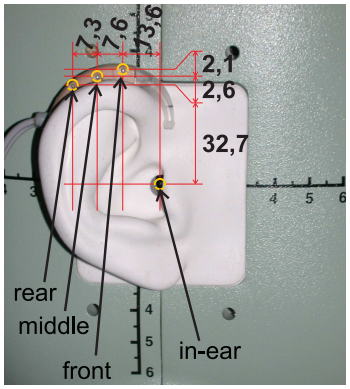


Figure 2.4: Dummy HA placed on a HATS [9].

acoustic environments such as an anechoic chamber, offices, and cafeterias. The HRIRs in the anechoic chamber are available for a source-microphone distance of 0.8 m and 3 m for near-field and far-field simulation of the HRTF, and will be used in this thesis. The HRIRs are sampled at a sampling frequency of 48 kHz. The HRIR were measured 360 ° around the head with an azimuth resolution of 5°. The HRIR was also measured at different elevations i.e. between -10° to 20° with a resolution of 5°. In this thesis we will focus on simulating the target and noise sources only from the azimuth angle. The sound waves may therefore impinge from a discrete set of 72 possible directions. The performance of beamformers studied in chapter 3 is affected by the array configuration such as the number of microphones and the distance between them.

2.3.2 Simulating the sound pressure at the left and right ear

In order to simulate the received sound, the HRTF must be used in combination with free-field ATF. From previous definitions of the ATF and HRTF, obtaining the transfer function between the source and the received signal at the microphones, must done by cascading the transfer functions in series i.e. $G_0(R)$ and $G_\nu(R_{\text{meas}}, \theta)$. Let R be the distance between the sound source and center of the head, and R_{meas} be the distance between the loudspeaker used to measure the HRIR and the center of the head. The complete ATF, $\tilde{a}_\nu^{(m)}$, between the source and the m'th microphone of ν'th HA is defined as

$$\begin{aligned}\tilde{a}_\nu^{(m)}(R,\theta) &= G_0(R)G_\nu^{(m)}(R_{\text{meas}},\theta) \\ &= \frac{\mathrm{e}^{-j\omega c^{-1}R}}{4\pi R}\frac{P_\nu^{(m)}(R_{\text{meas}},\theta)}{P_0(R_{\text{meas}})} \\ &= \frac{\mathrm{e}^{-j\omega c^{-1}R}}{4\pi R}\frac{4\pi R}{\mathrm{e}^{-j\omega c^{-1}R_{\text{meas}}}}P_\nu^{(m)}(R_{\text{meas}},\theta).\end{aligned} \tag{2.8}$$

If $R_{\text{meas}} = R$ then $G_0(R) = P_0(R_{\text{meas}})$ as they are both the free-field ATF between the source and the center of the head and (2.8) can be simplified to

$$\tilde{a}_\nu^{(m)}(R,\theta) = P_\nu^{(m)}(R_{\text{meas}},\theta), \quad \text{if } R_{\text{meas}} = R. \tag{2.9}$$

It is however not always possible to ensure that $R_{\text{meas}} = R$ as R_{meas} is limited by the distances the HRTFs were measured at. Using the HRTF database from [9] this limits

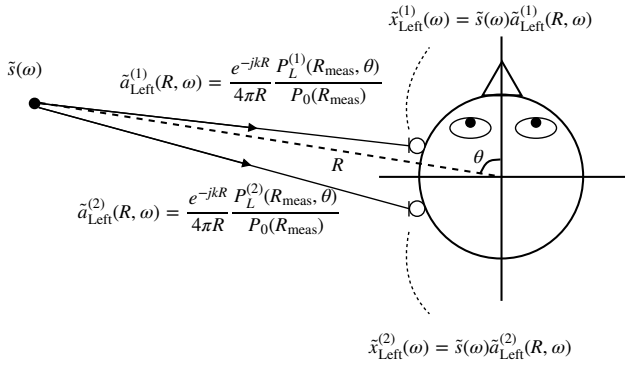


Figure 2.5: The ATF including the HRTFs between a source $s(\omega)$ and two microphones x_1 and x_2.

R_{meas} to be $R_{\text{meas}} = 0.8$ m or $R_{\text{meas}} = 3$ m. In case that $R_{\text{meas}} \neq R$, (2.9) becomes

$$\begin{aligned}\tilde{a}^{(m)}_{\nu}(R,\theta) &= \frac{R_{\text{meas}}}{R}\frac{\mathrm{e}^{-j\omega c^{-1}R}}{\mathrm{e}^{-j\omega c^{-1}R_{\text{meas}}}}P^{(m)}_{\nu}(R_{\text{meas}},\theta) \\ &= \frac{R_{\text{meas}}}{R}\mathrm{e}^{-j\omega c^{-1}(R-R_{\text{meas}})}P^{(m)}_{\nu}(R_{\text{meas}},\theta).\end{aligned} \tag{2.10}$$

We conclude from (2.10) that simulating the ATF between the source and microphone array at a distance of R_{meas} with $R_{\text{meas}} \neq R$ results in a gain of $\frac{R_{\text{meas}}}{R}$ and a linear phase-shift of $\mathrm{e}^{-j\omega c^{-1}(R-R_{\text{meas}})}$ which in the time domain translates to a delay of $c^{-1}(R - R_{\text{meas}})$ seconds. But since $\tilde{a}^{(m)}_{\nu}(R,\theta) \neq P^{(m)}_{\nu}(R_{\text{meas}},\theta)$ if $R_{\text{meas}} \neq R$, then this might cause misleading results in the simulation as the reflections and diffractions of the sound waves caused by the head, is a function of distance. One solution to avoid this issue, would be to obtain the HRTF at all possible distances such that $R_{\text{meas}} = R$ is always true. Fortunately according to [10], the HRTFs do not change substantially when the source to head distances are above 1 m as the impinging wave is approximately planar. This means that the error can be neglected when simulating source-microphone distances above 1 meter. Thus (2.9) can be used to simulate the ATF between the source and microphone at any distance above 1 m although the HRTFs are obtained at a source-microphone distance of 3 meters. Simulating the received signal at two microphones placed on a left HA on the head of a user, is illustrated in Figure 2.5.

2.4 Signal Model

The signal model, which will be used to simulate the received signals at the microphones, will be derived in the frequency domain for the following reasons. The first being that beamforming methods presented in [11, 12, 13, 14] assume that the impinging wave is narrow-band e.g. a sinusoidal wave which can be represented through complex baseband representation. For a sinusoidal wave, the complex baseband signal is simply the complex Fourier coefficient that represents the phase and magnitude of a carrier wave with frequency ω. Speech processing and acoustics, the impinging wave is in practice rarely only narrow-band signal but wide-band. In order to apply the methods developed for narrow-band signals, narrow-band decomposition of acoustic signals [13, 14] is performed

i.e. dividing the acoustic signal into multiple complex baseband signals at different frequencies.

Based on the narrow-band decomposition, $G_0(\omega, R)$ will denote the free-field ATF between the source and center of the head, and $G_L^{(m)}(\omega, R, \theta)$ is the HRTF at the left ear HA microphones. The ATF between the source and microphones, $\tilde{a}^{(m)}(\omega, R, \theta)$ is thus given as

$$\tilde{a}_\nu^{(m)}(\omega, R, \theta) = G_0(\omega, R) G_\nu^{(m)}(\omega, R_{\text{meas}}, \theta), \quad m \in \{1, 2, ..., M\}. \tag{2.11}$$

It is often seen in the literature that the ATF is both amplitude and phase normalized with respect to a reference microphone. After normalization, the ATF is referred to as the relative transfer function (RTF). If the front microphone is chosen as the reference microphone then the RTF is given as

$$\tilde{d}_\nu^{(m)}(\omega, R, \theta) = \frac{\tilde{a}_\nu^{(m)}(\omega, R, \theta)}{\tilde{a}_\nu^{(1)}(\omega, R, \theta)}, \quad \forall m \tag{2.12}$$

and $\tilde{d}_\nu^{(1)}(\omega, R, \theta) = 1$. In a cocktail party scenario for example, the target signal $\tilde{s}(\omega)$ and noise, denoted as $\tilde{\epsilon}_\nu^{(m)}(\omega)$, are the main components of the acoustic scene. Using the RTF from (2.12) the received signal $\tilde{x}_\nu^{(m)}(\omega)$ is given as

$$\tilde{x}_\nu^{(m)}(\omega) = \tilde{d}_\nu^{(m)}(\omega, R, \theta)\, \tilde{s}(\omega) + \tilde{\epsilon}_\nu^{(m)}(\omega) \in \mathbb{C}, \quad m \in \{1, 2, ..., M\}, \tag{2.13}$$

with the noise sources $\tilde{\epsilon}_\nu^{(m)}(\omega)$ being a superposition of interfering signals plus spatially white noise.[2] The vector notation of the signal model can be expressed as

$$\tilde{\mathbf{x}}_\nu(\omega) = \begin{bmatrix} 1 \\ \tilde{d}_\nu^{(2)}(\omega, R, \theta) \\ \vdots \\ \tilde{d}_\nu^{(\mathrm{M})}(\omega, R, \theta) \end{bmatrix} \tilde{s}(\omega) + \begin{bmatrix} \tilde{\epsilon}_\nu^{(1)}(\omega) \\ \tilde{\epsilon}_\nu^{(2)}(\omega) \\ \vdots \\ \tilde{\epsilon}_\nu^{(\mathrm{M})}(\omega) \end{bmatrix}, \tag{2.14}$$

where $\tilde{\mathbf{x}}_\nu(\omega) \in \mathbb{C}^M$ is the received signal at the ν'th HA. The indexing of R, θ in the vector notation of the RTF are omitted for a more compact notation. Given the signal model the next chapter will examine the acoustic scene in order to identify the parameters $\tilde{d}_\nu^{(m)}(\omega, R, \theta)$, $\tilde{s}(\omega)$, and $\tilde{\epsilon}^{(m)}(\omega)$ that will be used to simulate the acoustic scene. The noise, $\tilde{\epsilon}_\nu^{(m)}(\omega)$, that can be observed from an acoustic scene can vary greatly. The term noise is used slightly loosely in this thesis in the sense the term noise will cover any signal, that is not related to the clean target signal, however a clearer definition will be given in the next chapter.

The signal model has been derived and is given in Equation 2.14. This equation is used to simulate the sound received at the HA microphones in all of the studied acoustic scenes. The next chapter will cover an introduction to some of wide used beamformers.

[2] A formal definition of noise types will be given in chapter 3.

Chapter 3

Acoustic Beamforming

Modern HA devices often utilize more than one microphone per HA device to pick up sound from the acoustic environment [1, p. 111]. In acoustics scenes, where the overall goal is to enhance a speech signal of a target source, the received signal is most likely corrupted by noise originated from either space or generated by the microphones themselves. The basic concept of acoustic beamforming methods is therefore to create a directional microphone by combining the microphone signals into one signal in an optimal way, and obtain a directional microphone that is steered or focused towards a direction e.g. the target source. This chapter serves to introduce the concept of wideband acoustic beamforming which is a class of signal processing methods, that seeks to combine the temporal and spatial signals.

3.1 Spatio-Temporal Signals

Digital signal processing methods such as filtering [15] or spectral analysis [15] are often applied on temporal signals that are functions of time only. Well-known signal processing methods used for speech enhancement for temporal signals could be the Wiener filter that seeks the estimate a desired signal under some statistical assumptions on the received and desired signal, and then by the means of linear filtering, obtain an optimum estimate of the desired signal with a minimum mean squared error criterion [16].

When an array of microphones is available, this can be interpreted as adding another dimension to the received signal namely a spatial dimension, where the signals in the spatial dimension are generally correlated, if the received signal originates from space. Beamforming methods exploits this spatial correlation and utilize, it to obtain an optimum estimate of a target source signal. In many cases, methods and concepts found in temporal signal processing can be effectively reused for spatio-temporal signal processing. For example, the Wiener filter can be reformulated to estimate a desired signal over the spatial-dimension instead of temporal, and is then referred to as the spatial Wiener filter or multichannel Wiener filter [14, 17].

The HA user is typically only interested in sound reproduction of the temporal wave-

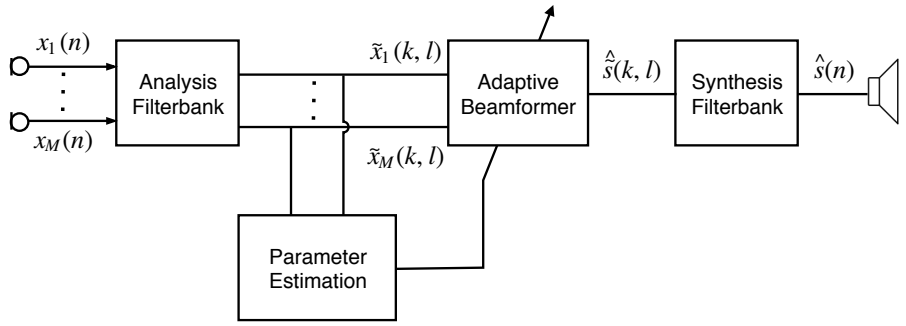


Figure 3.1: Overview of the elements included beamformer.

form in the eardrum of the spatio-temporal signal as it carries the sound information. For the HA devices, the spatial signal on the other hand, also reveals useful information about the acoustic environment such as the direction-of-arrival (DOA) of the target's source signal. Obtaining a clear interpretation of the spatial signal is yet not necessarily straight forward. For some applications, where the array aperture is a uniform linear array (ULA) placed in free-field, it is usually easier to see how spatial signals and its spatial frequency analysis in the wavenumber domain [18, p. 40] are related to DOA, since the microphones in a ULA can be seen as a discrete sampling of space. Furthermore, modeling the relation between microphones i.e. the array response for a ULA in free-field is also relatively simple (as it is simply a propagation delay between microphones). The interpretation of the spatial signal can, however, become tedious, when the array is no longer an ULA, and if placed behind the ear of a HA user, where the HRTF has to be accounted for. In order to obtain a sufficient array model, it is crucial to model the influence of e.g. the head-shadow (which is when the direct path between the microphones and the source is blocked by the head), reflection from the head and torso, and inter-person variations of the head and body shape.

Instead of modeling the array response mathematically, it is usually much simpler and accurate to simply measure the array response as a function of direction, when the microphones are placed behind the ear. In particular (see chapter 2) the spatial correlation between the microphones is given by the RTF-vectors.

Many beamformers utilize the RTF-vectors as they provide the necessary information the spatial correlation between microphones, and since these are available from [9], we will focus on beamforming methods that utilizes the RTF-vectors. In this chapter, beamformers that will be discussed are primarily the Bartlett [11], Minimum power distortionless response (MPDR) [19, p. 451], and Bayesian beamformer [17].

Before presenting the beamformers an overview block diagram of a beamforming system is illustrated in Figure 3.1. As beamforming algorithms often are applied in the frequency domain, we will start the discussion on analysis and synthesis of the spatio-temporal signals. The remaining of this chapter will then be on adaptive beamforming.

3.2 Pre and post processing

Beamforming is often applied on narrow-band signals. The common approach is to approximate the acoustic wideband signal (e.g. a noisy speech signal) into multiple narrow-

band signals using the Fourier transform. In practice, the short-time Fourier transform (STFT) is used for this purpose, where the concept is to divide the signal into l frames, Fourier transform the signal into the frequency domain (analysis), process the beamformer (modification), and inverse Fourier transform it into time (synthesis) [20, p. 230]. Let the frame length be N, hop-size D, then the short-time Fourier transform (STFT) is given as [20, p. 230]

$$\tilde{x}_\nu^{(m)}(k,l) = \mathrm{STFT}\{x_\nu^{(m)}(n)\} = \sum_{n=0}^{N-1} w(n) x_\nu^{(m)}(n+lD) \mathrm{e}^{\frac{-j2\pi kn}{N}}, \quad \nu \in \{\mathrm{Left}, \mathrm{Right}\}, \tag{3.1}$$

where $w_A(n)$ is a window function. Using the STFT, $\tilde{x}_\nu^{(m)}(k,l) \in \mathbb{C}$ becomes a function of discrete frequency k and time l. After analysis and modification, the processed signal $\tilde{y}_\nu(k,l)$ must be reconstructed into time domain by performing the inverse STFT. The inverse STFT is given as [20, p. 231]

$$\begin{aligned} \hat{s}(n+lD) &= \mathrm{iSTFT}\{\tilde{y}(k,l)\} \\ \hat{s}(n+lD) &= \frac{1}{K} \sum_{k=0}^{K-1} w(n) \tilde{y}(k,l) \mathrm{e}^{\frac{j2\pi kn}{N}}. \end{aligned} \tag{3.2}$$

The window function $w(n)$, frame length N, and hop-size D are parameters that have to be chosen. Typically, the windows are implemented as sliding overlapping windows. It is then important to ensure, that the selected window function has the so-called overlap-add property, which states that perfect reconstruction of the original signal must be possible after applying the window function [20, p. 232]. The property is

$$\sum_l w(n - lD) = 1. \tag{3.3}$$

Possible windows functions, that satisfy this perfect reconstruction property, are for example rectangular, triangular, and Hanning windows. In this thesis, we select the square-root Hanning window with a hop-size of $L = N/2$, and use the window function for both analysis and synthesis which will ensure perfect reconstruction. This choice may be motivated by the fact that the square-root Hanning window is often used in noise reduction and beamforming, when the target signal is speech [21, p. 50].

Signals such as speech has a time varying spectrum and its spectrum depends on the phonemes of a word. Ideally, it is desired that the window size is selected such that the PSD of signal in the frame can be assumed stationary. A common frame size for speech is approximately 20-30 ms [22]. If the sampling frequency of the temporal signal is 16 kHz, 20 ms will correspond to 320 samples. For convenience, the frame size is rounded to 256 samples.

3.3 Noise in the Acoustic Scene

Here we discuss more specifically what the term noise covers in an acoustic scene. Since a spatial dimension is added to the temporal signal, it is important to establish a clear distinction between noise in the temporal and spatial domain. In order to avoid confusion, a discussion on various noise types will be given in this section. Referring back

to the signal model previously introduced in chapter 2, the noise term in the signal model is $\tilde{\boldsymbol{\epsilon}}(k,l)$. We now assume that the noise can be decomposed into a sum of interference noise impinging from Q directions at distance R_q. As each interference noise source radiates a signal $\tilde{v}_q(k,l)$, which propagates in space before reaching the microphones of the HA, a RTF-vector $\tilde{\mathbf{d}}_\nu(k,l,R_q,\theta_q)$ is associated with each interference source. Moreover, a noise term $\mathbf{n}(k,l)$ is added to model the microphone self-noise. The noise $\tilde{\boldsymbol{\epsilon}}(k,l)$ can thus be decomposed into

$$\tilde{\boldsymbol{\epsilon}}(k,l) = \sum_{q=1}^{Q} \tilde{\mathbf{d}}_\nu(k,l,R_q,\theta_q)\tilde{v}_q(k,l) + \mathbf{n}(k,l). \tag{3.4}$$

A more detailed description of the noise is now provided.

3.3.1 Spatially White Noise

Noise that is said to be spatially white, is noise that is uncorrelated along the spatial dimension, i.e. uncorrelated between microphones. This is typically noise that is generated by the microphones themselves. This also implies that the noise does not have any spatial origin. In that case, the spatial coherency between microphones is zero. The spatial Cross-power spectrum density (CPSD) matrix, is a matrix which specifies the spatial correlation at a certain frequency k. For spatially WGN, the CPSD matrix is a diagonal matrix with equal diagonal elements σ_n^2, which is the variance of the noise. The microphone self-noise can be modeled in frequency domain as complex circular symmetric WGN and is given as

$$\mathbf{n}(k) \sim \mathcal{N}_C(\mathbf{0}, \sigma_n^2(k)\mathbf{I}), \tag{3.5}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{C}_n(k) = \sigma_n^2(k)\mathbf{I}$ is the noise CPSD matrix. If the noise is also temporally white, then the variance, $\sigma_n^2(k)$, is identical for all k.

3.3.2 Spatially Coherent Noise Sources

When noise originates from space, the signals received at the microphones are generally correlated. This noise type is very common in acoustic environment as reverberation, competing speakers, environmental noise, can be seen as coherent noise. This type of noise will be referred to as interference. The spatial CPSD matrix for interference is

$$\begin{aligned}
\mathbf{C}_v(k) &= \mathbb{E}\left[\left(g\sum_{q=1}^{Q}\tilde{\mathbf{d}}_\nu(k,l,R_q,\theta_q)v_q(k,l)\right)\left(g\sum_{q\in\mathcal{V}}\tilde{\mathbf{d}}_\nu(k,l,R_q,\theta_q)v_q(k,l)\right)^H\right], \\
&= g^2\sum_{q=1}^{Q}\tilde{\mathbf{d}}_\nu(k,l,R_q,\theta_q)\mathbb{E}\left[v_q(k,l)v_q(k,l)^*\right]\tilde{\mathbf{d}}_\nu(k,l,R_q,\theta_q)^H, \\
&= g^2\sum_{q=1}^{Q}\sigma_q^2(k,l)\tilde{\mathbf{d}}_\nu(k,l,R_q,\theta_q)\tilde{\mathbf{d}}_\nu(k,l,R_q,\theta_q)^H.
\end{aligned} \tag{3.6}$$

We will later see that the performance of some DoA estimation methods and beamformers depend on the noise type, as some algorithms are designed to perform better in specific noise types.

3.3.3 Isotropic noise field

The noise field found in acoustic environments can many times be modeled as being isotropic such as reverberation [23]. In an isotropic noise field the interference is impinging from all directions at equal power. Examples of noise fields that may be approximated as isotropic are pedestrian noise, reverberation, and cocktail party scenarios where competing speakers are present in all directions. For practical reasons we will limit ourselves to only approximately cylindrical isotropic noise fields as the HRTF in [9] are not provided in all elevation angles.

3.4 Linear Beamformers

After the analysis, i.e. applying the STFT to each noise microphone signal, comes the modification, where the beamformer is applied on the noisy signal. The beamformer is essentially just a linear combination between beamformer coefficients w_m and the noisy signal x_m, and can simply be expressed as an inner product between two vectors. This operation is then performed for each K frequency bins for a particular frame l. The output of the beamformer is given as

$$\tilde{y}(k,l) = \mathbf{w}^H(k,l)\tilde{\mathbf{x}}(k,l). \tag{3.7}$$

When determining the beamformer coefficients, the noisy CPSD matrix is often needed and is given as $\mathbf{C}_x(k) = \mathbb{E}[\tilde{\mathbf{x}}(k)\tilde{\mathbf{x}}(k)^H]$ and using the assumption that the target and noise signals are uncorrelated, we can then decompose the noisy CPSD matrix into

$$\mathbf{C}_x(k,l) = \mathbf{C}_s(k,l) + \mathbf{C}_v(k,l) + \mathbf{C}_n(k,l), \tag{3.8}$$

of which we choose to simplify $\mathbf{C}_\epsilon(k,l) = \mathbf{C}_v(k,l) + \mathbf{C}_n(k,l)$. The noisy CPSD may then be estimated as a moving average of outer products of the noisy signal over L frames as

$$\hat{\mathbf{C}}_x(k,l) = \frac{1}{L}\sum_{n=l-L-1}^{l} \tilde{\mathbf{x}}(k,n)\tilde{\mathbf{x}}(k,n)^H. \tag{3.9}$$

Determining the optimum number of frames L to estimate the CPSD over is not trivial. The more frames the CPSD matrix is estimated over, the lower variance the estimator has. However, in acoustic scenarios where the spatial position of either the target or interference is changing location, this will affect the estimates of the noisy CPSD and have an influence of the performance of beamformers presented later. Determining the number of frames L is a trade-off between better noise reduction and being reactive to spatial changes.

In the following sections, we derive the beamformer coefficients under various optimality criterions. To ease the notation, we will omit the indexing of k, l, ν, R_q and the tilde sign $\sim$ such that

$$\mathbf{d}(\theta_q) \triangleq \tilde{\mathbf{d}}_\nu(k,l,R_q,\theta_q), \quad \mathbf{C}_x \triangleq \mathbf{C}_x(k,l), \quad \mathbf{x} \triangleq \tilde{\mathbf{x}}(k,l). \tag{3.10}$$

3.4.1 Bartlett Beamformer

The Bartlett beamformer [11] seeks to minimize the output noise power under the assumption that the noise is spatially white. A constraint is added ensuring that the

target signal must pass through the beamformer undistorted. The optimization problem for estimating the beamformer coefficients is

$$\begin{aligned} \mathbf{w}_{\text{Bart}} = \arg\min_{\mathbf{w}} \mathbf{w}^H \left(\sigma^2 \mathbf{I}\right) \mathbf{w} = \arg\min_{\mathbf{w}} \sigma^2 \mathbf{w}^H \mathbf{w} \\ \text{subject to } \mathbf{w}^H \mathbf{d}(\theta_{i_{true}}) = 1, \end{aligned} \tag{3.11}$$

and the solution to the optimization is [11]

$$\mathbf{w} = \frac{\mathbf{d}(\theta_{i_{true}})}{\|\mathbf{d}(\theta_{i_{true}})\|_2^2}. \tag{3.12}$$

3.4.2 MPDR Beamformer

In cases where interference noise is present, the Bartlett beamformer may not reveal a satisfactory result because of its assumptions on the noise. If noise is coherent, the MPDR beamformer (sometimes also referred to as the minimum variance distortionless response (MVDR) [14] or Capon beamformer [11]) can reveal better noise reduction than the Bartlett beamformer. The optimization problem of the MPDR is to minimize the signal power under the constraint that the target is undistorted. The optimization problem for the MPDR beamformer is defined as [19, p. 451]

$$\begin{aligned} \mathbf{w}_{\text{MPDR}} = \arg\min_{\mathbf{w}} \mathbf{w}^H \mathbf{C}_x \mathbf{w} \\ \text{subject to } \mathbf{w}^H \mathbf{d}(\theta_{i_{true}}) = 1, \end{aligned} \tag{3.13}$$

and the closed-form solution to the optimization problem is [19, p. 451]

$$\mathbf{w}_{\text{MPDR}} = \frac{\mathbf{C}_x^{-1} \mathbf{d}(\theta_{i_{true}})}{\mathbf{d}(\theta_{i_{true}})^H \mathbf{C}_x^{-1} \mathbf{d}(\theta_{i_{true}})}. \tag{3.14}$$

The MPDR requires a matrix inversion, and can be numerically unstable. This would occur in simulation or theory if only the target is present without noise i.e. $\mathbf{C}_x$ is not full rank. It is evident that if the estimated DOA $\hat{\theta} \neq \theta_{i_{true}}$ such that $\mathbf{d}(\hat{\theta}) \neq \mathbf{d}(\theta_{i_{true}})$, then the MPDR beamformer will treat the target as an interference noise source and attenuate the target substantially.

3.4.3 Bayesian Beamformer

If the true RTF-vector is unknown, or equivalently if the DOA is unknown, using an MPDR beamformer lead to poor performance. In [19, p. 513], it has been analysed that a DOA mismatch for the MPDR can result in large amount of distortion on the target signal, as the beamformer treats the target as interference and attempts to place a spatial null at the direction of the target.

The goal of the Bayesian beamformer is to achieve robustness to DOA errors. To achieve this, the Bayesian beamformer first estimates a posterior probability for each direction defined in a discrete set of possible DOAs (e.g. $\Theta = \{-175°, -170°, ..., 180°\}$) and then form a linear combination between estimated posterior probabilities and MPDR

beamformers steered toward directions defined in the discrete set of possible DOAs [17]. The beamformer is given as

$$\mathbf{w}_B = \sum_{i=1}^{I} p(\theta_i|\mathbf{x}(k,l-L-1),\mathbf{x}(k,l-L),...,\mathbf{x}(k,l))\mathbf{w}_{\text{MPDR}}(\theta_i), \tag{3.15}$$

where we can view the computation of the Bayesian beamformer as weighted sums of MPDR beamformers. The full derivation of the Bayesian beamformer can be found in [17]. It is assumed that the DoA is a discrete random variable belonging to a finite set of possible DOAs. We make the assumption that the observed signals are realizations of a random process which is circular symmetric complex WGN, and that the observations are temporally independent. We form the likelihood function for a zero-mean complex normal distribution [17]

$$f(\mathbf{x}(k,l-L-1),...,\mathbf{x}(k,l)|\theta_i) = \prod_{j=1}^{l} \frac{1}{\pi^M \det(\mathbf{C}_x(\theta_i))} \exp\left(\mathbf{x}(k,j)^H \mathbf{C}_x^{-1}(\theta_i)\mathbf{x}(k,j)\right), \tag{3.16}$$

and using Bayes theorem we write the posterior probability $P(\theta_i|\mathbf{x}(k,l-L-1),...,\mathbf{x}(k,l))$ as

$$P(\theta_i|\mathbf{x}(k,l-L-1),...,\mathbf{x}(k,l)) = \frac{P(\theta_i) f(\mathbf{x}(k,l-L-1),...,\mathbf{x}(k,l)|\theta_i)}{\sum_{j=1}^{I} P(\theta_j) f(\mathbf{x}(k,l-L-1),...,\mathbf{x}(k,l)|\theta_j)}. \tag{3.17}$$

We let the prior probability be uniform and be in the range $[-175, 180]$ as no further prior information is available.

The expression in (3.16) is hard to evaluate as an exact expression of $\mathbf{C}_x(\theta_i)$ as a function of direction θ_i may not be available. In this case, it is desired to reduce the expression such that RTF-vectors are included in the expression. In order to simplify the expression, we use the assumption that no interference is within the range of directions defined by the discrete set Θ [17]. Under these assumptions, the likelihood function can be approximated to [17]

$$f(\mathbf{x}(k,l-L-1),...,\mathbf{x}(k,l)|\theta_i) \approx \exp^{L\gamma\left(\mathbf{d}(\theta_i)^H \mathbf{C}_x^{-1}\mathbf{d}(\theta_i)\right)^{-1}}, \tag{3.18}$$

where γ is a constant which is tuned according to the SNR and set small if the SNR is low and large if high [17]. The posterior probability can then be approximated as

$$P(\theta_i|\mathbf{x}(k,l-L-1),...,\mathbf{x}(k,l)) \approx \frac{P(\theta_i)\exp^{L\gamma\left(\mathbf{d}(\theta_i)^H \mathbf{C}_x^{-1}\mathbf{d}(\theta_i)\right)^{-1}}}{\sum_{j=1}^{I} P(\theta_j)\exp^{L\gamma\left(\mathbf{d}(\theta_j)^H \mathbf{C}_x^{-1}\mathbf{d}(\theta_j)\right)^{-1}}}. \tag{3.19}$$

It is expected that the Bayesian beamformer has a degraded performance than the MPDR using the true underlying RTF-vectors due to the optimality provided by the MPDR beamformer. However, as the SNR increases, it is expected that the posterior probability at the true DoA will approach 1 [17], which simplifies the Bayesian beamformer to the MPDR beamformer. An advantage of the Bayesian beamformer is if the DOA is not known with certainty, optimality of the MPDR beamformer cannot be guaranteed, and the Bayesian beamformer might perform better.

3.5 Beampattern and Beamformer Behavior

The purpose of this section is to examine the behavior of the beamformers in various noisy environments. For this purpose, the beampattern of the beamformers is evaluated. The beampattern provides an estimate of the noise reduction of the beamformer as a function of direction. The beampattern is defined as [12, p. 22]

$$B(\theta) = -10\log_{10}\left(\frac{|\mathbf{w}^H\mathbf{d}(\theta_{i_{true}})|^2}{|\mathbf{w}^H\mathbf{d}(\theta_i)|^2}\right). \tag{3.20}$$

The numerator and denominator are the output power of the beamformer when steered towards the target DOA $\theta_{i_{true}}$ and θ_i respectively. The RTF-vectors are obtained from the front and rear microphones of a left ear HA device [9], where each RTF-vector is associated with a direction θ in the range [-175,180] with a angle resolution of 5 degrees. The beampattern will be presented in a polar plot, where the angular axis is the azimuth direction, the radial axis is the frequency axis linearly spaced between 0 Hz to 8 kHz, and the color axis is the beampattern.

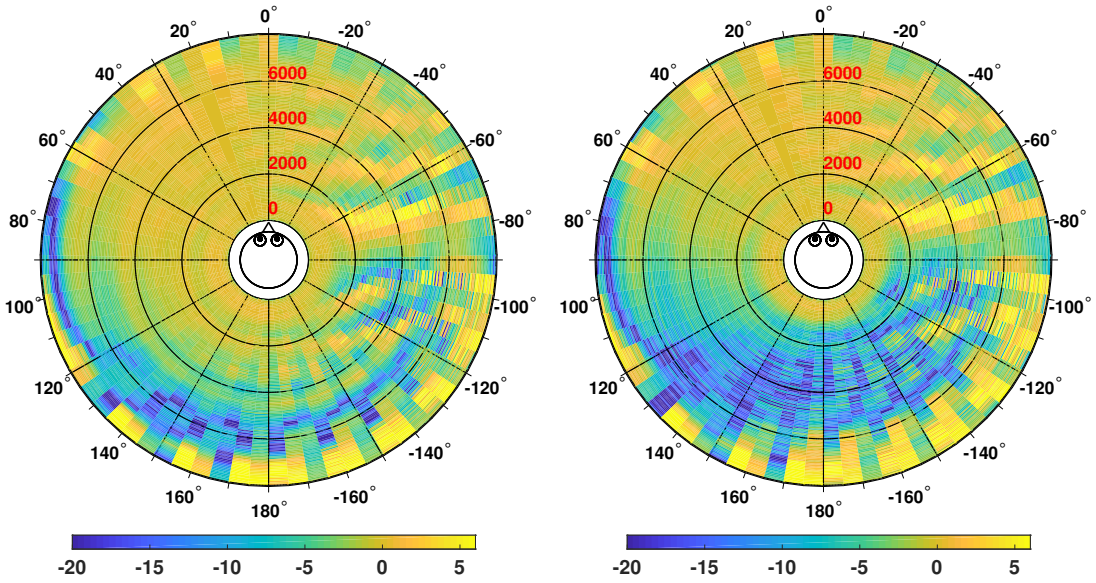
The Bayesian beamformer is expected to behave as the MPDR beamformer in high SNR as argued in subsection 3.4.3. However, in low SNR where the posterior probability approaches the prior probability [17], it is expected that the Bayesian beamformer will perform worse than the MPDR beamformers provided that the DOA is known. In order to emphasize the advantage of the Bayesian beamformer, DOA mismatch is included to show that the Bayesian beamformer is robust against DOA mismatches. In this case, it is expected that the Bayesian beamformer performs better than the other beamformers.

3.5.1 Bartlett vs. MPDR

Here we compare the beampattern of the Bartlett and MPDR beamformers given that the DOA is known, and the beamformer is provided with the true RTF-vector. The target source is temporally white Gaussian noise with a variance of $\sigma_s^2 = 1$ and is placed at a direction of 20°. A temporally white interference source is placed at 130° and we include spatially white noise. The SNR is set to -6 dB, where the power of the interference source and spatially white noise are set to be equal. The noisy CPSD matrix $\mathbf{C}_x$ is formed by averaging the outer product of $\mathbf{x}$ over L frames. It is seen that the MPDR attempts to place a spatial null at 130° to cancel the interference. The Bartlett beamformer does not use information about the noisy CPSD, and therefore it is expected that it does not attempt to place a null towards the direction of the interference.

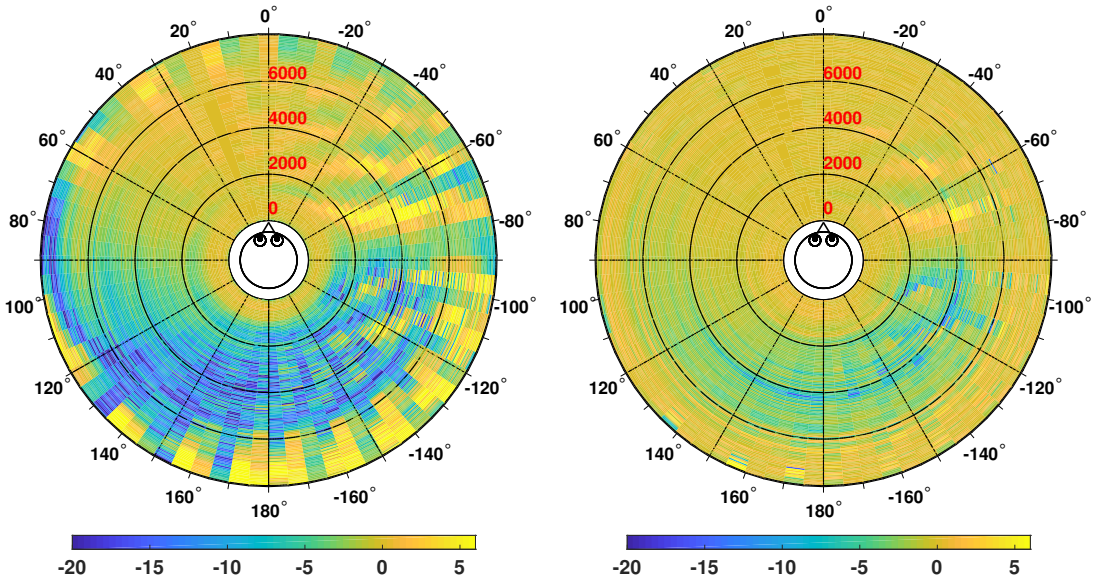
3.5.2 MPDR vs. Bayesian beamformer

Here we consider the beampattern between the MPDR and Bayesian beamformer. The prior probability of the Bayesian beamformer is set to be uniform and $\Theta = \{-175°, -170°, ..., 180°\}$. The setup is almost identical as the previous test, but here we the SNR = 0 dB. The target is placed at true DOA 20° and an interference source at 130°. The MPDR beamformer is provided the true DOA, while the Bayesian beamformer utilizes the estimated posterior probability. The beampattern of the MPDR and Bayesian



(a) Bartlett beampattern [dB] **(b)** MPDR beampattern [dB]

Figure 3.2: Beampatterns of Bartlett and MPDR beamformers, $L = 100$, $SNR = -6$ dB, true DOA= 20°. The true RTF-vector is provided the beamformers. The MPDR seem to emphasize noise reduction at the rear i.e. the direction of the interference.



(a) MPDR beampattern [dB] **(b)** Bayesian beampattern [dB]

Figure 3.3: Beampatterns of MPDR and Bayesian beamformers. $\gamma = 0.1$, $L = 100$, SNR= 0 dB, true DOA= 20°. Intereference at 130°.

beamformers are shown in Figure 3.3a and Figure 3.3b.

The comparison might not be entirely fair, as the MPDR is provided with the true DOA and RTF-vector. This is equivalent to letting the posterior probability of the Bayesian beamformer be equal to one at the true DOA. It however illustrates the trade-off

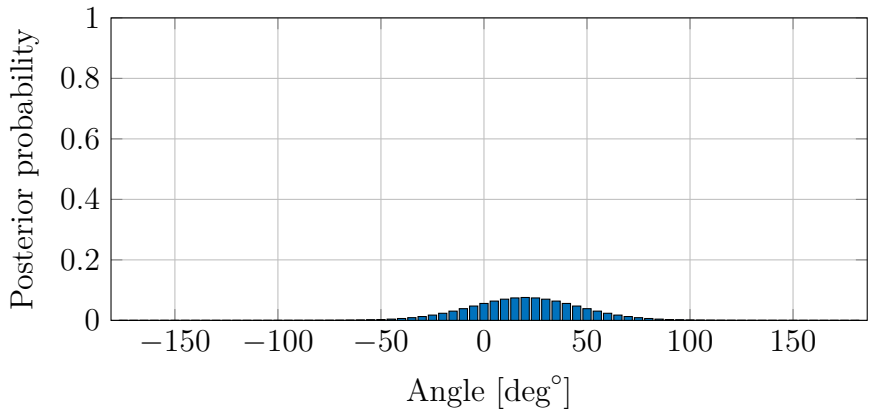


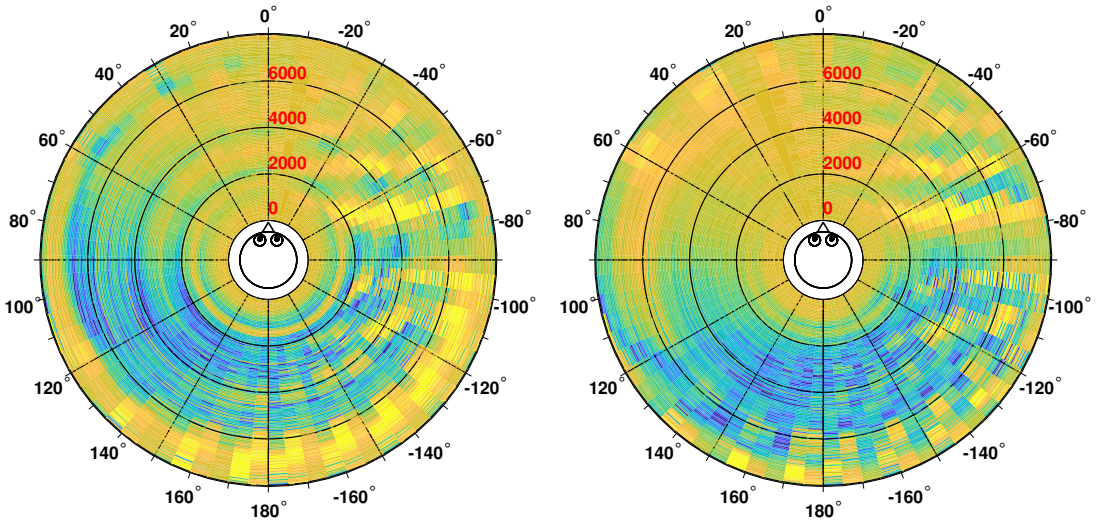
Figure 3.4: Sampled and normalized Von Mises distribution with mean at 20°.

between performance and robustness, where the MPDR beamformer seeks maximum noise reduction, at the risk of DOA mismatch and target distortion, while the opposite is true for the Bayesian beamformer.

3.5.3 MPDR vs. Bayesian beamformer with DOA mismatch

Here we compare the MPDR and Bayesian beamformer in a situation, where the MPDR beamformer is exposed to DOA mismatch, and is provided a false DOA of $-10°$ and true DOA of $20°$. We also demonstrate the potential performance of the Bayesian beamformer if good estimates of the posterior probability is available. To show the potential, we artificially generate a posterior probability obtained from a sampled and afterwards normalized Von Mises distribution as seen in Figure 3.4, .

In Figure 3.5 it is seen that the beampattern of the MPDR beamformer attempts to steer the spatial null from interference direction $130°$ to the target direction $20°$ due to the DOA mismatch. The result is, therefore, that the MPDR slightly distorts the target. The Bayesian beamformer on the other hand, seem to be less affected by the DOA mismatch as noise reduction is still primarily from behind.



(a) MPDR beampattern [dB]

(b) Bayesian beampattern [dB]

Figure 3.5: Beampatterns of MPDR and Bayesian beamformers with DOA mismatch. $\gamma = 0.1$, $L = 100$, SNR= 0 dB, true DOA= 20°. A false DOA of $-10°$ is provided to the MPDR beamformer.

It is seen that the MPDR and Bayesian beamformers are able to reduce interference noise in contrast to the Bartlett beamformer. Since realistic acoustic environments such as reverberant rooms contain interference, the Bartlett might not be suitable for hearing aids. We thus choose to only implement the MPDR and Bayesian beamformer for later for evaluations of a model-based beamformer and DNN supported beamformers.

Chapter 4

Model-based RTF Estimation

In practice the optimum performance with respect to the beamformers optimality criterion may not be achieved in real life due to model assumptions and model parameter estimation. In particular, the RTF-vector is a parameter needed for all the examined beamformers. Optimum noise reduction can therefore only be achieved if the beamformers are provided the true RTF-vectors. Therefore this chapter serves to investigate model-based methods for estimating the DOA. We will limit our search to common methods applied in frequency domain and only require the noisy observations $\tilde{\mathbf{x}}(k,l)$ as input.

4.1 DOA-based RTF Estimators

DOA estimation methods that will be covered in this section are based on the Bartlett and MPDR beamformers and the MUSIC algorithm. Since the DOA is a direction from a discrete set $\theta = \{-175°, -170°, ..., 180°\}$, the idea behind DOA-based RTF-estimation is map the estimated DOA into a RTF-vector associated with the direction.

4.1.1 Beamscan Algorithms

The basic concept of beamscanning methods is to make a beamformer, and sweep the beamformer through all possible directions [19, p. 1142]. For each direction, an estimate $\hat{E}(\theta_i, k)$ of the power received from direction θ_i and frequency bin k is made. For ease of notation, we omit the frequency bin index k. When an estimate of the output power is obtained for all directions, the direction with the highest power, is selected to be the estimated DOA [19, p. 1140]. We can formulate the optimization problem by

$$i^\star = \underset{i \in \{1,2,...,I\}}{\arg\max} \quad \hat{E}(\theta_i), \tag{4.1}$$

and the DOA-index $i^\star$ can then afterwards be mapped into a DOA in degrees. The output $\tilde{y}(k,l)$ of the beamformer is obtained by a linear combination of the beamformer

coefficients $\mathbf{w}(\theta_i)$ steered towards θ_i and the noisy signal $\mathbf{x}$ and is given as [1]

$$y = \mathbf{w}(\theta_i)^H \mathbf{x}, \tag{4.2}$$

and the expected output power of the beamformer is

$$E(\theta_i) = \mathbb{E}[yy^*|\theta_i] = \mathbf{w}(\theta_i)^H \mathbb{E}[\mathbf{x}\mathbf{x}^H]\mathbf{w}(\theta_i) = \mathbf{w}(\theta_i)^H \mathbf{C}_x \mathbf{w}(\theta_i), \tag{4.3}$$

where the beamformer coefficients can be obtained using either the closed-form solution for the Bartlett or MPDR beamformer. The Bartlett beamformer is given by $\mathbf{w}_{\text{Bart}} = \mathbf{d}(\theta_i)\,\|\mathbf{d}(\theta_i)\|_2^{-2}$ and inserting this into (4.3), leads to an estimate of the Bartlett pseudo-spectrum given as [11]

$$\hat{E}_{\text{Bart}}(\theta_i) = \frac{\mathbf{d}(\theta_i)^H \hat{\mathbf{C}}_x \mathbf{d}(\theta_i)}{\|\ \mathbf{d}(\theta_i)\ \|_2^4}, \tag{4.4}$$

for all i. The MPDR beamformer has a beamformer coefficient vector given by $\mathbf{w}_{\text{MPDR}} = \hat{\mathbf{C}}_x^{-1}\mathbf{d}(\theta_i)(\mathbf{d}(\theta_i)^H \hat{\mathbf{C}}_x^{-1}\mathbf{d}(\theta_i))^{-1}$, and similarly the MPDR psuedo-spectrum can be found as [11]

$$\hat{E}_{\text{MPDR}}(\theta_i) = \frac{1}{\mathbf{d}(\theta_i)^H \hat{\mathbf{C}}_x^{-1} \mathbf{d}(\theta_i)} \quad \text{for all } i. \tag{4.5}$$

4.1.2 MUSIC Algorithm

The MUltiple SIgnal Classification (MUSIC) algorithm is a subspace-based method and is a well-known method for estimating the DOA given its pseudo-spectrum [11]. The MUSIC algorithm exploits orthogonality between the eigenvectors from the signal and noise CPSD matrices. As both the target CPSD matrix $\mathbf{C}_s$ and the noise $\mathbf{C}_n$ are Hermitian matrices, all eigenvectors found in an eigenvalue decomposition (EVD) of the matrices are orthogonal. The MUSIC algorithm can be extended to estimate the DOA for multiple sources, but for our purpose we will assume that only a single source, namely the target, is of interest. In that case, the target CPSD matrix is $\mathbf{C}_s = \sigma_s^2 \mathbf{d}(\theta_{i_{true}})\mathbf{d}(\theta_{i_{true}})^H$ and $\text{rank}(\mathbf{C}_s) = 1$ as the space spanned by $\mathbf{C}_s$ is defined by $\mathbf{d}(\theta_{i_{true}})$. Let $\mathbf{Q}_s$ be a matrix whose column vectors are eigenvectors of $\mathbf{C}_s$ and let Λ_s be a diagonal matrix whose diagonal elements are eigenvalues $\lambda_{s,m}$, $m = 1, 2, ..., M$ of $\mathbf{C}_s$ then

$$\mathbf{C}_s = \mathbf{Q}_s \mathbf{\Lambda}_s \mathbf{Q}_s^H \in \mathbb{C}^{M \times M}. \tag{4.6}$$

Furthermore, $\mathbf{\Lambda}_s$ contains exactly one non-zero eigenvalue thus its remaining $M-1$ eigenvalues are zero. The MUSIC algorithm exploit the property that there are $M-1$ eigenvalues of $\mathbf{C}_s$ that are zero, and if we define $\mathbf{q}_{0,m}$ for $m = 1, ..., M-1$ to be the eigenvectors associated with the zero-eigenvalues, then

$$\mathbf{C}_s \mathbf{q}_{0,m} = 0, \quad m = 1, ..., M-1. \tag{4.7}$$

[1] We choose to omit the frequency and frame index such that $\mathbf{x} \triangleq \tilde{\mathbf{x}}(k,l)$ and $y \triangleq \tilde{y}(k,l)$.

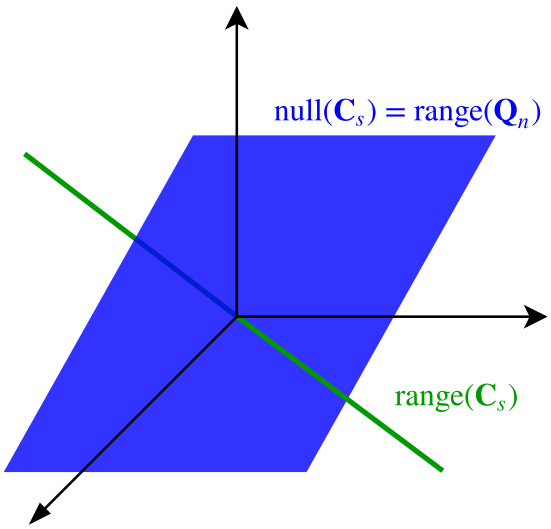


Figure 4.1: Space spanned by the signal subspace and null space.

Hence the set $\mathcal{Q} = \{\mathbf{q}_{0,1}, ..., \mathbf{q}_{0,M-1}\}$ forms a basis of the null-space of $\mathbf{C}_s$. Equivalently, if we form a matrix $\mathbf{Q}_0 = [\mathbf{q}_{0,1}, ..., \mathbf{q}_{0,M-1}]$, then the space spanned by its column vectors is in the null-space of the matrix $\mathbf{C}_s$. It then follows that null($\mathbf{C}_s$) = range($\mathbf{Q}_0$) from which it is evident that vectors in range($\mathbf{C}_s$) and vectors in range($\mathbf{Q}_0$) must be orthogonal, as illustrated in Figure 4.1. This leads to

$$\begin{aligned} \mathbf{Q}_0^H \mathbf{d}(\theta_{i_{true}}) &= \mathbf{0}, \\ \mathbf{d}(\theta_{i_{true}})^H \mathbf{Q}_0 \mathbf{Q}_0^H \mathbf{d}(\theta_{i_{true}}) &= 0, \end{aligned} \tag{4.8}$$

and the MUSIC pseudo-spectrum is defined as [13, p. 202]

$$E_{\text{MUSIC}}(\theta_i) = \frac{1}{\mathbf{d}(\theta_i)^H \mathbf{Q}_0 \mathbf{Q}_0^H \mathbf{d}(\theta_i)}, \tag{4.9}$$

and $E_{\text{MUSIC}}(\theta_i) = \infty$ if $i = i_{true}$. Therefore, the angle resulting in the largest peak in the MUSIC pseudo-spectrum, is the estimate of the DOA. In practice, the basis $\mathcal{Q}$ is not known in advance, and an estimate of the basis is needed. If the noise is spatially white with variance σ_n^2 then its CPSD matrix, $\mathbf{C}_n$, is a scalar matrix given as $\mathbf{C}_n = \sigma_n^2 \mathbf{I}$.[2] Given that the target and noise are uncorrelated, then the EVD of the noisy CPSD, $\mathbf{C}_x$, is

$$\mathbf{C}_x = \mathbf{Q}_s \mathbf{\Lambda}_s \mathbf{Q}_s^H + \mathbf{Q}_n \mathbf{\Lambda}_n \mathbf{Q}_n^H, \tag{4.10}$$

Further reduction reveals

$$\mathbf{C}_x = \mathbf{Q}_x (\underbrace{\mathbf{\Lambda}_s + \sigma_n^2 \mathbf{I}}_{\mathbf{\Lambda}_x}) \mathbf{Q}_x^H, \tag{4.11}$$

[2] A scalar matrix is a diagonal matrix where its diagonal elements are equal.

with

$$\mathbf{\Lambda}_x = \begin{bmatrix} \sigma_s^2 + \sigma_n^2 & 0 & \cdots & 0 \\ 0 & \sigma_n^2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_n^2 \end{bmatrix}, \tag{4.12}$$

where the diagonal elements are eigenvalues $\lambda_{x,m} = \mathbf{\Lambda}_x^{(m,m)}$ and $\lambda_{x,1} \geq \lambda_{x,2} = ... = \lambda_{x,M}$. The eigenvector associated with the eigenvalue $\sigma_s^2 + \sigma_v^2$ is therefore an eigenvector belonging to the signal subspace while the remaining $M-1$ belongs to the null space. The estimate of the basis vectors of the null space is therefore

$$\hat{\mathbf{q}}_{0,i-1} = \mathbf{q}_{x,i}, \quad i = 2, ..., M, \tag{4.13}$$

and $\hat{\mathbf{Q}}_0 = [\hat{\mathbf{q}}_{0,1}, ..., \hat{\mathbf{q}}_{0,M-1}]^T$. It follows that the estimate of the DOA using the MUSIC pseudo-spectrum is

$$i^\star = \underset{i \in \{1,2,...,I\}}{\arg\max} \quad \frac{1}{\mathbf{d}(\theta_i)^H \hat{\mathbf{Q}}_0 \hat{\mathbf{Q}}_0^H \mathbf{d}(\theta_i)}. \tag{4.14}$$

The partial disadvantage of the MUSIC algorithm is that assumptions of the noise are made, namely that the noise is spatially white. The MUSIC algorithm will perform poorly when coherent noise is present. To show this, we later test the MUSIC algorithm in spatially coherent noise by including an approximately isotropic noise field in the evaluation. The noise CPSD is

$$\mathbf{C}_\epsilon = \sum_{q=1}^{Q} \sigma_{v,q}^2 \mathbf{d}_v(\theta_q) \mathbf{d}_v^H(\theta_q) + \sigma_n^2 \mathbf{I}, \tag{4.15}$$

where $\mathbf{C}_\epsilon$ is used to specify that the noise consists of both spatially white noise and coherent noise. When spatially coherent noise is present, it is not possible to entirely separate the noise subspace from the signal subspace using the EVD, thus the eigenvectors associating with the $M-1$ smallest eigenvalues do to not span the signal subspace, meaning that the performance of the MUSIC algorithm will decrease. One could then treat interference noise sources as a desired signal, but this approach however requires an estimate of the number of sources in the acoustic scene.

4.2 Wideband Estimation of the DOA

The beamscans and MUSIC-pseudospectrum provide an estimate of the DOA for each frequency bin $\hat{\theta}^{(k)}$. In practice, there is only one true DOA under assumption that a single target is present, which radiates sound at all frequencies. A naive approach is to select an arbitrary frequency bin and use the beamscan or MUSIC algorithm to estimate the DOA. This might work when the SNR is high and under assumption that the target signal is temporally white but otherwise not. Instead, we will here discuss possible approaches to make an wideband extension to the narrow-band beamscan and MUSIC algorithms.

4.2.1 Peak-picking over frequency bins

We start by proposing a simple method, which is to peak-pick the DOA across frequency bins and can mathematically be described as

$$i^\star, k^\star = \underset{i,k\in\{1,2,...,I\}}{\arg\max}\ \hat{E}_\psi(\theta_i, k), \quad \psi \in \{\text{Bart,MPDR,MUSIC}\} \tag{4.16}$$

where i and k are the DOA-index and frequency bin-index respectively and the estimated DOA is $\theta_{i^\star}$. This approach however may however perform poorly, if noise is not temporally white. For example if the SNR is particularly low at a certain frequency range e.g. 10 Hz to 200 Hz, and the target signal is speech a peak-based approach will most likely pick a DOA to be towards an interference source.

4.2.2 Averaging logarithmic posterior probabilities over frequency bins

The second method we propose is similar to the concept of the Bayesian beamformer. Here we choose to interpret the pseudo-spectrums provided by the beamscan and MUSIC algorithm as estimates of the posterior probability of the DOA. Since we have K frequency bins, each frequency bin will then be associated with an estimated posterior probability distributions. Since the DOA is identical across frequency bins, the concept is to estimate the most likely observation, i.e. the most likely DOA to occur, given K probability distributions. The method presented here is not to be confused with maximum likelihood estimation.

The estimated power $\hat{E}(\theta, k)$ is desired to be interpreted as a posterior probability of the DOA. In order to do so, the softmax function is used to map the power to a probability-like sequence such that [5, p. 184]

$$P_\psi^{(k)}(\hat{\theta}_i) = \frac{\mathrm{e}^{E_\psi(\theta_i,k)}}{\sum_{j=1}^{I} \mathrm{e}^{E_\psi(\theta_j,k)}} \quad \text{and} \quad \sum_{i=1}^{I} P_\psi^{(k)}(\hat{\theta}_i) = 1, \quad \psi \in \{\text{Bart,MPDR,MUSIC}\}, \tag{4.17}$$

In case where we use the MPDR pseudospectrum, it is seen that $P_\psi^{(k)}(\hat{\theta}_i)$ reduces to an expression that is almost identical to the posterior probability estimated by the Bayesian beamformer found in subsection 3.4.3, given that the prior probability is uniform in the whole range of θ_i. The exact difference lays in $(L\gamma)$, where this constant is normalized i.e. set to $L\gamma = 1$. We then choose to model the posterior probability distribution for each frequency bin as a multinoulli distribution. The true DOA-vector over K frequency bins is then given as

$$t_i^{(k)} = \begin{cases} 1, & \text{if } i = i_{true} \\ 0, & \text{otherwise} \end{cases}, \quad i = 1, 2, ..., I, \quad k = 1, 2, ..., K, \tag{4.18}$$

where i_{true} is unknown and must be estimated. Moreover, $\mathbf{t}$ is a unit indicator vector and is 1 at the true DOA-index. The likelihood function is then given as [24, p. 209]

$$f_\psi(\mathbf{t}^{(1)}, \mathbf{t}^{(2)}, ..., \mathbf{t}^{(K)} | P_\psi^{(1)}(\boldsymbol{\theta}), P_\psi^{(2)}(\boldsymbol{\theta}), ..., P_\psi^{(K)}(\boldsymbol{\theta})) = \prod_{k=1}^{K} \prod_{i=1}^{I} P_\psi^{(k)}(\theta_i)^{t_i^{(k)}}, \quad \forall \psi. \tag{4.19}$$

and since the DOA is identical across frequency bins, this leads to

$$f_\psi(\mathbf{t} | P_\psi^{(1)}(\boldsymbol{\theta}), P_\psi^{(2)}(\boldsymbol{\theta}), ..., P_\psi^{(K)}(\boldsymbol{\theta})) = \prod_{k=1}^{K} \prod_{i=1}^{I} P_\psi^{(k)}(\theta_i)^{t_i}. \tag{4.20}$$

We now make an estimate of the most likely $\mathbf{t}$ to be observed under the constraint, that the norm of $\mathbf{t}$ must be equal 1 and that all elements except for one must be zero. The optimization problem is given as

$$\begin{aligned} j_\psi^\star = \underset{j \in \{1,2,...,I\}}{\arg\max} \quad & \prod_{k=1}^{K} \prod_{i=1}^{I} P_\psi^{(k)}(\theta_i)^{t_j} \\ \text{subject to:} \quad & \\ \|\mathbf{t}\|_2 &= 1, \\ t_j &= 1, \end{aligned} \tag{4.21}$$

and for numerical convenience, we use the natural logarithmic function to avoid rounding error from the product, which furthermore will not change the argument as the logarithm is a monotonous increasing function. Utilizing the natural logarithm reveals

$$\ln\left(\prod_{k=1}^{K} \prod_{i=1}^{I} P_\psi^{(k)}(\theta_i)^{t_j} \right) = \sum_{k=1}^{K} \sum_{i=1}^{I} t_j \ln P_\psi^{(k)}(\theta_i), \tag{4.22}$$

and the optimization problem becomes

$$\begin{aligned} j_\psi^\star = \underset{j \in \{1,2,...,I\}}{\arg\max} \quad & \sum_{k=1}^{K} \sum_{i=1}^{I} t_j \ln P_\psi^{(k)}(\theta_i) \\ \text{subject to:} \quad & \\ \|\mathbf{t}\|_2 &= 1, \\ t_j &= 1, \end{aligned} \tag{4.23}$$

which can be hard to solve analytically on this form. However, the constraints can actually be included into the optimization problem fairly simple by redefining the optimization problem slightly. Since the second constraint is $t_j = 1$ and the first $\|\mathbf{t}\|_2 = 1$, this essentially means that $t_i = 0$ for $i \neq j$. This results in $\left(t_j \ln P_\psi^{(k)}(\theta_i) = 0 \right)$ if $i \neq j$ and therefore the optimization problem can be reduced to

$$j_\psi^\star = \underset{j \in \{1,2,...,I\}}{\arg\max} \quad \sum_{k=1}^{K} \ln P_\psi^{(k)}(\theta_j), \tag{4.24}$$

and the optimum solution can be found by searching though all j and select the one that maximizes the objective function. In practice, we will omit the division by K as it does not

affect the estimate. In order obtain the RTF-vectors from the estimated DOA index $j^\star$, we assume that a database of RTF-vectors is provided such that $\mathcal{D} = (\mathbf{d}(\theta_1), \mathbf{d}(\theta_2), ..., \mathbf{d}(\theta_I))$ is an ordered tuple with the RTF-vectors for all $I = 72$ directions equidistantly spaced with a resolution of 5°. The estimated RTF-vector is then the $j^\star$'th element of the database i.e. $\hat{\mathbf{d}} = \mathcal{D}^{(j^\star)}$.

4.3 Evaluation of Model-Based Methods

In this section an evaluation on the performance of the DOA-based RTF estimators in different noise fields will be provided. We will in this performance evaluation define the target, interference, and noise signals to be temporally white (for convenience and generality) and their spatial position will remain fixed. Instead of explicitly simulate N observation of the received noisy signal, we instead directly obtain a noisy CPSD matrix $\mathbf{C}_x$ from the target CPSD $\mathbf{C}_s$ and noise $\mathbf{C}_\epsilon$. For evaluation, we use the front and rear microphones of the left hearing aid devices to obtain the RTF-vectors.

The target signal is temporally WGN with variance $\sigma_s^2(k)$ and is impinging from direction $\theta_{i_{true}}$ and has the CPSD matrix

$$\mathbf{C}_s(k) = \sigma_s^2(k)\mathbf{d}_s(\theta_{true}, k)\mathbf{d}_s(\theta_{true}, k). \tag{4.25}$$

The spatially white noise that will be included is circular symmetric complex WGN of which the CPSD is

$$\mathbf{C}_n(k) = \sigma_n^2(k)\mathbf{I}. \tag{4.26}$$

For interference, $\mathbf{C}_v(k)$, the CPSD matrix is formed by

$$\mathbf{C}_v(k) = \sum_{q=1}^{Q} \sigma_v^2(k)\mathbf{d}_v(\theta_q, k)\mathbf{d}_v^H(\theta_q, k), \tag{4.27}$$

where $Q = 72$ i.e. the interference is impinging from all 72 possible directions in order to approximately an isotropic noise field. The variance from the interference noise sources to also set to be temporally white. We first evaluate the performance on only spatially white noise i.e. $\mathbf{C}_\epsilon = \mathbf{C}_n(k)$ and afterwards evaluate on interference plus spatially white noise $\mathbf{C}_\epsilon = \mathbf{C}_v(k) + \mathbf{C}_n(k)$. In order to simulate the received signal at a specific SNR we set $\sigma_s^2 = 1$ and determine σ_v^2 and σ_n^2 so that

$$\sigma_\zeta^2(k) = 10^{\frac{\text{SNR}}{10}}, \quad \zeta \in \{n, v\}. \tag{4.28}$$

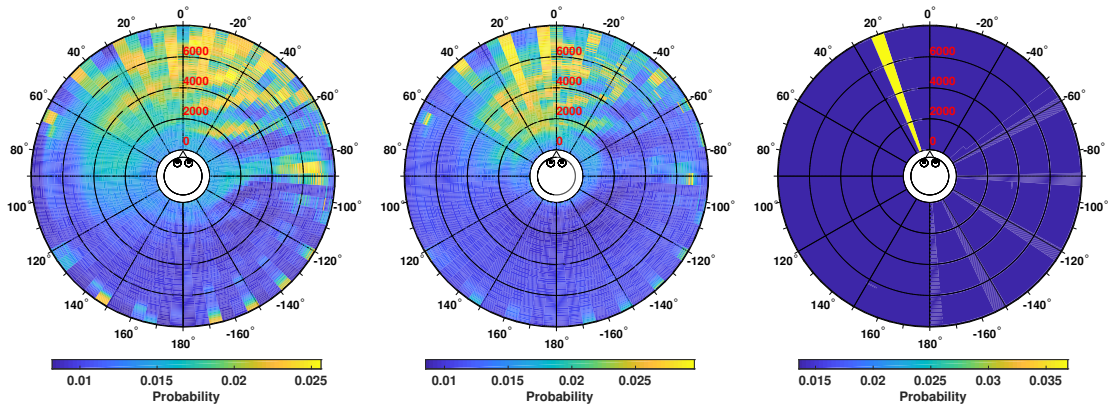
where we define[3]

$$\text{SNR} = \frac{\sigma_s^2(k)}{\sigma_v^2(k) + \sigma_n^2(k)}. \tag{4.29}$$

To control the ratio between interference and spatially white noise, a scaling factor g is used such that

$$\mathbf{C}_x(k) = \mathbf{C}_s(k) + g\mathbf{C}_v(k) + (1 - g)\mathbf{C}_n(k). \tag{4.30}$$

[3]Be aware that the SNR, in this case, may is also referred as the signal to-noise-plus-interference ratio (SINR).



(a) Bartlett **(b)** MPDR **(c)** MUSIC

Figure 4.2: Posterior probabilities as a function of frequency and angle. Spatially white noise only and SNR is 6 dB. The estimated DOAs are $\hat{\theta}_{Bart} = -10°$, $\hat{\theta}_{MPDR} = 20°$, $\hat{\theta}_{MUSIC} = 20°$.

We use $g = 0$ in the first test and $g = 0.5$ in the second.

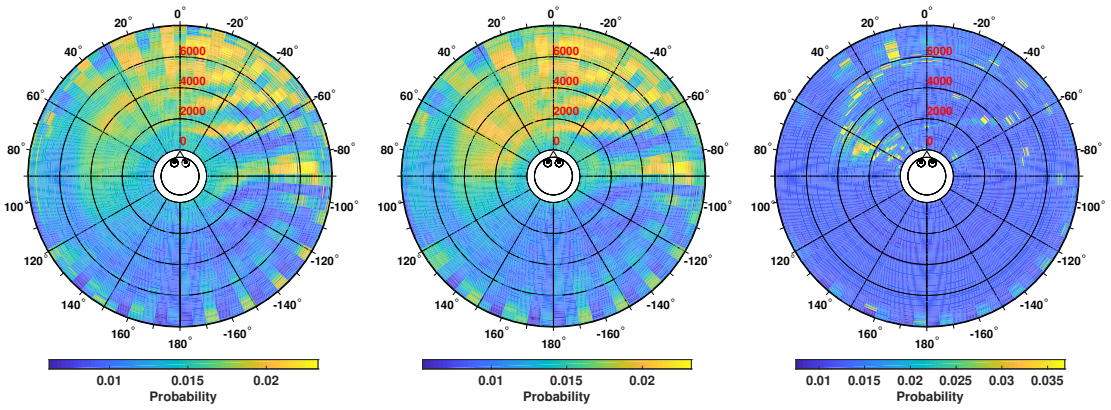
Initial tests of the wideband DOA estimation over frequency bins reveal that the peak-picking method performs very poorly compared to averaging the posterior probabilities over frequency bins at low SNR. Therefore, we choose to not include it in the evaluation.

4.3.1 DOA estimation performance in spatially white noise

In the first test, $\theta_{true} = 20°$, $g = 0$ and SNR = 6 dB and we use (4.17) to compute posterior probabilities for each frequency bin. The posterior probability is shown in a polar plot in Figure 4.2, where the radial axis is the frequency axis in the range 0 Hz to 8 kHz. The number of frequency bins is 129 without the mirrored response of the negative frequency. It is seen from the polar plots, that the posterior probability of the Bartlett beamformer provides the least accurate estimates in terms of providing a significant peak at the true DOA at $\theta_{true} = 20°$ compared to the other methods. The MPDR beamformer provides a decent posterior probability, with most of the estimated posterior probabilities peaking at $\theta_{true} = 20°$. The MUSIC algorithm outperforms both the Bartlett and MPDR beamformers, which is also expected since the noise is spatially white. The DOA is then estimated using (4.24). The estimated DOAs are $\hat{\theta}_{Bart} = -60°$, $\hat{\theta}_{MPDR} = 20°$, $\hat{\theta}_{MUSIC} = 20°$.

4.3.2 DOA estimation performance in spatially coherent noise

In the next test, we include spatially coherent noise by simulating an approximately isotropic noise field. The SNR is 0 dB and $g = 0.5$, meaning that interference is weighted as much as spatially white noise. The true DOA is as the same as previous test i.e. $\theta_{true} = 20°$ and the polar plot of the posterior probability is shown in Figure 4.3. The joint estimates of the DOAs are $\hat{\theta}_{Bart} = -10°$, $\hat{\theta}_{MPDR} = 5°$, and $\hat{\theta}_{MUSIC} = 60°$. From Figure 4.3 it seems that the posterior probability of all methods have less prominent peaks toward the true



(a) Bartlett **(b)** MPDR **(c)** MUSIC

Figure 4.3: Posterior probabilities as a function of frequency and angle. The SNR is 0 dB and $g = 0.5$. The estimated DOAs are $\hat{\theta}_{Bart} = -10°$, $\hat{\theta}_{MPDR} = 5°$, $\hat{\theta}_{MUSIC} = 60°$.

DOA. In order to determine the robustness and performance, we now choose to simulate at different SNRs between 20 dB to -6 dB with the same target and noise settings. The estimated DOA as a function of SNR is shown in Figure 4.4.

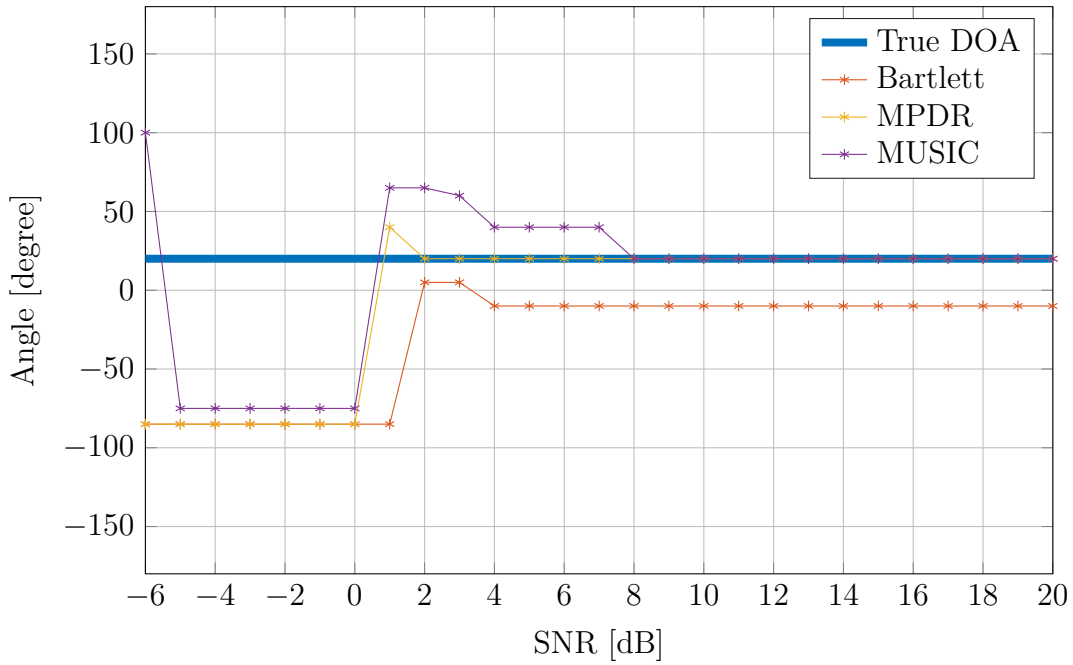


Figure 4.4: Estimated DOA as a function of SNR. The noise field is approximately isotropic, $g = 0.5$ and the true DOA is 20°.

Among all investigated methods, the Bartlett has the worst DOA estimation performance in approximately isotropic noise fields. The MUSIC algorithm outperforms both the Bartlett and MPDR beamformer when the noise is spatially white, but has a poor performance when noise field is approximately isotropic. Based on Figure 4.4 it seems

that the MPDR method provides the best performance in isotropic noise fields.

Because of the noise fields in real acoustic environment can be modeled as being approximately isotropic (such as reverberation) [23], we choose to use the MPDR beamscan method as baseline for model-based DOA estimation.

Chapter 5

Deep Learning-based RTF Estimation

The model-based methods examined in chapter 4 reveal decent performance under ideal conditions when assumptions made in their derivation are met or when the SNR is high. An example is the MUSIC algorithm studied in subsection 4.1.2. The MUSIC algorithm shows great performance under specific conditions i.e. when certain assumptions about the noise could be made, namely that the noise is spatially white. However, as soon as these conditions are not met as for example in realistic acoustic scenes, the performance of the MUSIC algorithm decreases significantly as the noise cannot be assumed spatially white. Other limitations of model-based methods are that they may not exploit all information and structures that can be found in the data and including them in the model-based methods but this will eventually result in an increased complexity.

Methods used to search for structures and dependencies found in data without relying on a complicated mathematical model, can often be found in machine learning as the general principle in machine learning is to teach a machine to recognize hidden patterns in the data generated by the unknown process. This way, one can avoid making poor assumptions that might eventually turn out to be invalid. Indeed, some methods make fewer assumptions than others and can turn out to be very robust in most scenarios. While this is true, the trade-off of robust methods is the cost of performance compared to when certain assumptions and conditions can be made in which methods such as MUSIC easily can outperform methods such as the MPDR beamscan.

Deep learning methods [5], have recently gained much attention due to increased accessibility, computational power, and big data. Furthermore, deep learning methods such as feedforward deep neural networks and convolutional neural networks (CNN) are becoming popular tools in many scientific communities as they sometime can outperform conventional methods significantly. The performance of deep learning methods is, however, bounded by the data used during training. In order to achieve a satisfactory performance, it is important to ensure that the network has obtained a good generalization of the data such that, when unseen data is presented, the network is still able to maintain a decent

performance. One of the reasons to use deep learning in acoustic beamforming is that potentially invalid assumptions can be avoided and the machine can effectively exploit structures found in the data.

This chapter is organized by first presenting an overview of current research related to multichannel speech processing using deep learning. Afterwards, this chapter will give a presentation of the proposed neural network.

5.1 Deep Learning in Acoustic Beamforming

Research in acoustic beamforming using deep learning has in the past few years gained much interest. Much of the interest is gained from the automatic speech recognition (ASR) research communities such as [25, 26, 27, 28, 29, 30, 31, 32] where the primary goal is to obtain better machine speech recognition performance in situations where more than one microphone is available. A natural choice is therefore to combine acoustic beamforming and deep learning where acoustic beamforming is a well-studied field and deep learning being a well-established method in ASR. We can roughly divide the research areas in acoustic beamforming combined with deep learning into following:

1. End-to-end multichannel ASR systems.
2. Approaches which estimate beamformer coefficients directly.
3. Approaches which estimate the parameters for beamformers.

with each group having their own advantages.

5.1.1 End-to-end multichannel ASR systems

Instead of explicitly using a linear beamformer to enhance the target signal researchers in group 1 such as [32] are designing neural networks that directly minimizes performance measures such as the word error rate (WER). This way the performance is not limited by a linear beamformer as for the other groups, as the neural networks may form outputs from non-linear combinations of the input data. The issue is however that it is hard to control the behavior of a network and if the target signal is desired undistorted at the reference microphone, this cannot be guaranteed with these networks.

5.1.2 Approaches which estimate beamformer coefficients directly

In group 2 e.g. [26, 27], the overall objective is to let a neural network estimate the beamformer coefficients. This approach, however, seems to be somewhat strange if the objective of the neural network is to estimate the coefficients for a linear beamformer as presented in for example [26] and [27]. If, for example, the objective is solely dedicated to predict the beamformer coefficient for a Bartlett beamformer, for example, then an equivalent network estimation the RTF-vector should be able to perform just as well.

Nonetheless, making a neural network solely for estimating the beamformer coefficient of a conventional beamformer might not be a good approach. For example, it can be shown that the MPDR beamformer is best beamformer among all linear beamformers that minimizes the noise power, while keeping the target distortionless because of its optimality criterion. The optimization problem of the MPDR beamformer, moreover, has a closed-form solution (see section 3.4). As $\mathbf{C}_x$ is guaranteed to be positive-semidefinite, the objective function is convex with linear constraints. A minimizer to the optimization problem is therefore also guaranteed to be the global minimizer. Thus rather than designing a network estimating the beamformer coefficients, one can instead create a network parameter estimating $\mathbf{d}$ to the MPDR or Bayesian beamformer. This way, one will both be able to obtain the desired beamformer coefficients while achieving important insights such as the direction of the target which might be useful for other subsystems on a hearing aid.

5.1.3 Approaches which estimate the parameters for beamformers

Finally, researchers in group 3 are in some sense more faithful to conventional beamforming methods as their approach is to limit the task of the neural network to only cover parameter estimation such as $\mathbf{d}$ and noise CPSD matrices $\mathbf{C}_\epsilon$. By limiting the neural network to only solving subproblems one might be able to obtain a more specialized neural networks which could result in a increased performance [33]. Some work related to beamformer parameter estimation using deep learning research are [25, 29, 34, 35]

In [28] they propose a method to estimate the time-frequency masks to determine whether a time-frequency bin is dominated by target speech or interference plus noise. Here they make two networks, where to estimates the time-frequency mask for the target and another for the noise. These masks are later used to estimate the CPSD matrices of the target and noise. For training, they use an ideal binary mask (IBM) with a threshold for target speech, $\text{thr}_s(k)$, and noise, $\text{thr}_\epsilon(k)$, to mark which time-frequency bin that are dominated by either speech or noise. The IBM is defined as

$$\text{IBM}_\epsilon(k,l) = \begin{cases} 1, \text{ if } \frac{\|\mathbf{d}(\theta_{true})s(k,l)\|}{\|\boldsymbol{\epsilon}(k,l)\|} < 10^{\text{thr}_\epsilon(k)} \\ 0, \text{ otherwise }, \end{cases} \tag{5.1}$$

and a similar IBM is defined for the target speech such that

$$\text{IBM}_s(k,l) = \begin{cases} 1, \text{ if } \frac{\|\mathbf{d}(\theta_{true})s(k,l)\|}{\|\boldsymbol{\epsilon}(k,l)\|} > 10^{\text{thr}_s(k)} \\ 0, \text{ otherwise }, \end{cases} \tag{5.2}$$

where $\text{thr}_\epsilon(k) \neq \text{thr}_s(k)$ and is chosen to limit the false rate [28]. For example, one can let $\text{thr}_s(k)$ be very large, and the network will classify a time-frequency bin to be dominated by target speech, when the SNR is sufficiently high. Equivalently, $\text{thr}_\epsilon(k)$ can be chosen to be very small, meaning that time-frequency bins will only be classified as noise dominant when the SNR is sufficiently low. The target and noise CPSD are then estimated as [28]

$$\mathbf{C}_\psi(k,l) = \sum_{m=1}^{l} \mathcal{M}_\psi(k,m)\mathbf{x}(k,m)\mathbf{x}^H(k,m), \quad \psi \in \{s,\epsilon\}. \tag{5.3}$$

The estimate of the noise CPSD $\mathbf{C}_\epsilon$ can then be used to estimate the beamformer coefficients of an MVDR beamformer [29]

$$\mathbf{w}_{\text{MVDR}} = \frac{\mathbf{C}_\epsilon^{-1}\mathbf{d}}{\mathbf{d}^H\mathbf{C}_\epsilon^{-1}\mathbf{d}} = \frac{\mathbf{C}_x^{-1}\mathbf{C}_s\mathbf{i}}{\text{trace}\left(\mathbf{C}_x^{-1}\mathbf{C}_s\right)}, \tag{5.4}$$

where $\mathbf{i}$ is an unit vector with 1 at the reference microphone index. The first equality uses the RTF vectors, and thus requires that one must estimate these. Otherwise, one may use the second equality. This approach is, however, hard to guarantee that the target will remain undistorted as it requires a very good estimate of the target CPSD $\mathbf{C}_s$. Furthermore, this approach does not ensure that the target CPSD is $\text{rank}(\mathbf{C}_s) = 1$ as the space spanned by the target vectors are defined by RTF-vector. If these requirements are not met then the MPDR beamformer may distort the target.

Other researchers attempt to estimate the RTF-vector have been proposed in [30, 35, 36]. The authors in these articles are focused on using deep learning to obtain an estimate of a discrete set of possible DOAs. Although they are not explicitly estimating the RTF-vector, their methods can easily be extended to estimating the RTF-vector if a database of RTF-vectors associated with each DOA is available. In [30, 36] they seek to estimate the DOA based on an eigenvalue decomposition (EVD) of the noisy CPSD matrices estimates. Their approach is very similar to the concept of MUSIC but instead of forming the MUSIC-pseudospectrum (see chapter 4), they design a neural network whose inputs are the noise eigenvectors used in the MUSIC algorithm. Their issues remain the same as for the MUSIC algorithm. First, they must perform some preprocessing which determines the number of sound sources in the acoustic scene. Secondly, their method might perform less well in reverberant rooms and in realistic acoustic scenes due to coherent noise sources, if their preprocessing in determining the number of target sources is poor.

Finally, in [35] they seek to estimate the DOA based on the phase of the STFT coefficients across frequency bins. The STFT coefficients of the received signal $x^{(m)}(k,l)$ can be written into the following form with magnitude $A_m(k,l)$ and phase $\phi_m(k,l)$ for the m'th microphone [35]

$$x^{(m)}(k,l) = A_m(k,l)\mathrm{e}^{j\phi_m(k,l)}. \tag{5.5}$$

The magnitude part $A_m(k,l)$ is then ignored (similarly to the phase transform (PHAT) method [13, p. 192]), and the network is trained on the input feature $\phi_m(k,l)$ to classify the DOA from a discrete set of possible DOAs.

There are two potential drawbacks with their approach. First, they choose to ignore the magnitude part which contains the magnitude response of the RTF and since the magnitude response is different between DOA and microphones when mounted on a head. Secondly, they limit their network to only predict the DOA on one frame and under the assumption that the target's spatial position does not change, using multiple frames for estimating the DOA will provide a better performance. The trade-off is however that using more frames might lead to the network being slow reacting to changes if the target is changing spatial position.

Inspired by the work of [35], the objective of the proposed DNN is to estimate the RTF-vectors. The proposed network will, in contrast to [35], seek to utilize 1) the magnitude part of the observed signals $\mathbf{x}$ and 2) more than one frame to estimate the RTF-vector. Related to this, there will be two approaches to estimate the RTF-vectors. The

first approach is to estimate the DOA based on a classification problem with a finite set of possible DOAs i.e. $(\Theta \in \{-175°, -170°, ..., 180°\})$. This network will be referred to as DNN-DOA. Afterwards, the corresponding RTF-vector associated with the estimated DOA, will be selected from a database of RTF-vectors and the network will be referred to as DNN-RTF.

5.2 Proposed DNN Architecture

In this section an overview and discussion of the proposed DNNs input, output and the overall structure is provided.

5.2.1 Input of the Network

The choice of input feature is partly inspired by [35]. It is believed that better performance can be achieved if more past information and the magnitude of the STFT coefficients are provided the network under assumption that the spatial position of the target is only slowly time varying. A simple extension to the network in [35] is to feed the past L frames into the network. This, however, results in an increased input dimension. An alternative input, which will be used, are the noisy CPSD matrices $\mathbf{C}_x$. As the noisy CPSD matrices are time averages of the outer product of the observed signal in the frequency domain, the CPSD matrices contains past information. One can also interpret the noisy CPSD matrices as a form of smoothing of the STFT-coefficients since the diagonal elements of the estimated CPSD are smoothed power spectrums of the observed signal at each microphone.

Another advantage is to also utilize the magnitude part. However, instead of feeding the network, the complex numbers in polar form i.e. magnitude and phase, we choose to feed the network rectangular form i.e. real and imaginary values resulting in low amount of preprocessing.

5.2.2 Input normalization

The magnitude of the CPSD matrices might vary depending on the gain applied to the microphone signals (e.g. amplifier in a hearing aid). Thus in order to increase the robustness, the CPSD matrices are normalized with respect to the average input power between microphones. We can do this by averaging the trace of the CPSD matrices across frequency bins such that the normalized CPSD matrices are

$$\hat{\mathbf{C}}_x(k,l) \leftarrow \frac{\hat{\mathbf{C}}_x(k,l)}{\frac{1}{K}\sum_{k=1}^{K} \operatorname{tr}\left(\hat{\mathbf{C}}_x(k,l)\right)}. \tag{5.6}$$

The exact derivation is shown in Appendix C. We can now formulate the input to the proposed DNNs. Let M be the number of microphones and $\mathbf{B}(k,l)$ be a block matrix

given as

$$\mathbf{B}(k,l) = \begin{bmatrix} \mathrm{Re}\{\hat{\mathbf{C}}_x^{(1,1)}(k,l)\} & \mathrm{Im}\{\hat{\mathbf{C}}_x^{(1,1)}(k,l)\} & \cdots & \mathrm{Re}\{\hat{\mathbf{C}}_x^{(1,M)}(k,l)\} & \mathrm{Im}\{\hat{\mathbf{C}}_x^{(1,M)}(k,l)\} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathrm{Re}\{\hat{\mathbf{C}}_x^{(M,1)}(k,l)\} & \mathrm{Im}\{\hat{\mathbf{C}}_x^{(M,1)}(k,l)\} & \cdots & \mathrm{Re}\{\hat{\mathbf{C}}_x^{(M,M)}(k,l)\} & \mathrm{Im}\{\hat{\mathbf{C}}_x^{(M,M)}(k,l)\} \end{bmatrix}, \tag{5.7}$$

where $\mathbf{B}(k,l) \in \mathbb{R}^{M \times 2M}$ then the input matrix to the neural network is structured as follows

$$\mathcal{C}_x(l) = \begin{bmatrix} \mathbf{B}(1,l) \\ \mathbf{B}(2,l) \\ \vdots \\ \mathbf{B}(K,l) \end{bmatrix}, \tag{5.8}$$

where $\mathcal{C}_x(l) \in \mathbb{R}^{KM \times 2M}$.

5.2.3 Output of the Network

The DNN architecture between DNN-DOA and DNN-RTF will be almost identical with the only difference being at the output layer. For DNN-DOA the desired outputs are posterior probabilities of a discrete set of possible DOAs. The set is $\Theta = \{-175, -170, ..., 180\}$ with $I = 72$ elements and $\theta \in \Theta$. The notation θ_i will be used to refer to the DOA from i'th element in the set Θ i.e. $\theta_1 = -175$. The posterior probabilities estimated by the DNN-DOA will be denoted as $\hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x, \mathcal{A})$ where $\mathcal{A}$ is the set of all parameters of the neural network. A natural choice to obtain an estimate of the DOA is to use a maximum a posterior (MAP) estimate given the network output i.e.

$$\hat{i} = \underset{i \in \{1,2,...,I\}}{\arg\max}\ \hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x, \mathcal{A}) \tag{5.9}$$

where $\hat{i}$ is the estimated DOA index which can used to obtain a RTF-vector from a look-up table e.g. $\mathcal{D} = (\mathbf{d}(\theta_1), \mathbf{d}(\theta_2), ..., \mathbf{d}(\theta_I))$ with the RTF-vector being $\hat{\mathbf{d}} = \mathcal{D}^{(\hat{i})}$. In order to ensure that the output of the neural network can be interpreted as a probability the activation function at the output layer is selected to be the softmax function as it maps the output to a value between [0,1] while ensuring that the sum of all outputs equals one.

For the DNN-RTF, the network must directly map the input into an RTF-vector $\hat{\mathbf{d}}$ at the output layer. The activation function at the output layer in this case is a linear function. Since the output of the network is real valued the output must be reconstructed into a complex RTF-vector again. The output of the network will be trained to output all the $M \cdot K$ values of the real part of the estimated RTF-vector followed by another $M \cdot K$ values of the imaginary part.

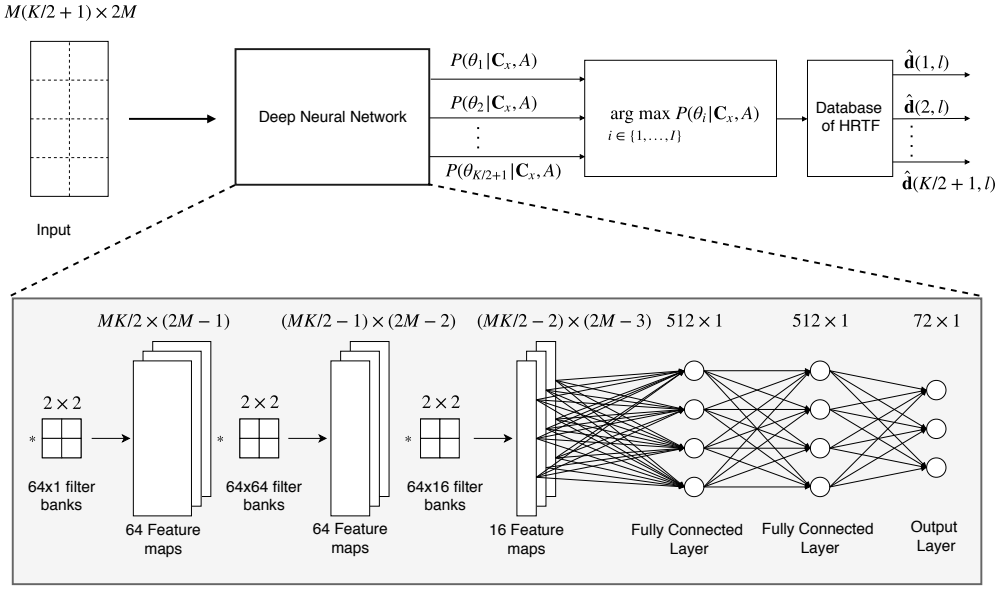


Figure 5.1: The architecture of the proposed DNN-DOA.

5.2.4 Structure of Neural Network

The design of the proposed DNNs is based on an iterative process with trial and error, until a satisfactory result is obtained. However, the starting point of the DNN architecture was the structure of the DNN proposed by [35] thus the overall structure of neural network is very similar to the one proposed in [35]. The first three layers of the neural network are convolutional layers followed by three feedforward networks. The architecture of the DNN-DOA is shown in Figure 5.1.

Equivalently, the DNN architecture for the DNN-RTF is shown in Figure 5.2. We would like to stress out, that the primary difference between DNN-DOA and DNN-RTF is that DNN-DOA estimates the DOA of the target of a discrete set and maps the estimated DOA to RTF-vectors with a database of RTF-vectors associated with each direction. The DNN-RTF instead estimates the RTF-vectors without relying on a database of predefined RTF-vectors. By using a convolutional network, the DNN can search for local structures that might occur in the data. A convolutional layer as input also contains relatively low amount of parameters compared to a fully connected layer because of parameter sharing [5, p. 335].

For the DNN-DOA the convolutional layer is followed by three fully connected layers and an output layer with 72 units where each output gives an estimate of the posterior probability of the target direction. The DNN-RTF outputs $M \cdot K$ units, where each output is an estimate of the real or imaginary part of the RTF-vectors. An overview of the network is given in Table 5.1.

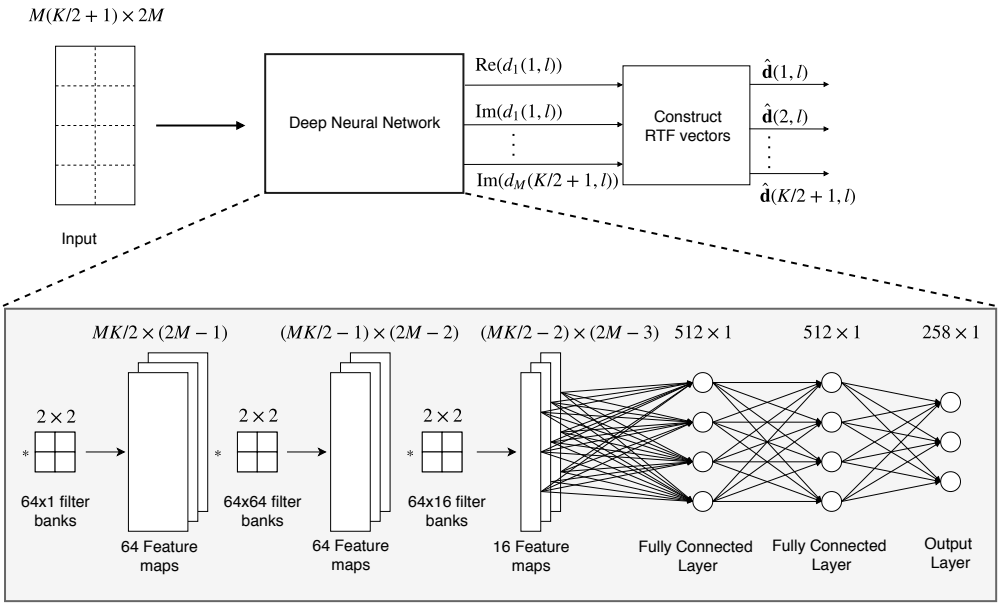


Figure 5.2: The architecture of the proposed DNN-RTF.

Table 5.1: Table of network architecture

	Type	Units/feat. maps	Kernel	Act. function	# parameters
Layer 1	Conv.	64	2×2	ReLU	320
Layer 2	Conv.	64	2×2	ReLU	16,448
Layer 3	Conv	16	2×2	ReLU	4,112
Layer 4	FC.	512	-	ReLU	2,089,472
Layer 5	FC.	512	-	ReLU	262,656
Layer 6	FC.	512	-	ReLU	262,656
Layer 7	FC.	72/258	-	Softmax/Linear	36,864/132,354

5.3 Training and Testing the Network

Training the neural network is one of the most crucial tasks when developing neural networks. It is eventually desired to train the neural network, such that the network is able to recognize the appropriate patterns in the data and obtain a good generalization. The generalization of the network is closely related to the training data and network capacity. Too little training data and the network might eventually not be able to recognize the pattern in the data. However, variety in the data is also important, to avoid overfitting and maintain the performance when unseen data is presented. In this section we will discuss how the datasets are generated and how the network is trained in order to obtain good generalization.

5.3.1 Overivew of the training and test datasets

Recall that estimates of noisy CPSD matrices are used as input to the DNNs (see (5.8)). Furthermore, recall that the noisy CPSD matrices are estimated based on a number of past L noisy frames (see (3.9)). This number of frames is crucial as this will affect the behavior in situations where the DOA is changing. The trade-off as previously discussed was that a CPSD matrix estimated over few frames is capable of tracking fast changes of the DOA at the cost of having an estimation error with a higher variance compared to a CPSD estimates with a large number of frames but at the cost of being slow reacting to changes in the DOA. Because of this, the DNN-DOA and DNN-RTF are trained on CPSD formed over various number of past noisy frames.

For training purposes the true DOA is fixed and the tracking capabilities of the DNN-DOA and DNN-RTF will be verified in a separate test. To generate the training and test data, the following signal model is used:

$$\tilde{\mathbf{x}}(k,l) = \tilde{\mathbf{d}}\left(R_s, \theta_s\right) \tilde{s}(k,l) + g \sum_{q=1}^{Q} \tilde{\mathbf{d}}\left(R_q, \theta_q\right) \tilde{v}_q(k,l), \tag{5.10}$$

where the parameters are defined in chapter 2. When generating the data, parameters that will be randomized are $R_s, \theta_s, R_q, \theta_q, \tilde{v}_q, \tilde{s}, g, Q$. These parameters will however by randomized differently depending on the noise field and SNR. The RTF-vectors are obtained from HRTFs of the front and rear microphones on the left hearing aid placed on a dummy head [9].

The target source, $\tilde{s}$, is always speech and is obtained from the TIMIT database [37]. The TIMIT database consists of recorded sentences from human subjects. The recordings are concatenated into a single recording for each subject which afterwards is normalized with respect to the peak value. The speech absent frames are then sorted out with a short-time energy voice activity detector [38] to only include speech present frames.

Possible interferences that are used to create the noise field in the acoustic scene are speech/babble (BBL), speech shaped noise (SSN), buses (BUS), cafeterias (CAF), pedestrian (PED), and streets (STR). The babble noise is obtained from the TIMIT database and processed like for the target speech, while the BUS, CAF, PED, and STR noises are obtained from the CHiME3 challenge database [39]. Finally, the speech shaped noise is artificially generated by filtering WGN with the impulse response of an all-pole model of the speech. A 12'th-order all-pole model is estimated by concatenating 100 speech recordings from the TIMIT database and then use the Yule-Walker equations to obtain the filter coefficients as described in [33, 40].

5.3.2 Generating acoustic scenes with isotropic Noise Field

We generate acoustic scenes where the noise field is approximately isotropic, as the noise field in many realistic acoustic scenes may be approximately isotropic such as reverberation from a room and competing speakers in a cocktail party. Since the HRTF database is limited to HRTFs measured at 72 different azimuth angles on a circle at a distance of 3 meters, the isotropic noise field is approximated to include 72 uncorrelated

noise sources placed on a circle such that exactly all HRTF are used to generate the approximated isotropic noise field.

In this acoustic setup, all $Q = 72$ noise sources, $\tilde{v}_q(k,l)$ for $q = 1, ..., Q$, are placed at a distance of $R_q = 3$ meters from the center of the head, where $\theta_q = 5q - 180°$ for $q = 1, ..., Q$ is the direction of the q'th noise source.[1] In order to control the SNR the variable gain of the noise sources g is introduced. The SNR is randomized with uniform distribution in the range -6 dB to 10 dB.

5.3.3 Randomly placed noise sources

Realistic acoustic scenes may also be anisotropic, for example when a single competing speaker is present. Here we generate acoustic scenes whose noise field is anisotropic. First, we randomize the number of noise sources Q to a value between 1 and 40 with a uniform distribution. After selecting the number of noise sources, the distance and azimuth angle of both the target and noise sources are randomized with a uniform distribution $\mathcal{U}(\cdot,\cdot)$ such that

$$R_s \sim \mathcal{U}(0.5, 1),\ R_q \sim \mathcal{U}(1, 4),\ \theta_s \sim \mathcal{U}(-90°, 90°),\ \theta_q \sim \mathcal{U}(-180°, 180°). \tag{5.11}$$

The distance between the noise sources and the head has an upper bound of 4 meters to ensure that the noise signal will not become too attenuated when placed far away, making it redundant in the simulation. Since the azimuth angle of the HRTF is limited to a resolution of 5 degrees, θ_s and θ_q are rounded to the nearest available azimuth angle. Again, the gain g is adjusted to the desired SNR between -6 dB and 10 dB. There is however a drawback by adjusting the SNR with this approach. When for example 1 noise source is present and the SNR is -6 dB the noise source will be louder than the target speech. In an acoustic scene, this is equivalent to physically place the noise source at a closer distance to head than the target source.

5.3.4 The generated data set

In the developed system we will be using two microphones i.e. $M = 2$. A table of the generated training set with specified noise type and field is given in Table 5.2. It is seen that babble noise and speech shaped noise occurs more often in the training set compared to other noise types. We hypothesize that the target will primarily be speech, and a common acoustic scene where the hearing aid user will benefit from a beamformer are cocktail parties and restaurant environments [1, p. 122]. We choose to model the noise in such acoustic scenes as babble and speech shaped noise and include larger number of these acoustic scenes in the training and test set. In Table 5.2 a overview of the training set is provided.

[1] Frontal is when the azimuth angle is 0°, left is positive and right is negative angles.

	BBL	BUS	CAF	PED	SSN	STR
Isotropic						
T:200 ms	50000	10000	10000	10000	50000	10000
T:500 ms	50000	10000	10000	10000	50000	10000
T:1000 ms	0	0	0	0	20000	0
Anisotropic						
T:200 ms	20000	10000	10000	10000	50000	10000
T:500 ms	20000	10000	10000	10000	20000	10000
T:1000 ms	20000	0	0	0	0	0

Table 5.2: Number of generated acoustics scenes in various noise types and noise fields. "T" refers to the number of past frames the CPSD was formed over (in seconds, where 200 ms = 25 frames). For example T:200 ms means that the CPSD matrices were estimated over 25 past frames. There are in total 510,000 examples of acoustic scenes and the total amount of training hours is 55.55 hours.

Out of the 510,000 examples, 10 percent are reserved to the validation set which is not used for training but to evaluate if the network is over-fitting on the fly. For the test set, we generate 51,000 examples with the same distribution of noise type, time, and noise fields. It is important that the network is tested on unseen data, and therefore the we use unseen speech signals from the TIMIT database and unseen noise (but same noise type) from the CHiME3 database. To avoid that the network becomes biased, training data is shuffled each time a training epoch is completed.

The network parameters are determined through an optimization problem. For the DNN-DOA we use cross-entropy and for the DNN-RTF we use mean-squared error as objective functions. A detailed explanation of network optimization is provided in Appendix A. The selected iterative solver is the Adam optimizer [41] with a learning rate of 0.001. The minibatch size is 50 examples and we ran the training set for 20 epochs. Dropout and regularization were not added to the network as experiments indicated that they were not needed.

Chapter 6

Evaluation and Experimental Results

This chapter serves to evaluate the performance of the proposed DNN supported beamformers and DOA estimators, and compare them with model-based equivalences, to examine if a DNN-based approach is able to outperform a model-based in low SNR. In section 6.1 a comparison between the proposed DNN for DOA estimation (DNN-DOA) and a model-based MPDR beamscan method (MPDR-DOA) will be provided. The comparison will be in terms of confusion matrices and mean-absolute error (MAE) as a function of SNR.

Afterwards, in section 6.2, the performance of the proposed DNN supported beamformers will be compared with a model-based Bayesian beamformer (Bayes-Model) in terms of Extended Short-Time Objective Intelligibility (ESTOI), Perceptual Evaluation Speech Quality (PESQ), and segmental SNR (segSNR). The proposed DNN supported beamformers are: 1) a Bayesian beamformer, where posterior probabilities are estimated with the DNN-DOA (Bayes-DNN-DOA), 2) an MPDR beamformer where the DOA is estimated with the DNN-DOA (MPDR-DNN-DOA), and 3) an MPDR beamformer where the RTF-vectors are estimated by a DNN (MPDR-DNN-RTF).

The performance of the DOA estimation and beamformer algorithms are evaluated in simulated acoustic scenes where the front and rear microphones of a left hearing aid, device mounted on a dummy head [9], is used. The number of parameters that can be adjusted in an acoustic scene, such as type of noise field and noise type, is large, and for convenience some of the parameters will be fixed when generating the acoustic scenes. Among the variable parameters that must be considered, are:

1. **Noise field**
   The noise field can either be anisotropic or isotropic. Anisotropic noise field can easily be implemented and synthesized, but might be less suitable for evaluation and comparison as we must further consider a) the number of noise sources, b)the spatial distribution of the noise sources, and c) the power of each noise source. Simulating an approximately isotropic noise field can, however, be more convenient as

the number of noise sources is fixed (72 sources in this case), the spatial distribution of noise sources is uniform, and the power radiated from each noise source is equal.

2. **Noise type**
   The noise type may also range from being speech shaped noise, bus noise, cafeterias, babble noise, street noise, and pedestrian to mention a few. Speech shaped noise is however artificially generated and does not occur in real acoustic environments. Natural acoustic scenes where the hearing user will benefit from a beamformer, can be a noisy environment such a cocktail party or a restaurant [1, p. 122] for which reason, we choose the noise type of the acoustic scene to be babble noise. The babble noise received at the microphones is formed as a superposition of speech impinging uniformly from all 72 directions. The speech signals are recordings from the TIMIT database, but were not included in the training of the DNNs. Furthermore, an isotropic model serves as a reasonable model natural noise sources e.g. long-term reverberation [23].

3. **SNR**
   For the SNR dimension, since one of the motivations behind this project is to develop a DNN supported acoustic beamformer able to outperform a model-based beamformer in low SNRs, the proposed and baseline beamformers are tested under a variety of different low SNRs.

4. **Target direction**
   The target direction will play a role in DOA estimation and beamforming performance. For example, the power received from a source of which the direct path is blocked by the head, might be lower than if the target source were placed at a direction where the head is not blocking the sound. Therefore, the performance of DOA estimation and beamforming must be conducted at different target directions.

6.1 DOA Estimation

Here we compare the MPDR-DOA and the DNN-DOA algorithms. Both algorithms output estimated posterior probabilities of the DOA, and the estimated DOA for both algorithms is selected to be the direction with the largest posterior probability. The systems are tested in artificially generated acoustic scenes and the settings are

Noise Field	Noise Type	SNR	Target direction
Isotropic	Babble	0, -6, -12 dB	-175°, -170°, ..., 180°

Table 6.1: The configuration of the acoustic scenes for testing the DOA estimation of MPDR-DOA and DNN-DOA. The MPDR-DOA and DNN-DOA are tested for 0 dB, and -6 dB and -12 dB SNR. Results for -6 dB and -12 dB SNR are given in Appendix B

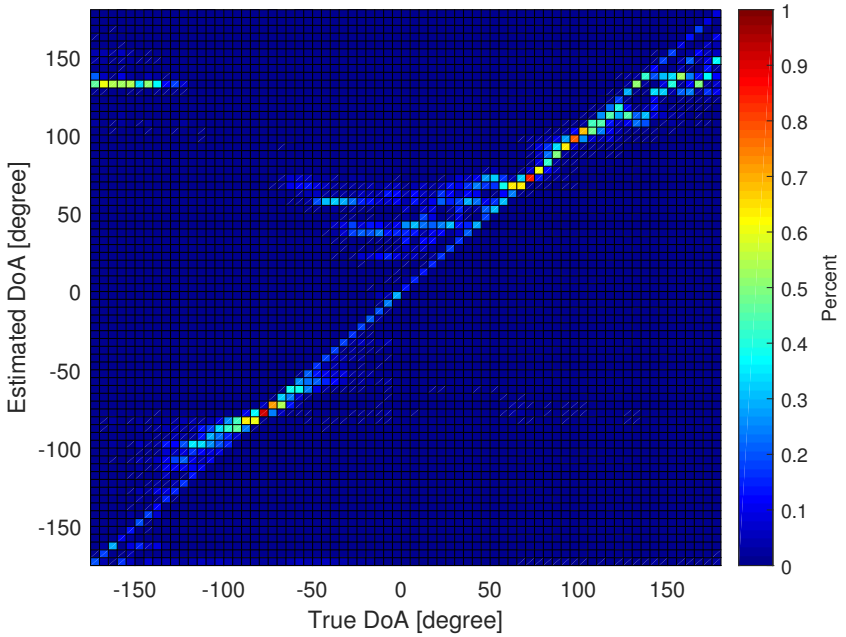


Figure 6.1: Confusion matrix for MPDR-DOA at 0 dB SNR. 1000 realization of acoustic scenes with different target talkers, different noise waveforms, are generated and tested for each column.

Furthermore, each CPSD matrix is estimated over $L = 25$ frames and estimated with (3.9). For DOA estimation, the results are presented in forms of confusion matrices seen in Figure 6.1 for the MPDR-DOA and Figure 6.2 for the DNN-DOA for an SNR of 0 dB for isotropic babble noise. The confusion matrices show the estimated DOA versus the true DOA, where the column k for angle θ_k in the confusion matrix, indicates the distribution of estimated DOA in the range -175 to 180 degrees for angle θ_k. A red diagonal line in the confusion matrix implies, that the majority of estimated DOA are correct. It is seen that the diagonal line for the DNN-DOA is more prominent than the MPDR-DOA, meaning that the DNN-DOA is more robust than the MPDR-DOA at estimating the DOA at 0 dB SNR. This was also expected as the DNN-DOA, has been trained on similar acoustic scenes, where the noise field is approximately isotropic and with babble noise.

The mean-absolute error (MAE) is shown in Figure 6.3. The MAE can be computed by averaging the absolute error between the estimated and true DOA. It is seen that the MPDR-DOA has a low MAE at approximately -100 to -70 degrees and equivalently around 70 to 120 degrees, but is otherwise high. At 70 to 120 degrees, the target is closest to the microphones and the power received might be slightly higher compared to other directions, which may explain the low MAE. The increased performance at -100 to -70 degrees may not appear obvious, as target is blocked by the head. The low MAE might however be explained by diffraction caused by the head [2, p. 201] [42]. The DNN-DOA is seen to have a lower MAE on average, than the MPDR-DOA indicating a better performance at 0 dB SNR. Results for -6 dB and -12 dB SNR are shown in Appendix B. The comparison concludes that the DNN-DOA is more robust at estimating the DOA at 0 dB, -6 dB and -12 dB SNR.

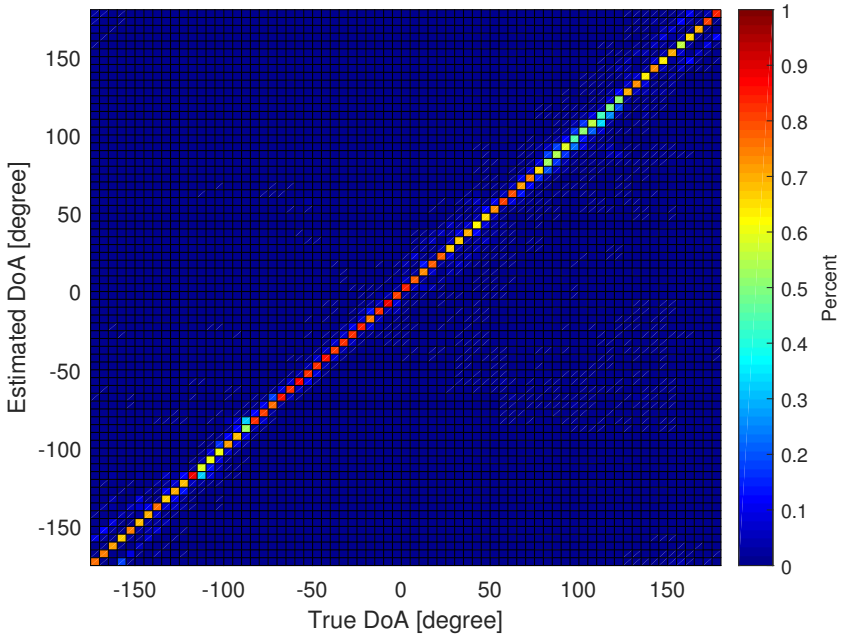


Figure 6.2: Confusion matrix for DNN-DOA at 0 dB SNR. 1000 realization of acoustic scenes with different target talkers, different noise waveforms, are generated and tested for each column.

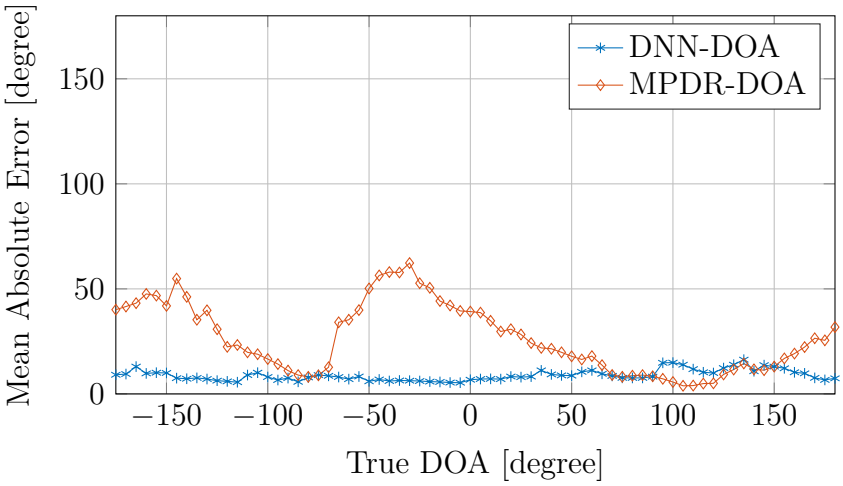


Figure 6.3: Mean absolute error for isotropic babble noise with an SNR of 0 dB at the reference microphones.

6.2 Beamformer Performance

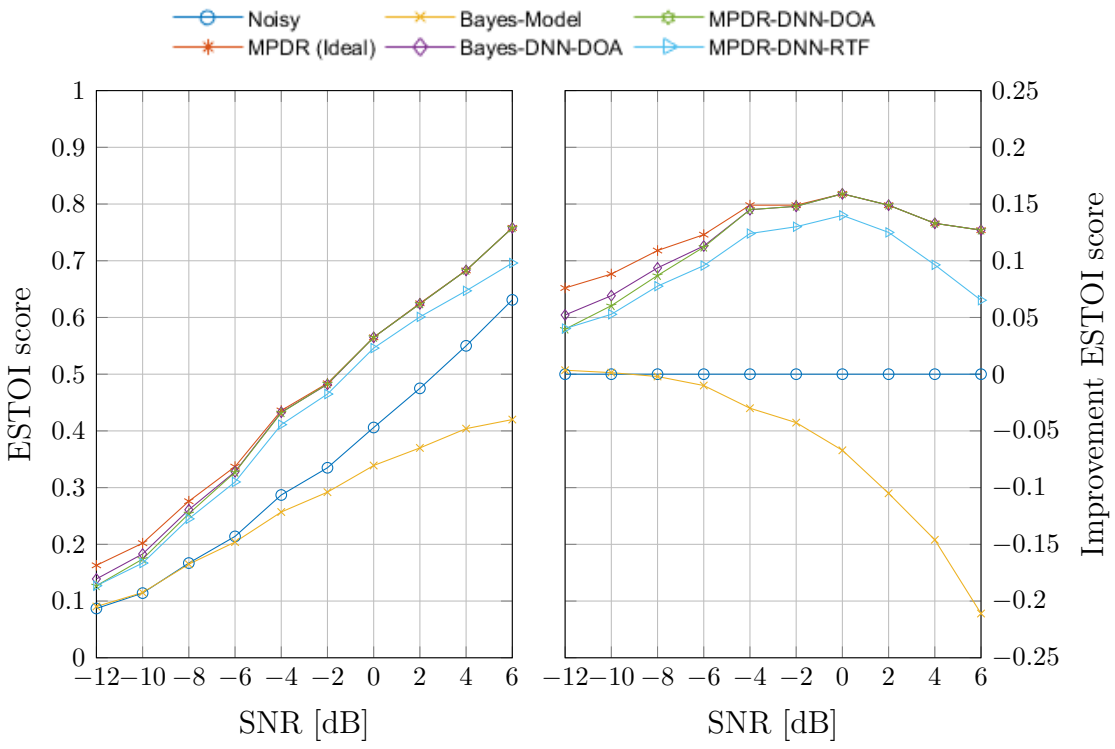
In this section, the proposed and baseline beamformers are compared in terms of ESTOI, PESQ, and segSNR scores. For reference, the unprocessed noisy observations and an ideal MPDR beamformer with the true DOA, are included. The ESTOI, PESQ, and segSNR are computed as a function of SNR and for target directions -90, 0, 90, and 180 degrees. The acoustic scene settings are:

Noise Field	Noise Type	SNR	Target direction
Isotropic	Babble	6,4,2,...,-12 dB	-90°, 0°, 90°, 180°

Table 6.2: The configuration of the acoustic scenes for evaluating the performance of the proposed beamformers.

6.2.1 Speech intelligibility

Here we evaluate the speech intelligibility predicted by ESTOI [43] of the output of the beamformers. The details of ESTOI will not be covered in this thesis, but can be found in [43, 44]. The input to the predictor are the clean target signal, and the output of the beamformer. The score of ESTOI is between -1 and 1, where a higher score indicates high speech intelligibility. The ESTOI scores for SNRs between -12 dB and 6 dB for a target direction of 0° is shown in Figure 6.4a and the improvement in Figure 6.4b. The ESTOI scores for target directions -90, 90 and 180 are shown in Appendix B.



(a) ESTOI score. **(b)** ESTOI improvement relative to noisy.

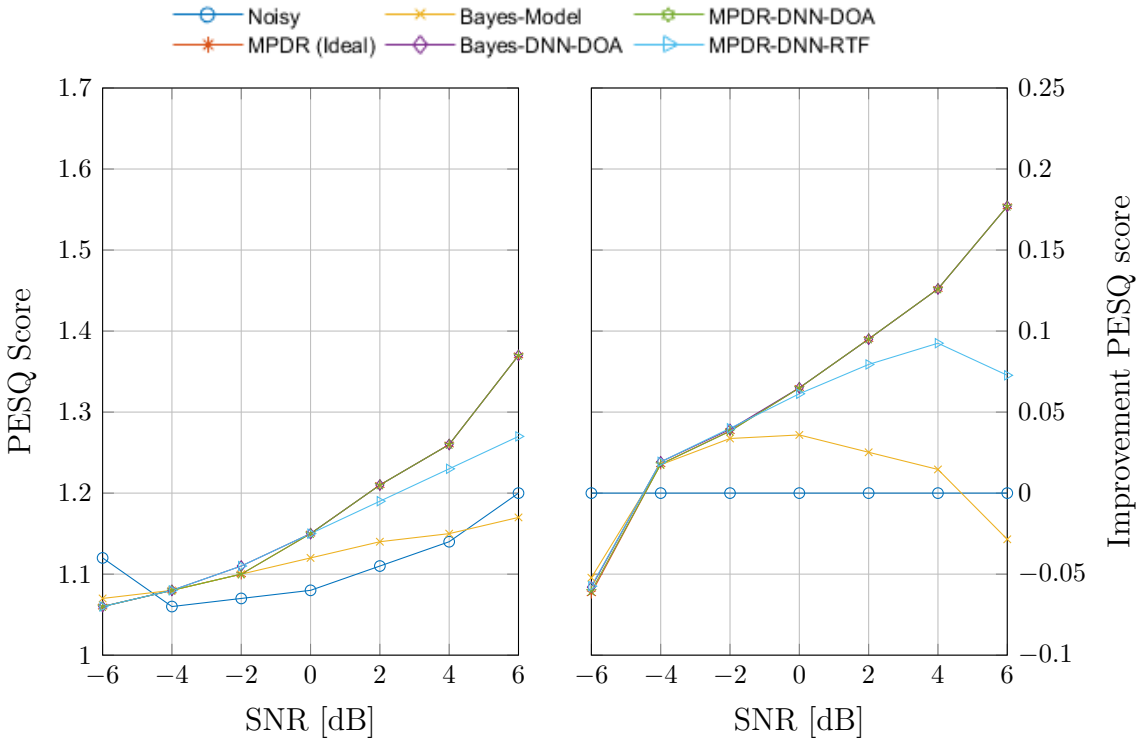
Figure 6.4: Speech intelligibility estimated with ESTOI. The SI for each SNR is averaged over 20 runs.

It is seen that all DNN supported beamformers achieve a higher ESTOI score than the unprocessed noisy signal for all SNRs. In high SNRs (-2 to 6 dB), the Bayes-DNN-DOA and MPDR-DNN-DOA have almost identical ESTOI score to the ideal MPDR beamformer. The similar performance between the Bayes-DNN-DOA and MPDR-DNN-DOA

is expected as the posterior probability in high SNR estimated by the DNN-DOA, most likely will be close to 1 at the true DOA. The MPDR-DNN-RTF beamformer does not obtain a high ESTOI score compared to other DNN supported beamformers. This is possible due to the fact that the DNN-RTF algorithm has to estimate the RTF-vector instead estimating the DOA from a discrete set of directions. The model-based Bayesian beamformer, Bayes-Model, seems to have a poor performance. The poor performance might be due to poor estimates of the posterior probabilities. Furthermore, in order to estimate the model-based posterior probability, the parameters γ has to be tuned. According to [17], γ is a function of SNR, and should be small at low SNR and large at high SNR for good performance. However, since the SNR is unknown by the beamformers in our particular application, γ had to be set to a fixed scalar, that remained identical for all SNRs. This resulted in poor ESTOI scores at high SNRs as seen in Figure 6.4.

6.2.2 Perceptual Evaluation of Speech Quality

Here we compare the PESQ score between the beamformers at different SNRs. PESQ is a standardized method to predict the perceived speech quality from subjects [45] [46, p. 13], and will be used as a black box for evaluation. The inputs of PESQ are the clean speech signal and the degraded speech signal and the PESQ algorithm returns a score between -0.5 and 4.5, where a high score means high predicted speech quality [45]. For this case the degraded signal is the output of the beamformer. The PESQ score of the beamformer output will then be compared to PESQ scores of the noisy signal. The PESQ score as a function of SNR with a target source located at 0 degrees (i.e. frontal) is shown in Figure 6.5.



(a) PESQ score. **(b)** PESQ improvement relative to noisy.

Figure 6.5: PESQ score. The PESQ score for each SNR is averaged over 20 runs.

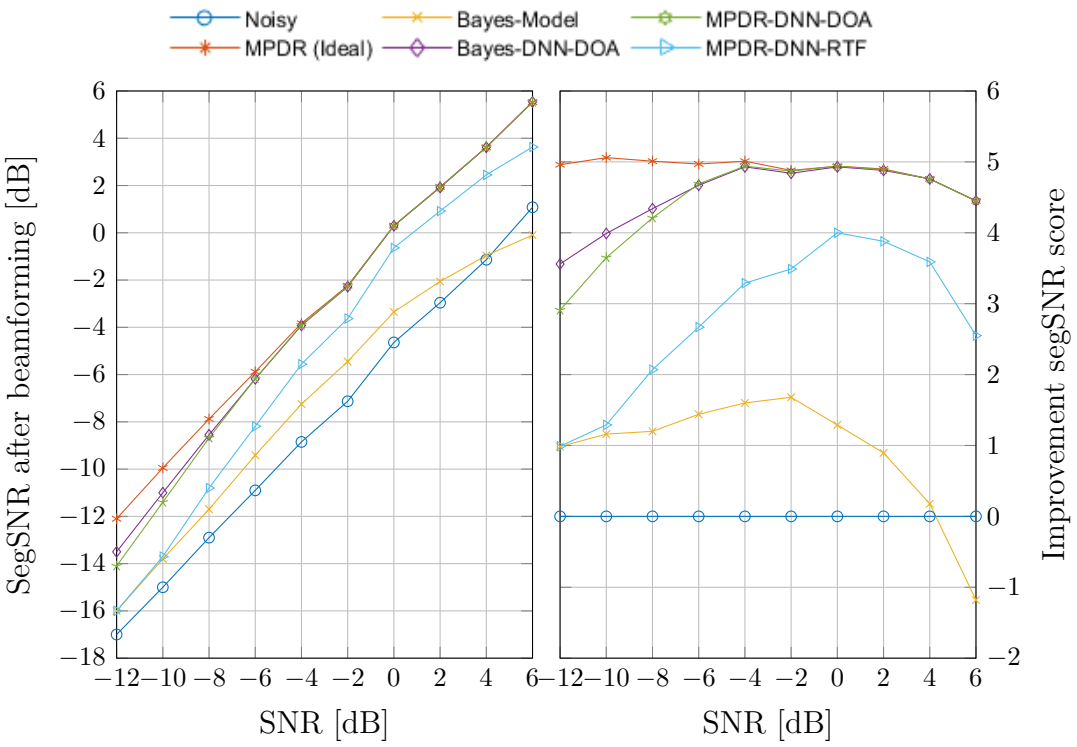
It is seen that the PESQ score for the ideal MPDR, Bayes-DNN-DOA, and MPDR-DNN-DOA are close to being identical for all tested SNR's between -6 db to 6 dB. The MPDR-DNN-RTF and Bayes-Model score a lower PESQ score than the other DNN supported methods until -2 dB, where all methods scores equally. From this evaluation, it is hard to determine which method has the best performance.

6.2.3 Segmental SNR

Segmental SNR (segSNR) is a simple method to evaluate the speech quality and is similar to computing the SNR, except that the signal is segmented into smaller frames of N samples, and the SNR is computed and averaged over all frames [46, p. 9]. The segSNR is given as [46, p. 9]

$$\mathrm{SNR}_{\mathrm{seg}} = \frac{1}{M}\sum_{m=1}^{M} 10\log_{10}\frac{\sum_{n=Nm}^{Nm+N-1} s(n)^2}{\sum_{n=Nm}^{Nm+N-1}\left[s(n)-\hat{s}(n)\right]^2} \tag{6.1}$$

The segSNRs are computed after beamforming and are plotted in Figure 6.6 with the target direction at 0°. The remaining results for −90°, 90°, and 180° are provided in Appendix B.



(a) segSNR score. **(b)** segSNR improvement relative to noisy.

Figure 6.6: Segmental SNR. The segmental SNR for each SNR is averaged over 20 runs.

The Bayes-DNN-DOA and MPDR-DNN-DOA appears to offer similar performance in SNR's above -2 dB. However it is then seen, that below -2 dB Bayes-DNN-DOA returns a higher segSNR score than the MPDR-DNN-DOA. The MDPDR-DNN-DOA and Bayes-Model have degraded performances in segSNR compared the Bayes-DNN-DOA and MPDR-DNN-DOA with Bayes-Model performing the worst.

It can be concluded from the results, that the MPDR-DNN-RTF and Bayes-Model perform the worst among all tested beamformers in terms of ESTOI, PESQ, and segSNR scores. The MPDR-DNN-DOA and Bayes-DNN-DOA share similar performance at high SNR, but it seems that the Bayes-DNN-DOA is slightly more robust at low SNRs.

Chapter 7

Discussion

In this chapter, we will discuss the results found in chapter 6. Moreover, we will examine the possible limitations and issues, that might appear when the DNN-based methods are implemented on a hearing aid in real life i.e. outside simulation.

7.1 DOA Estimation

The DOA estimation performance of the MPDR-DOA and DNN-DOA was compared in an approximately isotropic noise field with babble noise at SNRs of 0 dB, -6 dB, and -12 dB. The results, given in form of confusion matrices, suggest that the DNN-DOA algorithm on average is more accurate at estimating the DOA than the MPDR-DOA. One explanation is, that the DNN is able to recognize features in the observed data, that also appeared during training. Yet in acoustic scenes where the SNR is high, it has been observed that the MPDR-DOA and DNN-DOA approach the same performance. From the confusion matrices and the mean absolute error, we conclude the DNN-DOA is able to outperform the model-based MPDR-DOA in the studied acoustic scenes in low SNRs.

7.2 Beamformer performance

From the beamformer evaluation in section 6.2, it is evident that the MPDR-DNN-DOA and Bayes-DNN-DOA outperform Bayes-Model and MPDR-DNN-DOA in isotropic noise fields with babble noise. The performance of the MPDR-DNN-DOA and Bayes-DNN-DOA are almost identical to the ideal MPDR (i.e. known DOA) in high SNRs, as the posterior probabilities estimated by the DNN-DOA return a probability close to 1 at the true the DOA as we observed during experiments. It moreover appears that the Bayes-DNN-DOA offers slightly higher robustness in low SNR compared to MPDR-DNN-DOA, which might be due to the fact, that the Bayes-DNN-DOA is able to utilize the posterior probability estimated by DNN-DOA to weighting the beamformer coefficients.

Among the proposed DNN supported beamformers (i.e. MPDR-DNN-DOA, Bayes-DNN-DOA, and MPDR-DNN-RTF), the MPDR-DNN-RTF beamformer performs the

worst. In contrast to the other beamformers, the DNN-RTF has to estimate the RTF-vector directly from the noisy CPSD matrices $\mathbf{C}_x$. A potential advantage of this, is that the DNN-RTF is not bounded by selecting a predefined RTF-vector associated with a direction from a discrete set. However, this also means that the DNN-RTF has a higher degree of freedom, which may result in a DNN which may be more difficult to train. The objective functions are also different, meaning that the solution space of DNN-RTF might contain more local minimas and thus be more prone to obtain a poor optimum during training. However, successfully training the DNN-RTF may potentially result in a DNN, that is still able to estimate the RTF-vector, when an RTF-vector from an unseen direction is presented in contrast to the DNN-DOA. A more fair comparison between DNN-DOA-based beamformers and the MPDR-DNN-RTF beamformer would be in environments where the DOA is from a direction unknown to the DNN-DOA.

The Bayes-Model beamformer has the worst performance of all the examined beamformers. The poor performance is at least partly due to poor estimates of the posterior probabilities of the DOA, as the only difference between the Bayes-Model and Bayes-DNN-DOA is their method to estimate the posterior probabilities. Other aspects of Bayes-Model that contribute to the poor performance, are its tuning of the parameter γ. In principle, the γ parameter is a function a SNR. The SNR is, however, unknown for the beamformers in our application, and our simulation experiments suggest that determining a fixed γ that performs good for a large range of SNRs is difficult. In the evaluation, the γ parameter is set to be small (i.e. $\gamma = 1$) as the lowest SNR is -12 dB, which in return gives a poor performance at higher SNR such as 6 dB. Moreover, in principle, γ might be a function of frequency for noise and target signals which are not temporally white. This is due to the fact, that if the target signal is speech, then it is expected that the SNR at e.g. 800 Hz in general is higher than the SNR at 100 Hz, where ambient noise might dominate in the acoustic scene. These difficulties are hard to take into account, if additional prior information is not provided to the posterior probability estimates of Bayes-Model.

7.3 Challenges Faced in Real World Implementation

In the performance evaluation, the noise field is approximately isotropic, but in real acoustic scenarios, the noise field may also be anisotropic. An example of such a situation is an acoustic scene with a target speaker and a single competing speaker. In this case, if the SNR is high such that the target speaker is "louder" than the competing speaker, the DNN will not have any issues in estimating the true DOA or RTF-vector. However, as the SNR decreases until the competing speaker becomes louder, the DNN will estimate the DOA to be towards the competing speaker, and as a consequence, the beamformer will distort the true target speaker. If the DNN should be robust against such cases, the DNN requires additional prior information about the target's position.

Other aspects that most likely occurs in a real world acoustic scene is changes in the direction of the target and changes in the spatial coherence of the noise. In the evaluation, the direction of the target speaker is fixed and the spatial coherence of the noise is identical for all time frames. If these parameters changes over time, the DNN-based method may perform less well. We previously discussed in chapter 5 that the performance of DNN-based method will depend the number of frames the noisy CPSD matrices are formed

over. The trade-off is that the fewer frames that are used to estimate the CPSD matrices, the faster the DNN-based method will be at adjusting to changes at the cost of possible worse steady-state performance.

7.4 Limitation of the HRTF database

The RTF-vectors used to simulate the received signal at the HA microphones are obtained from a hearing aid mounted on a dummy head. We have assumed in the signal model, that the HRTFs are not a function of distance. Unfortunately, this may be a potential problem, when simulating the propagation of sound waves from a distance below 1 meter, where the HRTFs differ substantially with distance [10]. As the DNN is trained on RTF-vectors obtained from HRTFs measured at a distance of 3 m from the head, we expect that the DNN will perform worse in acoustic scenes where the target is at distances below 1 meter. HRTFs measured at a distance below 0.8 m were however not available in the database of HRTFs.

Another issue is that the elevation angle is not included in the training of the DNN, which eventually also will cause a decrease in performance. Furthermore, differences in the shape of the head and torso between person, and the placement of HA behond the ear of the user will certainly also influence, the performance of the DNN-based method. These are some of the hypothesized challenges related to the HRTF the DNN might face, and further experiments are needed for verification.

Chapter 8

Conclusion

We have in this thesis explored the possible of applying a DNN to support acoustic beamformers for a hearing aids. Specifically, the goal of this thesis was first and foremost to seek the answer to the following question:

> How can a DNN be applied to support an acoustic beamformer and can it potentially outperform a model-based acoustic beamformer in terms of speech intelligibility and sound quality in acoustic scenes with low SNR?

To answer this question, we proposed three DNN supported acoustic beamformers, namely an:

- MPDR beamformer supported by a DNN estimating the DOA (MPDR-DNN-DOA).
- MPDR beamformer supported by a DNN estimating the RTF-vector (MPDR-DNN-RTF).
- Bayesian beamformer with the posterior probabilities estimated by a DNN (Bayes-DNN-DOA).

We then compared the performance of the proposed DNN-supported beamformers with a Bayesian beamformer with a model-based estimate of the posterior probabilities of the DOA (Bayes-Model). From the results presented in chapter 6, it is evident that the DNN-supported acoustic beamformers outperform the Bayes-Model in terms of ESTOI, PESQ, and segSNR scores in the studied acoustic scenes. Of the three proposed DNN-supported acoustic beamformers, the MPDR-DNN-DOA and Bayes-DNN-DOA performed similarly with Bayes-DNN-DOA being slightly more robust at low SNRs.

The MPDR-DNN-RTF performed worse than the MPDR-DNN-DOA and Bayes-DNN-DOA, but this is possibly due to the fact that the MPDR-DNN-RTF had to estimate the RTF-vector from a continuous solution set in contrast to MPDR-DNN-DOA and

Bayes-DNN-DOA which selected their solutions from a discrete set. The hypothesized potential of the MPDR-DNN-RTF is that, it might outperform the other DNN-supported beamformers when the DOA is not from a predefined discrete set of directions, which is generally the case in real life.

From the results, we can thus concludes that the proposed DNN-supported beamformers are able to outperform a model-based acoustic beamformer in terms of predicted speech intelligibility and predicted sound quality in low SNR.

Chapter 9

Future Work

We have in this thesis assumed that no additional prior information about the DOA of the target is available. However, in practice, prior information can possible be obtained by making an acoustic scene analysis in order to classify the type of acoustic environment e.g. a cocktail party or car cabin noise. Then based on the classified acoustic scene, it might be possible to apply a prior probability to the direction of the target. Related to this, it might be reasonable to design a DNN, which task is solely to classify the type of acoustic environment based on the spatial coherence found in the noisy CPSD matrix for instance.

It is inevitably required to evaluate the performance of the DNN-supported beamformers in real acoustic scenes for example in reverberant rooms in order to verify that the DNNs are able to generalize to real acoustic environments. Furthermore, the proposed beamformers must be able to be implemented on a hearing aid. It is then necessary to further consider the amount of parameters in the DNN and the execution time as hearing aids have limited resources. In case that, the DNNs seem to perform poorly in real acoustic environments, one possible way to improve the generalization of the DNN, is to include a wider range of RTF-vectors obtained from multiple HRIR measurements. Furthermore, generally performance may be increased by extending the training set to include more representative noise fields and noise types.

Bibliography

[1] G. R. Popelka, B. C. Moore, R. R. Fay, and A. N. Popper, *Hearing Aids*, 1st ed. Springer International Publishing, 2016.

[2] H. Dillon, *Hearing Aids*, 2nd ed. Thieme Medical Pub, 2012.

[3] Beth McCormick, "What's Inside a Hearing Aid," https://www.starkey.com/blog/2014/07/whats-inside-a-hearing-aid, 2014.

[4] Oticon, "Oticon's webpage," https://www.oticon.dk, 2017.

[5] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016, http://www.deeplearningbook.org.

[6] F. Jacobsen and P. M. Juhl, *Fundamentals of General Linear Acoustics*, 1st ed. John Wiley & Sons Ltd, 2013.

[7] F. B. Jensen, W. A. Kuperman, M. B. Porter, and H. Schmidt, *Wave Propagation Theory*. New York, NY: Springer New York, 2011, pp. 65–153.

[8] B. Xie, *Head-Related Transfer Function and Virtual Auditory Display*, 2nd ed. J. Ross Publishing, 2013.

[9] H. Kayser, S. D. Ewert, J. Anemuller, T. Rohdenburg, V. Hohmann, and B. Kollmeier, "Database of multichannel in-ear and behind-the-ear head-related and binaural room impulse responses," *EURASIP Journal on Advances in Signal Processing*, vol. 2009, no. 1, Jul 2009.

[10] A. Kan, C. Jin, and A. van Schaik, "A psychophysical evaluation of near-field head-related transfer functions synthesized using a distance variation function," *The Journal of the Acoustical Society of America*, vol. 125, no. 4, pp. 2233–2242, 2009.

[11] H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE Signal Processing Magazine*, vol. 13, no. 4, pp. 67–94, Jul 1996.

[12] M. Brandstein and D. Ward, *Microphone Arrays*, 1st ed. Springer-Verlag Berlin Heidelberg, 2001.

[13] J. Benesty, J. Chen, and Y. Huang, *Microphone Array Signal Processing*, 1st ed. Springer-Verlag Berlin Heidelberg, 2008.

[14] S. Gannot, E. Vincent, S. Markovich-Golan, and A. Ozerov, "A consolidated perspective on multimicrophone speech enhancement and source separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 4, pp. 692–730, April 2017.

[15] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*, 2nd ed. Prentice Hall, 1999.

[16] Alan V. Oppenheim and George C. Verghese, "Wiener Filtering," https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-011-introduction-to-communication-control-and-signal-processing-spring-2010/readings/MIT6_011S10_chap11.pdf, 2010, downloaded May 22, 2018.

[17] K. L. Bell, Y. Ephraim, and H. L. V. Trees, "A bayesian approach to robust adaptive beamforming," *IEEE Transactions on Signal Processing*, vol. 48, no. 2, pp. 386–398, Feb 2000.

[18] D. H. Johnson and D. E. Dudgeon, *Array Signal Processing: Concepts Techniques*, 1st ed. Prentice Hall, 1993.

[19] H. L. V. Trees, *Optimum Array Processing. Part IV of Detection, Estimation, and Modulation Theory*, 1st ed. John Wiley & Sons Ltd, 2002.

[20] J. Benesty, M. M. Sondhi, and Y. Huang, *Springer Handbook of Speech Processing*, 1st ed. Springer-Verlag Berlin Heidelberg, 2008.

[21] R. C. Hendriks, T. Gerkmann, and J. Jensen, *DFT-Domain Based Single-Microphone Noise Reduction for Speech Enhancement*, 1st ed. Morgan and Claypool, 2013.

[22] Z. H. Tan and B. Lindberg, "Low-complexity variable frame rate analysis for speech recognition and voice activity detection," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 5, pp. 798–807, Oct 2010.

[23] A. Kuklasiński, S. Doclo, S. H. Jensen, and J. Jensen, "Maximum likelihood psd estimation for speech enhancement in reverberation and noise," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 9, pp. 1599–1612, Sept 2016.

[24] C. M. Bishop, *Pattern Recognition And Machine Learning*, 1st ed. Springer, 2006.

[25] J. Heymann, L. Drude, and R. Haeb-Umbach, "A generic neural acoustic beamforming architecture for robust multi-channel speech processing," *Computer Speech & Language*, vol. 46, pp. 374 – 385, 2017.

[26] X. Xiao, S. Watanabe, H. Erdogan, L. Lu, J. Hershey, M. L. Seltzer, G. Chen, Y. Zhang, M. Mandel, and D. Yu, "Deep beamforming networks for multi-channel speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 5745–5749.

[27] B. Li, T. N. Sainath, R. J. Weiss, K. W. Wilson, and M. Bacchiani, "Neural network adaptive beamforming for robust multichannel speech recognition," 2016.

[28] J. Heymann, L. Drude, and R. Haeb-Umbach, "Neural network based spectral mask estimation for acoustic beamforming," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 196–200.

[29] X. Xiao, S. Zhao, D. L. Jones, E. S. Chng, and H. Li, "On time-frequency mask estimation for mvdr beamforming with application in robust speech recognition," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 3246–3250.

[30] R. Takeda and K. Komatani, "Discriminative multiple sound source localization based on deep neural networks using independent location model," in *2016 IEEE Spoken Language Technology Workshop (SLT)*, Dec 2016, pp. 603–609.

[31] J. Heymann, L. Drude, C. Boeddeker, P. Hanebrink, and R. Haeb-Umbach, "Beamnet: End-to-end training of a beamformer-supported multi-channel asr system," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 5325–5329.

[32] Y. Liu, P. Zhang, and T. Hain, "Using neural network front-ends on far field multiple microphones based speech recognition," in *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2014, pp. 5542–5546.

[33] M. Kolbæk, Z. H. Tan, and J. Jensen, "Speech intelligibility potential of general and specialized deep neural network based speech enhancement systems," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 1, pp. 153–167, Jan 2017.

[34] C. Boeddeker, P. Hanebrink, L. Drude, J. Heymann, and R. Haeb-Umbach, "Optimizing neural-network supported acoustic beamforming by algorithmic differentiation," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 171–175.

[35] S. Chakrabarty and E. A. P. Habets, "Broadband doa estimation using convolutional neural networks trained with noise signals," in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, Oct 2017, pp. 136–140.

[36] R. Takeda and K. Komatani, "Sound source localization based on deep neural networks with directional activate function exploiting phase information," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 405–409.

[37] Garofolo and John S, "TIMIT Acoustic-Phonetic Continuous Speech Corpus," https://catalog.ldc.upenn.edu/ldc93s1, 1993.

[38] M. Jalil, F. A. Butt, and A. Malik, "Short-time energy, magnitude, zero crossing rate and autocorrelation measurement for discriminating voiced and unvoiced segments of speech signals," in *2013 The International Conference on Technological Advances in Electrical, Electronics and Computer Engineering (TAEECE)*, May 2013, pp. 208–212.

[39] J. Barker, R. Marxer, E. Vincent, and S. Watanabe, "The third ???chime??? speech separation and recognition challenge: Dataset, task and baselines," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, Dec 2015, pp. 504–511.

[40] R. G. Sæderup, P. Hoang, S. Winther, M. Bøttcher, J. Struijk, S. Schmidt, and J. Østergaard, "Estimation of the second heart sound split using windowed sinusoidal models," *Biomedical Signal Processing and Control*, vol. 44, pp. 229 – 236, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1746809418300831

[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[42] U. H. Kim, T. Mizumoto, T. Ogata, and H. G. Okuno, "Improvement of speaker localization by considering multipath interference of sound wave for binaural robot audition," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2011, pp. 2910–2915.

[43] J. Jensen and C. H. Taal, "An algorithm for predicting the intelligibility of speech masked by modulated noise maskers," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 11, pp. 2009–2022, Nov 2016.

[44] C. H. Taal, R. C. Hendriks, R. Heusdens, and J. Jensen, "A short-time objective intelligibility measure for time-frequency weighted noisy speech," in *2010 IEEE International Conference on Acoustics, Speech and Signal Processing*, March 2010, pp. 4214–4217.

[45] ITU-T, *Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs*, 1st ed. INTERNATIONAL TELECOMMUNICATION UNION, 2001.

[46] K. Kondo, *Speech Quality*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 7–20. [Online]. Available: https://doi.org/10.1007/978-3-642-27506-7_2

[47] Y. H. Hu and J.-N. Hwang, *Handbook of Neural Network Signal Processing*, 1st ed. CRC Press, 2002.

Part I

Appendix

Appendix A

Network Optimization

The network parameters $\mathcal{A}$ are obtained by solving an optimization problem, which, however, tend to be non-convex [5, p. 282] meaning a global minimizer cannot be guaranteed if a minima is found. Additionally the solution space of the objective function might be highly non-linear with many local minimas in the solution space [5, p. 284]. Often, the objective function includes an expectation operator such as the mean squared error (MSE). The issue with the expectation operator is that it requires a statistical model of random variables which is not available. It is, fortunately, possible to approximate the expectation given the data from the dataset.

Iterative solvers

Related to non-convex optimization, is that no closed-form solution exists that can be used to determine the network parameters. Therefore, the common approach is to use an iterative solver to update the network parameters. Let a_n be the current n'th approximation of the optimum solution, η_n is the step size or learning rate, and δ_n is the step direction, then an iterative solver has the form

$$a_{n+1} = a_n + \eta_n \delta_n. \tag{A.1}$$

The learning rate is usually chosen at the beginning of the optimization while δ_n is determined on the fly. One of the most simple and widely used step direction is the negative gradient. As the gradient provides a direction of the steepest ascent, the negative gradient points towards the steepest descent. If δ_n is a realization of a random variable, then the method of steepest decent is referred to as stochastic steepest descent as the estimates of the gradient will fluctuate which will occur if minibatches are used.[1] The learning rate η_n might be set according to the batch size, where a small learning rate should be selected if the batch sizes is small [5, p. 279]. For practical implementation, the back-

[1] Minibatches refer to a smaller subset of the full training set, used to compute the gradient for one iteration.

propagation algorithm is an efficient algorithm to obtain the partial derivatives of the objective function with respect to some parameters in the neural network [5, p. 204].

Cross-entropy as objective function for classification

In this section we will discuss how the network parameters can be obtained through a maximum likelihood approach using the cross-entropy error function for multiclass classification [24, p. 209]. As previously discussed the output of the classification network can be interpreted as posterior probabilities if the output activation function is chosen to be a softmax function which is given as [5, p. 184]

$$f_i(\mathbf{z}) = \frac{\mathrm{e}^{z_i}}{\sum_{j=1}^{I} \mathrm{e}^{z_j}} \quad \text{and} \quad \sum_{i=1}^{I} f_i(\mathbf{z}) = 1, \tag{A.2}$$

where $f_i(\mathbf{z})$ is the output of the softmax activation function of the i'th unit in the layer and I is the number of units in the layer. The outputs of the neural network are estimated posterior probabilities $\hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x, \mathcal{A})$ from a true unknown probability distribution which will be referred to as $P(\theta_i|\boldsymbol{\mathcal{C}}_x)$. For a more convenient notation, let $t_i = P(\theta_i|\boldsymbol{\mathcal{C}}_x)$ and $\mathbf{t} \in \mathbb{R}^I$ will be referred to as the target vector. We define

$$t_i = P(\theta_i|\boldsymbol{\mathcal{C}}_x) = \begin{cases} 1, & \text{if } i = i_{true} \\ 0, & \text{otherwise} \end{cases}, \quad i = 1, 2, ..., I, \tag{A.3}$$

with $I = 72$. To obtain a maximum likelihood estimate of the network parameters, a likelihood function is needed. If the batch size for training consists of 1 example then the likelihood function is $f(\mathbf{t}|\boldsymbol{\mathcal{C}}_x, \mathcal{A})$. Since the estimates of the posterior probabilities of the network is discrete with I possible outcomes, the multinoulli distribution is used to model the likelihood function such that

$$f(\mathbf{t}|\boldsymbol{\mathcal{C}}_x, \mathcal{A}) = \prod_{i=1}^{I} \hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x, \mathcal{A})^{t_i} \tag{A.4}$$

and for batch sizes with N examples, then the likelihood function is given as [24, p. 209]

$$f(\mathbf{t}^{(1)}, \mathbf{t}^{(2)}, ..., \mathbf{t}^{(N)}|\boldsymbol{\mathcal{C}}_x^{(1)}, \boldsymbol{\mathcal{C}}_x^{(2)}, ..., \boldsymbol{\mathcal{C}}_x^{(N)}, \mathcal{A}) = \prod_{n=1}^{N} \prod_{i=1}^{I} \hat{P}(\theta_i|\mathbf{C}_x^{(n)}, \mathcal{A})^{t_i^{(n)}}. \tag{A.5}$$

To obtain the maximum likelihood estimate, the likelihood function is maximized i.e.

$$\mathcal{A}^\star = \arg\max_{\mathcal{A}} f(\mathbf{t}^{(1)}, \mathbf{t}^{(2)}, ..., \mathbf{t}^{(N)}|\boldsymbol{\mathcal{C}}_x^{(1)}, \boldsymbol{\mathcal{C}}_x^{(2)}, ..., \boldsymbol{\mathcal{C}}_x^{(N)}, \mathcal{A}). \tag{A.6}$$

We then use the logarithmic function to obtain the log-likelihood function for mathematical and numerical convenience, and as the logarithm is a monotonically increasing function, the argument does not change and the solution remains identical. Furthermore, we turn

the maximization problem into a minimization problem by minimizing the negative log-likelihood function which leads to

$$\mathcal{A}^\star = \arg\min_{\mathcal{A}} \; -\ln f(\mathbf{t}^{(1)}, \mathbf{t}^{(2)}, ..., \mathbf{t}^{(N)} | \boldsymbol{\mathcal{C}}_x^{(1)}, \boldsymbol{\mathcal{C}}_x^{(2)}, ..., \boldsymbol{\mathcal{C}}_x^{(N)}, \mathcal{A}), \tag{A.7}$$

and

$$-\ln\left(\prod_{n=1}^{N}\prod_{i=1}^{I} \hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x^{(n)}, \mathcal{A})^{t_i^{(n)}}\right) = -\sum_{n=1}^{N}\sum_{i=1}^{I} t_i^{(n)} \ln \hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x^{(n)}, \mathcal{A}), \tag{A.8}$$

which is also called the cross-entropy error function for multiclass classification [24, p. 209]. We then substitute (A.8) into (A.7) which leads to the optimization problem

$$\mathcal{A}^\star = \arg\min_{\mathcal{A}} \; -\sum_{n=1}^{N}\sum_{i=1}^{I} t_i^{(n)} \ln \hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x^{(n)}, \mathcal{A}), \tag{A.9}$$

meaning that the maximum likelihood estimate of the network parameters can be found by minimizing the cross-entropy between the estimated posterior probability and the expected target probability distribution.

An alternative way to interpret the optimization problem, is that the Kullback-Leibler (KL) divergence is minimized. The KL divergence is a method to measure the dissimilarities between two probability distributions, namely an observed $\hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x, \mathcal{A})$ and an expected probability distribution $P(\theta_i|\boldsymbol{\mathcal{C}}_x)$ [5, p. 74]. For discrete probability distributions the KL divergence is given as [47, p. 322]

$$\begin{aligned} D\{P \parallel \hat{P}\} &= \sum_{i=1}^{I} P(\theta_i|\boldsymbol{\mathcal{C}}_x) \ln\left(\frac{P(\theta_i|\boldsymbol{\mathcal{C}}_x)}{\hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x, \mathcal{A})}\right) \\ &= -\sum_{i=1}^{I} P(\theta_i|\boldsymbol{\mathcal{C}}_x) \ln \hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x, \mathcal{A}) + \sum_{i=1}^{I} P(\theta_i|\boldsymbol{\mathcal{C}}_x) \ln P(\theta_i|\boldsymbol{\mathcal{C}}_x) \\ &= H\{P \parallel \hat{P}\} + H\{P\}, \end{aligned} \tag{A.10}$$

where $H\{P \parallel \hat{P}\}$ is the cross-entropy between P and $\hat{P}$ and $H\{P\}$ is the entropy of P. The objective is to minimize the KL divergence in an expected sense such that

$$\mathcal{A}^\star = \arg\min_{\mathcal{A}} \; -\mathbb{E}\left[\sum_{i=1}^{I} P(\theta_i|\boldsymbol{\mathcal{C}}_x) \ln \hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x, \mathcal{A}) - \sum_{i=1}^{I} P(\theta_i|\boldsymbol{\mathcal{C}}_x) \ln P(\theta_i|\boldsymbol{\mathcal{C}}_x)\right], \tag{A.11}$$

and since the last term is not a function of the network parameters $\mathcal{A}$ then this can be omitted in the optimization leading to

$$\mathcal{A}^\star = \arg\min_{\mathcal{A}} \; -\mathbb{E}\left[\sum_{i=1}^{I} P(\theta_i|\boldsymbol{\mathcal{C}}_x) \ln \hat{P}(\theta_i|\boldsymbol{\mathcal{C}}_x, \mathcal{A})\right]. \tag{A.12}$$

Since the expectation cannot be exactly evaluated, it is approximated by the data in the minibathces such that

$$\mathcal{A}^\star = \arg\min_{\mathcal{A}} \; -\sum_{n=1}^{N}\sum_{i=1}^{I} P(\theta_i^{(n)}|\boldsymbol{\mathcal{C}}_x^{(n)}) \ln \hat{P}(\theta_i^{(n)}|\boldsymbol{\mathcal{C}}_x^{(n)}, \mathcal{A}), \tag{A.13}$$

which is equivalent to (A.9).

As mentioned in the beginning of this section, the optimization problem is solved by an iterative solver. Each time the iterative solver is updated, a new minibatch of data is used to compute the gradient, and thus, in practice, one will never obtain an exact maximum likelihood estimate the of network parameters except for when the data in the batches are exactly identical with all previous batches.

MMSE estimator of network parameters

A well-known type of estimator in statistical signal processing is an estimator that minimizes the mean squared error between the true parameter and the estimated parameter and is referred to as the minimum mean squared error (MMSE) estimator. For the regression network, the network parameters $\mathcal{A}$ are updated such that the MSE between the estimated RTF-vector and true RTF-vector is minimized. The optimization problem is given as

$$\mathcal{A}^\star = \arg\min_{\mathcal{A}} \mathbb{E}\left[\left\|\mathbf{d}(\theta) - \hat{\mathbf{d}}(\boldsymbol{\mathcal{C}}_x, \mathcal{A})\right\|_2^2\right]. \tag{A.14}$$

The expectation is then approximated which leads to

$$\mathcal{A}^\star = \arg\min_{\mathcal{A}} \frac{1}{N}\sum_{n=1}^{N}\left\|\mathbf{d}(\theta^{(n)}) - \hat{\mathbf{d}}(\boldsymbol{\mathcal{C}}_x^{(n)}, \mathcal{A})\right\|_2^2. \tag{A.15}$$

where N is the size of the minibatch. Finally, the activation function of the output layer for the regression network is linear i.e. $f_i(\mathbf{z}) = z_i$.

Appendix B

Additional Results

Here we provide the remaining results from the DOA estimation and beamformer performance evaluations.

B.1 DoA estimation

The acoustic scene settings for the DOA estimation comparison is seen in Table B.1.

Noise Field	Noise Type	SNR	Target direction
Isotropic	Babble	0, -6, -12 dB	-175°, -170°, ..., 180°

Table B.1: The configuration of the acoustic scenes for testing the DOA estimation of MPDR-DOA and DNN-DOA.

Here, we only provide the results for the SNRs -6 dB and -12 dB as 0 dB is already shown and discussed in chapter 6. The confusion matrix for SNRs of -6 dB for the MPDR-DOA is shown in Figure B.1 and in Figure B.2 for the DNN-DOA. The MAE as a function of true DOA direction is shown in Figure B.3. It is seen from Figure B.1, that the the MPDR-DOA seems biased towards estimating the DOA to be approximate 100 degrees, 70 degrees, and -80 degrees when noise is isotropic babble noise and the SNR is -6 dB. Comparing the confusion matrices in Figure B.1 and in Figure B.2, it seems to be the case that the DNN-DOA is more robust at estimating the DOA. This is also verified in Figure B.3 where the the MAE on average is lower for the DNN-DOA. The MAE for MPDR-DOA, however, seem to be lower approximately in the range $[75, 150]$ degrees. The lower MAE might although be explained by biased estimates by the MPDR-DOA at this range.

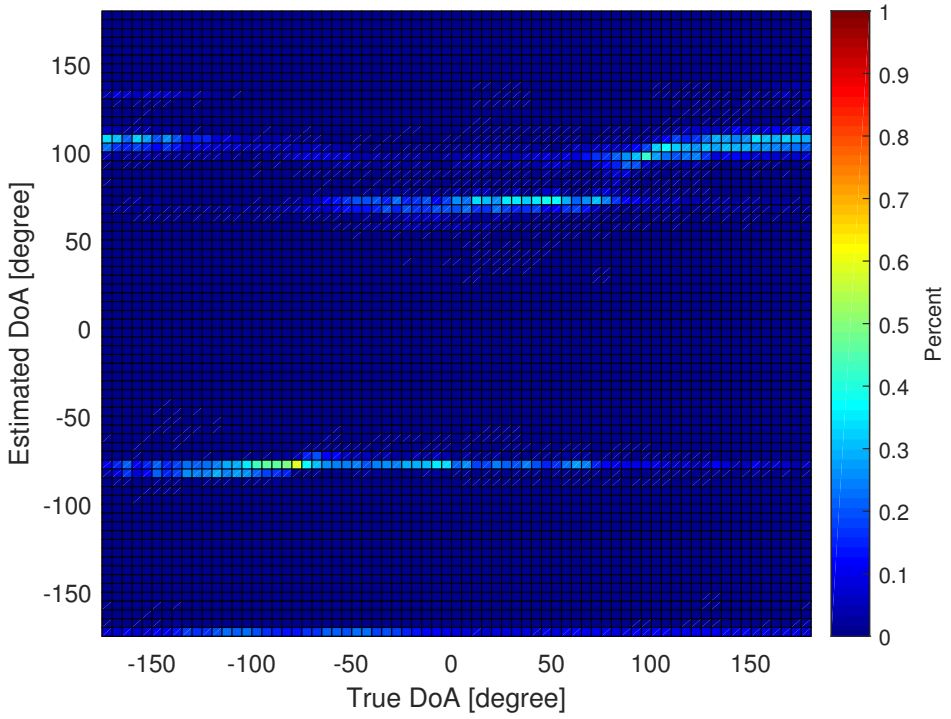


Figure B.1: Confusion matrix for MPDR-DOA at -6 dB SNR. 1000 realization of acoustic scenes with different target talkers, different noise waveforms, are generated and tested for each column.

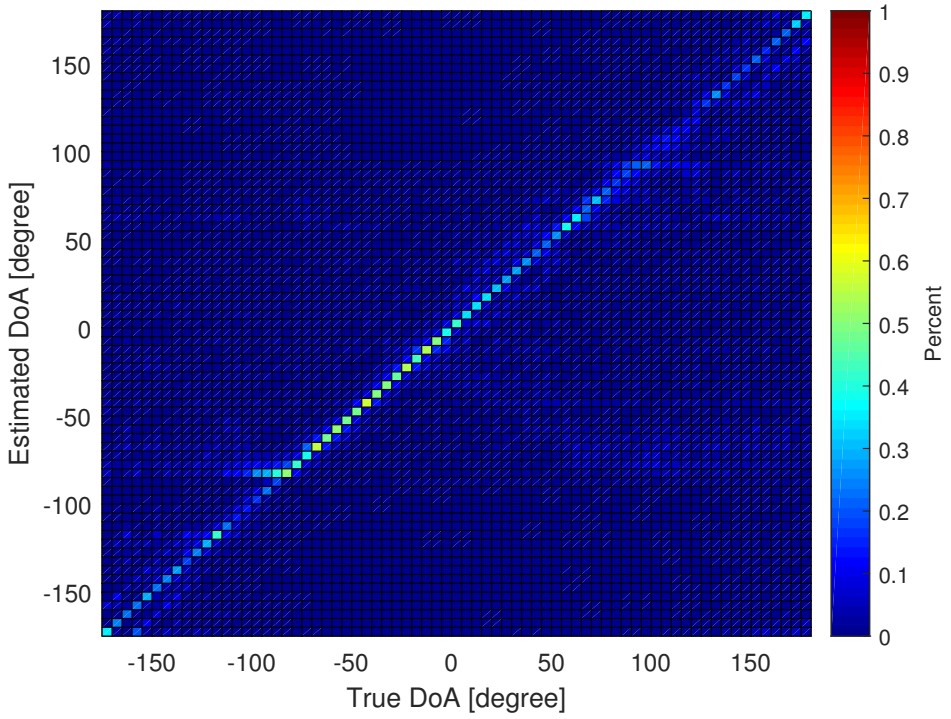


Figure B.2: Confusion matrix for DNN-DOA at -6 dB SNR. 1000 realization of acoustic scenes with different target talkers, different noise waveforms, are generated and tested for each column.

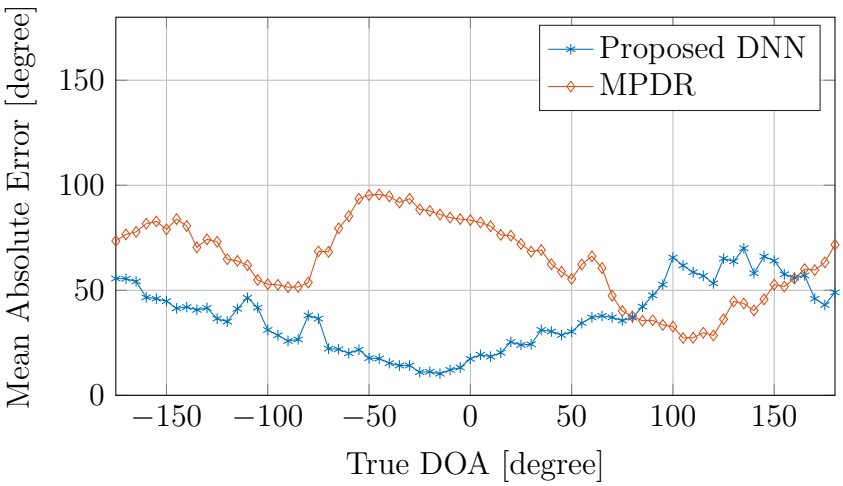


Figure B.3: Mean absolute error for isotropic babble noise with an SNR of -6 dB at the reference microphones.

The confusion matrices for -12 dB SNR is shown in Figure B.4 and Figure B.5. Again, it seems to be the case, that the MPDR-DOA makes biased estimates of the DOA, although at -12 dB SNR isotropic babble noise, the majority of the DOA estimates are at -175 degrees. The DNN-DOA at -12 dB SNR is also biased at estimating the DOA to be approximately in the range $[-90, 90]$. This can also be seen in the MAE as a function of direction in Figure B.6, where the MAE is lowest close to 0 degree for the DNN-DOA and -175 degrees for the MPDR-DOA.

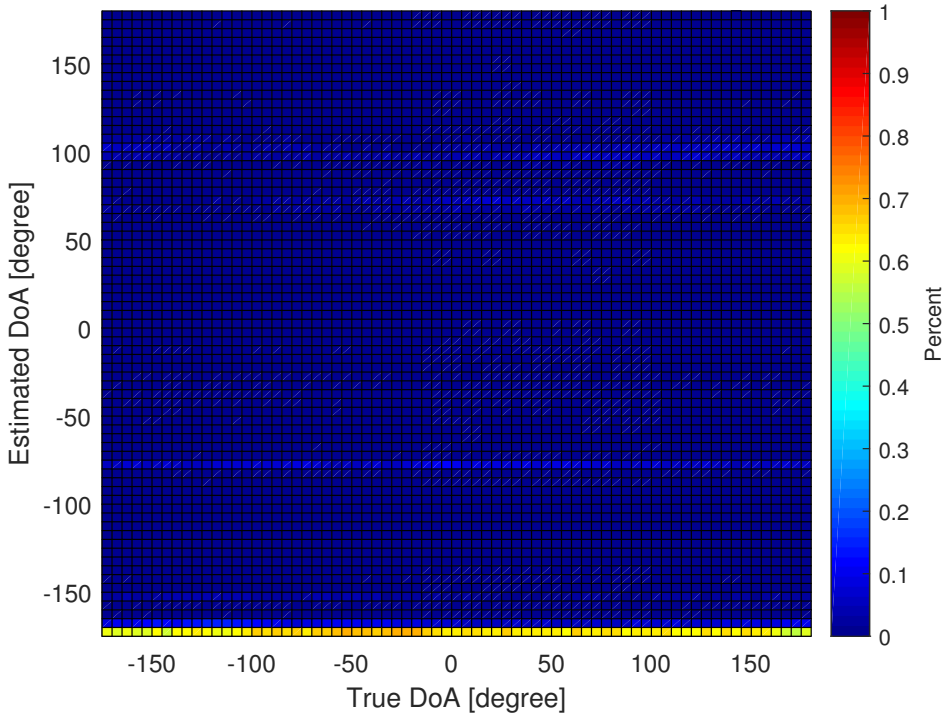


Figure B.4: Confusion matrix for MPDR-DOA at -12 dB SNR. 1000 realization of acoustic scenes with different target talkers, different noise waveforms, are generated and tested for each column.

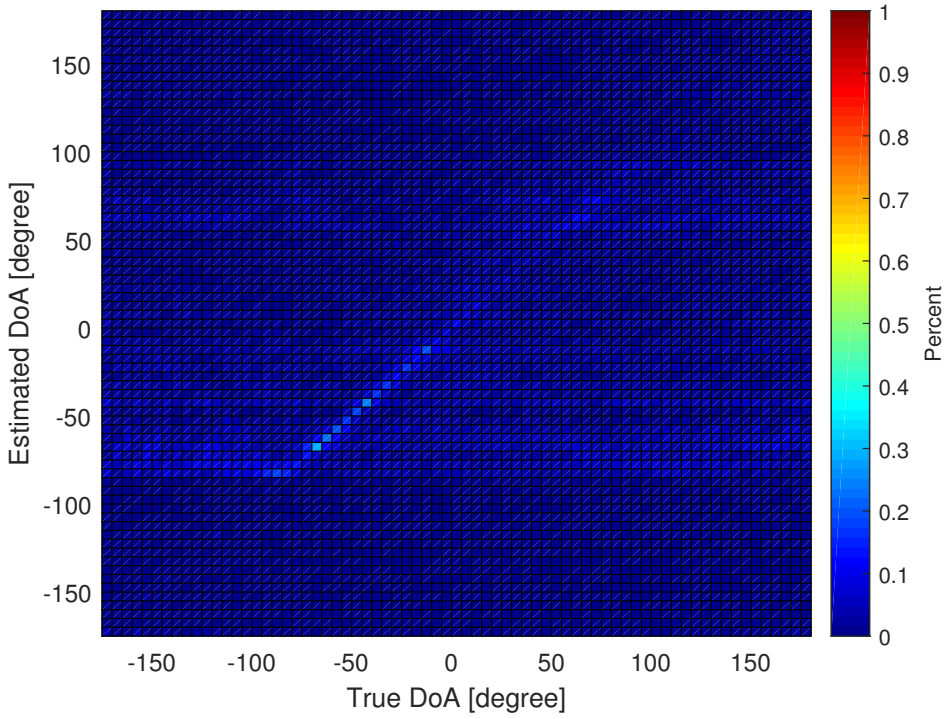


Figure B.5: Confusion matrix for DNN-DOA at -12 dB SNR. 1000 realization of acoustic scenes with different target talkers, different noise waveforms, are generated and tested for each column.

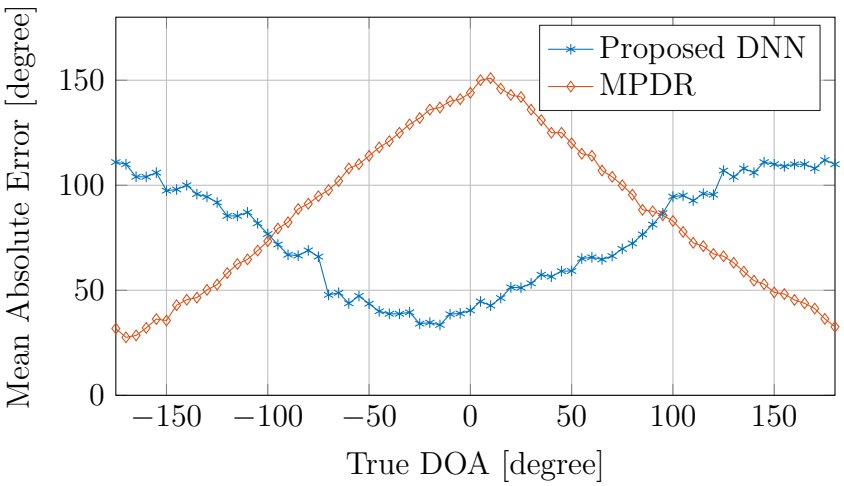


Figure B.6: Mean absolute error for isotropic babble noise with an SNR of -12 dB at the reference microphones.

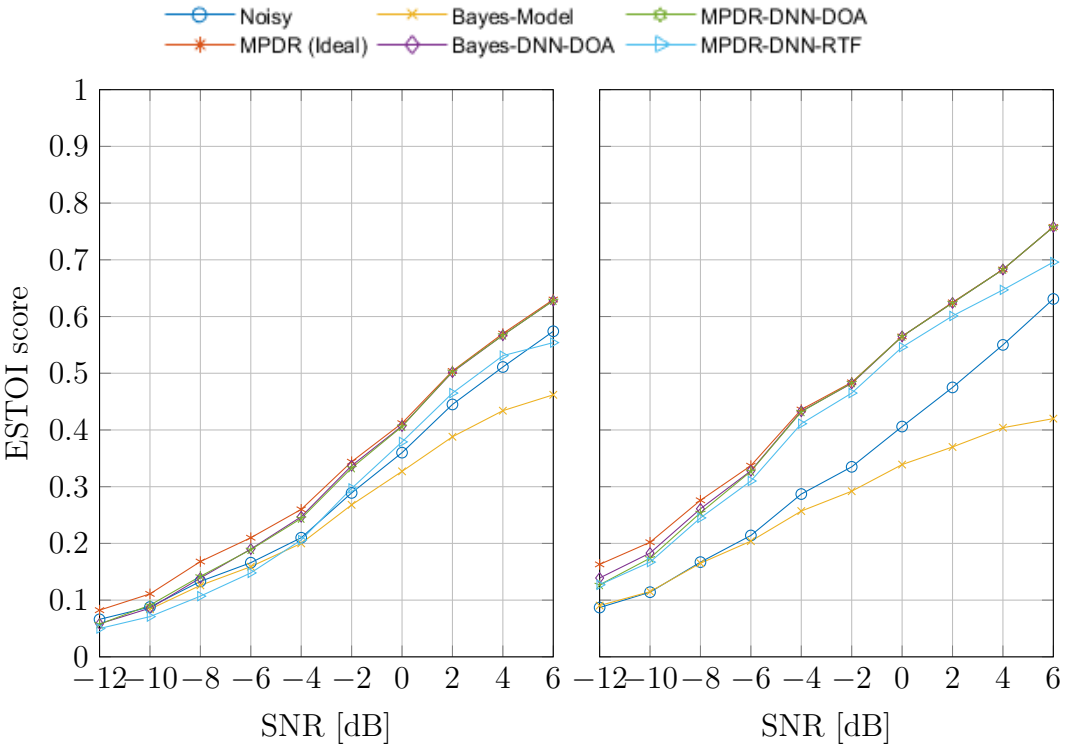
B.2 Beamformer Performance

In this section, the remaining results for the beamformer evaluation are shown for DOAs of -90, 90, and 180 degrees. The setting of the acoustic scenes is shown in Table B.2

Noise Field	Noise Type	SNR	Target direction
Isotropic	Babble	6,4,2,...,-12 dB	-90°, 0°, 90°, 180°

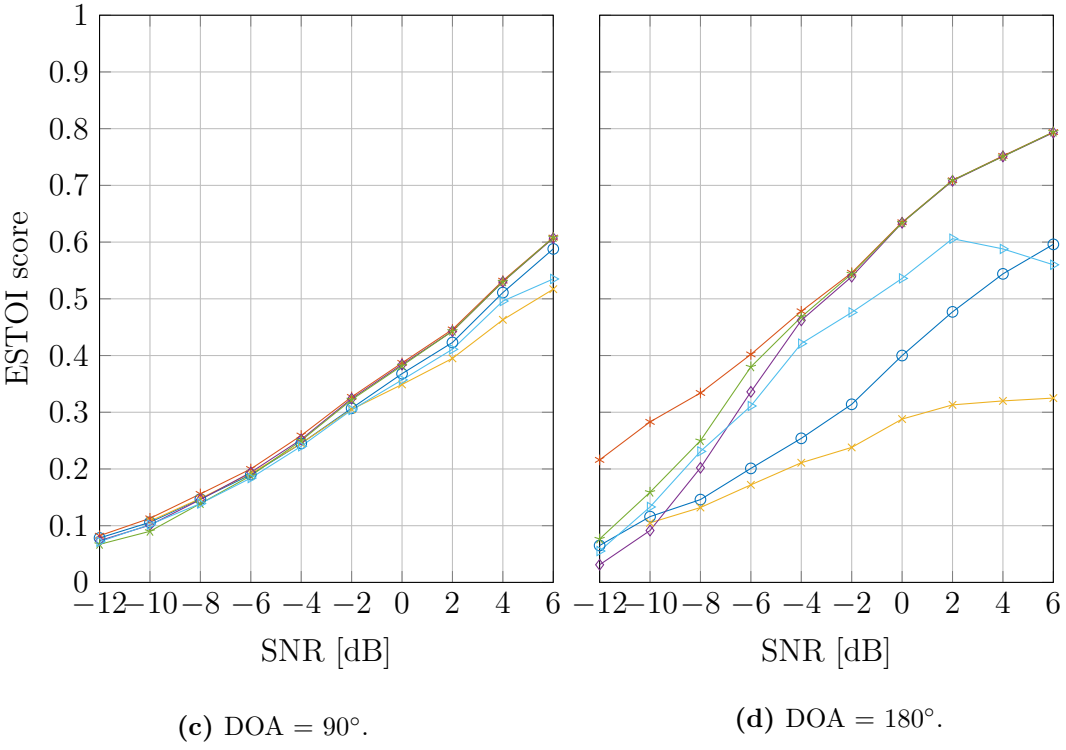
Table B.2: The configuration of the acoustic scenes for evaluating the performance of the proposed beamformers.

The ESTOI scores are shown in Figure B.7 as a function of SNR in isotropic babble noise for -90, 0, 90, and 180 degrees. It is seen that the MPDR-DNN-DOA and Bayes-DNN-DOA have similar performance. It however seems that MPDR-DNN-DOA and Bayes-DNN-DOA have much degraded ESTOI score at 180 degrees in very low SNR. In high SNR, the MPDR-DNN-RTF has the lowest ESTOI score among all DNN-based methods. At very low SNR the MPDR-DNN-RTF perform similar to the MPDR-DNN-DOA and Bayes-DNN-DOA. The Bayes-Model has the worst performance in high SNR, but gives an ESTOI score close to the noisy in very low SNRs.



(a) DOA = −90°.

(b) DOA = 0°.



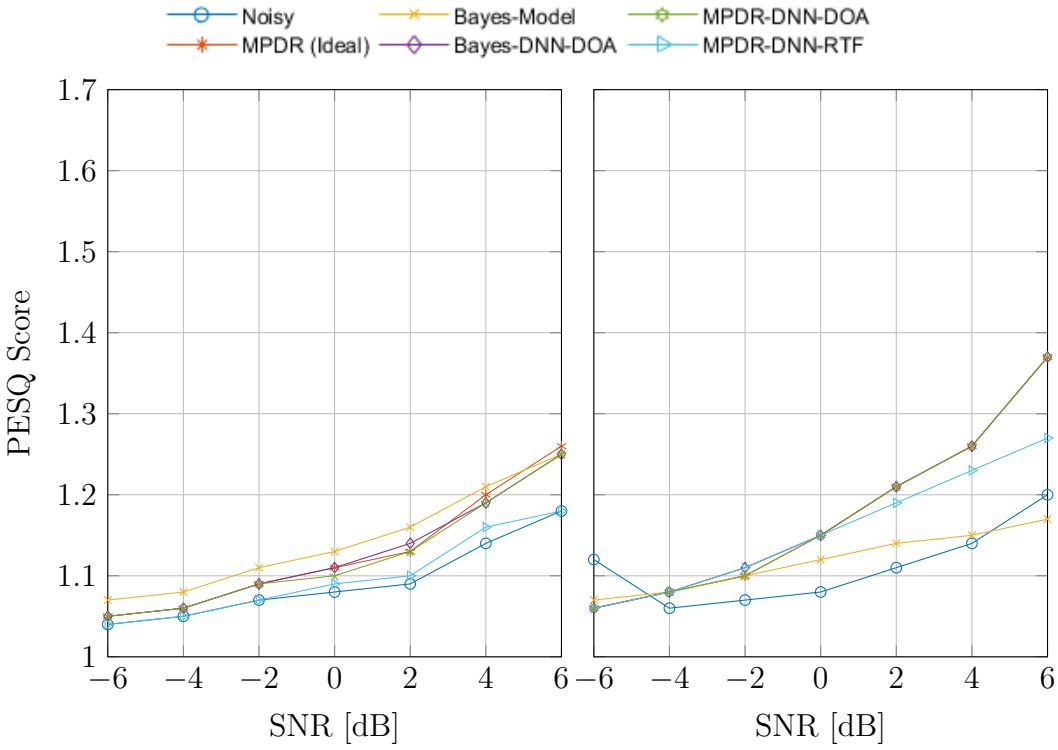
(c) DOA = 90°.

(d) DOA = 180°.

Figure B.7: Speech intelligibility estimated with ESTOI. The SI for each SNR is averaged over 20 runs.

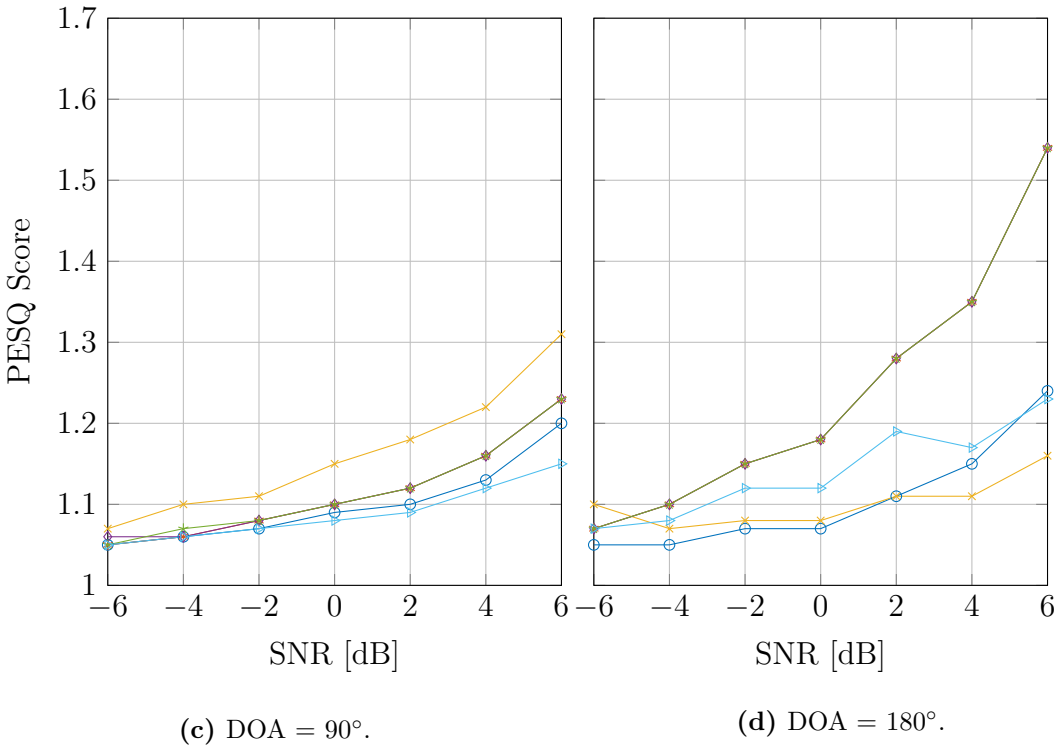
B.2.1 Perceptual Evaluation of Speech Quality

The PESQ score for -90, 0, 90, and 180 degrees are shown in Figure B.8 in isotropic babble noise. Noteworthy, is that the Bayes-Model seem to score a much higher PESQ score, at -90 and 90 degrees, than all examined methods including the ideal MPDR where the DOA is known, but otherwise poorly at 0 and 180 degrees. This exact reason to the high PESQ score is unfortunately unknown. Otherwise, the MPDR-DNN-DOA and Bayes-DNN-DOA again seem to have similar PESQ scores in isotropic babble noise. The MPDR-DNN-RTF performs less well in high SNRs than the MPDR-DNN-DOA and Bayes-DNN-DOA, but approaches the same PESQ score in low SNRs.



(a) DOA = −90°.

(b) DOA = 0°.



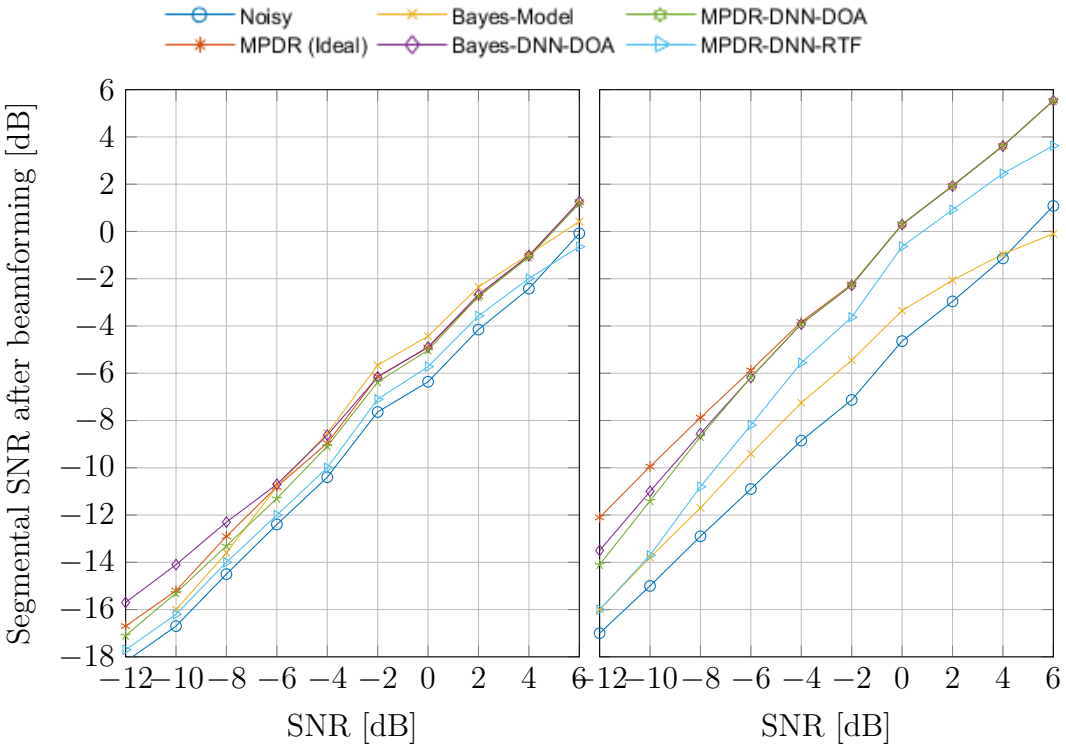
(c) DOA = 90°.

(d) DOA = 180°.

Figure B.8: PESQ score. The PESQ score for each SNR is averaged over 20 runs.

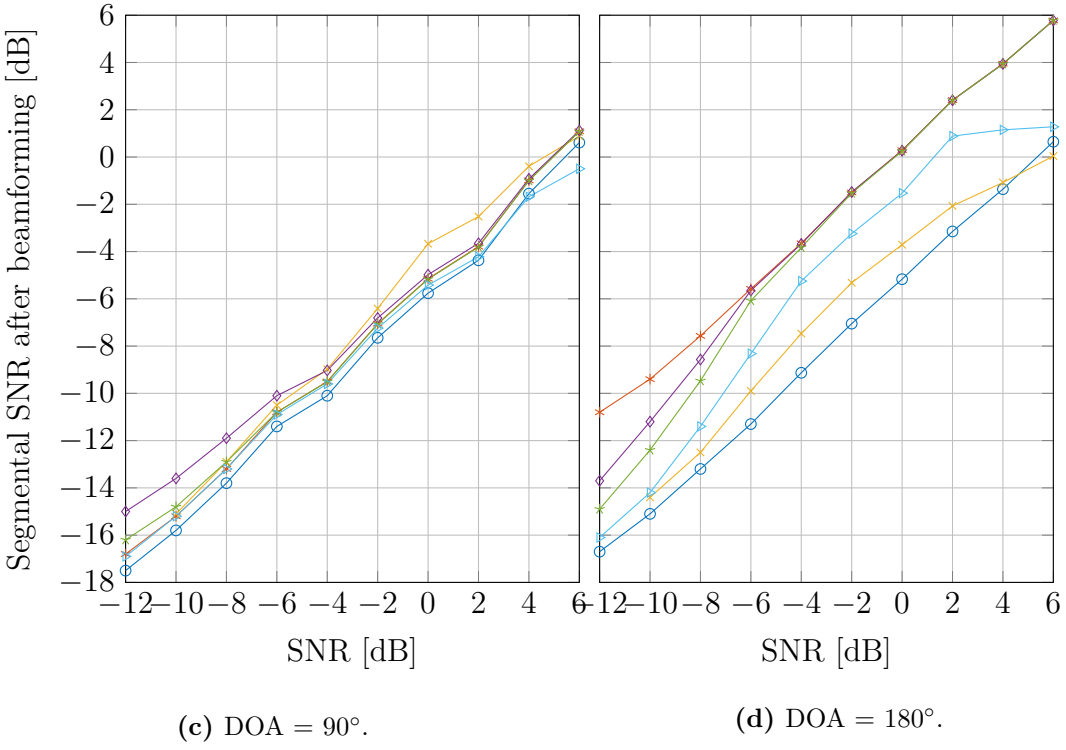
B.2.2 Segmental SNR

The segSNR scores for -90, 0, 90, and 180 degrees as a function of SNR are shown in Figure B.9. It is again, observed that the MPDR-DNN-DOA and Bayes-DNN-DOA have similar performance in high SNR, however, it appears that the Bayes-DNN-DOA has a higher segSNR score than the MPDR-DNN-DOA at low SNRs. It appears that the Bayes-DNN-DOA is able to score a high segSNR score than the ideal MPDR at -90 and 90 degrees at low SNRs in isotropic babble noise, and equivalently Bayes-Model that obtains a higher segSNR at high SNR at -90 and 90 degrees. The exact reason remains unknown but one must consider that minimizing the objective function of the MPDR beamformers, does not necessary translate directly to optimum segSNR performance. The MPDR-DNN-RTF also shows worse performance compared to the other DNN-based approaches.



(a) DOA = −90°.

(b) DOA = 0°.



(c) DOA = 90°.

(d) DOA = 180°.

Figure B.9: Segmental SNR. The segmental SNR for each SNR is averaged over 20 runs.

Appendix C

Input normalization

Here we derive the input normalization that is applied to the noisy CPSD matrices before being fed into the DNNs. The average power over all microphones is

$$P_x = \sum_{m=1}^{M} P_x^{(m)} = \sum_{m=1}^{M} \sum_{n=1}^{N} x_m^2(n), \tag{C.1}$$

where $x_m(n)$ is the time domain noisy signal of the m'th and N is the total number of samples observed. Using Parseval's theorem, the Fourier transformed microphone signal is $X_m(k)$ with $K = N$ frequency bins. We then have

$$P_x = \frac{1}{K} \sum_{k=1}^{K} \sum_{m=1}^{M} |X_q(k)|^2. \tag{C.2}$$

Since we are processing the signal frame by frame, the average received power over all frames is

$$\bar{P}_x = \frac{1}{L} \sum_{l=1}^{L} P_x(l) = \frac{1}{L} \sum_{l=1}^{L} \sum_{m=1}^{M} \left(\frac{1}{K} \sum_{k=1}^{K} |X_m(k,l)|^2 \right) = \frac{1}{LK} \sum_{k=1}^{K} \sum_{l=1}^{L} \sum_{m=1}^{M} |X_m(k,l)|^2. \tag{C.3}$$

We recall that the CPSD matrices are estimated by

$$\hat{\mathbf{C}}_x(k,l) = \frac{1}{L} \sum_{l=1}^{L} \mathbf{X}(k,l)\mathbf{X}^H(k,l), \tag{C.4}$$

thus the diagonal elements of the CPSD matrices are

$$\hat{C}_x^{(m,m)}(k,L) = \frac{1}{L} \sum_{l=1}^{L} |X_m(k,l)|^2, \tag{C.5}$$

and computing the trace of the CPSD matrices reveals

$$\mathrm{tr}\left(\hat{\mathbf{C}}_x(k,L)\right) = \frac{1}{L} \sum_{l=1}^{L} \sum_{m=1}^{M} |X_m(k,l)|^2. \tag{C.6}$$

Taking the average trace across all frequency bins gives

$$\frac{1}{K}\sum_{k=1}^{K}\mathrm{tr}\left(\hat{\mathbf{C}}_x(k,L)\right)=\frac{1}{LK}\sum_{k=1}^{K}\sum_{l=1}^{L}\sum_{m=1}^{M}|X_m(k,l)|^2, \tag{C.7}$$

thereby showing that the average power received at all microphones is

$$P_x(L)=\frac{1}{K}\sum_{k=1}^{K}\mathrm{tr}\left(\hat{\mathbf{C}}_x(k,L),\right) \tag{C.8}$$

and the normalized CPSD matrices are therefore,

$$\hat{\mathbf{C}}_x(k,L)\leftarrow\frac{\hat{\mathbf{C}}_x(k,L)}{\frac{1}{K}\sum\limits_{k=1}^{K}\mathrm{tr}\left(\hat{\mathbf{C}}_x(k,L)\right)}. \tag{C.9}$$

SPEECH ENHANCEMENT WITH DNN SUPPORTED ACOUSTIC BEAMFORMING

Poul Hoang

Modern hearing aids often have more than one microphone available for each device. It has been shown that substantial gains in speech intelligibility can to obtained by applying multichannel signal processing methods (e.g. beamformers) to noisy observations in noisy environments such as cocktail parties or restaurant-like environments. Model-based signal processing methods might, however, perform less well in acoustic environments where the SNR is low as the unknown parameters needed for the beamformers are harder to estimate. The motivation behind the work presented in this thesis, is thus to explore the possibility of applying a deep neural network (DNN) to support an acoustic beamformer as an alternative to the model-based methods. The DNN will in this thesis specifically estimate the direction-of-arrival (DOA) and the relative transfer function (RTF) vector needed for the examined beamformers.

We have proposed three types of DNN supported beamformers in this thesis: 1) A minimum power distortionless response (MPDR) beamformer supported by a DNN for DOA estimation, 2) an MPDR beamformer supported by a DNN estimating RTF-vectors, and 3) a Bayesian beamformer where the posterior probabilities are estimated by a DNN. The experimental results show that the DNN-supported beamformers are able to outperform a model-based Bayesian beamformer in acoustic scenes with isotropic babble noise in terms of ESTOI, PESQ, and segSNR scores.

Aalborg University
Department of Electronic Systems
Fredrik Bajers Vej 7
DK-9220 Aalborg Øst