Graz University of Technology

Master's Thesis

Circular Differential Microphone Arrays

conducted at the
Signal Processing and Speech Communications Laboratory
Graz University of Technology, Austria

by
Thomas Clemens Pichler, 0731144

Supervisor:
Dipl.-Ing. Dr. techn. Martin Hagmüller

Graz, June 20, 2016

Affidavit

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

date	(signature)

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Arbeit selbstständig verfasst, andere als die angegebenen Quellen/Hilfsmittel nicht benutzt, und die den benutzten Quellen wörtlich und inhaltlich entnommenen Stellen als solche kenntlich gemacht habe. Das in TUGRAZonline hochgeladene Textdokument ist mit der vorliegenden Masterarbeit identisch.

Graz, am	(Unterschrift)

Acknowledgement

I would like to express my deepest gratitude to my supervisor, Martin Hagmüller for his guidance and patience throughout the whole thesis and many previous projects. Without his knowledge and support this thesis would not have been possible.

My sincere thanks go to Erik Leitinger, his help and advice about localisation, tracking and array processing was greatly appreciated.

Further I want to thank Hannes Pessentheiner, his knowledge about beamforming was very helpful in the course of this work.

I also want to thank Andreas Läßer for his help in building the microphone arrays and doing the recordings.

Last but not least I want to thank all the members of the SPSC lab, working with all of you was a a very pleasant and unique experience.

Abstract

The use of multichannel speech enhancement systems offers some significant advantages over single channel solutions. Speakers can be localised, tracked and unwanted noise can be spatially supressed to improve perceived audio quality in recordings or for telephone conference particpants. Differential microphone arrays have gained a lot of attention in the last years because of their ability to keep their directional response pattern constant over a wide frequency range and their small sensor spacings.
Focusing on circular geometries this thesis will explore algorithms to localise speakers and electronically steer a beamformer to improve the quality of the speech signal. The first part of the thesis is focused on the design of data independent time-invariant beamformers for circular geometries. Different design algorithms will be shown and their advantages and drawbacks will be discussed.
In the second part the possibilities of sound source localisation with small aperture arrays will be explored.
To test the proposed algorithms for beamforming and localising, real world measurements were done in a reverberant environment. The performance of the proposed algorithms is evaluated and errors of the localisation algorithms and noise suppression ability of the beamformers are measured.

Kurzfassung

Die Verwendung von Mehrkanalsystemen zur Verbesserung der Sprachverständlichkeit bietet im Vergleich zu einkanaligen Lösungen einige Vorteile. Sprecher können lokalisiert werden und unerwünschte Nebengeräsche können mittels Beamforming eliminiert werden um die Sprachverständlichkeit zu erhöhen. In diesem Bereich haben sich Differentielle Mikrofon Arrays in den letzten Jahren sehr hervorgetan. Die kleinen Abstände der Mikrofone und die über große Bereiche konstante Richtcharakteristik der differentiellen Arrays machen diese sehr interessant für die Verwendung in kompakten Geräten wie zum Beispiel Tischsprechstellen.
In der vorliegenden Arbeit werden ausschließlich kreisförmige array Anordnungen besprochen. Ausgehend von diesen wird das Design von frequenz- und zeit-invarianten Beamformern besprochen. Verschiedene Lösungsmöglichkeiten für diese Beamformer werden gezeigt und ihre Vor- und Nachteile werden besprochen.
Der zweite Teil der Arbeit beschäftigt sich mit der Lokalisation von aktiven Schallquellen. Hier werden Algorithmen auf ihre Funktionalität mit Arrays mit sehr kleinen Aperturen untersucht. Um die besprochenen Algorithmen zu validieren wurden Aufnahmen in einem realistischen, reflexionsbehaftetem Raum durchgeführt. Die Lokalisationsfehler und die Störgeräuschunterdrückung der ausgewählten Algorithmen werden mit den aufgenommenen Daten evaluiert.

Contents

1 Introduction

1.1 Motivation

In this thesis the use of circular microphone arrays as frontend for speech enhancement systems will be studied. Differential microphone arrays (DMA) only need a very small amount of microphones and can be built with small microphone apertures compared to conventional microphone arrays but still keep a constant beampattern over a large frequency range.
Because of these advantages DMAs can be used in compact table mounted devices for telephone conferences like in Fig. 1.1. It demonstrates a situation where people are positioned around a table and are talking. The enhancement system on the table should now be able to identify a speaking person of interest and focus the microphone beam onto that person while supressing occuring noise.
That way noise in the room can be supressed so that only the signal from the person of interest is recorded or sent to telephone conference participants.

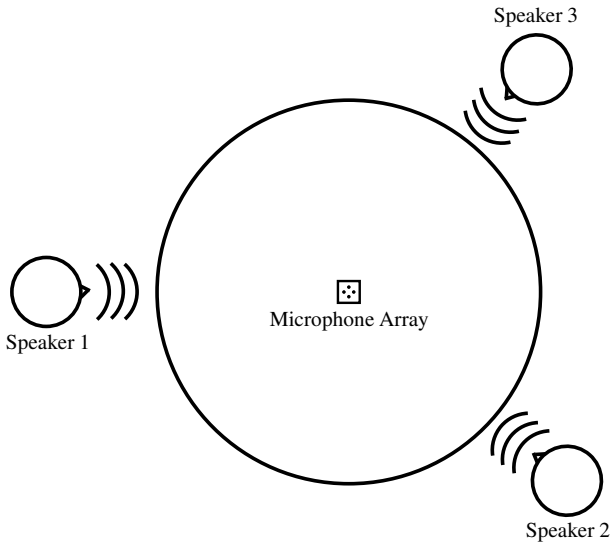


Figure 1.1: Conference situation

1.2 System Overview

To be able to achieve the requirements set up in Section 1.1 the system in Fig. 1.2 is proposed. The noisy signal from the room is captured by a circular microphone array that is situated somewhere in between the speakers and the noise. The signals from the microphone array are filtered by a beamformer to get only the desired signal at the output of the system. To get the direction of the main speaker and steer the mainlobe into that direction a sound source localiser (SSL) is needed. A voice activity detector decides what frames should be used for the SSL. This is needed because even for continous speech the proportion of frames that can reliably be used for SSL is about 30-50%. [1]

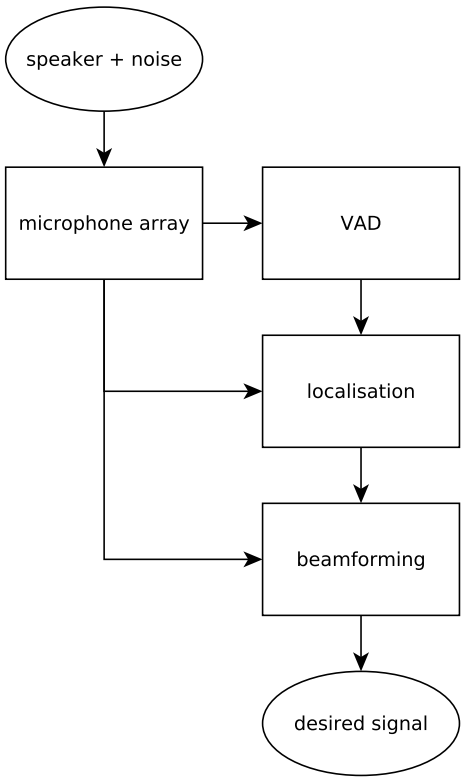


Figure 1.2: Speech enhancement system overview

1.3 Outline

In the following Section 2 the design of fixed beamformers will be shown. Various algorithms will be evaluated using MATLAB [2] and their interesting properties will be discussed. Five promising array geometries are chosen and evaluated further in detail for all algorithms. The third section will show a simple way to improve the performance of the beamformer and mitigate possible mechanical imperfections of the arrays using an adaptive algorithm. Section 4 shows various possibilities of SSL and give an easy but still effective way for the VAD. Chapter 5 will give an overview over the conducted recordings and document the parameters used. The results of those measurments can be seen in chapter 6. There the performance of the beamformers is evaluated in terms of noise energy reduction in an reverberant environment. The SSL is tested against the known ground truth and the localisation error is computed.
At the end of the thesis some conclusive thoughts and an outlook to further improvements will be given.

2 Circular Differential Microphone Arrays

In this chapter an overview regarding various design methods for fixed CDMA beamformers is given. Many design algorithms with closed form solutions can be found in [3].
Since all of the algorithms there are based on a two-dimensional signal model, at the end of the chapter the design method will be extended to a three-dimensional model and a solution to this problem will be presented.

2.1 CDMA Signal Model and Problem Formulation

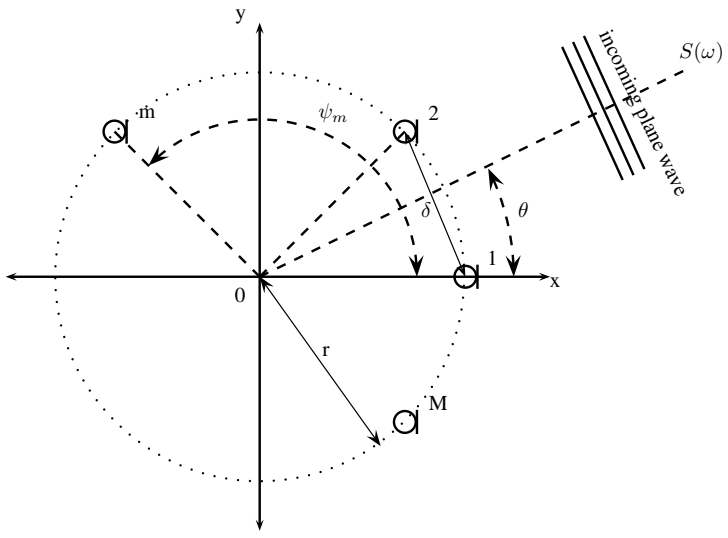


Figure 2.1: Signal model

In Fig. 2.1 the signal model used for the beamformer design can be seen. Since the source signal is assumed to be in the farfield of the array, because of the small array apertures, the incoming signal will be a plane wave that impinges on a uniform circular microphone array (UCA). [3]
The array has radius r and consists of M omnidirectional microphones at angles ψ_m. Because we assume the speaker is in the farfield the only relevant information from the speaker is the direction which is denoted by θ. All angles are measured anti-clockwise from the x-axis and the

center of the coordinate system is in the center of the microphone array.
The time delay τ_m between a microphone m and the origin of the array is now given by [3]

$$\tau_m = \frac{r}{c}\cos(\theta - \psi_m), \quad m = 1, 2, \cdots, M \tag{2.1}$$

with the angles of microphones ψ_m being

$$\psi_m = \frac{2\pi(m-1)}{M} \tag{2.2}$$

and the speed of sound i.e. $c = 343\,\mathrm{m/s}$.
The resulting steering vector in the frequency domain that describes the array geometry is then described as

$$\mathbf{d}(\omega, \theta) = \begin{bmatrix} \mathrm{e}^{j\omega\tau_1} & \cdots & \mathrm{e}^{j\omega\tau_M} \end{bmatrix}^T \tag{2.3}$$

$$\mathbf{d}(\omega, \theta) = \begin{bmatrix} \mathrm{e}^{j\omega rc^{-1}\cos(\theta-\psi_1)} & \cdots & \mathrm{e}^{j\omega rc^{-1}\cos(\theta-\psi_M)} \end{bmatrix}^T$$

where ^{T} denotes the matrix transposition and ω is the angular frequency ω=2πf.

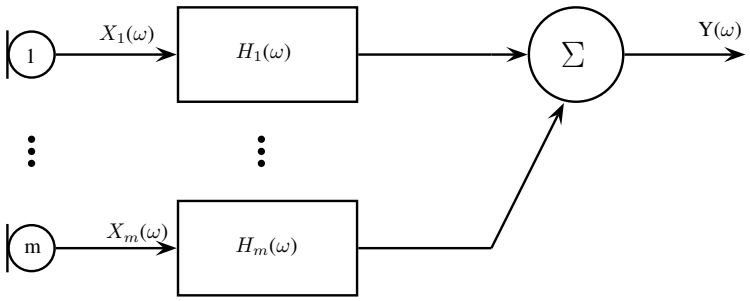


Figure 2.2: Signal model with filters

In Fig. 2.2 the signal model for the fixed beamformer can be seen. With the input signal $S(\omega)$ and additive noise $V_m(\omega)$ at each microphone the Mth input signal in the frequency domain can be described as

$$X_m(\omega, \theta) = \mathrm{e}^{j\omega rc^{-1}\cos(\theta-\psi_m)}S(\omega) + V_m(\omega), \quad m = 1, 2, \cdots M \tag{2.4}$$

and simplified to vector notation as

$$\begin{aligned} \mathbf{x}(\omega, \theta) &= \begin{bmatrix} X_1(\omega, \theta) & X_2(\omega, \theta) & \cdots & X_M(\omega, \theta) \end{bmatrix}^T \\ &= \mathbf{d}(\omega, \theta)S(\omega) + \mathbf{v}(\omega) \end{aligned} \tag{2.5}$$

where the noise-vector is

$$\mathbf{v}(\omega) = \begin{bmatrix} V_1(\omega) & V_2(\omega) & \cdots & V_M(\omega) \end{bmatrix}^T \tag{2.6}$$

The input signal $X_m(\omega)$ of every microphone m is filtered with the frequency weights $H_m(\omega)$

and then summed up to the output signal $Y(\omega)$. So the beamformer output results as

$$\begin{aligned} Y(\omega,\theta) &= \sum_{m=1}^{M} H_m(\omega) X_m(\omega,\theta) \\ &= \mathbf{h}^T(\omega)\mathbf{x}(\omega,\theta) \\ &= \mathbf{h}^T(\omega)\mathbf{d}(\omega,\theta)S(\omega) + \mathbf{h}^T(\omega)\mathbf{v}(\omega) \end{aligned} \tag{2.7}$$

with the filter transfer functions

$$\mathbf{h}(\omega,\theta_s) = \begin{bmatrix} H_1(\omega,\theta_s) & H_2(\omega,\theta_s) & \cdots & H_M(\omega,\theta_s) \end{bmatrix}^T \tag{2.8}$$

As mentioned in [3], the goal is now to design an undistorted beam to the steering direction θ_s and set a number of nulls to various directions. This differs from the design of filter-and-sum or additive beamformers where the filters are desined to steer the mainlobe of the beamformer to a target direction.
The problem for linear differential microphone arrays (LDMA) is that the optimal direction of the target is at the end-fire-direction of the array. This means that only physical steering of the whole array is viable when the beampattern of the beamformer should be conserved.
CDMAs are more flexible in this regard since the beam that is designed for one direction θ_s can easily be transformed to another direction by simple permutation of the transfer functions H_m. So it is possible to design a beampattern for one direction and steer the mainlobe to M directions while conserving the desigend beampattern.
The objective is now to get beamformers with patterns that are very close to beampatterns of ideal DMAs (Section 2.2) for the target direction. For this we select a certain number of fundamental constraints from a well-defined beampattern of a DMA to design $\mathbf{h}(\omega,\theta_s)$ for a number of microphones M [3].

2.2 DMA patterns

In this chapter beampatterns of ideal frequency independent differential microphone arrays are presented. The beampatterns of these DMAs are often used in the design of circular differential arrays. (Section 2.6, Section 2.8) The definition of the frequency independent beampattern is given in [3] as

$$\mathcal{B}_N(\theta-\theta_s) = \sum_{n=0}^{N} a_{N,n}\cos^n(\theta-\theta_s) \tag{2.9}$$

Depending on the values of the real coefficients $a_{N,n}, n = 0, 1, \cdots, N$ different directivity beampatterns can be designed for the Nth-order DMA.
In Fig. 2.3 three cardioids of different orders can be seen. To plot all the beampatterns a modified version of [4] was used. Since in the direction $\theta = \theta_s$ the pattern should be 1 we always choose the first coefficient to be

$$a_{N,0} = 1 - \sum_{n=0}^{N} a_{N,n} \tag{2.10}$$

so the three beampatterns in Fig. 2.3 are given as

$$\begin{aligned}
\mathcal{B}_1(\theta-\theta_s) &= (1-a_{1,1}) + a_{1,1}\cos(\theta-\theta_s) \\
\mathcal{B}_2(\theta-\theta_s) &= (1-a_{2,1}-a_{2,2}) + a_{2,1}\cos(\theta-\theta_s) + a_{2,2}\cos^2(\theta-\theta_s) \\
\mathcal{B}_3(\theta-\theta_s) &= (1-a_{3,1}-a_{3,2}-a_{3,3}) + a_{3,1}\cos(\theta-\theta_s) + a_{3,2}\cos^2(\theta-\theta_s) + a_{3,3}\cos^3(\theta-\theta_s)
\end{aligned} \tag{2.11}$$

where the real valued coefficients for first order are

$$a_{1,1} = \frac{1}{2} \tag{2.12}$$

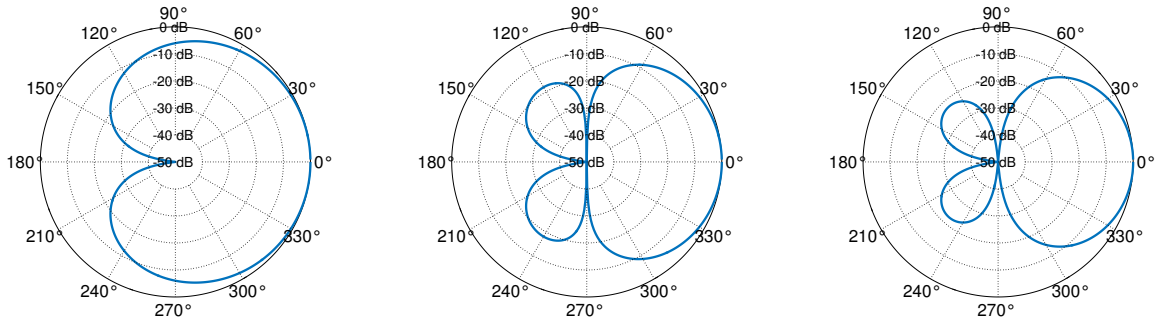
for second order

$$a_{2,1} = \frac{1}{2} \quad a_{2,2} = \frac{1}{2} \tag{2.13}$$

and for third order

$$a_{3,1} = 0 \quad a_{2,2} = \frac{1}{2} \quad a_{2,2} = \frac{1}{2} \tag{2.14}$$

In this thesis mostly first and second order designs will be considered. The reason for this is that high order designs are very difficult to implement since microphones with low inherent noise and low tolerances would be needed. Further the relative gain in directivity is small as the order of microphones increases. [5]



(a) First order cardioid $\mathcal{B}_1$ (b) Second order cardioid $\mathcal{B}_2$ (c) Third order cardioid $\mathcal{B}_3$

Figure 2.3: Ideal frequency independent DMA patterns for a steering angle $\theta_s = 0°$

2.3 Quality measures

Here the three main performance measures that are used to evaluate and compare different beamformer designs will be described.

Beam Pattern

The beampattern describes the spatial sensitivity of the beamformer. It evaluates the sensitivity of the array to a plane wave impinging on the UCA from direction θ.

It is defined as [3]

$$\begin{aligned}\mathcal{B}[\mathbf{h}(\omega),\theta] &= \mathbf{h}^H(\omega)\mathbf{d}(\omega,\theta) \\ &= \sum_{m-1}^{M} H_m^*(\omega)\mathrm{e}^{(j\omega r c^{-1}\cos(\theta-\psi_m))}\end{aligned} \tag{2.15}$$

Where * denotes a complex conjugation. The beampattern is evaluated for one frequency but should be constant for all frequencies.

White Noise Gain

The white noise gain describes the change of the SNR from the beamformer input to the output for the desired speaker direction θ_s and is evaluated as

$$\mathcal{G}_{\mathrm{wn}}[\mathbf{h}(\omega)] = \frac{|\mathbf{h}^H(\omega)\mathbf{d}(\omega)|^2}{\mathbf{h}^H(\omega)\mathbf{h}(\omega)} \tag{2.16}$$

Assuming the distortionless constraint of the beamformer

$$\mathbf{h}^H(\omega)\mathbf{d}(\omega) = 1 \tag{2.17}$$

we can get the white noise gain as

$$\mathcal{G}_{\mathrm{wn}}[\mathbf{h}(\omega)] = \frac{1}{\mathbf{h}^H(\omega)\mathbf{h}(\omega)} \tag{2.18}$$

Directivity Factor

The evaluation of the SNR gain for diffuse noise is called the directivity factor. In [3] the pseudo-coherence matrix $\Gamma_v(\omega)$ for diffuse noise and between all microphones is given as

$$[\mathbf{\Gamma}_{\mathrm{v}}(\omega)]_{ij} = [\mathbf{\Gamma}_{\mathrm{dn}}(\omega)]_{ij} \quad = \mathrm{sinc}\left(\frac{\omega\delta_{ij}}{c}\right) \tag{2.19}$$

where

$$\mathrm{sinc}(x) = \frac{\sin x}{x} \tag{2.20}$$

and

$$\delta_{ij} = 2r\left|\sin\frac{\pi(i-j)}{M}\right| \tag{2.21}$$

is the distance between the microphones i and j.
Similar to 2.18 the directivity gain can now be evaluated as

$$\mathcal{G}_{\mathrm{dn}}[\mathbf{h}(\omega)] = \frac{1}{\mathbf{h}^H(\omega)\mathbf{\Gamma}_{\mathrm{dn}}(\omega)\mathbf{h}(\omega)} \tag{2.22}$$

Further the directivity index for a certain frequency can be determined with

$$\mathcal{D}[\mathbf{h}(\omega)] = 10log_{10}\mathcal{G}_{\mathrm{dn}}[\mathbf{h}(\omega)] \tag{2.23}$$

2.4 Minimum-Norm Solution (MNS) Beamformers

In this section one approach to design the transfer functions of the filter in equation 2.8 mentioned in Section 2.1 is shown.
The straight forward solution to the problem of designing a CDMA for a given number of microphones and a fixed order presented in [3] has a severe problem with white noise gain at low frequencies. There a linear system of equations is solved for a CDMA with order $N = \lfloor \frac{M}{2} \rfloor$ for the minimum needed number of microphones given the desired order of the beamformer. A better method that can mitigate the white noise amplification is the minimum norm solution (MNS).
Since the MNS solution yields the same linear system anyways the solution should be the same as for the direct solution so the details to these solutions are left out here. One big advantage of the MNS is that the white noise gain can be controlled by increasing the number of microphones. This way the white-noise-gain at low frequencies can be reduced to an acceptable level.

The general way to design a CDMA would be to solve the following system of $N' = N+1+M'$ linear equations

$$\boldsymbol{A}(\omega,\theta)\boldsymbol{h}(\omega) = \boldsymbol{b} \tag{2.24}$$

where $\boldsymbol{A}(\omega,\theta)$ is the constraint matrix size $N' \times M$

$$\boldsymbol{A}(\omega,\theta) = \begin{bmatrix} \boldsymbol{D}(\omega,\boldsymbol{\theta}) \\ \boldsymbol{C} \end{bmatrix} \tag{2.25}$$

with $\boldsymbol{D}(\omega,\boldsymbol{\theta})$ being the matrix of steering vectors as

$$\boldsymbol{D}(\omega,\boldsymbol{\theta}) = \begin{bmatrix} \mathbf{d}^H(\omega,0) \\ \mathbf{d}^H(\omega,\theta_{N,1}) \\ \vdots \\ \mathbf{d}^H(\omega,\theta_{N,N}) \end{bmatrix} \tag{2.26}$$

The steering vector $\mathbf{d}^H(\omega,\theta_{N,n})$ is the steering vector of length M for the N angles $\theta_{N,n}$ of the desired nulls.
So the matrix $\boldsymbol{D}(\omega,\boldsymbol{\theta})$ has size $(N+1)\times M$. The matrix C contains the corresponding symmetry conditions for the array that are derived from the symmety conditions of the filter in equation

2.8 and is given as

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{c}_{M,1}^T \\ \boldsymbol{c}_{M,2}^T \\ \vdots \\ \boldsymbol{c}_{M,M'}^T \end{bmatrix} \tag{2.27}$$

where the vector $\boldsymbol{c}_{M,m'}^T$ contains $M' = M - \lfloor \frac{M}{2} \rfloor - 1$ constraints that ensure the symmetry of the beampattern and is determined as

$$\boldsymbol{c}_{M,m'}^T = \begin{cases} 1 & \text{at } m'+1 \\ -1 & \text{at } M-m'+1 \\ 0 & \text{otherwise} \end{cases} \tag{2.28}$$

In other words every $(m'+1)$th element of $\boldsymbol{c}_{M,m'}^T$ is 1, while every $(M-m'+1)$th element is -1. The rest of the elements is zero. This matrix describes the repetition of frequency responses in equation 2.8 that occurs due to the circular shape of the array.
So $\boldsymbol{c}_{M,m'}^T$ is a matrix of size $M' \times M$

The filter vector $\mathbf{h}(\omega)$ of length M is described as

$$\mathbf{h}(\omega) = \begin{bmatrix} H_1(\omega) & H_2(\omega) & \cdots & H_M(\omega) \end{bmatrix}^T \tag{2.29}$$

$$\boldsymbol{b} = \begin{bmatrix} \boldsymbol{\beta^T} & 0 & \cdots & 0 \end{bmatrix}^T \tag{2.30}$$

is a vector of length N' and the vectors

$$\boldsymbol{\theta} = \begin{bmatrix} 0 & \theta_{N,1} & \cdots & \theta_{N,N} \end{bmatrix}^T \tag{2.31}$$

$$\boldsymbol{\beta} = \begin{bmatrix} 1 & \beta_{N,1} & \cdots & \beta_{N,N} \end{bmatrix}^T \tag{2.32}$$

are the design coefficients for the directivity pattern of length $N+1$. The values of $\boldsymbol{\beta}$ are the solutions of the given linear equations and $\boldsymbol{\theta}$ are the angles of those solutions. So with these coefficients the positions of N desired nulls can be designed. Unlike the straight forward solution to this problem where we would choose $N = \lfloor \frac{M}{2} \rfloor$ we want to choose M larger than needed. After minimizing the residual noise at the output of the beamformer the minimum norm solution for spatially white noise and $M > N'$ is obtained as

$$\mathbf{h}(\omega, \boldsymbol{\theta}, \boldsymbol{\beta}) = \boldsymbol{A}^H(\omega, \theta)[\boldsymbol{A}(\omega, \theta)\boldsymbol{A}^H(\omega, \theta)]^{-1}\boldsymbol{b} \tag{2.33}$$

The MNS solution for differential microphones is one way to overcome the problem of white noise gain at low frequencies. If there are more microphones than needed to construct a certain DMA pattern the spare microphones can be used to reduce the amout of white noise gain.
In Section 2.9.1 the effect of more microphones on the white noise gain can be seen. The

beampattern that is created will deviate from the ideal DMA pattern a little bit depending on the number of microphones used. But in general the designed pattern will still be very stable with an acceptable amount of white noise gain.

2.5 Superdirective Beamformers

A different approach to differential beamforming presented in [3] based on [6] is the design of superdirective beamformers. In the following section the three different designs for superdirective circular beamformers given in [3] are presented.

2.5.1 Superdirective Beamformer without Symmetry Constraint

Using the array gain in diffuse noise given in Section 2.3

$$\mathcal{G}_{\text{dn}}[\mathbf{h}(\omega)] = \frac{|\mathbf{h}^H(\omega)\mathbf{d}(\omega,0)|^2}{\mathbf{h}^H(\omega)\mathbf{\Gamma}_{\text{dn}}(\omega)\mathbf{h}(\omega)} \tag{2.34}$$

the superdirective beamformer is given as

$$\mathbf{h}_{max}(\omega) = \frac{\mathbf{\Gamma}_{\text{dn}}^{-1}(\omega)\mathbf{d}(\omega,0)}{\mathbf{d}^H(\omega,0)\mathbf{\Gamma}_{\text{dn}}^{-1}(\omega)\mathbf{d}(\omega,0)} \tag{2.35}$$

Based on the derivation of the robust superdirective beamformer for a uniform linear array (ULA) the first beamformer is the superdirective beamformer without symmetry constraint given as

$$\mathbf{h}_{\epsilon_r}(\omega) = \frac{[\mathbf{\Gamma}_{\text{dn}}(\omega) + \epsilon_r \boldsymbol{I}_M]^{-1}\mathbf{d}(\omega,0)}{\mathbf{d}^H(\omega,0)[\mathbf{\Gamma}_{\text{dn}}(\omega) + \epsilon_r \boldsymbol{I}_M]^{-1}\mathbf{d}(\omega,0)} \tag{2.36}$$

which is a regularized version of 2.35. The parameter ϵ_r can be used to find a compromise between supergain and white noise gain. Small ϵ_r lead to a large directivity factor with a lot of white noise amplification, while large ϵ_r lead to low directivity factors with less white noise.
It can be seen that for $\epsilon_r = 0$ equation 2.36 leads to

$$\mathbf{h}_0 = \mathbf{h}_{max}(\omega) \tag{2.37}$$

and

$$\mathbf{h}_\infty = \frac{\mathbf{d}(\omega,0)}{M} \tag{2.38}$$

where 2.38 yields the maximum possible white noise gain of

$$\mathcal{G}_{\text{wn}}[\mathbf{h}_\infty(\omega)] = M \tag{2.39}$$

Since the unregularized superdirective beamformers suffer from extremely high white noise amplification it can be advantageous to make the regularization factor frequency dependent. ϵ_r can be high for low frequencies where the white noise amplification is very high and get lower with increasing frequency to get a higher directivity factor.

2.5.2 Superdirective Beamformer with Symmetry Constraint

Using the constraint vectors $c_{M,m}$ already introduced in 2.28 and the distortionless constraint we get

$$\boldsymbol{C}(\omega)\mathbf{h}(\omega) = \boldsymbol{i}_1 \tag{2.40}$$

where

$$\boldsymbol{C}(\omega) = \begin{bmatrix} \mathbf{d}^H(\omega, 0) \\ \boldsymbol{c}_{M,1}^T \\ \boldsymbol{c}_{M,2}^T \\ \vdots \\ \boldsymbol{c}_{M,M'}^T \end{bmatrix} \tag{2.41}$$

is a matrix of size $(M'+1) \times M$ and

$$\boldsymbol{i}_1 = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T \tag{2.42}$$

is a vector of length $(M+1)$.
After solving the optimisation problem

$$\min_{\mathbf{h}(\omega)} \mathbf{h}(\omega)^H \boldsymbol{\Gamma}_{\text{dn}}^{-1}(\omega)\mathbf{h} \quad \text{subject to} \quad \boldsymbol{C}(\omega)\mathbf{h}(\omega) = \boldsymbol{i}_1 \tag{2.43}$$

which yields the superdirective beamformer

$$\mathbf{h}_{max}(\omega) = \boldsymbol{\Gamma}_{\text{dn}}^{-1}(\omega)\boldsymbol{C}^H(\omega)[\boldsymbol{C}(\omega)\boldsymbol{\Gamma}_{\text{dn}}^{-1}(\omega)\boldsymbol{C}^H(\omega)]^{-1}\boldsymbol{i}_1 \tag{2.44}$$

the robust beamformer can be derived as

$$\mathbf{h}_{\epsilon_r}(\omega) = [\boldsymbol{\Gamma}_{\text{dn}}(\omega) + \epsilon_r \boldsymbol{I}_M]^{-1}\boldsymbol{C}^H(\omega)[\boldsymbol{C}(\omega)[\boldsymbol{\Gamma}_{\text{dn}}(\omega) + \epsilon_r \boldsymbol{I}_M]^{-1}\boldsymbol{C}^H(\omega)]^{-1}\boldsymbol{i}_1 \tag{2.45}$$

where we get

$$\mathbf{h}_0(\omega) = \mathbf{h}_{max}(\omega) \tag{2.46}$$

and

$$\mathbf{h}_\infty(\omega) = \boldsymbol{C}^H(\omega)[\boldsymbol{C}(\omega)\boldsymbol{C}^H(\omega)]^{-1}\boldsymbol{i}_1 \tag{2.47}$$

Compared to the first derived superdirective beamformer the symmetry constraints in the matrix $\boldsymbol{C}$ ensure the symmetry of the beampattern around θ_s.

2.5.3 Superdirective Beamformer with Maximum of Nulls

Another superdirective beamformer presented in [3] is the beamformer with the maximum of nulls. Here we want to have zeros at the microphone positions $\psi_i, \quad i = 2, 3, \cdots, M$ and a one

at the first microphone $\psi_1 = 0$. The linear system of equations that has to be solved is then

$$\boldsymbol{C}_N(\omega)\mathbf{h}(\omega) = \boldsymbol{i}_{M,1} \tag{2.48}$$

where

$$\boldsymbol{C}_N(\omega) = \begin{bmatrix} \mathbf{d}^H(\omega, 0) \\ \mathbf{d}^H(\omega, \psi_2) \\ \vdots \\ \mathbf{d}^H(\omega, \psi_M) \end{bmatrix} \tag{2.49}$$

is a constraint matrix of size $M \times M$ containing the steering vectors of the microphone positions ψ_i and $\boldsymbol{i}_1$ is a vector of length M.

$$\boldsymbol{i}_{M,1} = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^T \tag{2.50}$$

The superdirective beamformer with the maximum of nulls can then be found by solving

$$\mathbf{h}_N(\omega) = \boldsymbol{C}_N^{-1}(\omega)\boldsymbol{i}_{M,1} \tag{2.51}$$

2.6 Jacobi-Anger Expansion

One additional approach for designing circular differential microphone arrays that is mentioned in [3] is the design using the Jacobi-Anger expansion.
In this design the beampattern is obtained from the approximation of the general definition of the beampattern through the Jacobi-Anger expansion. The details of the approximation are skipped here and only the design principle is presented.
Starting from an alternative description of the frequency independent beampattern

$$\begin{aligned} \mathcal{B}_{Ch,N}(\theta) &= \sum_{n=0}^{N} b_{N,n} T_n(\cos\theta) \\ &= \sum_{n=0}^{N} b_{N,n} \cos(n\theta) \end{aligned} \tag{2.52}$$

where

$$T_n(\cos\theta) = \cos(n\theta) \tag{2.53}$$

are Chebyshev polynomials of the first kind, the describing beampattern is approximated and a relation between the coefficients of that expansion and those of a beamforming filter is shown. Ultimately the system of linear equations that has to be solved is described as

$$\boldsymbol{\Psi}_{N+1}\mathbf{h}'(\omega) = \boldsymbol{b}_{N+1}^*(\omega) \tag{2.54}$$

where

$$\mathbf{\Psi}_{N+1} = \begin{bmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \cos\psi_2 & \cos\psi_3 & \cdots & \cos\psi_{N+1} \\ 1 & \cos 2\psi_2 & \cos 2\psi_3 & \cdots & \cos 2\psi_{N+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \cos N\psi_2 & \cos N\psi_3 & \cdots & \cos N\psi_{N+1} \end{bmatrix} \tag{2.55}$$

is a matrix of size $(N+1) \times (N+1)$ and * denotes a complex conjugation. The resulting filter with length $(N+1)$ of this system of equations is

$$\mathbf{h}'(\omega) = \begin{bmatrix} H'_1(\omega) & H'_2(\omega) & \cdots & H'_{N+1}(\omega) \end{bmatrix}^T \tag{2.56}$$

and the vector containing the design coefficients is given as

$$\boldsymbol{b}^*_{N+1}(\omega) = \frac{1}{2}\left[\frac{b_{N,0}}{J'^*_0(\overline{\omega})} \quad \frac{b_{N,1}}{J'^*_1(\overline{\omega})} \quad \cdots \quad \frac{b_{N,N}}{J'^*_N(\overline{\omega})}\right]^T \tag{2.57}$$

where $\overline{\omega}$ is the scaled angular frequency

$$\overline{\omega} = \frac{\omega r}{c} \tag{2.58}$$

$$J'_n(\overline{\omega}) = \begin{cases} J_0(\overline{\omega}), & n = 0 \\ 2j^n J_n(\overline{\omega}), & n = 1, 2, \cdots, N \end{cases} \tag{2.59}$$

and $J_n(\overline{\omega})$ is the nth order Bessel function of the first kind. The final filter for the microphone array has then to be found from the vector $h'(\omega)$ like

$$H_1(\omega) = 2H'_1(\omega) \tag{2.60}$$

$$H_m(\omega) = H'_m(\omega), \quad m = 2, 3, \cdots, \mathcal{M} - 1 \tag{2.61}$$

$$H_{\mathcal{M}}(\omega) = \begin{cases} H'_{\mathcal{M}}(\omega) & \text{if } M\text{odd} \\ 2H'_{\mathcal{M}}(\omega) & \text{if } M\text{even} \end{cases} \tag{2.62}$$

Where $\mathcal{M} = N + 1$ is the number of calculated transfer functions. The resulting transfer functions have then to be arranged depending of the symmetry of the array to get the whole filter matrix for the array.

2.6.1 Second-Order Differential Arrays

To gain deeper insight into the design with this method the example of a second order arrays is given here. The design of first order arrays is not discussed here since the derivation is very similar. Also since the other designs all have at least four microphones for comparability the second order design was choosen since it uses also either four or five microphones.

The first step is to determine the coefficients $b_{2,n}, n = 0, 1, 2$ from the coefficients of the ideal

beampattern $a_{2,n}, n = 0, 1, 2$.

$$\begin{aligned}\mathcal{B}_2(\theta) &= a_{2,0} + a_{2,1}\cos\theta + a_{2,2}\cos^2\theta \\ &= (1 - a_{2,1} - a_{2,2}) + a_{2,1}\cos\theta + a_{2,2}\cos^2\theta\end{aligned} \tag{2.63}$$

$$\begin{aligned}\mathcal{B}_{Ch,2}(\theta) &= b_{2,0} + b_{2,1}\cos\theta + b_{2,2}\cos(2\theta) \\ &= (b_{2,0} - b_{2,2}) + b_{2,1}\cos\theta + 2b_{2,2}\cos^2\theta\end{aligned} \tag{2.64}$$

we get

$$\begin{aligned}b_{2,0} &= 1 - a_{2,1} - \frac{a_{2,2}}{2} \\ b_{2,1} &= a_{2,1} \\ b_{2,2} &= \frac{a_{2,2}}{2}\end{aligned} \tag{2.65}$$

so using 2.55 and 2.57, 2.54 the solution for 4 microphones results as

$$\begin{bmatrix}1 & 1 & 1 \\ 1 & 0 & -1 \\ 1 & -1 & 1\end{bmatrix}\begin{bmatrix}H_1'(\omega) \\ H_2'(\omega) \\ H_3'(\omega)\end{bmatrix} = \frac{1}{2}\begin{bmatrix}\frac{b_{2,0}}{J_0'^*(\overline{\omega})} \\ \frac{b_{2,1}}{J_1'^*(\overline{\omega})} \\ \frac{b_{2,2}}{J_2'^*(\overline{\omega})}\end{bmatrix} \tag{2.66}$$

With 2.60,2.61 and 2.62 the complete filter for all microphones can then be arranged as

$$H_1(\omega) = 2H_1'(\omega), H_2(\omega) = H_2'(\omega), H_3(\omega) = 2H_3'(\omega) \tag{2.67}$$

$$\mathbf{h}(\omega) = \begin{bmatrix}H_1(\omega) & H_2(\omega) & H_3(\omega) & H_2(\omega)\end{bmatrix}^T \tag{2.68}$$

This approach can be easiliy extended to DMAs with different orders by using their ideal patterns and deriving the needed coefficients from them. Everything else needed is the matrix Ψ_{N+1} which only depends on the angular microphone positions.

2.7 Extension to Three Dimensional Signal Model

For now all beampatterns, optimisations and evaluations were only made for the two-dimenional case. Since for real world applications the three-dimensional beampattern is also of interest a three-dimensional coordinate is introduced in this chapter. The steering vector is extended to a third dimension and beampatterns of some designed arrays are explored using this new steering vector.

In Fig. 2.4 the used coordinate system can be seen. The two-dimenional coordinates stay the same as already defined in Fig. 2.1. The system is only extended by an additional angle γ that is counted upwards from the x/y plane.
The steering vector used for this coordinate system then results from equation 2.3 as

$$\mathbf{d}_{3\mathrm{d}}(\omega, \theta, \gamma) = \begin{bmatrix}\mathrm{e}^{j\omega r c^{-1}\cos(\theta - \psi_1)\cos\gamma} & \dots & \mathrm{e}^{j\omega r c^{-1}\cos(\theta - \psi_M)\cos\gamma}\end{bmatrix}^T \tag{2.69}$$

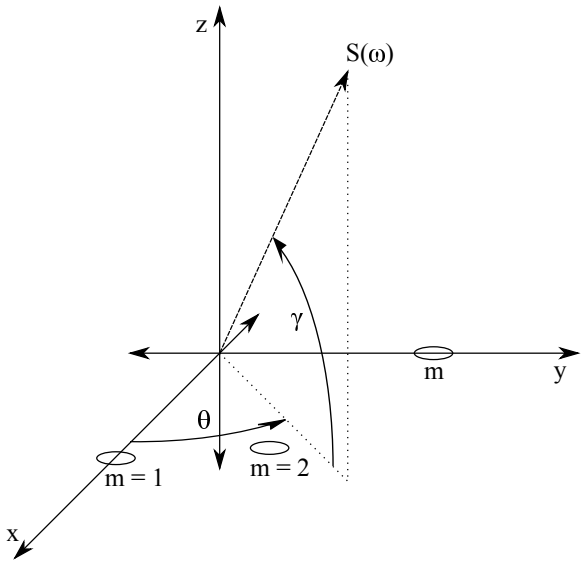
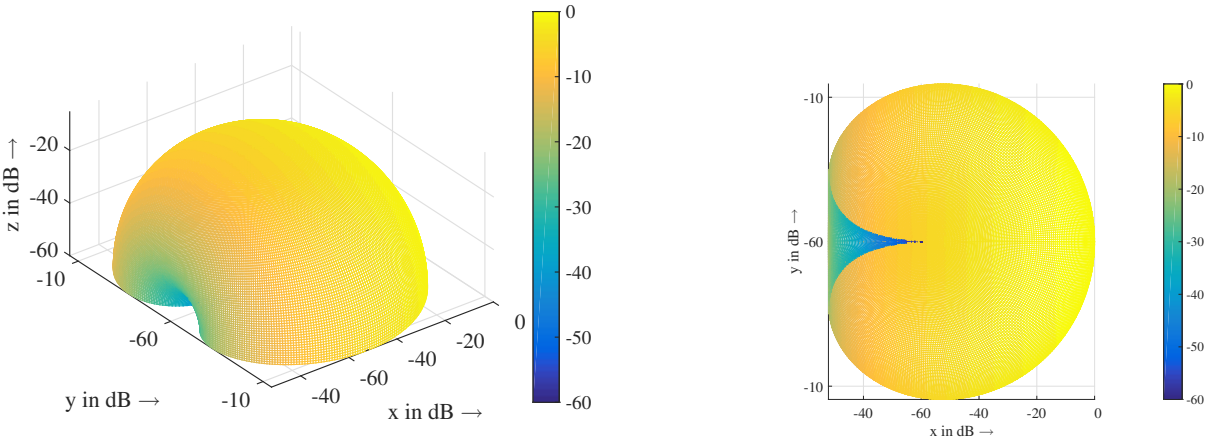


Figure 2.4: Three-dimensional coordinate system

Using this steering vector the designed beampatterns are evaluated as

$$\begin{aligned}\mathcal{B}[\mathbf{h}(\omega),\theta,\gamma] &= \mathbf{h}^H(\omega)\mathbf{d}_{3\mathrm{d}}(\omega,\theta,\gamma)\\ &= \sum_{m-1}^{M} H_m^*(\omega)\mathrm{e}^{j\omega r c^{-1}\cos(\theta-\psi_m)\cos\gamma}\end{aligned} \tag{2.70}$$

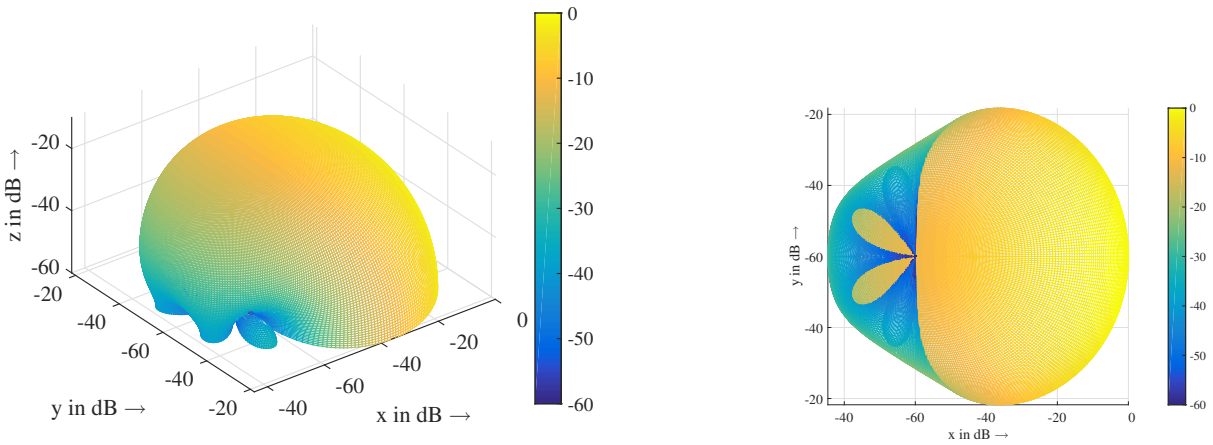


(a) Three-dimensional beampattern

(b) Bottom view

Figure 2.5: MNS first order cardioid for $\delta = 10mm$ and $M = 4$ microphones, $f = 1875Hz$

In Fig. 2.5 and Fig. 2.6 two examples of beamformers designed with the MNS approach are depicted for $\delta = 10\,\mathrm{mm}$ and $M = 4$ and $M = 6$ microphones and a frequency of $f = 1875\,\mathrm{Hz}$. It can be seen that while the desired beampattern is as expected in the plane where it was designed depending on the elevation the beampatterns lose some of their properties.
Especially in Fig. 2.6(b) we can see that the designed zeros are vanishing quite fast and there are no destinctive zeros left in the beampattern. Besides degrading the effectiveness of the designed zeros of the beampattern this could also impact the performance of an adaptive beamfomer that tries to minimize the signal output like in Section 3.2 since there are no clear zeros to minimize the output energy of the system.



(a) Three-dimensional beampattern

(b) Bottom view

Figure 2.6: MNS third order cardioid for $\delta = 10mm$ and $M = 6$ microphones, $f = 1875Hz$

2.8 Differential Beamforming as Convex Problem

The design of differential beamformers as an convex problem can offer some interesting options considering the probelms mentioned in the privious chapters. In [7] it was already shown that superdirective beamformers can be designed with a constraint on the white noise gain using Sequential Quadratic Programming and using the beampattern of a delay and sum beamformer as pattern to optimise. Further in [8] a beamformer is developed using an ideal target pattern consisting only of ones in the desired direction and zeros elsewhere.
So the question arises if it is viable to design a beampattern for arrays with small apertures that have a DMA pattern or a pattern that improves on the problematic elevation dependency of the beampattern shown in Section 2.7.

2.8.1 Problem Definition

In this section the problem of designing planar circular differential beamformers will be described as a convex problem that can be solved under various constraints with a convex solver.
Similar to [8] the beampattern will be designed describing the problem in vector notation and solving the least squares problem

$$\min_{\mathbf{h}(\omega)} \|\boldsymbol{G}(\omega) \cdot [\mathbf{h}(\omega) \otimes \boldsymbol{I}] - \boldsymbol{\mathcal{B}}_{\boldsymbol{des}}\|_F \tag{2.71}$$

where $\otimes$ descibes the Kronecker product, $\boldsymbol{I}$ is the identity matrix of rank N_γ being the number of discrete elevation angles that should be optimised. The vector $\mathbf{h}(\omega)$ are the weights that have to be optimised with vector length beeing M and $\boldsymbol{G}(\omega)$ is a matrix size $N_\theta \times (M \cdot N_\gamma)$ where N_θ is the number of azimutal angles to optimise.
The matrix $\boldsymbol{G}(\omega)$ is constructed from the three-dimenional steering vector $\mathbf{d}_{3\mathrm{d}}(\omega)$ to match the result of the Kronecker product.

$$\mathbf{d}_{3\mathrm{d},m}(\omega) = \begin{bmatrix} \mathbf{d}_{3\mathrm{d},m=1}(\omega) & \mathbf{d}_{3\mathrm{d},m=2}(\omega) & \cdots & \mathbf{d}_{3\mathrm{d},m=M}(\omega) \end{bmatrix} \tag{2.72}$$

where $\mathbf{d}_{3\mathrm{d},m}(\omega)$ is a matrix of the steering vector calculated for the discrete azimuthal and elevation angles of size $N_\theta \times N_\gamma$.

$$\mathrm{d}_{3\mathrm{d},m}(\omega) = \mathrm{e}^{j\omega r c^{-1} \cos(\theta - \psi_m) \cos\gamma} \tag{2.73}$$

The matrix $\boldsymbol{\mathcal{B}_{des}}$ is the desired beampattern with size $N_\theta \times N_\gamma$. The optimisation is then done for each frequency bin ω independently.

2.8.2 Desired Beampattern

In this section the two beampatterns that were used for the convex optimisation are presented and discussed. It is impossible to design a beamformer for an elevation up to $\gamma = 90°$. The reason for this is, the higher the elevation angle, the lower are the time differences that the microphone array can measure. If the impinging wave is perpendicular to the array there is no time difference of arrival between the microphones and spacial filtering becomes very difficult. For this reason the elevation angle was only optimised up to $\gamma = 15°$ in 50 steps. In Fig. 2.7 it can be seen that for a speaker sitting in front of a table an optimisation angle of 15° can be sufficient.

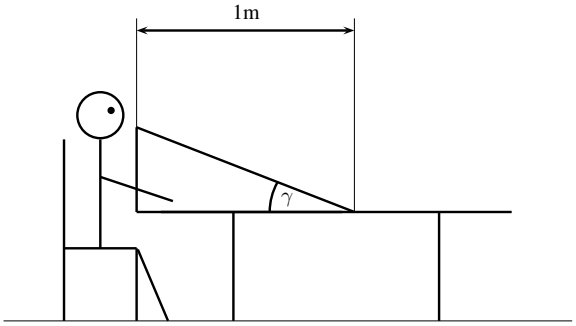
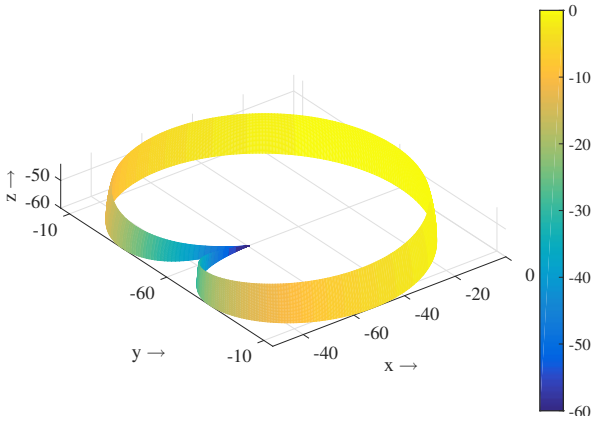
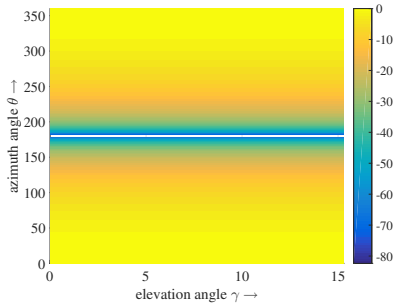


Figure 2.7: Speaker in front of the compact array

In Fig. 2.8 and Fig. 2.9 two of the used optimisation patterns for $\boldsymbol{\mathcal{B}_{des}}$ can be seen. Fig. 2.8 uses a first order DMA pattern while in Fig. 2.9 a binary mask is used just as proposed in [8]. The circle in Fig. 2.9 indicates that the angles that do not belong to the mainlobe were not set to zero but to $-40\,\text{dB}$ to give the optimisation process more freedom when finding a beampattern.



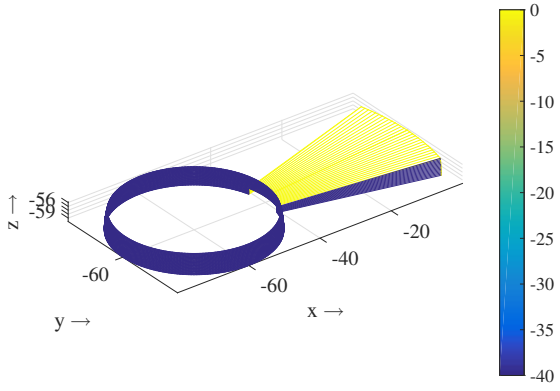
(a) Desired beampattern for first order DMA 3d in dB



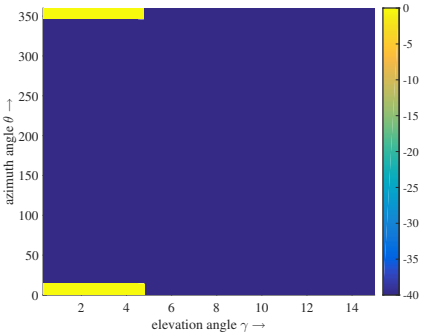
(b) Desired beampattern for first order DMA surface plot in dB

Figure 2.8: Desired beampattern to optimise for DMA pattern

To ensure the distortionless constraint the solution for the target vector $[\theta_s = 0, \gamma_s = 0]$ was constrained to 1. The minima of the beampattern were constrained to be smaller than $-40\,\text{dB}$ and not zero, again to give the optimisation process more freedom.
Also to improve the behaviour of the white noise amplification the optimisation was subjected to the WNG. Since a low noise gain can be tolerable when high SNR microphones are used the



(a) Desired beampattern for first order Bitmask 3d in dB



(b) Desired beampattern for first order Bitmask surface plot in dB

Figure 2.9: Desired beampattern to optimisze for Bitmask pattern

WNG was constrained to be higher than $-10\,\mathrm{dB}$. So all constraints can be summed up as

$$\mathcal{G}_{\mathrm{wn}}[\mathbf{h}(\omega)] = \frac{1}{\mathbf{h}^H(\omega)\mathbf{h}(\omega)} \quad \mathbf{d}_{3\mathrm{d}}(\omega, \theta_s, \gamma_s)\mathbf{h}(\omega) = 1 \quad \boldsymbol{V}(\omega)\mathbf{h}(\omega) = 1 \tag{2.74}$$

Where $\boldsymbol{V}(\omega)$ is a matrix containing the steering vector data for the desired minima of the beampattern. The combination of all constraints can lead to an infeasible problem where no viable solution can be found. The results of the calculations and what constraints are working are discussed in Section 2.9.4 in detail.
The optimisation was implemented with the convex optimisation toolbox CVX in MATLAB. In the table below the pseudo code of the optimisation is shown for one frequency bin.

Algorithm 1 CVX optimisation code

```
cvx_begin
    variable h(M) complex
    minimize(norm(G · kron(h, eye(N_γ)) − B_des), 'fro')
    subject to
        d_3d · h = 1
        abs(V · h) <= 0.01
        h' · h <= 1/0.1
cvx_end
```

2.9 Designed Beamformers

In this section the proposed beamformers for the project are presented. The number of microphones as well as the diameters of the circles were not only choosen by the performance of the CMDA but also regarding the acoustic source localisation.
The requirements for beamforming and source localisation are contradicting. To get a constant beampattern over frequency for the CDMA it is best to have a small inter-element spacing δ. For localising it is better to have a large δ since it makes time delay estimation easier. (Section 4.1)
Beamformers and localising have one requirement in common. For arrays with larger apertures

the white noise gain at low frequencies is less.
This means that a balance between WNG/localising and frequency independent beampatterns has to be found.

Considering the requirements of the CDMAs five different circular geometries were choosen.

- 4 microphones with 10 mm microphone distance
- 6 microphones with 10 mm microphone distance
- 12 microphones with 10 mm microphone distance
- 4 microphones with 20 mm microphone distance
- 6 microphones with 20 mm microphone distance

All design methods will be tested with those arrays and in Section 5.1, real world recordings are presented to evaluate the beamformer algorithms with real data.

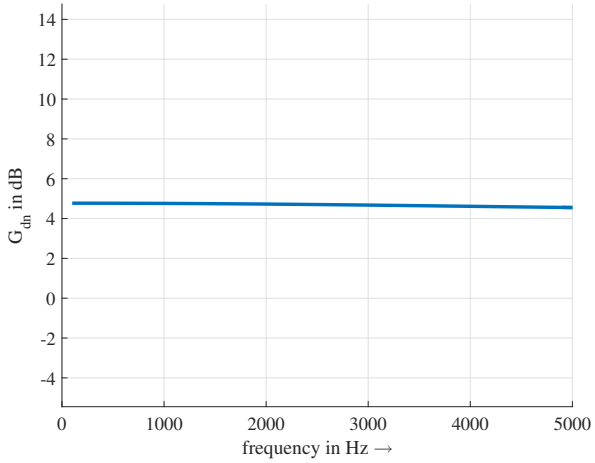
2.9.1 MNS Beamformers

Here the beamformers designed with the minimum norm solution are presented. All the designed beampatterns are first order cardioids so the differences of the choosen arrays can be explored. Although with the used MNS method it is easiliy possible to design arbitray DMA orders. In Fig. 2.15 a designed third order pattern with three distinctive nulls can be seen.
In the following plots the measures discussed in Section 2.3 are plotted up to a frequency of 5 kHz. Since the frequency responses of DMAs tend to have a lot of notches at high frequencies it is advisable to apply a lowpassfilter to the signal that is processed. For speech signals this is not a major problem since most information is in this frequency band so the intellibility should be uncompromised.

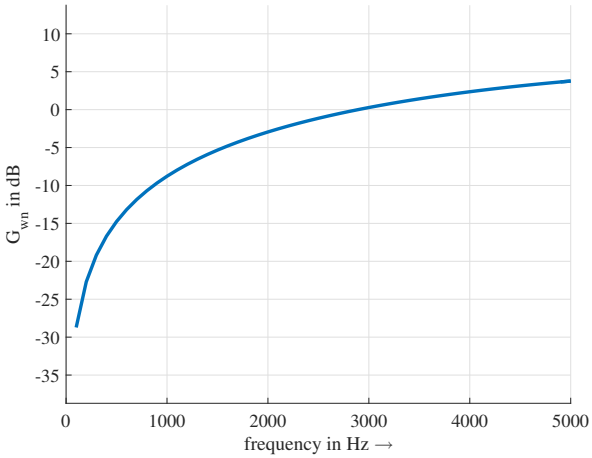
The plots in Fig. 2.10 show the directivity pattern, the white noise gain and the beampattern of the minimum norm solution for four microphones with $\delta = 10\,\text{mm}$. In Fig. 2.10(a) the directivity pattern is plotted and we can see that in the considered frequency band it is very stable.
For few microphones and small element distances the white noise gain is still problematic as demonstrated in Fig. 2.10(b). For low frequencies the white noise gain starts at about $-25\,\text{dB}$ which indicates a severe increase of white noise at the beamformer output.
Since the differencial approximation of the array works best for small element distances the beampattern in Fig. 2.10(c) is very stable up to 5 kHz. The beampattern in plotted there for 5 frequencies up to 5 kHz and the patterns are almost completely overlapping.

In Fig. 2.11 a first order cardiod designed for an array of 6 microphones can be seen. Compared to the 4 microphone version in Fig. 2.10 the white noise gain improves already. The point where the white noise gain gets positive moves from about 2.9 kHz down to 1.7 kHz.
The beampattern and the directivity pattern are still constant in the whole frequency region.

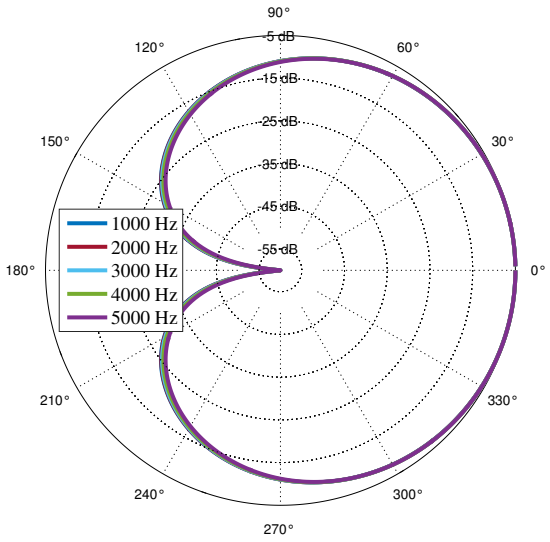
One limit for increasing the number of microphones is shown in Fig. 2.12. While the white noise gain continues to improve in Fig. 2.12(b) the beampattern in Fig. 2.12(c) starts to develop unwanted zeros at about 90°. This behaviour can also be noticed in the directivity pattern in Fig. 2.12(a) since for high frequencies the value increases a little bit.
So even if the point where the white noise gain gets to positive values shifts down to about 600 Hz the beampattern already gets frequency variant which could be a problem if this fixed beamformer is used for adaptive beamforming as in chapter Section 3.2. This behaviour can be traced back to the increase of element spacing due to more microphones. Even in this



(a) Directivity pattern



(b) White noise gain



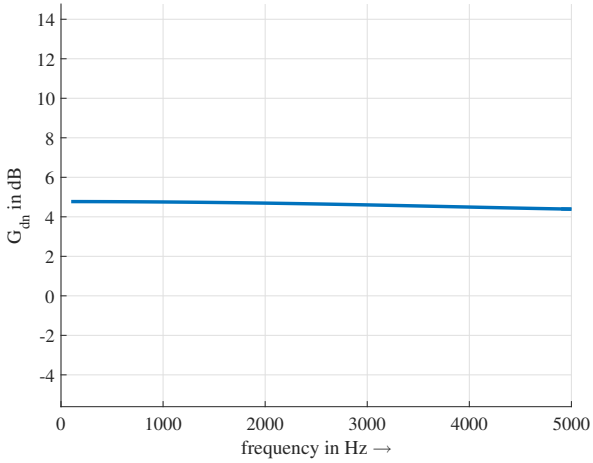
(c) Polar plot in dB

Figure 2.10: MNS first order cardioid for $\delta = 10mm$ and $M = 4$ microphones

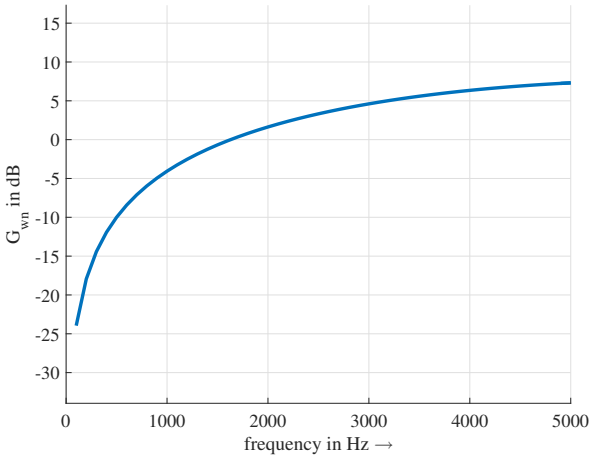
configuration the designed zero at 180° is stable and for frequencies lower than 4 kHz the pattern is stable.

In Fig. 2.13 and Fig. 2.14(c) the first order cardioid design can be seen for arrays with 4 and 6 microphones with $\delta = 20$ mm element distance. Compared to the arrays with $\delta = 10$ mm it is visible that the WNG is better for a larger δ but the beampattern develops unwanted zeros an uncontrolled angles faster.
This makes sense considering the beahaviour already seen at high frequencies for the array with 12 microphones. Here the directivity pattern already develops sidelobes with 4 microphones since the element spacing is larger from the beginning. But just as for the arrays with $\delta = 10$ mm the beampattern is almost frequency independent under 4 kHz and the designed zero is stable for the whole considered frequency range.

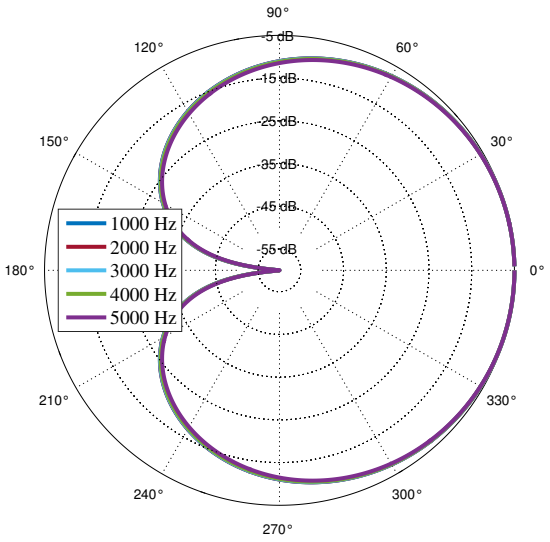
Compared to the first order cardioids we can see that the white noise gain is worse for the higher order design in Fig. 2.15. This is a behavior that all differential beamformers share. The higher the amount of distincitve nulls is, the higher the amount of white noise gets.
For the third order design the white noise gain starts at about -80 dB. This means that this design is practically not usable since there would only be noise at the beamformer output.
This is a pity espacially since the designed 3 zeros are constant in the considered frequency



(a) Directivity pattern



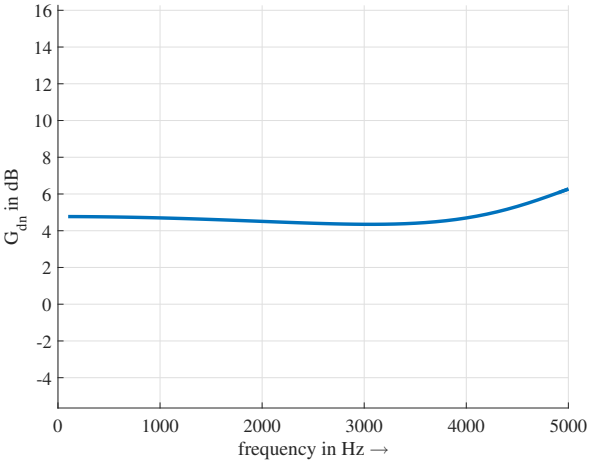
(b) White noise gain



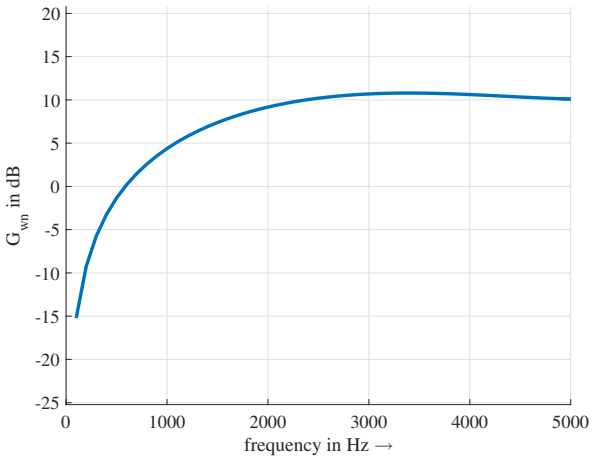
(c) Polar plot in dB

Figure 2.11: MNS first order cardioid for $\delta = 10\,\text{mm}$ and $M = 6$ microphones

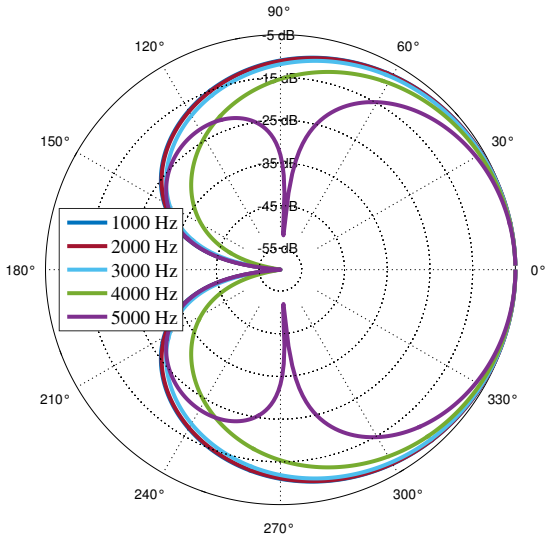
regions and the beampattern does not develop any additional zeros.



(a) Directivity pattern

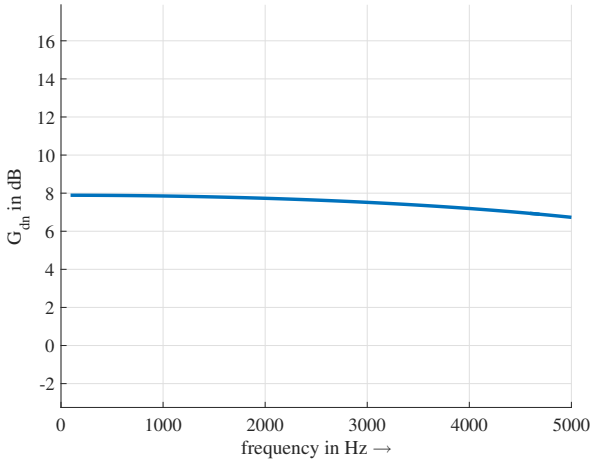


(b) White noise gain

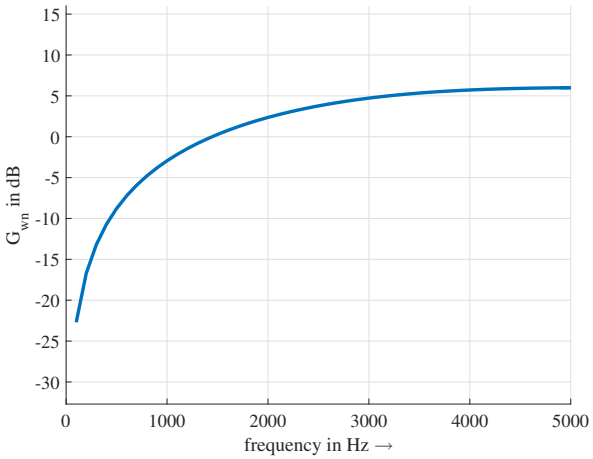


(c) Polar plot in dB

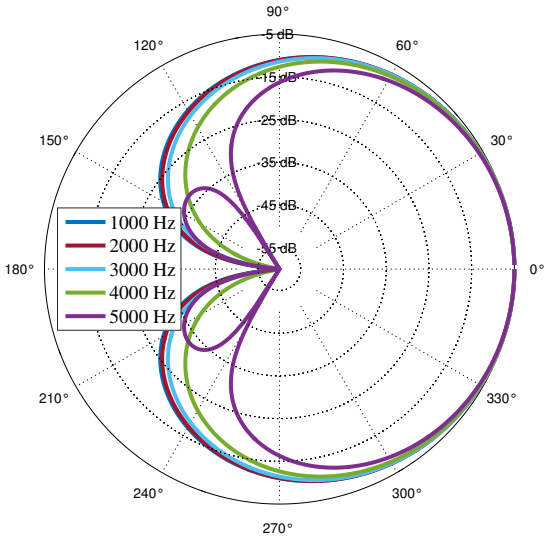
Figure 2.12: MNS first order cardioid for $\delta = 10\,\mathrm{mm}$ and $M = 12$ microphones



(a) Directivity pattern

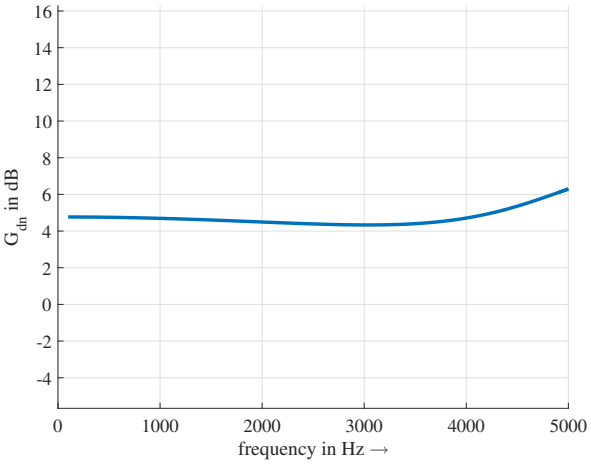


(b) White noise gain

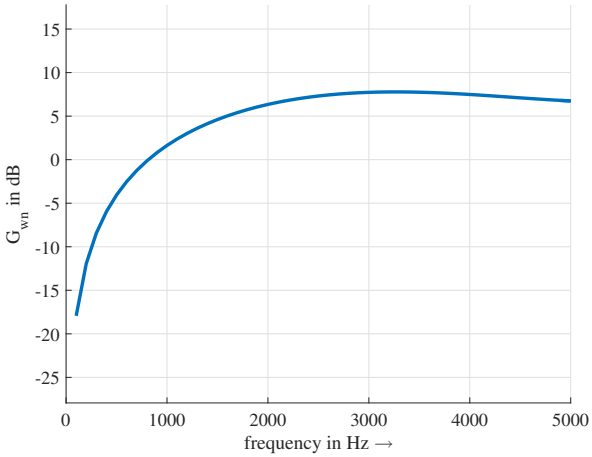


(c) Polar plot in dB

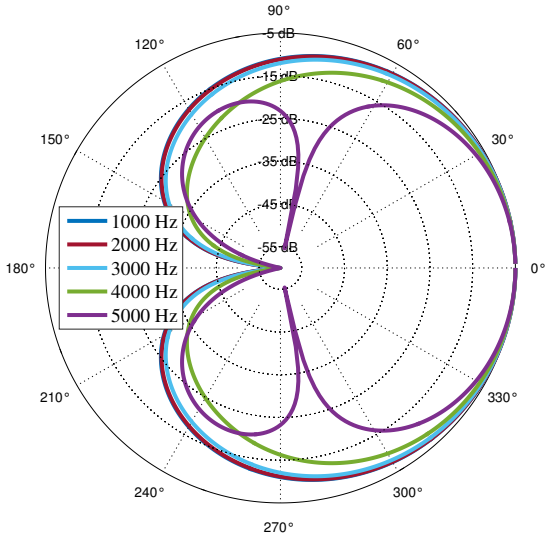
Figure 2.13: MNS first order cardioid for $\delta = 20\,\mathrm{mm}$ and $M = 4$ microphones



(a) Directivity pattern

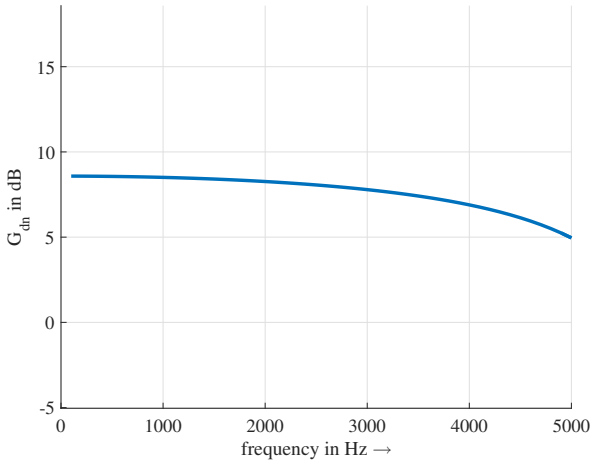


(b) White noise gain

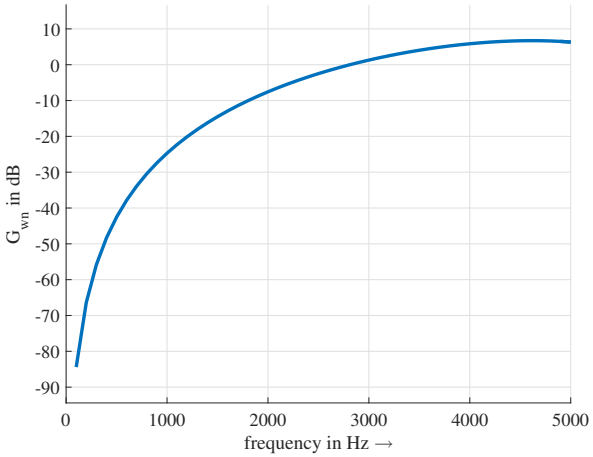


(c) Polar plot in dB

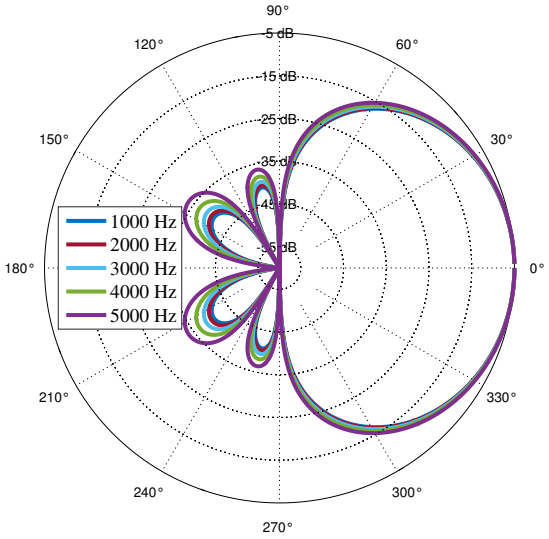
Figure 2.14: MNS first order cardioid for $\delta = 20\,\text{mm}$ and $M = 6$ microphones



(a) Directivity pattern



(b) White noise gain



(c) Polar plot in dB

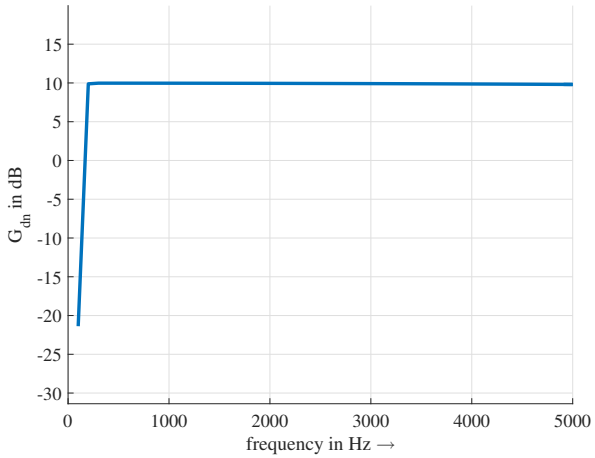
Figure 2.15: MNS third order cardioid for $\delta = 20\,\text{mm}$ and $M = 6$ microphones

2.9.2 Superdirective Beamformers

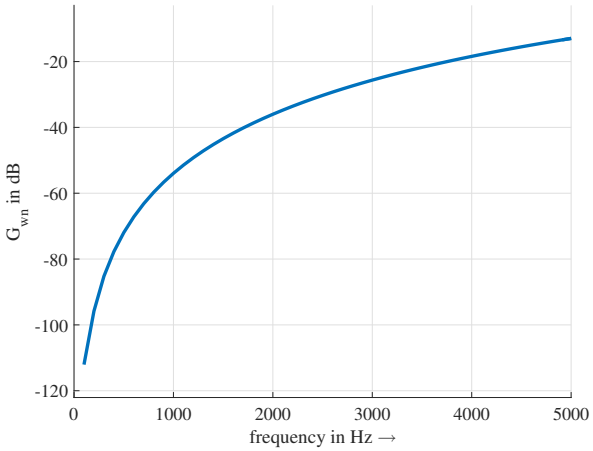
In this section some designed superdirective beamformers are presented. To keep the chapter clear only beamformers for one proposed geometry are shown. Considering the microphone array with $M = 6$ microphones and a sensor distance of $\delta = 10\,\text{mm}$ the differences between the designed beamformers will be explored.

Without Symmetry Constraint

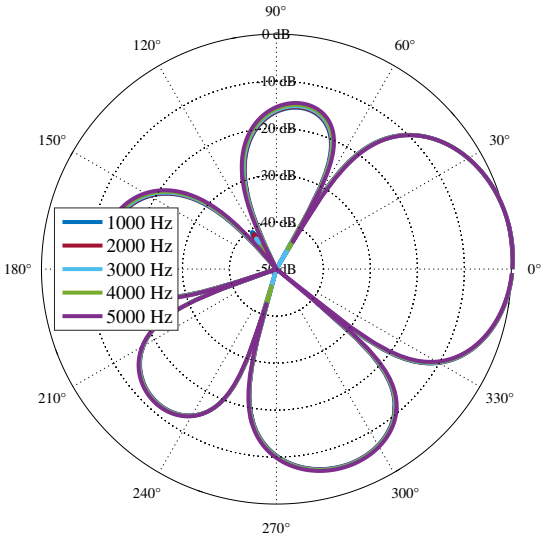
In Fig. 2.16 the designed superdirective beamformer without symmetry constraint can be seen. The desired mainlobe θ_s was set to be at 15°. Looking at the beampattern clearly there is no guarantee of a symmetric beampattern around θ_s.
Further we see that the directivity pattern is very stable over the whole considered frequency band. Yet since the white noise amplification is extremely high for low frequencies this beamfomer is not usable for practicle applications.



(a) Directivity pattern



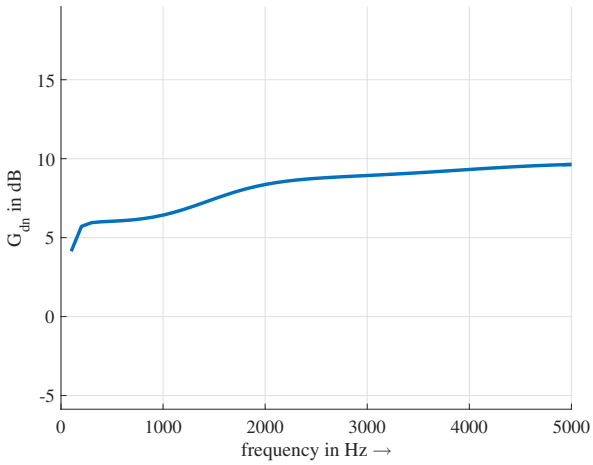
(b) White noise gain



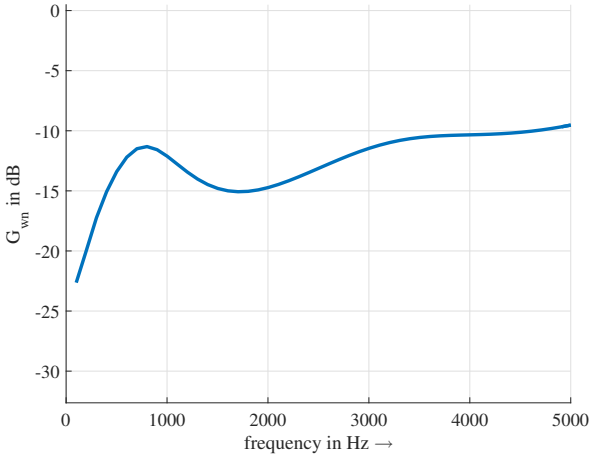
(c) Polar plot in dB

Figure 2.16: Superdirective beamformer for $\delta = 10\,\text{mm}$ and $M = 6$ microphones and $\theta_s = 15°$

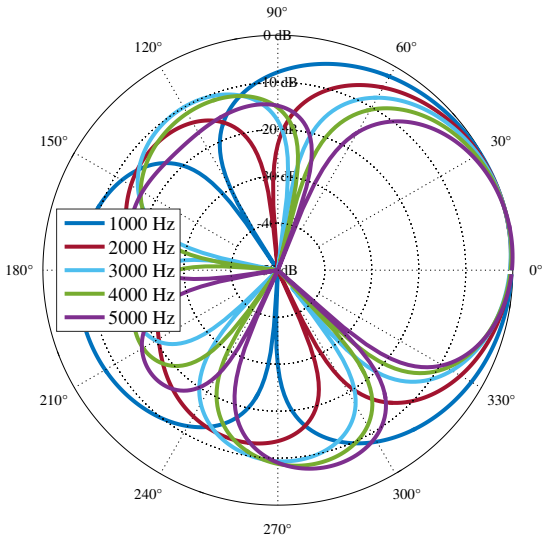
The plots in Fig. 2.17 show the result of the robust superdirective beamformer without symmetry constraint. This design tries to overcome the problematic white noise gain by optimising the beamformer with a constraint on the white noise gain. Looking at Fig. 2.17(b) the white



(a) Directivity pattern



(b) White noise gain



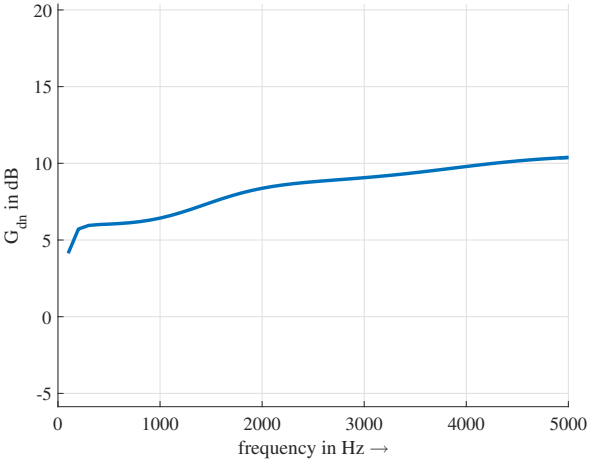
(c) Polar plot in dB

Figure 2.17: Robust superdirective beamformer for $\delta = 10\,\text{mm}$ and $M = 6$ microphones and $\theta_s = 15°$, $\epsilon_r = 0.001$

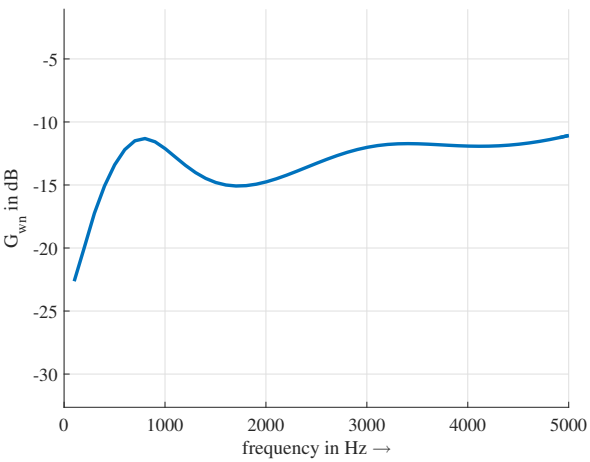
noise gain is a lot higher than for the not robust beamfomer.
The better white noise gain is at the expense of a frequency dependent beampattern. While the mainlobe stays at the desired 15° the zeros of the beampattern are changing with the frequency. This can also be observed in the plot of the directivity pattern which is not as stable anymore as for the not robust beamformer.
The regularization factor was set to $\epsilon_r = 0.001$ as proposed in [3].

With Symmetry Constraint

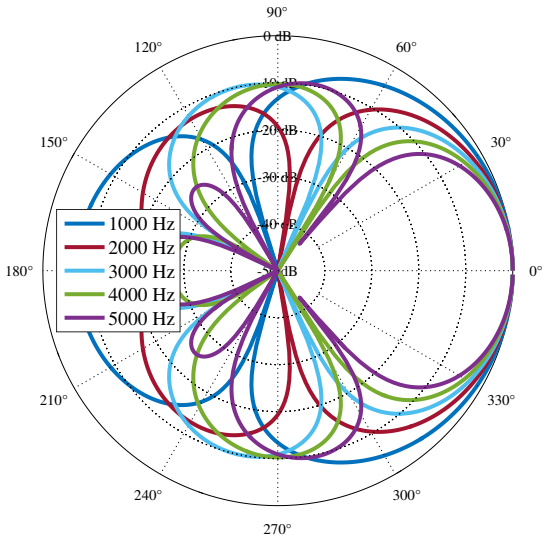
In Fig. 2.18 the robust superdirective beamformer with symmetry constraint around θ_s is shown. The resulting white noise gain and directivity pattern are very similar to the robust beamformer designed without the symmetry constraints. Only the beampattern is now constrained to be symmetric around the desired mainlobe which is now set to be $\theta_s = 0$ again.
The regularization factor is again $\epsilon_r = 0.001$ which leads to a beampattern that is frequency variant.



(a) Directivity pattern



(b) White noise gain

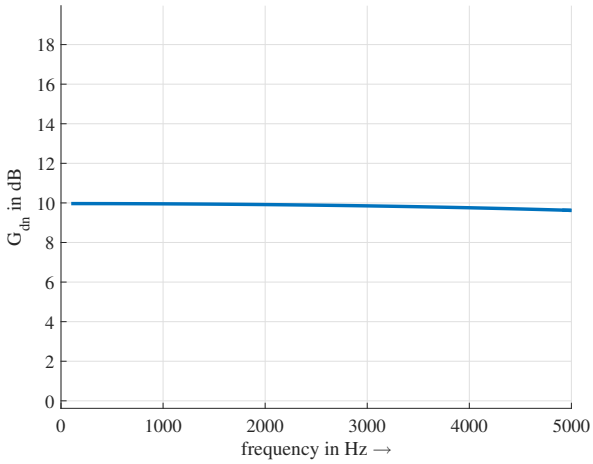


(c) Polar plot in dB

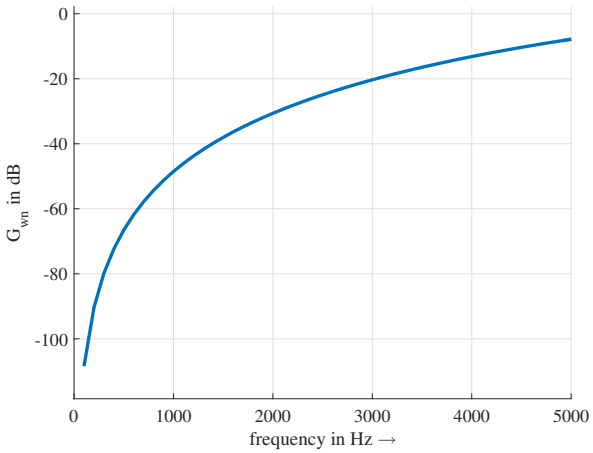
Figure 2.18: Robust superdirective beamformer with symmetry constraint for $\delta = 10\,\text{mm}$ and $M = 6$ microphones, $\epsilon_r = 0.001$

With Maximum of Zeros

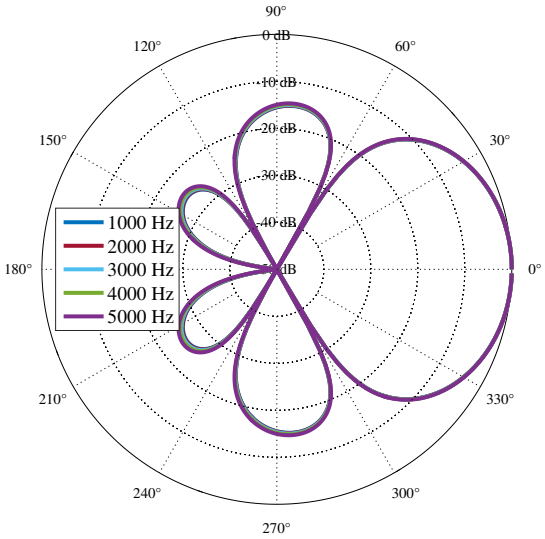
In Fig. 2.19 and Fig. 2.20 two designes for the beamformer with maximum zeros are shown. The directivity and beam-patterns of both beamformers demonstrate that those designes are very stable with frequency. The big drawback of this design is the extremly high white noise amplification in low frequencies that can be observed in the plots of the white noise gain.
Another problem that can appear similar in the other superdirective designs when increasing the number of microphones can be seen in Fig. 2.20(a). The low values at low frequencies of the directivity factor indicate that beamforming is not working at those regions and the beampattern is most likely omnidirectional. This is probably due to the increasing diameter of the used circular array when increasing the number of microphones while keeping the element distance.



(a) Directivity pattern

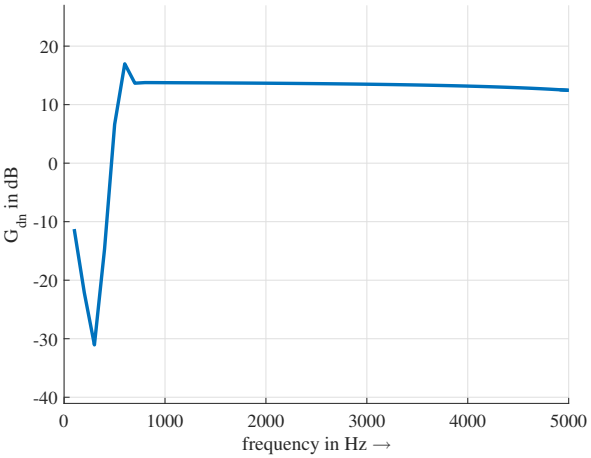


(b) White noise gain

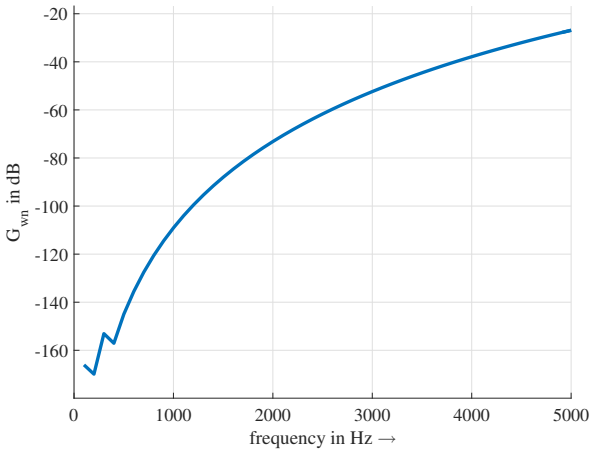


(c) Polar plot in dB

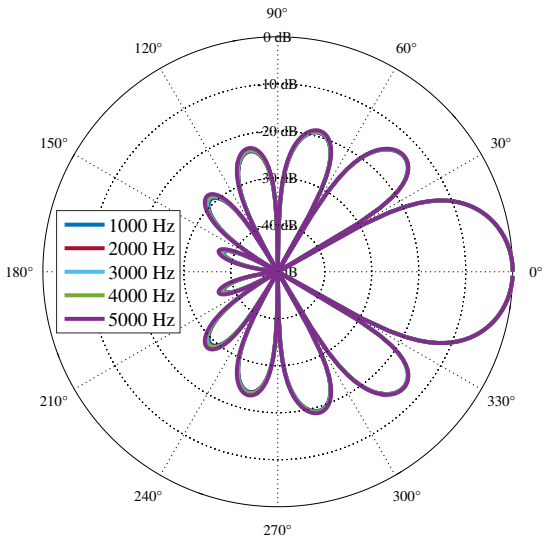
Figure 2.19: Superdirective beamformer with maximum number of zeros for $\delta = 10\,\text{mm}$ and $M = 6$ microphones



(a) Directivity pattern



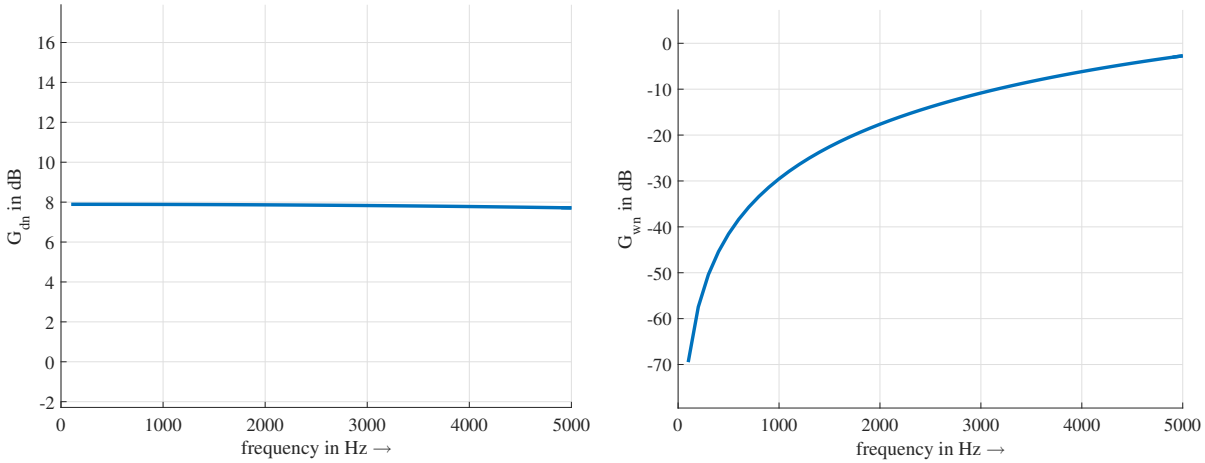
(b) White noise gain



(c) Polar plot in dB

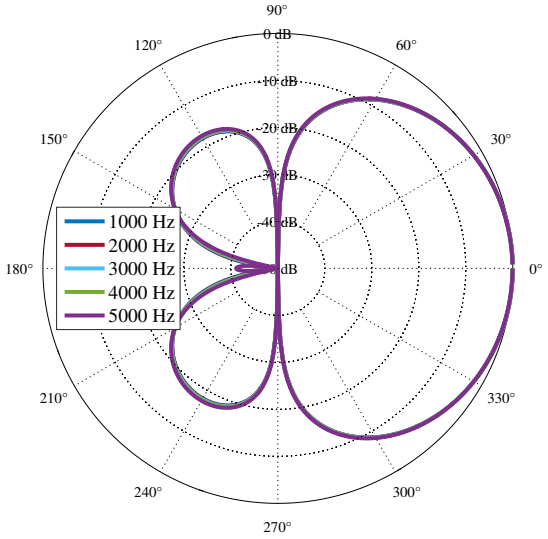
Figure 2.20: Superdirective beamformer with maximum number of zeros for $\delta = 10\,\text{mm}$ and $M = 12$ microphones

2.9.3 Jacobi-Anger Beamformers



(a) Directivity pattern

(b) White noise gain



(c) Polar plot in dB

Figure 2.21: Designed beamformer with Jacobi-Anger expansion for $\delta = 10\,\text{mm}$ and $M = 4$ microphones

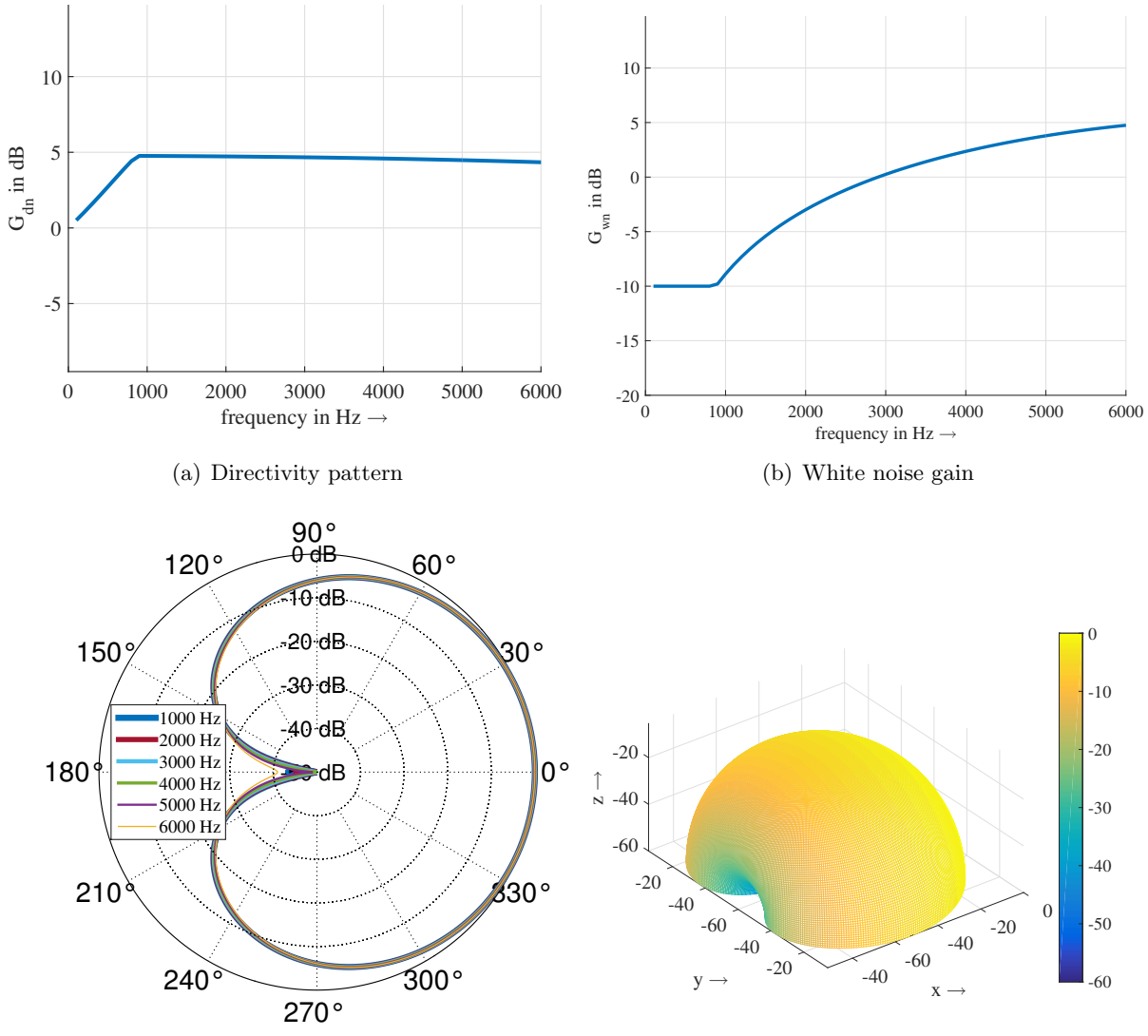
The plots in Fig. 2.21 show the second order design with the Jacobi-Anger expansion. In Fig. 2.21(a) it can be seen that the directivity pattern is very stable over frequency. Unfortunately the white noise amplification is still very high for this approach which is why this design is not very applicable for pracical use.
The performance is comparable to the solution of the MNS beamformer for second order with 4 microphones.

2.9.4 CVX Beamformers

In this section the results of the beamformers designed with the CVX toolbox are shown. The chapter is divided into four parts. First the results for the two design methods used are shown and compared to the design methods studied in the previous chapters. Then the advantages of this method are used to design differential beamformers with very small apertures. In conclusion a summary is given and some interesting behaviours of this method are discussed.

Beamformers Optimised with DMA pattern

This section demonstrates the results of the convex optimised beamformers with DMA patterns. The desired beampattern used to solve the minimisation problem from Fig. 2.71 is an ideal DMA pattern. The optimisation was calculated for first and second order DMAs, with and without a constraint on the white noise gain and with and without constraints on the minima of the pattern.
In Fig. 2.22 the results for an ideal first order DMA is shown for an array with 4 microphones and a sensor spacing of $\delta = 10\,\text{mm}$. The white noise gain was constrained to be higher than $\mathcal{G}_{\text{wn}} = -10\,\text{dB}$. This constraint can be seen in the low frequency range of Fig. 2.22(b).
Fig. 2.22(c) shows the polar plot for the designed beamformer at different frequency values up to $f = 6kHz$. The beampattern is very stable over the whole considered frequency range. But in Fig. 2.22(d) it can be seen that the resulting three-dimensional beampattern is not very different to the one of the MNS beamformer Fig. 2.6.



(a) Directivity pattern

(b) White noise gain

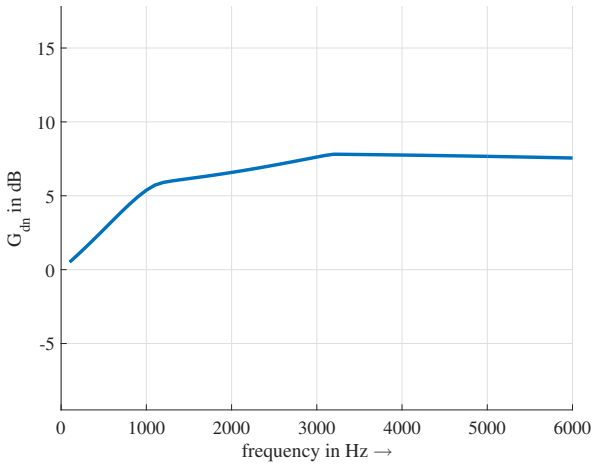
(c) Polar plot in dB

(d) 3d beampattern in dB for 1 kHz

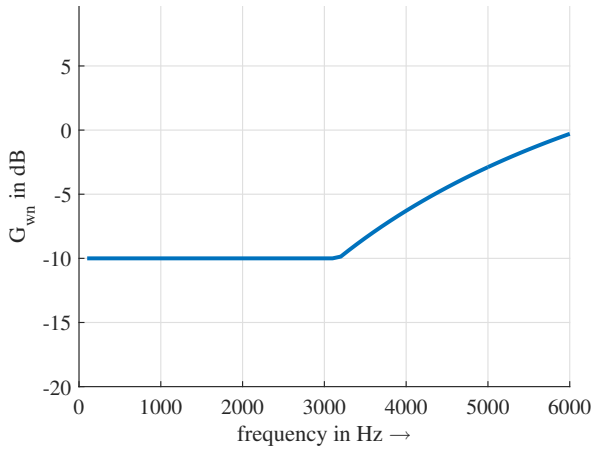
Figure 2.22: designed beamformer with CVX with 1st order DMA pattern and no constraint for the zeros for $\delta = 10\,\text{mm}$ and $M = 4$ microphones

In Fig. 2.22(a) a problem that arises with the constraint on the white noise gain can be seen. The lower values of the directivity pattern for low frequencies indicate that the beampattern

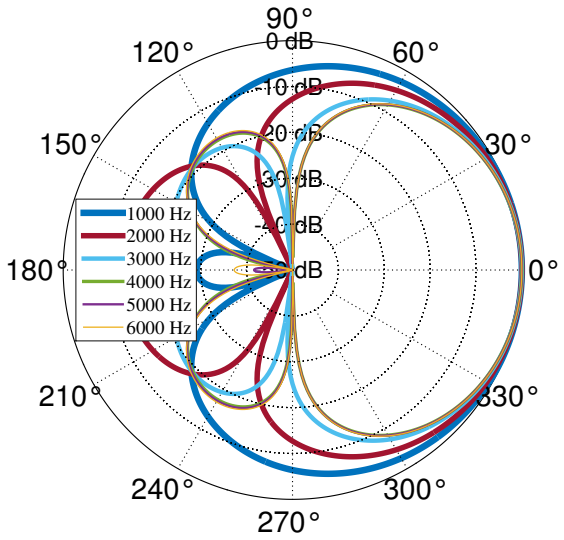
changes more and more into a omnidirectional pattern. In Fig. 2.22(c) this can't be seen since the lowest plotted frequency is 1 kHz. Nevertheless this can pose a problem for speech signals since the degradation of the beampattern starts already at about 900 Hz.



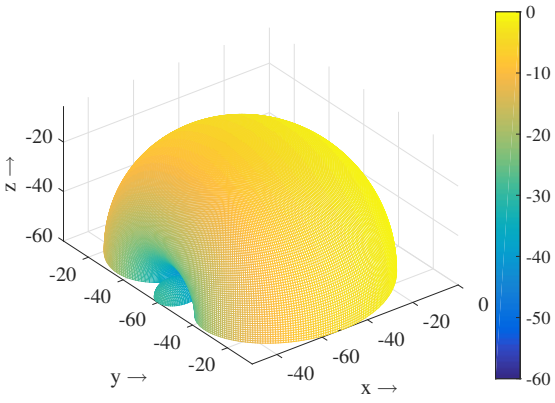
(a) Directivity pattern



(b) White noise gain



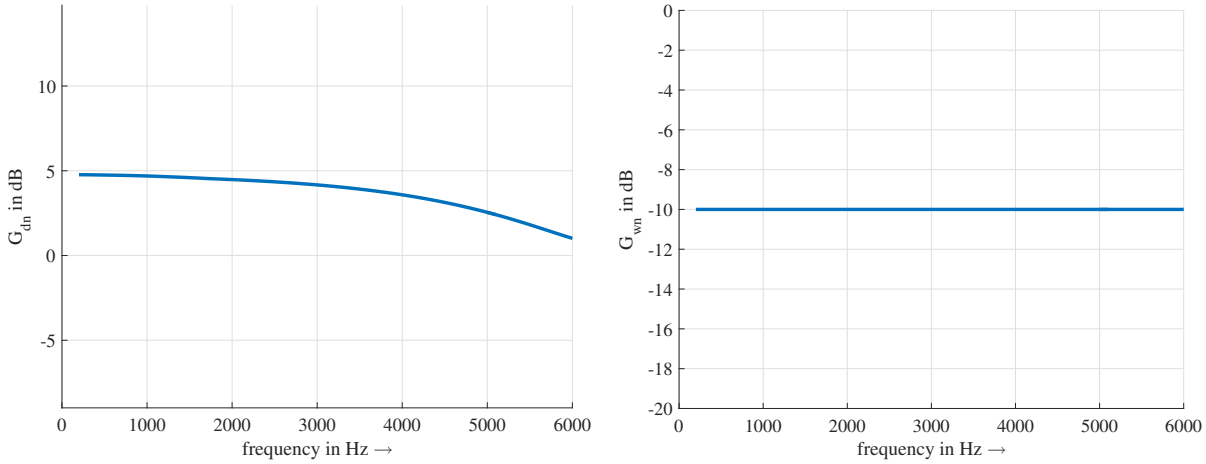
(c) Polar plot in dB



(d) 3d beampattern in dB for 1 kHz

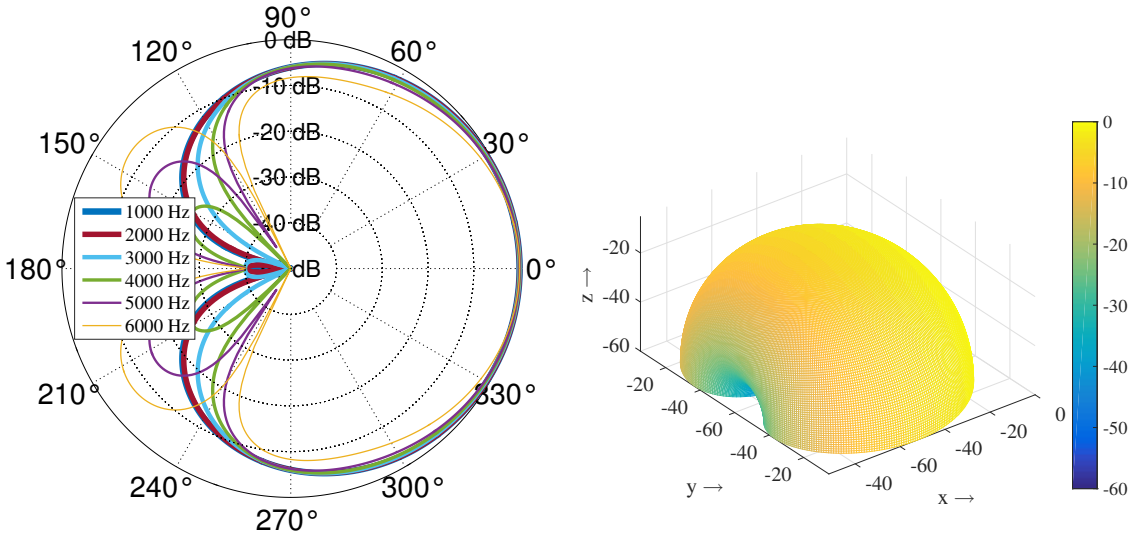
Figure 2.23: designed beamformer with CVX with 2nd order DMA pattern and no constraint for the zeros for $\delta = 10\,\text{mm}$ and $M = 4$ microphones

Fig. 2.23 shows the results for a design with a 2nd order DMA pattern. Again the white noise gain was constrained to be higher than $\mathcal{G}_{\text{wn}} = -10\,\text{dB}$. The consequence of this can be seen in the polar plot of the beampattern in Fig. 2.23(c). The beampattern for low frequencies is differing from the ideal desired 2nd order beampattern.
In Fig. 2.23(a) it can also be seen that just as the design for the first order DMA, the directivity pattern gets very low at low frequencies. There the zeros of the designed pattern shift from the wanted 90/270° backwards. The desired minimum at 180° is completely lost.
In Fig. 2.23(d) the tree-dimensional beampattern is plotted for 1 kHz. The pattern resembles that of a first order design more than a second order design for this frequency.
To improve on this problem the minima of the beampattern were also set as constraints for the optimisation. This can be problematic since the combination of all constraints leads to an infeasible problem for many if not all frequency bins that should be optimised.
One array geometry that yields acceptable results is shown in Fig. 2.24.



(a) Directivity pattern

(b) White noise gain



(c) Polar plot in dB

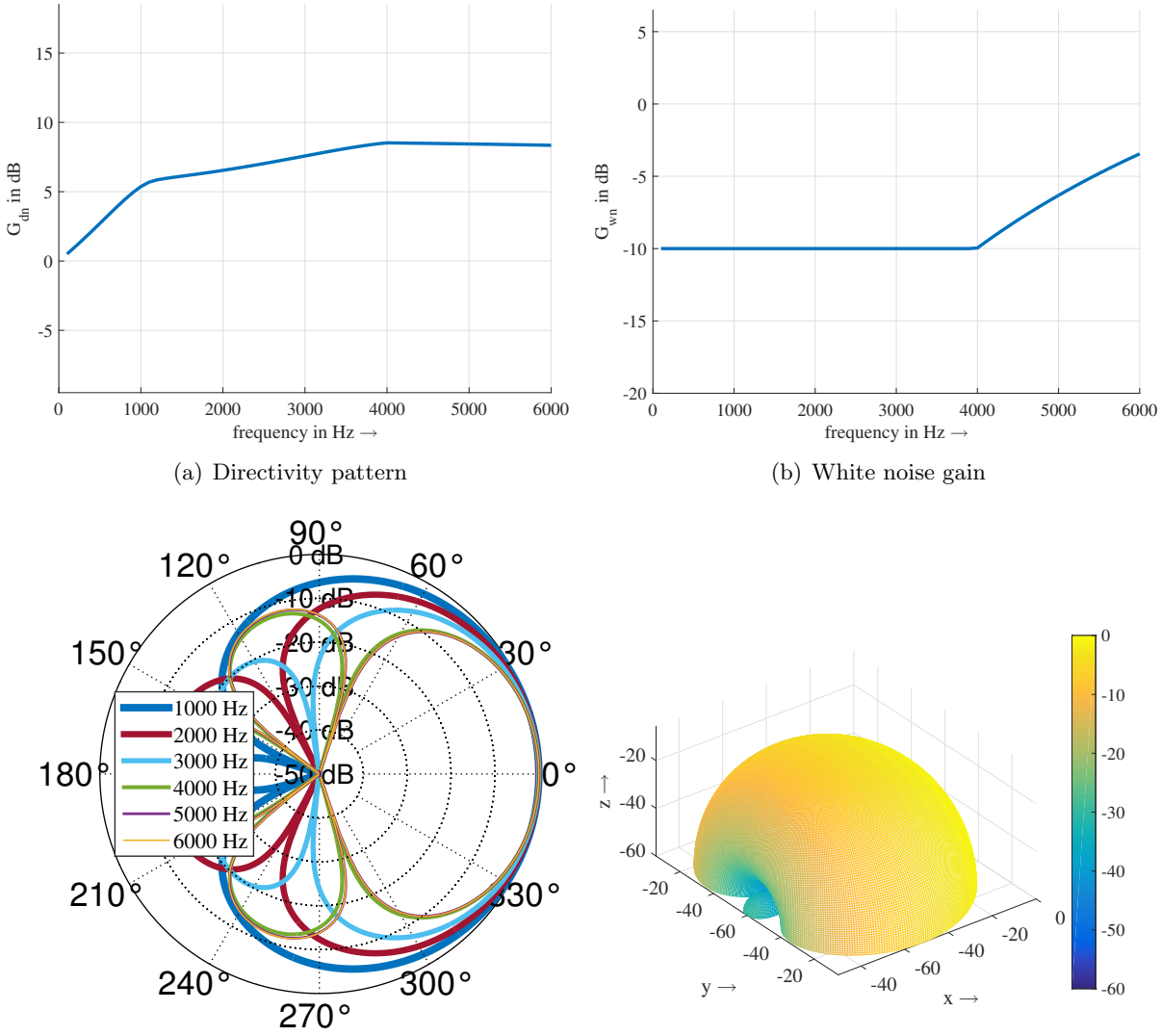
(d) 3d beampattern in dB for 1 kHz

Figure 2.24: designed beamformer with CVX with 1st order DMA pattern and constraint for the zeros for $\delta = 10\,\text{mm}$ *and* $M = 12$ *microphones*

The design for the array with 12 microphones and $\delta = 10\,\text{mm}$ could be calculated using all given constraints. In Fig. 2.24(c) it can be seen that the desired zero is kept nicely even for high frequencies while the constraint on the white noise gain is still in tact.
One drawback that is not easy to spot in the plots is that for the first two frequency bins the problem is still infeasible and can not be solved with cvx. But since these are frequencies below 200 Hz it does not necessarily pose a problem for speech signals.
Compared to it's MNS counterpart Fig. 2.12 the beamformer designed with CVX has smaller sidelobes at high frequencies. One drawback of this design could be that the white noise gain is constantly held at $\mathcal{G}_{\text{wn}} = -10\,\text{dB}$ by the optimisation while the MNS design for first order very soon has positive values for the white noise gain. This means that the CVX beamformer will always add white noise to the signal while the MNS beamformer improves the SNR up to 11dB Fig. 2.12(b).

Beamformers Optimised with Bitmask

Another approach to design the desired beampattern taken from [8] is presented here. The idea is to optimise the weights not towards a desired ideal DMA pattern but to an ideal beamformer pattern that only consists of zeros and ones. As already mentioned in the previous chapter the lower bound of the beamformer is not set to strict zeros but only to be smaller than −40 dB. The beamformers are designed with and without constraints on the zeros. Without any constraint the optimisation should try to minimize the beampattern to reach a optimal beampattern depicted in Fig. 2.9. The desired mainlobe is ±15° wide and ±15° high.
If there are explicit constraints on zeros they are the same as for a first or second order cardioid. So for the first order case there should be one zero at 180°, for the second order case there are three desired zeros at 90°, 180°, 270°.



(a) Directivity pattern

(b) White noise gain

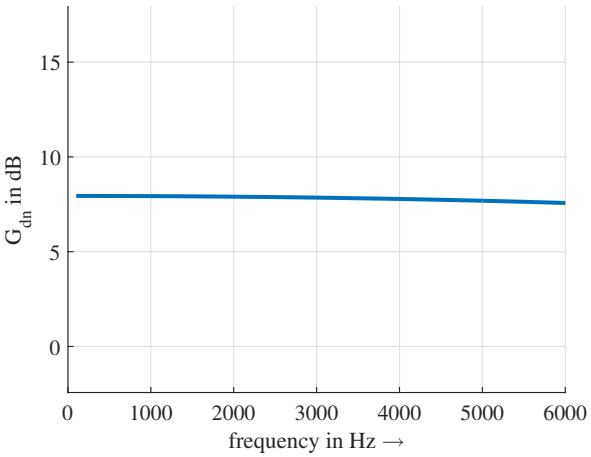
(c) Polar plot in dB

(d) 3d beampattern in dB for 1 kHz

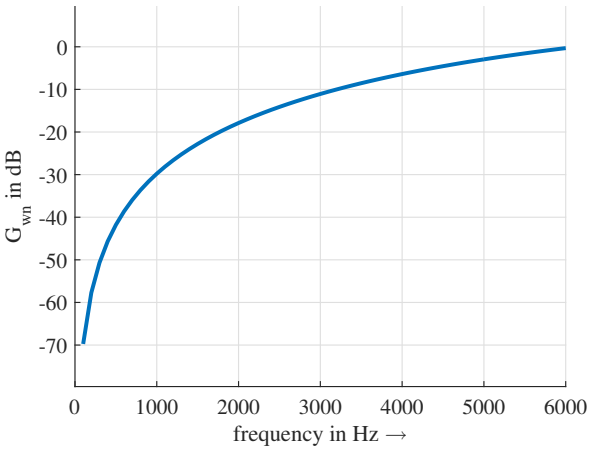
Figure 2.25: designed beamformer with CVX with bitmask and no constraint for zeros for $\delta = 10\,\text{mm}$ and $M = 4$ microphones

In Fig. 2.25 the design with a bitmask for an array of 4 microphones and $\delta = 10\,\text{mm}$ can be seen. The patterns although the desired pattern is completely different from the DMA pattern the result of the optimisaton looks a lot like the solution of the design for ideal DMAs for the same array.
This can mostly be traced back to the constraint on the white noise gain. For low frequencies

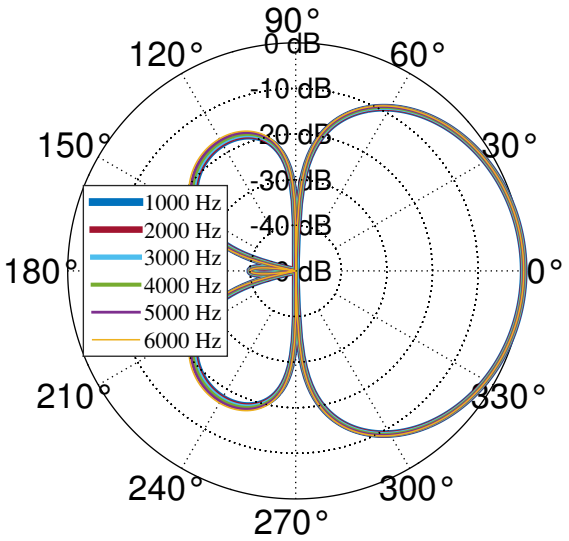
the solution is exactly the same for both variations. The difference is that the design using a bitmask has a lower white noise gain and so the constraint has to be longer enforced than in Fig. 2.22(b). This can also be seen in Fig. 2.25(a) where the second bend in the plot has shifted to higher frequencies than in Fig. 2.22(a).



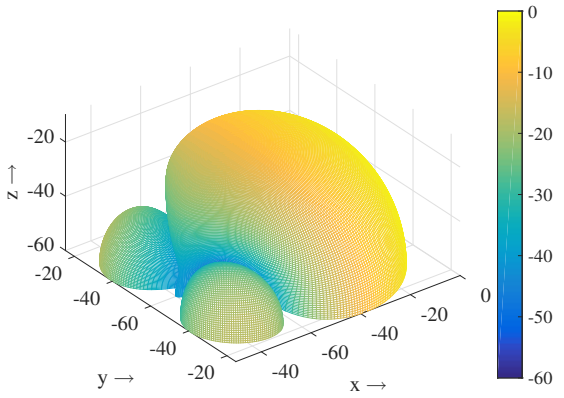
(a) Directivity pattern



(b) White noise gain



(c) Polar plot in dB



(d) 3d beampattern in dB for 1 kHz

Figure 2.26: designed beamformer with CVX with bitmask, constraint for zeros and no constraint on WNG for $\delta = 10\,\text{mm}$ and $M = 4$ microphones

As last example there is the design of a second-order beampattern for $M = 4$ microphones and $\delta = 10\,\text{mm}$ in Fig. 2.26. Compared to the previous examples here the white noise gain was not constrained. As a result the beampattern is constant over frequency. The minima were constrained to get a second order like beampattern. In Fig. 2.26(d) it can be seen that the three-dimensional beampattern is still very close to the solution of the second order MNS beamformer even with optimisation up to $\gamma = 15°$.
The real drawback of this solution is the very high white noise amplification that can be seen in Fig. 2.26(b). Since the value of $\mathcal{G}_{\text{wn}}$ starts at $-70\,\text{dB}$ for low values and does not get positive in the whole considered frequency band this beamformer is not usable for practical applications.

Summary

To sum up the design of differential beamformers with CVX some concluding considerations. One problem that arises with differential microphones is the high white noise gain. This problem can be tackled with CVX by constraining the WNG. The problem is that, as also shown with the superdirective beamfomers, this happens at cost of a frequency invariant beampattern. If the minima of the pattern are conserved using additional constraints the convex problem gets infeasible very often. Even for the cases with up to $M = 12$ microphones the first few frequency bins normally don't have a viable solution. Depending on the used frequency band this can pose a problem or can be ignored if the speech signal is filtered anyways. Another problem that should have been solved when solving the beamforming problem as a convex problem was the optimisation for the elevation angle. This is only possible in a very limited way. On the one side optimising a three-dimensional beampattern with a two-dimesional array is physically not possible at every direction. The larger the elevation, the smaller the time delay between the microphones. So the optimisation for the elevation angle was only done for the first 15° of the elevation. The results were similar to those of the MNS beamfomers. Some differences were that the CVX beamformers tend to set the minimum of the three-dimensional pattern not to the x/y-plane but some degrees above it.
This is not very suprising because if the optimisation is done for more elevation angles, the optimal point of the minimum will be above the plane of the array.

3 Adaptive Beamforming with Circular Differential Microphone Arrays

So far the design of fixed beamformers has been shown and their advantages and disadvantages have been discussed. In this chapter the fixed beamformers are used as source for an adaptive beamformer. The outputs of the fixed beamformers are used to adaptively minimize the system output and therefore cancel one or several noise sources.
Since the used array geometries are symmetric the direction of the mainlobe can be switched to every microphone direction by simple permutation of the transfer functions.

3.1 Simulation Setup

To evaluate the desiged beamformers and localisation algorithms MATLAB [2] simulations were conducted. The parameters used were roughly the same for all algorithms so that all of them could be used in a single system. Here the parameters of the used block processing framework for beamforming are presented.

- blocklength = 512 samples
- overlap = 50%
- window type = Hanning
- sample frequency = 48 kHz
- FFT length = 512 samples

For beamforming the STFT was computed using the MATLAB spectrogram function.

3.2 ACDMA with First-Order Forward/Backward Cardioid

In this section an adaptive beamformer that uses first order cardioids for adaptive beamforming is presented. It is based on the idea shown in [9]. A minor difference lies in the construction of the fixed beamformer. Here no direct implementation of the fixed beamformer using delays

is used but the solution mimics a lot more the adaptive beamfomer using the minimum norm solution shown in [10].
The microphone inputs are fitered with the transfer functions designed with an design algorithm similar to the ones described in the previous chapters. In this way a forward facing cardioid $C_f(\omega)$ that has the steering direction θ_s and a backwards facing cardioid that has its null in that direction $C_b(\omega)$ are constructed. Those two cardioid signals are then used to adaptively minimize the system output.
In Fig. 3.1 the calculation of the two fixed beamformers for the case of $M = 4$ is shown. It can be seen that the ouptut for the backward facing cardioid $C_b(\omega)$ can be constructed by simply circular shifting the vector with the transfer functions of $H(\omega)$ Fig. 2.8. Since the chosen array geometries for the project all have an even number of microphones the backwards cardioid can easily be calculated by simply circular shifting the elements of $H(\omega)$ by $M/2$.

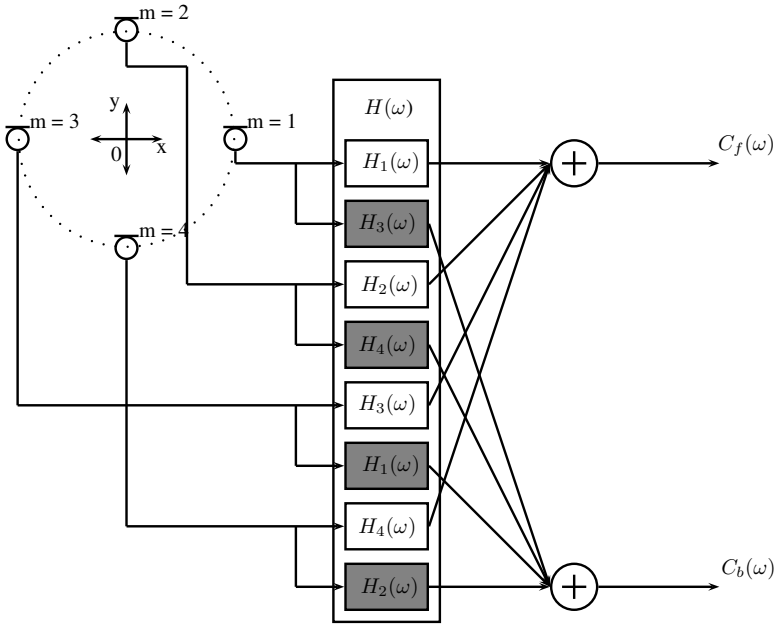


Figure 3.1: Calculation of the fixed beamformers for an array with $M = 4$

In Fig. 3.2 a sketch of the adaptation algorithm can be seen. The two cardioids are added using the real factor β. The spectrum of the system output normalized by the input spectrum can be written as

$$\left|Y(\omega)\right| = |C_f(\omega) - \beta C_b(\omega)| \tag{3.1}$$

This means that β should be adapted in a way to minimise the beamfomer output. To achieve this the value of every frequency bin is minimized. The complete adaptation algorithm is implemented in the frequency domain. As a result the adaptation happens for the whole frequency vector at once. This means that every frequency bin is treated individually. So theoretically even a first order adaptive filter can supress more than one noise source as long as the noise sources have a very different spectrum.

The adaptation is done using the NLMS algorithm to calculate the value of β adaptively in a time-varying environment. This algorithm is easy to implement and it has been shown that it works very well with this setup [10]. The used algorithm in the frequency domain is

$$\beta_{t+1} = \beta_t + \mu \frac{Y_{out}(\omega) C_b(\omega)}{\|C_b(\omega)^2\| + \Delta} \tag{3.2}$$

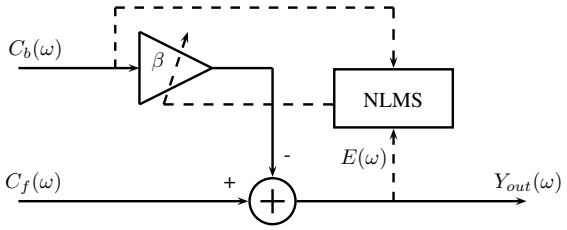


Figure 3.2: Adaptive minimizing of the output

where μ is the stepsize of the NLMS and Δ is a regularization factor that ensures stability for low energies of $C_b(\omega)$. The index t denotes the index of the adapted frame and the error input signal for the NLMS $E(\omega)$ equals the output signal $Y_{out}(\omega)$. The adaptation coefficients of the NLMS are constrainted to $[0, 1]$. This way a noise source in the back of the array can be supressed. In Fig. 3.3 a simulation of the adaptation can be seen. The mono signal in the upper plot was convolved with the steering vector of the array to get the multichannel array data. For the simulations the dry mono signals were taken from headset speaker signals, from speakers 8 and 12, recorded in the AMISCO corpus. [11]
The speaker moves from 0 to 360° around the array. In the lower plot of Fig. 3.3 the result of the adaptation can be seen. From about 1.5 s to 4.5 s when the speaker is at the backside of the array. The cancellation works really well.
In a real world secenario the performance of the noise cancellation will of course be degraded by sensor imperfection and reflections.

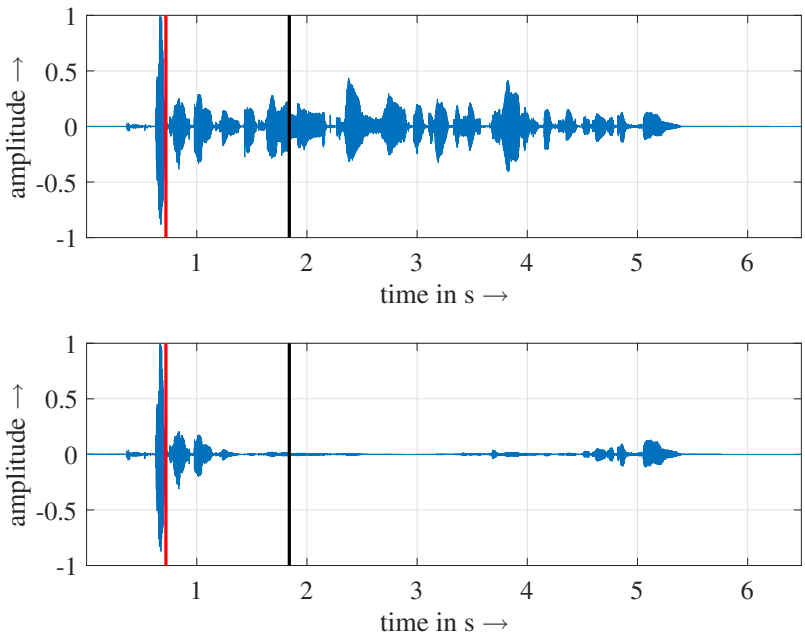
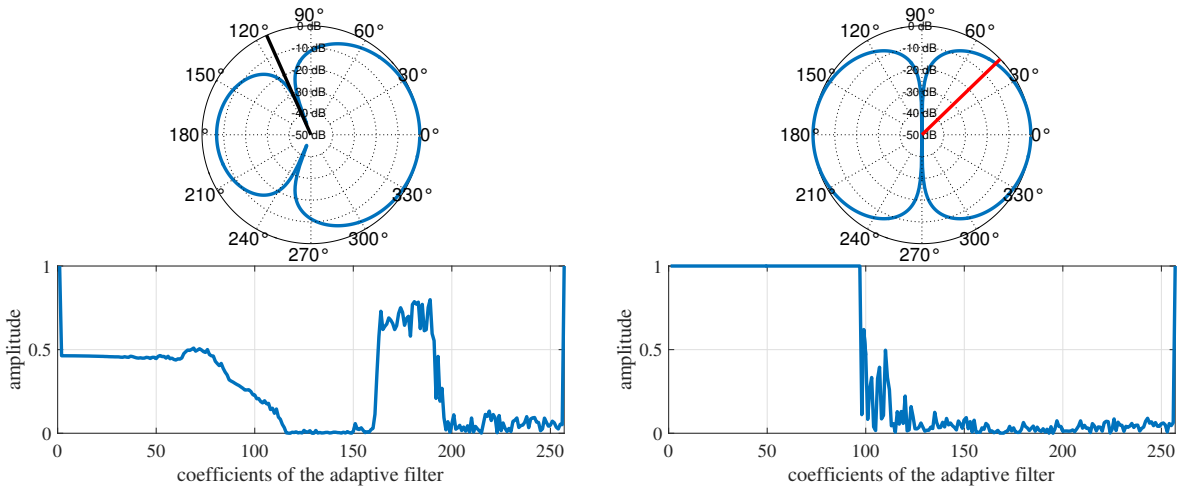


Figure 3.3: Input and output signal of the speaker in the timedomain

The two plots in Fig. 3.4 shows the adaptation of the beampattern for two different timesteps. In Fig. 3.3 the corresponding timestep is marked by the black and red lines at about 1.9 s. In the polar plots the angle of the speaker is depicted by the same colours. The lower plots show the adapted coefficients of the vector β. In Fig. 3.4(b) it can be seen that because of the constraint that was set on the coefficients of beta the overall beampattern can produce no zero in the front half of the array. Only from 90° to 270° the zero can follow the noise speaker.
Using this method various adaptive beamformers like 2nd order designs or hybrid beamformers shown in [9], [12] and [10] can be designed. For a 2nd order adaptive microphone array it would



(a) Beampattern for a speaker position of $\theta_s = 115°$

(b) Beampattern for a speaker position of $\theta_s = 45°$

Figure 3.4: Beampattern and adaptive coefficients for a first-order ADMA with $M = 4$ microphones and MNS patterns, $f = 1031Hz$

be necessary to design an additional toroid and introduce a second adaptation vector. This can be done quite straight forward for the given array structures by simply designing a dipole pattern for a circular array and rotate it to 90° by permutation of the transfer functions. The implementation of higher order designs was not done in this thesis and is only mentioned for complenteness.

4 Acoustic Source Localisation

Like already mentioned in the previous chapters the used planar cirular arrays can be steered electronically in every microphone direction. To automatically steer the mainlobe of the beamformer to a speaking person of interest it is necessary to localise and track the speaker. In this chapter possibilities to do that will be explored.
Based on the idea of a small compact microphone array algorithms that could potentially be used in a real time system will be shown and their advantages and drawbacks regarding complexity, performance will be discussed.

4.1 System Overview

In Fig. 4.1 a possible sound source localisation (SSL) system like described in [1] can be seen. The fact that most SSL algorithms are working in the time domain is fine considering that all fixed beamformers were also designed as filters in the frequency domain so the same blocking, windowing and FFT can be used for beamforming and localisation.
The voice activity detector (VAD) is needed to only do the SSL on frames that contain speech signals. Without the VAD the system would try to localise sources in frames that contain only noise and would generate a lot of useless localisation estimates that can seriously degrade the performance of the final location estimate. Even in contiuous speech the proportion of frames where SSL yields reliable results lies around 30-50%, since pauses are an integral part of human speech [1].
The SSL block in Fig. 4.1 uses the FFT transformed microphone signals to give a location estimate for the current frame. The output of that block can be one or more position estimations, with their confidence levels, or the probability of sound source presence as a function of direction. For the localisation the SSL block uses geometric information of the array to calculate the angle of arrival and if possible also the distance. In the case of differential microphone arrays where the distance between the microphones is very small the estimation of the distance to the speaker is not sensible at all.
Even the sole estimation of the bearing angle can already be challenging because time delay of the impinging wave between microphones is very small. This can be problematic when using a time discrete system since it could happen that the sampling rate of the system is too low and violates the Nyquist-Shannon sampling theorem. In Section 4.4 several algorithms are explored and evaluated since there is always a tradeoff between performance and complexity.
The last block called tracking gathers more frame estimates to stabilize the estimation over

a larger time. This block can depending on the given situation cluster the frame estimates, calculate signal dynamics, movement direction and so on. If more than one speaker is in the room, labeling of the datapoints can happen here to get distinctive tracks of the speakers.

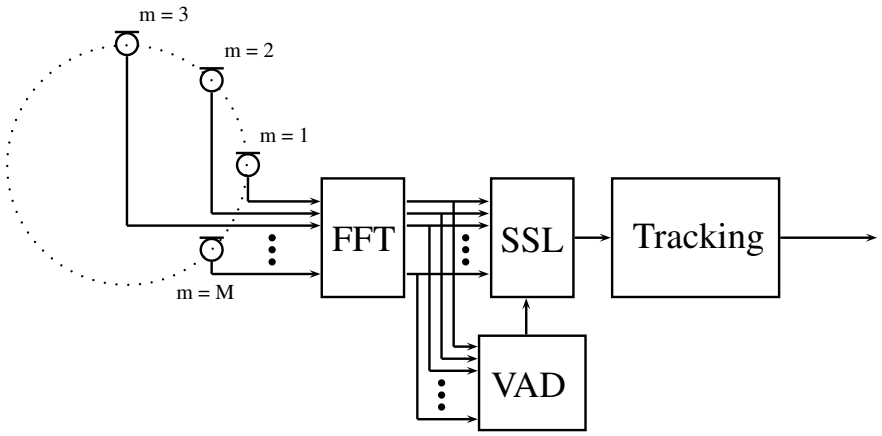


Figure 4.1: Block diagram of a localisation system

4.2 Simulation Setup

For the localisation algorithms the framesize was reduced compared to Section 3.1 because the timeframes were zeropadded to avoid circular convolution when calculating the correlations. So different than in Section 3.1 the spectrogram function was not used here.

- blocklength = 256 samples
- overlap = 50%
- window type = Hanning
- sample frequency = 48 kHz
- FFT length = 512 samples

4.3 Voice Activity Detection

This section shows the VAD used for the localiser. The algorithm used is an adaption of the simple VAD with dual-time-constant integrator shown in [1]. It assumes that the noise floor only changes slowly compared to the speech envelope and uses two different time constants for tracking it. One low when the current level is higher than the estimate, and one high when the level is lower than the estimate

$$L_{min}^{(t)} = \begin{cases} (1 - \frac{T}{\tau_{up}})L_{min}^{(t-1)} + \frac{T}{\tau_{up}}L^{(t)} & L^{(t)} > L_{min}^{(t-1)} \\ (1 - \frac{T}{\tau_{down}})L_{min}^{(t-1)} + \frac{T}{\tau_{down}}L^{(t)} & L^{(t)} > L_{min}^{(t-1)} \end{cases} \tag{4.1}$$

where t is the number of the current frame. $L_{min}^{(t)}$ is the estimated noise floor for the t-th frame, $L^{(t)}$ is the estimated signal level, T is the frame duration and τ_{down} and τ_{up} are the time

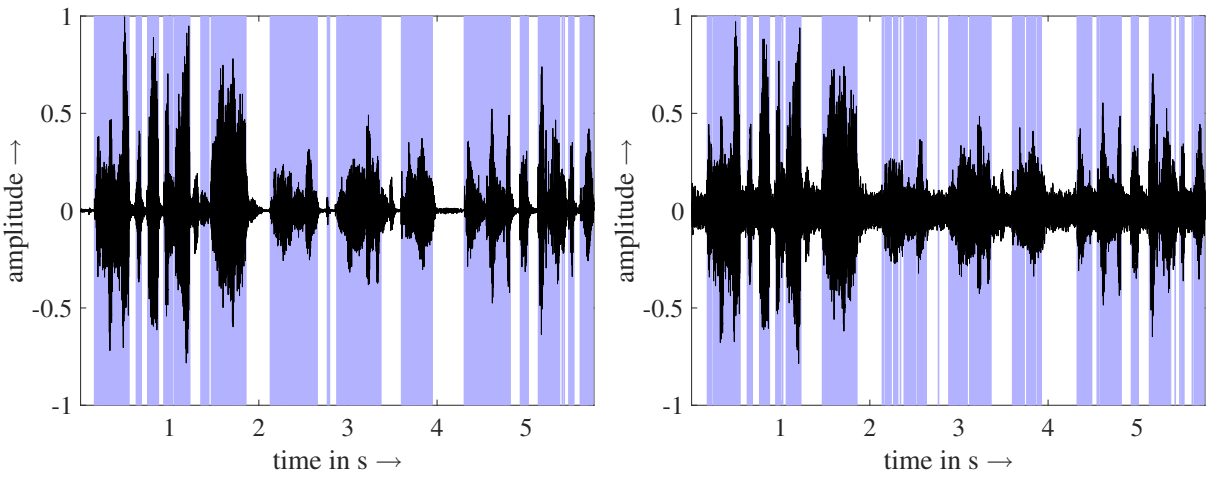
constants for tracking the noise floor level.
The signal level is estimated using a weighted RMS

$$L^{(t)} = \sqrt{\frac{1}{K}\sum_{k=0}^{K-1}(W_k \cdot |Y_k^{(t)}|)^2} \tag{4.2}$$

for every frame. Since the noise energy is usually situated in the lower frequency bands the used weights W_k can just be a high-pass filter. Here the C-message or ITU-T Recommendation O.41 weighting is used. It also supresses the higher frequencies since the speech energy decreases there anyways.
The decision if there is speech or no speech is then made based on the estimated noise floor $L_{min}^{(t)}$, the signal level $L^{(t)}$ and the previous value of the voice activity flag V

$$V^{(t)} = \begin{cases} 0 & \text{if} \quad \frac{L^{(t)}}{L_{min}^{(t)}} < T_{down} \\ 1 & \text{if} \quad \frac{L^{(t)}}{L_{min}^{(t)}} > T_{up} \\ V^{(t-1)} & \text{otherwise} \end{cases} \tag{4.3}$$

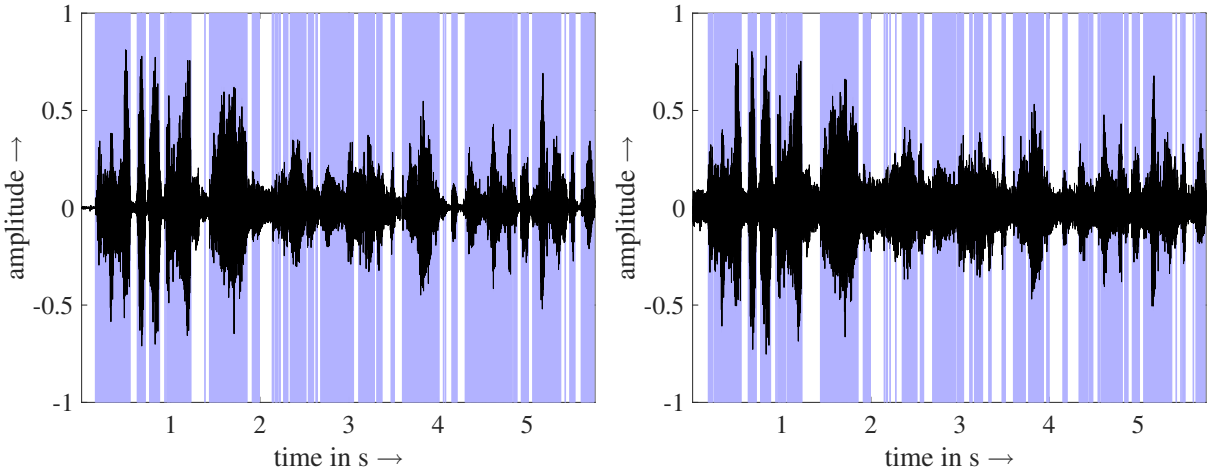
Where T_{down} and T_{up} are the two thresholds for speech and noise. This means that there is a hysteresis to switch between noise and speech states to stabilize the VAD.



(a) $SNR = 60\,\text{dB}$ (b) $SNR = 20\,\text{dB}$

Figure 4.2: Result of the simple VAD for a single speaker

In Fig. 4.2 and Fig. 4.3 the performance of the VAD can be seen. The input signal for the VAD was constructed in a similar way like in Section 3.2. A mono signal was taken convolved with the steering vector of an differential microphone array. This was done for two speakers here and additionally to the steeringvector, white noise was added to the signal to reduce the SNR. As input signal the first microphone of the simulated array was used.
Fig. 4.2 shows a single speaker with high and low SNR. Even in the case of $SNR = 20\,\text{dB}$ the detection of the speech frames seems to work quite nice. The plots in Fig. 4.3 show the detection of speech frames for two different speakers. Here the speech detection also seems to work fine for most speech frames. In Fig. 4.2 it can be seen that for some low energy cases the ends of the words are not detected by the VAD. Further the same can happen for low energy consonants at the beginning of word because of the hysteresis in Fig. 4.3. Since it is better to reject frames that could potentially yield erroneous localisation estimates this is not really a problem. It is more important here to reject frames with noise only than to do localisation of every speech frame.



(a) $SNR = 60\,\text{dB}$ (b) $SNR = 20\,\text{dB}$

Figure 4.3: Result of the simple VAD for two speakers

Especially if the speech frame is low energy anyways so that SSL won't work well anyways. The used parameters for the VAD were

- $\tau_{up} = 10.0$
- $\tau_{down} = 0.008$
- $T_{down} = 3.2$
- $T_{up} = 3.5$

4.4 Bearing estimation

In this section several possibilities to estimate the direction of a speaker (bearing estimation) will be presented. All algorithms will be tested with synthesised data to check their ability to work with microphone arrays with small apertures. The microphone distance is a essential limiting factor when doing SSL.
Since the distance between the array elements gets as low as $\delta = 1\,\text{cm}$ for the considered geometries the resolution of the SSL will be severly impacted. A soundwave traveling with $c = 343\,\text{m/s}$ at room temperature needs $t = 29.15\,\mu\text{s}$ to travel from microphone to microphone. This corresponds to only 1.39 samples when sampling with $fs = 48\,000\,\text{Hz}$.
Since δ is only the microhone distance between one pair of microphones, the maximal usable microphone distance will be larger because of the circular geometry depending on the amount of microphones used. But still the estimation of the time delays between singular sensors will be extremely inaccurate and localising in general will only be viable using a sufficient amount of microphones.

In Fig. 4.4 the direction of arrival is estimated using two microphones at a distance δ. If the time delay of the impinging soundwave from microphone to microphone can be measured the direction of arrival can be estimated as

$$\theta = \arcsin \frac{\tau_D \cdot c}{\delta} \tag{4.4}$$

where τ_D is the time delay between the two microphones. Since for differential microphones the sound source is always assumend to be in the far-field, $\rho >> \delta$. So the problem of estimating the

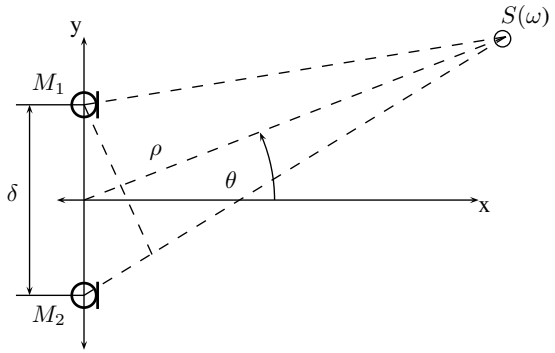


Figure 4.4: Time delay estimation with 2 microphones based on [1]

angle of arrival is converted to one of measuring the time delay between the two microphones with known position.
To do this the generalized cross-correlation introduced in [13] is used.

$$\boldsymbol{R_{12}} = iFFT[\boldsymbol{\Psi} \cdot \boldsymbol{G_{X_1X_2}}] \tag{4.5}$$

where

$$\boldsymbol{G_{X_1X_2}} = \boldsymbol{X_1X_2^H} \tag{4.6}$$

is the cross-power spectral density function of the two microphone signals.
For the weighting function $\boldsymbol{\Psi}$ there are several possibilities, all optimal in one or another way, many of them are discussed in [1]. Here the very commonly used phase transform (PHAT) weighting is used.

$$\Psi_{PHAT}(f) = \frac{1}{|G_{X_1X_2}(f)|} \tag{4.7}$$

It just eliminates the magnitude from the spectrum and gives equal weight to the phases in each bin. After the PHAT weighting we should see the sharp peaks in the spectrum making it practical for detecting multiple sources or working in a reverberant environment [1]. With one microphone pair only sources in a range from $\theta = [-90°, +90°]$ can be detected. Also there are ambiguities between the front and the back of the array, which is true for all linear microhone arrays.
Since the considered arrays all have planar circular geometries it is possible to combine the results of all the microphone pairs and get a localisation estimate for 360°, or if using enough microphones even the elevation. Although the case of estimating elevation angles has been left out here for simplicity.
Using the signal model already introduced in Fig. Section 2.1 in the next chapters some algorithms to combine all microphones will be discussed.

4.4.1 TDE Interpolation

The first algorithm used for combining the microphone pairs presented in [1] is taken from [14] where the authors use a planar four-element array for localising sources in the upper hemisphere. The geometry used there is very similar to the ones considered in this these except for the coordinate system and the used sensor spacing.
Since the smallest distance between the microphones is very small and time delay estimation

can get difficult, an adaptation of the GCC-PHAT proposed in [15] was used here.

$$\Psi_{PHAT}(f) = \frac{1}{|G_{X_1X_2}(f)|^{\lambda}} \tag{4.8}$$

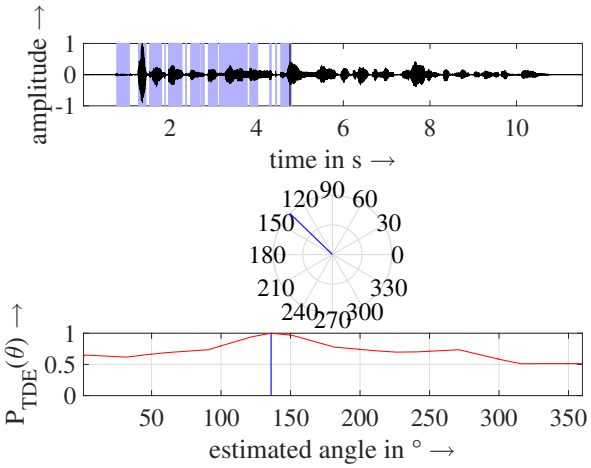
The idea here is to introduce an SNR dependent exponent λ that reduces the impact of the magnitude elimination of the PHAT weighting. It is shown in [15] that the detection of maxima can be improved for high SNRs with this exponent.
The modified GCCs calculated like this are then converted to functions of azimuth and elevation to get a hypothesis vector $h_i(\theta, \gamma)$ for every unique microphone pair. The index i describes the microphone pair and the indices θ and γ are the discrete latitudes and longitudes of the hemisphere.
Using linear interpolation an estimate for every value of θ and γ is computed from the GCCs. The final estimate of the direction is then calculated by getting the maximum of the summed hypotheses.

$$(\theta, \gamma) = \arg\max_{\theta,\gamma} \left(\sum_{i=1}^{\frac{M(M-1)}{2}} h_i(\theta, \gamma) \right) \tag{4.9}$$

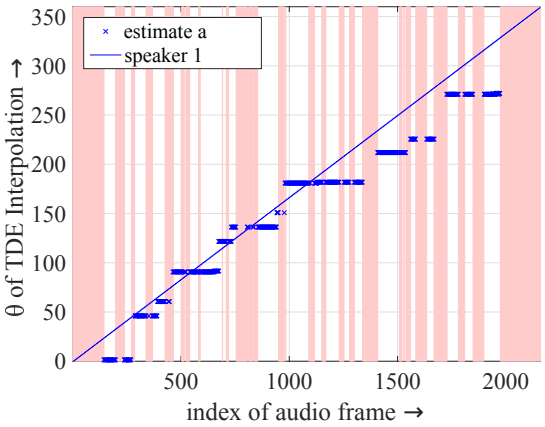
In our case the localisation was done only in the plane of the array so the parameter γ is not estimated.
There are two major drawbacks with this algorithm. One is the required computational power that is needed, since for every frame one inverse fourier transform has to be calculated for every microphone pair. The other is that the resolution of the GCCs is not very high. So the computation at low microphone distances will need high sampling frequencies and interpolation both increasing the computational load and the interpolation can introduce errors.
Simulations were done with MATLAB to evaluate this algorithm for the chosen array geometries. The setup is about the same as for simulating the adaptation of the CADMAs and the VAD.
The speaker signals are mono wav files that are convolved with the steering vector of an array to get the multichannel output of the array. The unique microphone pairs were computed using [16]. In this case the scenario was done for one and two speakers to also test the localisation abilities of the algorithms in multispeaker scenarios. Again there are no room reflections in the simulation and the convolution implies a geometric prefectly build array with spectral flat, omidirectional microphones. This won't happen in real-world scenarios so the localisation will not work as well as in the simulations. The maxima of the localisation functions were retrieved using the peak picker function [17]. This peak finder additionally does a thresholding to supress low power peaks. The value of the threshold was set to thresh = 0.05 for the normalised localisation functions.

In Fig. 4.5 the simulation results for one speaker and an array of $M = 4$ microphones with $\delta = 0.01\,\mathrm{m}$ can be seen. The upper part Fig. 4.5(a) shows the signal in the time domain at a speaker position of 150°. The blue colored parts are the frames that were classified by the VAD as speech frames. In the polar plot the blue line indicates the estimated angle of the target speaker. The lower part of the plot shows the result of Fig. 4.9 normalised by its maximum value, again the blue line in the plot is the estimated position of the speaker. As shown in the lowest plot the resolution of this algorithm is not very high.
This can also be seen in Fig. 4.5(b) where the result of all localisation estimates over time is plotted. There the ground truth of the speaker and the estimates are drawn. The red parts in the plot are the frames that contain no speech signal. It can be seen that because of the low angular resolution of this algorithm the detected angles can deviate quite a lot from the ground truth.

The plots in Fig. 4.6 show the results for the same array as in Fig. 4.5 but for two speakers.

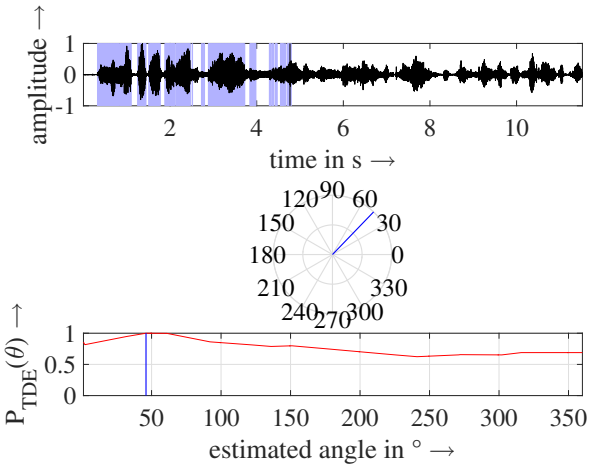


(a) Time domain signal, estimated angle and detection function at $\theta_s = 150°$

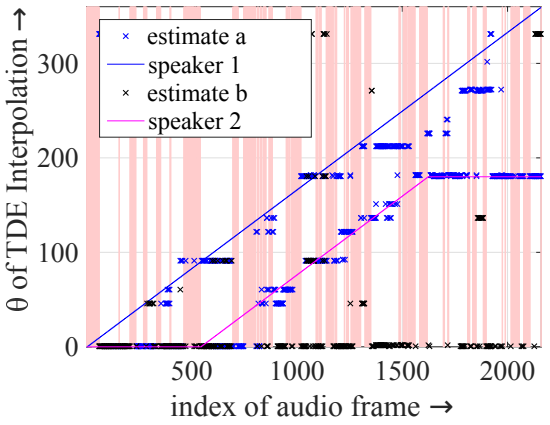


(b) Picked maxima and ground truth

Figure 4.5: TDE interpolation result for one speaker moving around the array, $M = 4$



(a) Time domain signal, estimated angle and detection function at $\theta_{s1} = 150°$, $\theta_{s2} = 60°$



(b) Picked maxima and ground truth

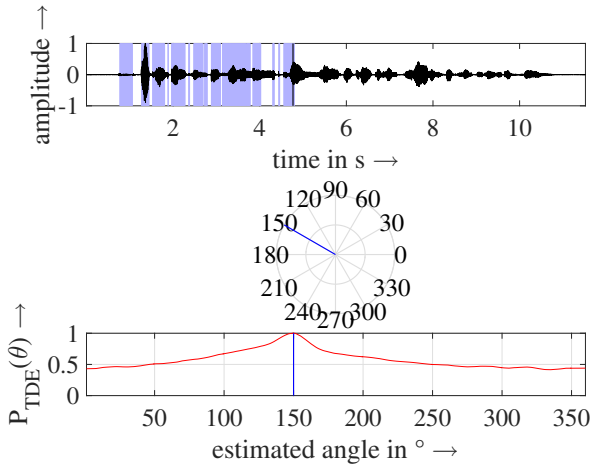
Figure 4.6: TDE interpolation result for two speakers moving around the array, $M = 4$

The amount of wrong detections gets a lot higher than for one speaker. This can also be seen in the polar plot of Fig. 4.6(a) where the estimate of the second speaker should be seen in magenta. But at the considered timeslot of the signal there is only one distinctive peak of $P_{TDE}(\theta)$ so the peak used peak picking function could not identify a second speaker.

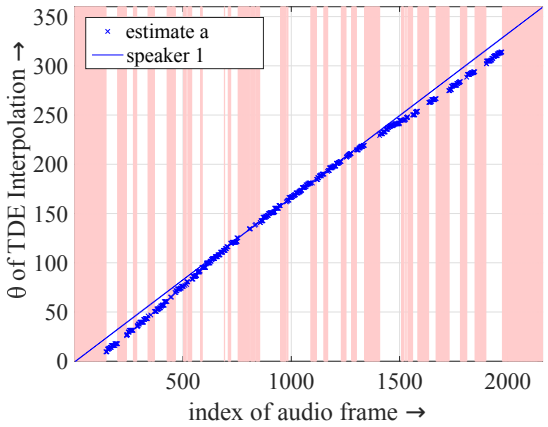
In Fig. 4.7 the results for one speaker and an array with $M = 12$ microphones can be seen. Here the SSL works already very well since the amount of unique microphone pairs that can be used for the estimation is very high. The detection function plot in Fig. 4.7(a) also shows that the angular resolution is a lot better and the maximum is also more distinct as for the 4 microphone array.

At last a simulation for the microphone array with $M = 12$ microphones and two speakers was done. As the plots in Fig. 4.8 show, even if the amount of false detections is higher compared to the single speaker the two tracks of the speakers are already clearly visible in Fig. 4.8(b). This is mainly because the two speakers are not always talking at the same time and so by picking the highest two maxima the two speakers can already be detected.

The algorithm is very dependent on the amount of microphones, the sensor spacing and is

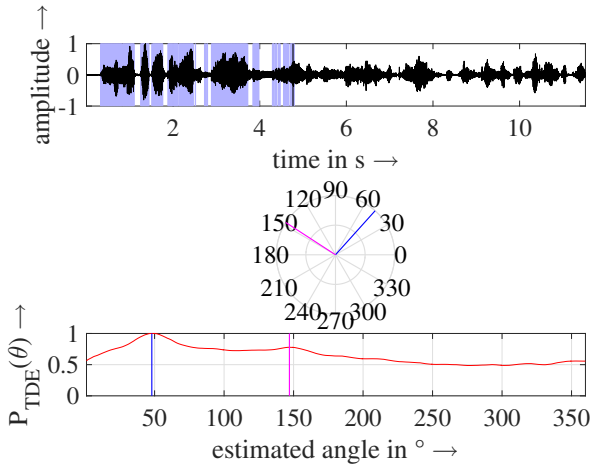


(a) Time domain signal, estimated angle and detection function at $\theta_s = 150°$

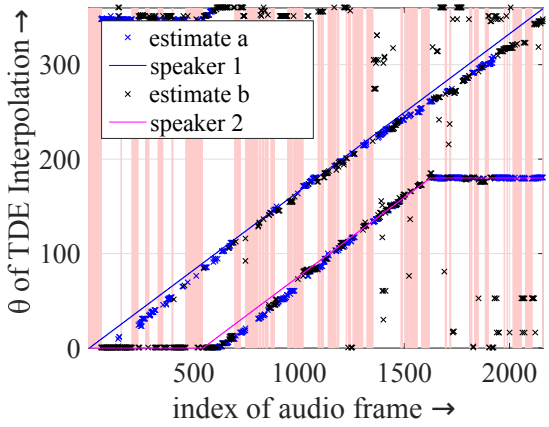


(b) Picked maxima and ground truth

Figure 4.7: TDE interpolation result for one speaker moving around the array, $M = 12$



(a) Time domain signal, estimated angle and detection function at $\theta_{s1} = 150°$, $\theta_{s2} = 60°$



(b) Picked maxima and ground truth

Figure 4.8: TDE interpolation result for two speakers moving around the array, $M = 12$

not really meant to track multiple speakers, it is not sensible to use this method for tracking simultaneously active targets. Also in all of the simulated examples a seemingly systematic error can be seen that reaches its maximum at $\theta_s = 0/360°$. This is most likely because for a single pair of microphones as in Fig. 4.4 the resolution of the localisation is not uniform over the whole possible search range of 180°. The resolution is worst at ±90°. The bias to negative errors could be expained by the use of the floor function when computing the linear interpolated hypothesis functions.

4.4.2 SRP BF/ SRP PHAT

Another type of SSL methods discussed in [1] are the steered-response power (SRP) algorithms. Those types of algorithms form a conventional beam, scan it over the appropriate region of the working space and plot the magnitude squared output. The maxima of the resulting function are then the resulting locations of the speakers.

Other than the TDE-interpolation in Section 4.4.1 this type of algorithms are defined in only one frequency bin. To get a broadband estimate of the speaker location it is necessary to combine several computed frequency bins. Here the PHAT weighting 4.7 will be applied to the SRP to make the peaks in the detection function more distinct.

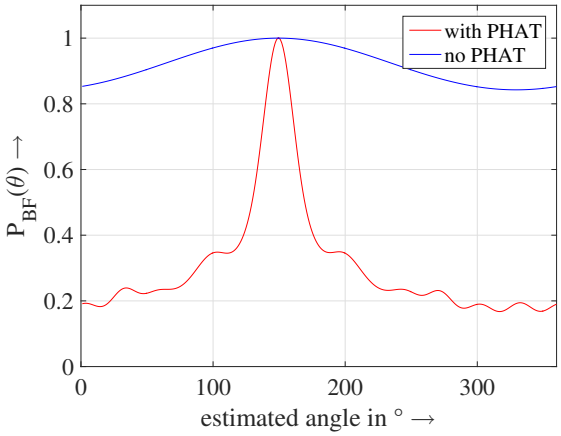


Figure 4.9: SSL with and without PHAT weighting

In Fig. 4.9 a comparison of SRP with and without PHAT weighting can be seen. It is clear that with the weighting the detection of the maximum becomes a lot easier since the peak is a lot narrower.
As proposed in [1] the cross-power matrix of the input signal is averaged over $N = 10$ frames to increase the stability of the estimation. So the localisation function can be given for a single frequency bin as

$$P_{BF}(\theta) = \boldsymbol{D}(\theta)^H \boldsymbol{S} \boldsymbol{D}(\theta) \tag{4.10}$$

where $\boldsymbol{D}(\theta)$ is computed from the steering vector 2.3. $\boldsymbol{S}$ is the averaged cross-power matrix of the input samples.

$$\boldsymbol{S} \equiv \boldsymbol{X}\boldsymbol{X}^H \tag{4.11}$$

with

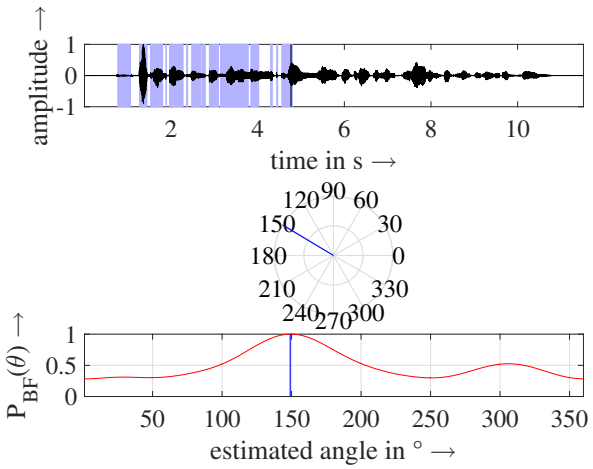
$$\boldsymbol{X} = \frac{1}{\sqrt{N}} \begin{bmatrix} X_1^{(n)} & X_1^{(n-1)} & \cdots & X_1^{(n-N+1)} \\ X_2^{(n)} & X_2^{(n-1)} & \cdots & X_2^{(n-N+1)} \\ \vdots & \vdots & \ddots & \vdots \\ X_M^{(n)} & X_M^{(n-1)} & \cdots & X_M^{(n-N+1)} \end{bmatrix} \tag{4.12}$$

Since the algorithm above is only given for one frequency bin the frequecy index was omitted there. With the already mentioned applied PHAT weighting the detection function is given as

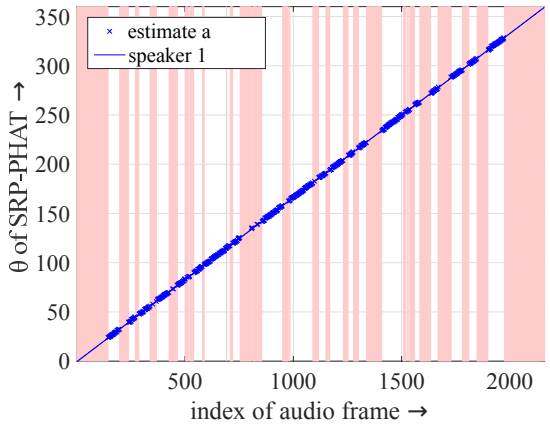
$$P_{SSL}(\theta) = \frac{1}{K} \sum_{k=1}^{K} \frac{M}{\boldsymbol{X}_k^H \boldsymbol{X}_k} P_{BF}(\theta, k) \tag{4.13}$$

where k is the frequency index and X_k is here the vector of averaged input samples used for the calculation of $P_{BF}(\theta, k)$.
In the following plots the same simulation setup was used as in Section 4.4.1 to evaluate the localisation performance for the proposed array geometries.
The capturing matrix $\boldsymbol{D}(\theta)$ that was used to calculate the localisation function 4.10 was calculated for a number of 360 angles, so the angular resolution is very high. For practical solutions

the amount of angles that are used for the localisation has to be reduced to save computational power. Also the localisation was done for the whole available frequency range for $fs = 48\,\text{kHz}$ so all frequency bins up to the nyquist frequency of 24 kHz are used to calculate 4.13. Since most of the speech energy is in the lower frequecy band this could also be reduced to save CPU time.



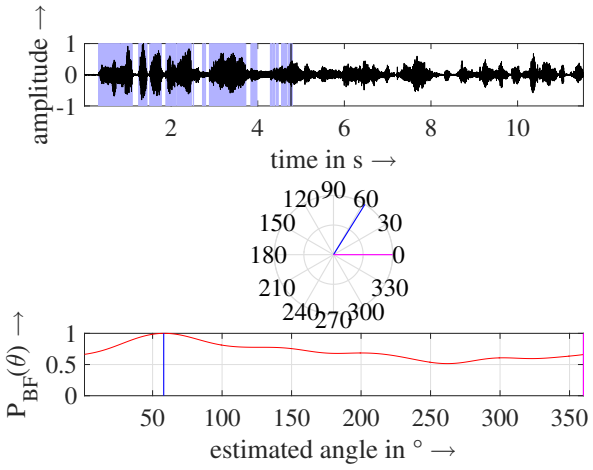
(a) Time domain signal, estimated angle and detection function at $\theta_s = 150°$



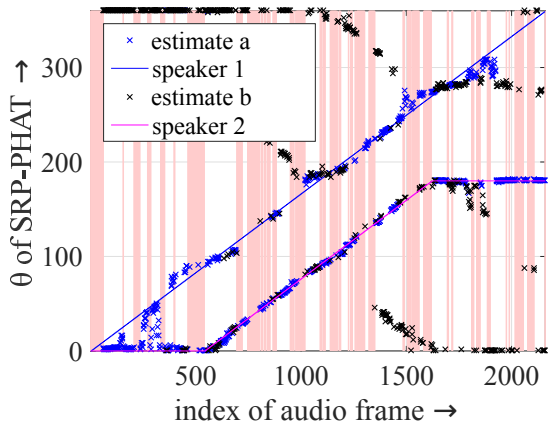
(b) Picked maxima and ground truth

Figure 4.10: SRP-PHAT result for one speaker moving around the array, $M = 4$

In Fig. 4.10 the results of the simulation for one speaker moving around the array can be seen. The plot in Fig. 4.10(b) shows that the localisation for a single speaker works very well already for the 4 microphone array with $\delta = 10\,\text{mm}$.
Also the localisation function in Fig. 4.10(a) shows a very distincitve peak that can be used for picking the maximum.



(a) Time domain signal, estimated angle and detection function at $\theta_{s1} = 150°$, $\theta_{s2} = 60°$

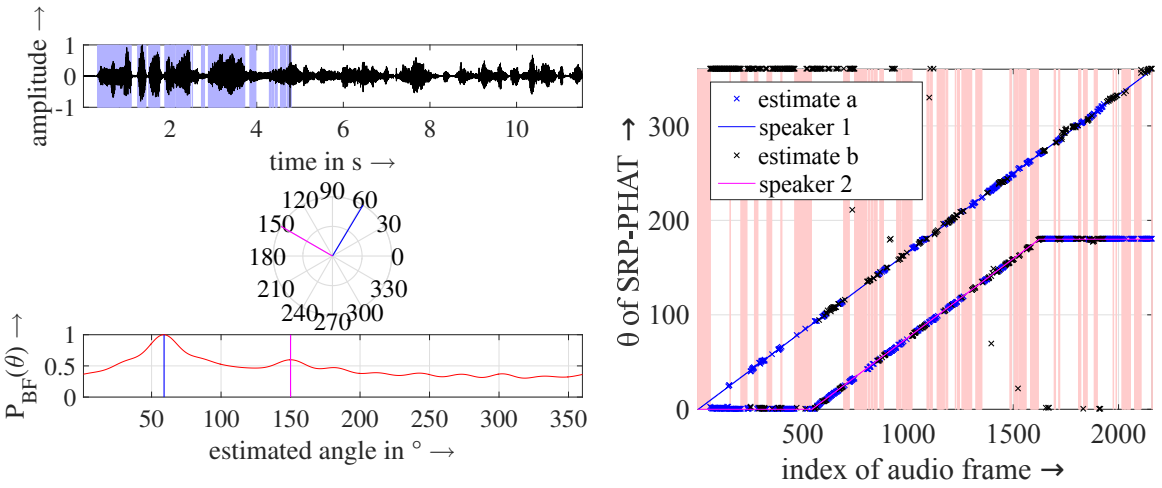


(b) Picked maxima and ground truth

Figure 4.11: SRP-PHAT result for two speakers moving around the array, $M = 4$

For two speakers and an array of $M = 4$ microphones the SSL still yields quite a lot of erroneous detections as can be seen in the plots of Fig. 4.11. Since the maximum of the second speaker is often not very high in the localisation function Fig. 4.11(a) the second picked maximum

is often at the wrong position.



(a) Time domain signal, estimated angle and detection function at $\theta_{s1} = 150°$, $\theta_{s2} = 60°$

(b) Picked maxima and ground truth

Figure 4.12: SRP-PHAT result for two speakers moving around the array, $M = 12$

In Fig. 4.12 the simulation for an array with $M = 12$ microphones is shown. Here the SSL already works pretty well also for two speakers. Even if the second maximum of the considered localisation function in Fig. 4.12(a) is not very high, the picked maxima are the exact postions of the two speakers. The plot of all detections in Fig. 4.12(b) shows that there are only few false localisations. The additional points at the upper left side of the plot seem to be just detections that jump between 0° and 360° from the second speaker who is standing at 0°.
The simulations for a single speaker and an array with 12 microphones are left out here since the SSL already works very well for 4 microphones and the results are not very different from the plots in Fig. 4.10, the localisation function has a more distinct peak because more microphone pairs are used for the computation.

4.4.3 SRP MUSIC

The last SSL algorithm that will be discussed here is the multiple signal classification (MUSIC) algorithm proposed in [18]. Provided that the number of sound sources that should be localised is known, this algorithm can provide a localisation function with very sharp peaks even at small sensor distances. In [19] it has been shown that using MUSIC it is possible to do SSL for arrays with very small apertures even smaller than $\delta = 1\,\text{cm}$. There the localised sources exclusivly were vehicles in a non-closed environment. Still it seems like MUSIC can provide a viable solution for SSL for arrays with small microphone distances.
Given J sound sources F_J the sound capturing equation is given as in [1]

$$\begin{bmatrix} X_1 \\ X_2 \\ \vdots \\ X_M \end{bmatrix} = \begin{bmatrix} \boldsymbol{D}(\theta_1) & \boldsymbol{D}(\theta_2) & \cdots & \boldsymbol{D}(\theta_J) \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ \vdots \\ F_J \end{bmatrix} + \begin{bmatrix} N_1 \\ N_2 \\ \vdots \\ N_M \end{bmatrix} \tag{4.14}$$

with $\boldsymbol{D}_j$ beeing the capturing vector of the source locations. The spectral matrix of the input vector $\boldsymbol{X}$ can than be written in terms of eigenvalues and eigenvectors as

$$\begin{aligned} \boldsymbol{S}_x &\equiv \boldsymbol{X}\boldsymbol{X}^H \\ \boldsymbol{S}_x &= \sum_{i=1}^{N} \lambda_i \boldsymbol{\Phi_i}\boldsymbol{\Phi_i}^H = \boldsymbol{\Phi_i}\boldsymbol{\Lambda}\boldsymbol{\Phi_i} \end{aligned} \tag{4.15}$$

where $\Lambda = \mathrm{diag}[\lambda_1, \lambda_2, ..., \lambda_N]$. If the eigenvalues are sorted in decreasing size, then the first eigenvalues correspond to the signal subspace eigenvalues and the first eigenvectors are the signal subspace eigenvectors

$$\boldsymbol{U_S} \equiv \begin{bmatrix} \boldsymbol{\Phi_1} & \boldsymbol{\Phi_2} & \cdots & \boldsymbol{\Phi_J} \end{bmatrix} \tag{4.16}$$

The rest of the eigenvectors define the noise subspace

$$\boldsymbol{U_N} \equiv \begin{bmatrix} \boldsymbol{\Phi_{J+1}} & \boldsymbol{\Phi_{J+2}} & \cdots & \boldsymbol{\Phi_N} \end{bmatrix} \tag{4.17}$$

So the steering function is given as

$$P_{mus}(\theta) = \frac{1}{\boldsymbol{D}(\theta)^H \boldsymbol{U}_N \boldsymbol{U}_N^H \boldsymbol{D}(\theta)} \tag{4.18}$$

or

$$P_{mus}(\theta) = \frac{1}{\boldsymbol{D}(\theta)^H [\boldsymbol{I} - \boldsymbol{U}_S \boldsymbol{U}_S^H] \boldsymbol{D}(\theta)} \tag{4.19}$$

The largest J maxima of P_{mus} are then the locations of the sound sources. Compared to the previous methods for SSL here the peaks for the estimation are very sharp. Even in the case of two speakers Fig. 4.13 the localisation works very well already for 4 microphones and small microphone distances $\delta = 10\,\mathrm{mm}$.
The only drawback is that in general the number of speakers has to be known so that the number of eigenvectors needed for the calculation of 4.19 is known. The number of source (NOS) could eventually be estimated from the eigenvalue profiles as proposed in [20]. There the authors estimate the number of eigenvalues and reach a SSL performance similar to if the number of sources is known.
For this thesis the number of similar speaking sources is assumed to be known so no further investigation was done into the estimation of the number of active sound sources.
Since the peaks in the localisation function are already very sharp no further weigthing was done when the frequency bins were combined. The localisation functions of different frequency bins were simply averaged.

In Fig. 4.14 and Fig. 4.15 the results for SSL with the MUSIC algorithm can be seen for a 4 microphone array.
It can clearly be seen that compared to the previous SSL algorithms MUSIC is working best for small sensor distances compared to TDE-Interpolation and SRP-PHAT.

Given one or two speakers the estimation is working very well in the simulations. Because of the very sharp peaks, tracking of multiple sound sources should be no problem with this algorithm. The results for the 12 microphone array are not given here since the SSL does not improve much more there. In the case of 2 speakers the erroneous localisations become even less but as can be seen in 4.15(b) the tracks of the speakers are already very distinct.

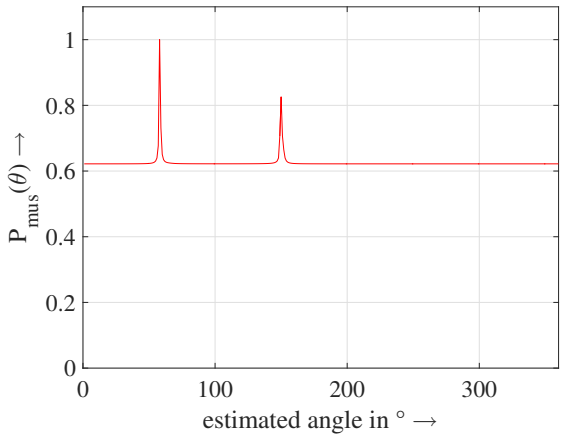
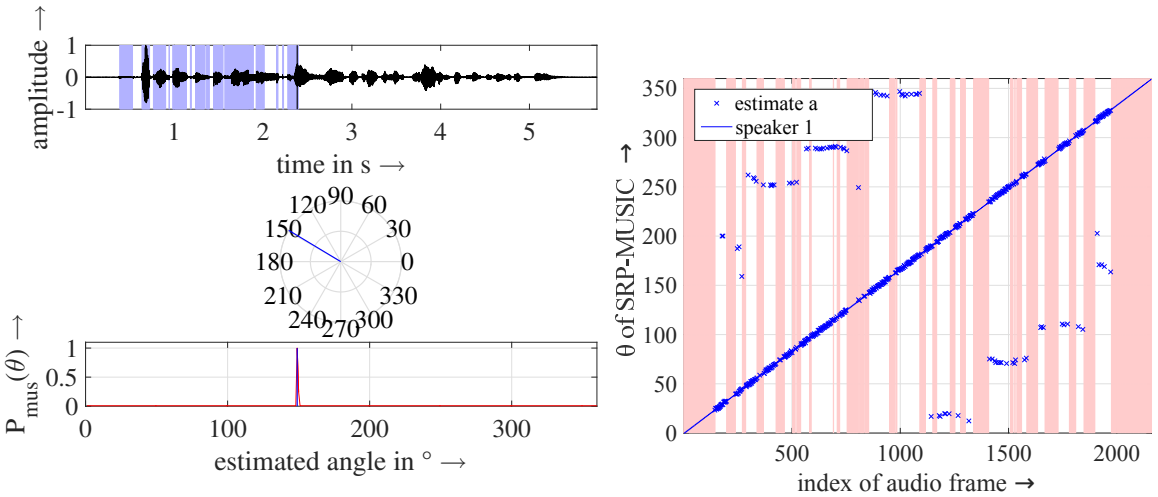


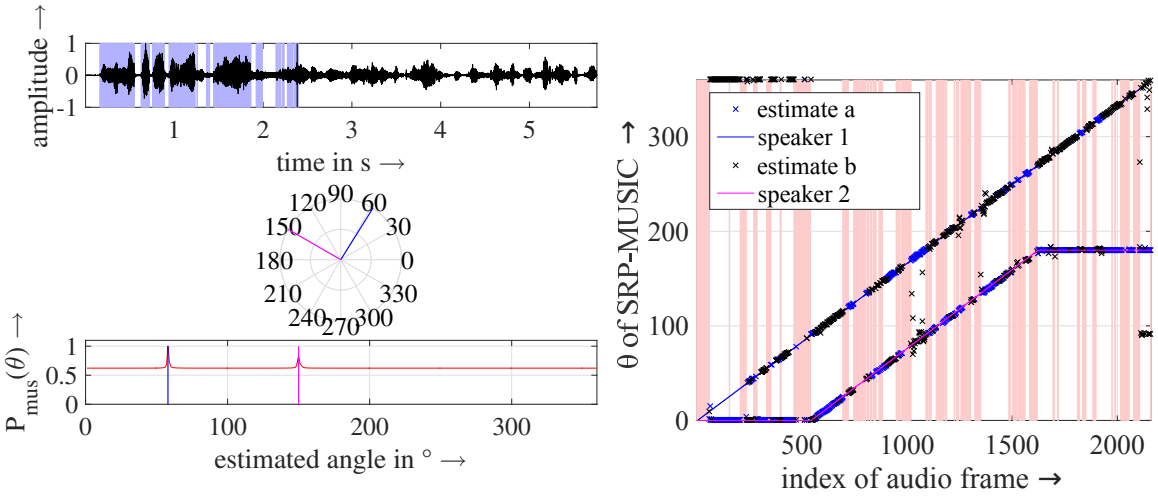
Figure 4.13: $P_{mus}(\theta)$ at speakers at 150° and 60°



(a) Time domain signal, estimated angle and detection function at $\theta_s = 150°$

(b) Picked maxima and ground truth

Figure 4.14: SRP-MUSIC result for one speaker moving around the array, $M = 4$



(a) Time domain signal, estimated angle and detection function at $\theta_{s1} = 150°$, $\theta_{s2} = 60°$

(b) Picked maxima and ground truth

Figure 4.15: SRP-MUSIC result for two speakers moving around the array, $M = 4$

5 Recording

To evaluate the performance of the proposed beamformers and localisation algorithms, recordings in a real environment were conducted. In this chapter the recording setup is described. The used equipment, test signals and recording parameters are shown and documented.

5.1 Recording Setup

For the recordings the same small conference room (cocktail party room/ CPR) already used in [10] for the realistic scenarios was used. The room hast a reverberation time of $T_{60} = 0.5\,\mathrm{s}$. In Fig. 5.1 the setup can be seen. Eight loudspeakers were placed at a circle with radius $r = 1\,\mathrm{m}$ around the microphone array. Various different scenarios described in detail in Section 5.3.2 were played back and recorded at the same time.
The loudspeakers were mounted at a height of $h_{LSP} = 1.30\,\mathrm{m}$ measured from their bottom. The arrays were mounted at $h_{mic} = 1.38\,\mathrm{m}$ which corresponds to the middle of the lower loudspeaker of the used speakers. To play and record the signals the PC that can be seen in the lower right corner of Fig. 5.1 was used.
During the measurments the window and door of the room were closed and no person was in the room.

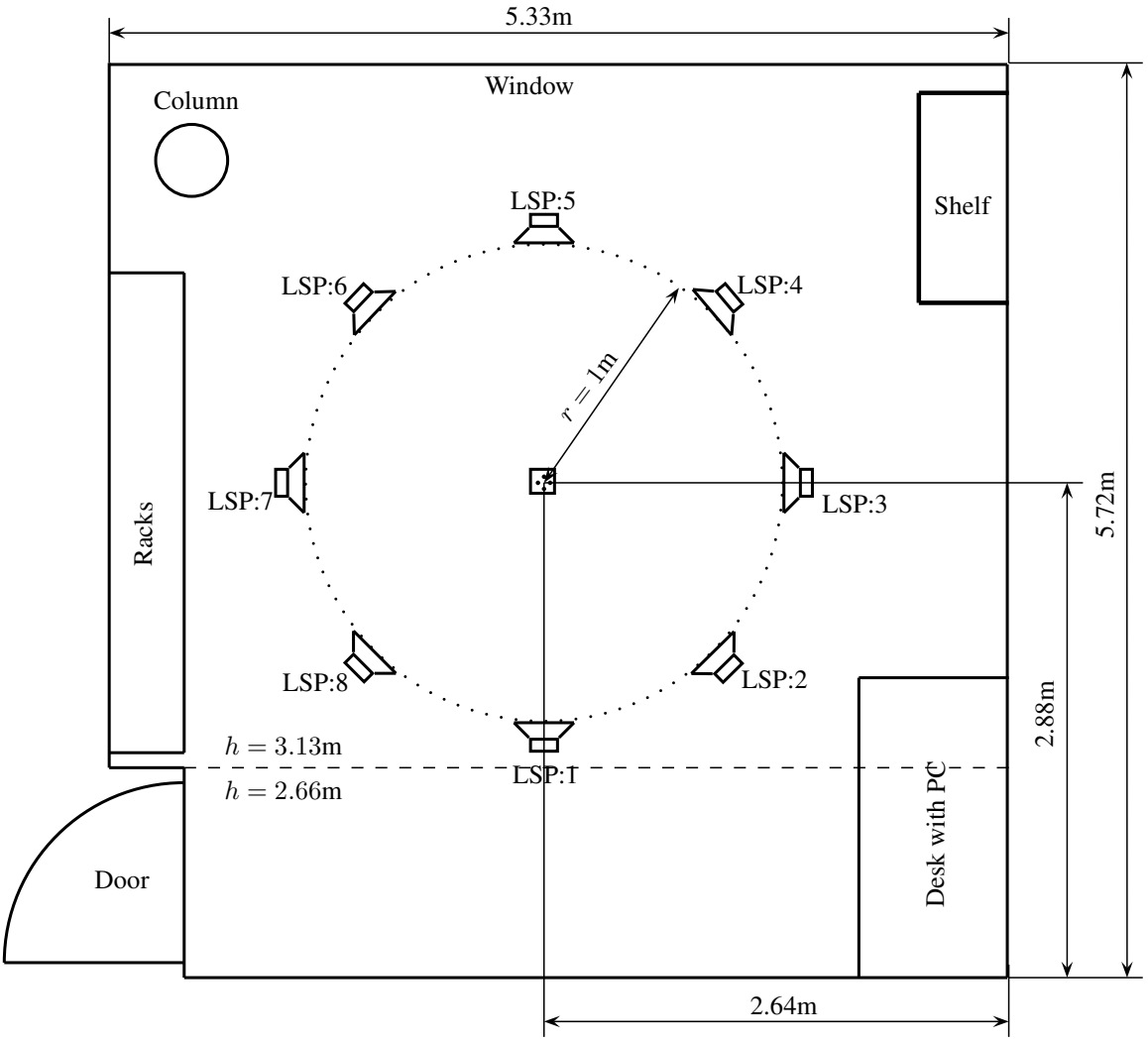


Figure 5.1: Recording setup in the conference room

5.2 Recording Equipment

5.2.1 Playback

For playback eight Yamaha MSP5 Studio loudspeakers were used. They were connected to the outputs of one Focusrite Octopre MKII microphone preamplifier that was connected via ADAT with the recording PC. The soundcard in the PC was a RME HDSPe RayDAT PCI express card. The multichannel audiofiles for playback were generated with MATLAB and automatically played back with a shell script using aplay.

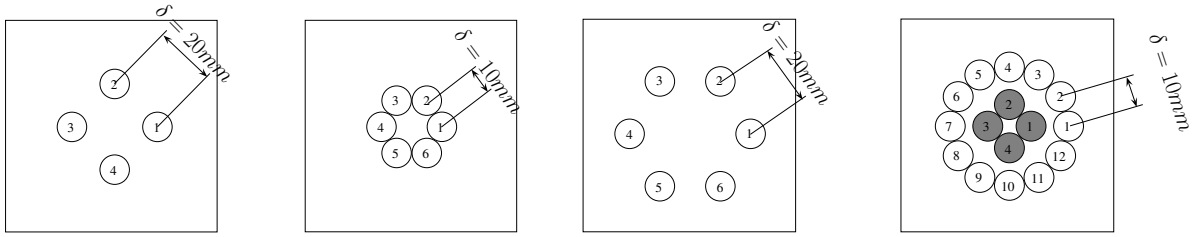
5.2.2 Recording

The recording was done with the same microphone preamplifier that was used for playback. An additional Focusrite Octopre MKII was used since there was a maximum of 12 microphones used for one array. The microphone preamplifiers and the recording PC were synchronized using the wordclock output of the PC.
As microphones Primo EM172 electret condenser microphones were used. They were connected to the preamplifiers via Audix APS-910 phantom power adapters.
The microphone capsules have a diameter of $d = 10\,\text{mm}$ and feature a very high SNR of 80 dB

at 1 kHz. As recording software arecord was used. The recording was started in the same script as the playback so no considerable latency should be between playback and recording. For routing and to ensure constant latency for recording jack audio was used as sound server. The samplerate was set to $fs = 48\,\text{kHz}$, the framesize (Frames/Period) was set to $N = 1024$ and the frame buffer (Periods/Buffer) was set to $K = 2$. With these settings a constant latency of 42.7 ms should be provided by jack audio.

5.2.3 Microphone Arrays Configurations

In Fig. 5.2 sketches of the constructed microphone arrays can be seen. Four different carrier plates were constructed to hold the microphone capsules. The dimensions of the mounting plates are 8 cm × 8 cm with a thickness of 5 mm. With these array geometries all proposed beamformers from the previous chapters were evaluated.



(a) 4 microphones $\delta = 20\,\text{mm}$

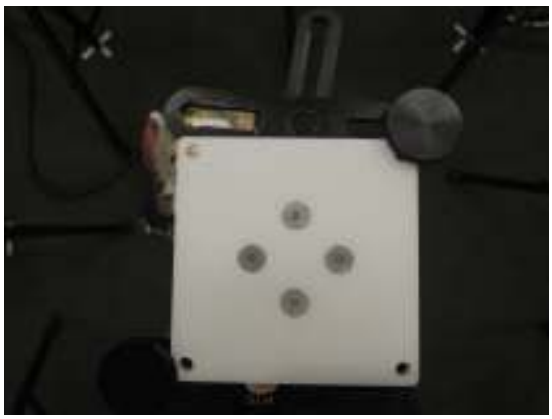
(b) 6 microphones $\delta = 10\,\text{mm}$

(c) 6 microphones $\delta = 20\,\text{mm}$

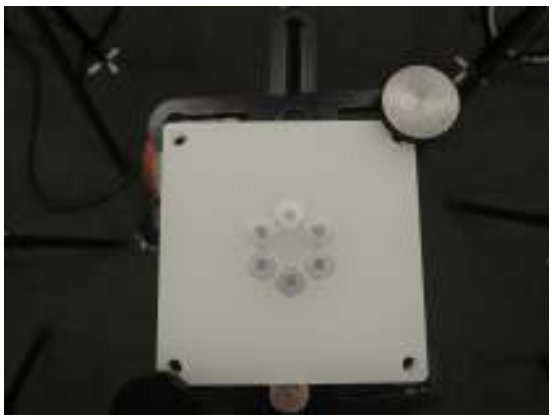
(d) 4 and 12 microphones $\delta = 10\,\text{mm}$

Figure 5.2: Designed CDMAs

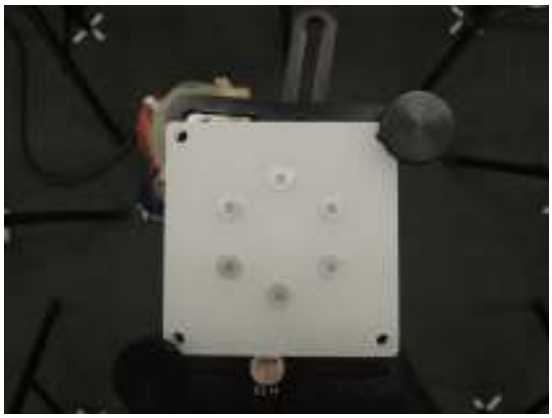
The pictures in Fig. 5.3 show the mounted arrays. For recording the arrays were fixed on a microphone stand using three stereo mount bars. This way the arrays could be mounted at the same height of the loudspeakers without masking any of the loudspeakers.



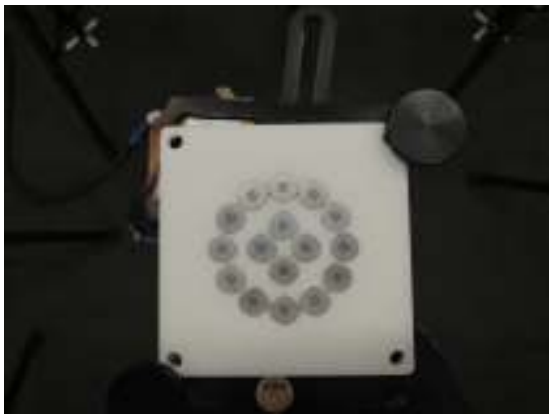
(a) 4 microphones $\delta = 20\,\text{mm}$



(b) 6 microphones $\delta = 10\,\text{mm}$



(c) 6 microphones $\delta = 20\,\text{mm}$



(d) 4 and 12 microphones $\delta = 10\,\text{mm}$

Figure 5.3: Designed CDMAs

5.3 Recordings

(a) Recording setup in conference room

(b) Detail of mounted microphone array

Figure 5.4: Recording secenario in conference room

5.3.1 Calibration

To calibrate the output level of the loudspeakers white gaussian noise generated with audacity [21] was played back at every loudspeaker. The sound preassure level was measured using a NTI Audio: XL2 sound level meter. The level of the loudspeakers was adjusted to reach $L_{A,eq} = 80\,\text{dB}$ at the middle of the microphone array. Measured results can be seen in Tab. 5.1.

LspNum	$L_{A,eq}$
1	80.0 dB
2	79.9 dB
3	80.7 dB
4	79.8 dB
5	80.5 dB
6	79.8 dB
7	79.8 dB
8	79.8 dB

Table 5.1: Measured SPL for loudspeaker calibration

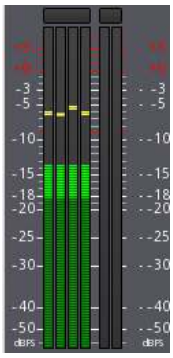


Figure 5.5: Calibration for the 4 microphone array with $\delta = 10mm$

To calibrate the microphone gain of the preamplifiers white gaussian noise was played at the loudspeakers. The input of the microphones was monitored with the ardour [22] metering tool and the microphones were adjusted to reach a level of $-14\,\text{dBFS}$. In Fig. 5.5 a screenshot of the microphone calibration for the 12 microphone array can be seen. The settings of the metering tool were set to measure RMS+Peak and the decay rate was set to slowest to make the manual gain adjustment easier.

5.3.2 Test Signals

Four different scenarios were created using data from the GRASS corpus [23] and multichannel files were created with MATLAB. In a part of the Grass corpus the participants were told to describe a picture in their own words. These recordings last for about 2 min and contain mostly continuous, spontaneous speech. Those files were taken and split into segments of 10s and distributed to the loudspeakers.

- scenario 0: one speaker walks around the array clockwise
- scenario 1: one speaker is fixed at $\theta_{s1} = 0°$, one speaker is walking around the array clockwise
- scenario 2: the same as scenario 1 but with an additional speaker walking around the array anti-clockwise
- scenario 3: three speakers are continously talking at $\theta_{s1} = 0°$, $\theta_{s1} = 135°$ and $\theta_{s1} = 225°$ for 15 s

To avoid unwanted noise when switching loudspeakers the beginning and the end of the 10 s segments were multiplied with a 25 ms long ramp function.

Five different scenarios were created and played back using the speakers in Tab. A.1 from the GRASS corpus. A detailed list of the used speakers can be found in appendix A.

6 Results

In this chapter the results of the recordings are presented. The noise supression performance of the proposed beamformers is evaluated by simply measuring the energy reduction of the recorded signal in different directions.
The performance of the localisation will be determined by comparing the localised data to the expected ground truth of the speaker positions.

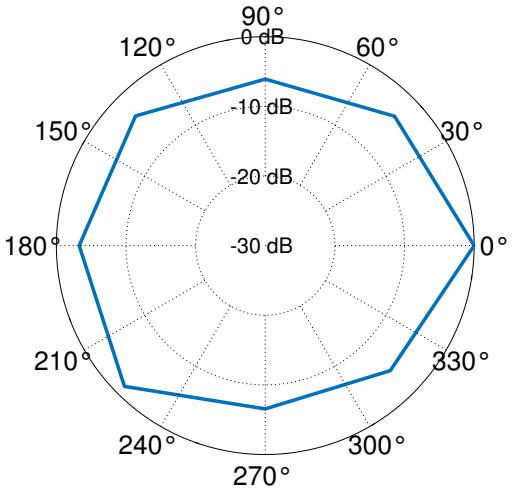
6.1 Beamformers

To evaluate the noise supression capabilities of the adaptive beamformer the angle dependent beamformer output was evaluated. Adaptive beamforming like described in chapter Section 3.2 was done for scenario 0. The time slots of different loudspeaker angles were split and their energy was calculated for the 10 s when the signal was at a single speaker. Then the ratio between main direction and the other angles is calulated. To ensure that there is no overlap from the previous speaker due to latency of the recording system 50 ms of the 10 s signal parts were cut off, in the beginning and the end of the audio frame.
The energy values were averaged over all recordings to get results for every beamformer. Three different design methods for the used fixed beamformer were tested and evaluated.

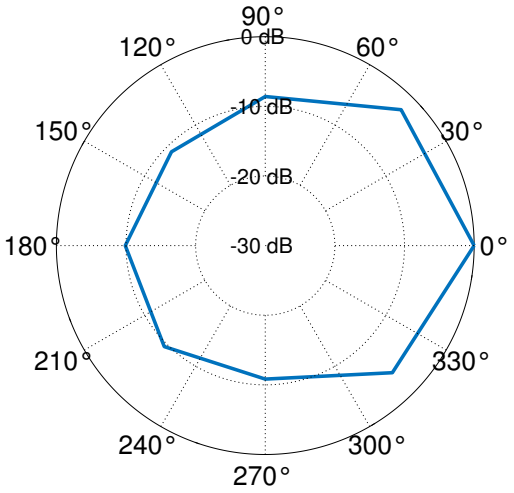
- MNS
- CVX designed beamformer with bitmask
- CVX designed beamformer with DMA pattern

In Fig. 6.2 some interesting examples are given as polar plots. As expected the energy reduction in the real world scenario is not very large. This is mostly due to the strong reflections of the walls in the room that was used for the recordings. In general an energy reduction of almost 10 dB can be seen in the best cases.
To decrease the influence of unwanted spikes in the transfer functions of the microphones the signal was filtered after the adaptive beamformer. To also reduce the influence of the directivity pattern reduction that has been seen for the CVX designed beamformers the output signal was bandpass filtered from 300 Hz-6 kHz. The used filter was an equiripple bandpass of order 327 designed with the fda-tool of MATLAB with a stopband attenuation of 60 dB. The low cutoff frequency of 300 Hz certainly influences the speech quality but increases the energy reduction of

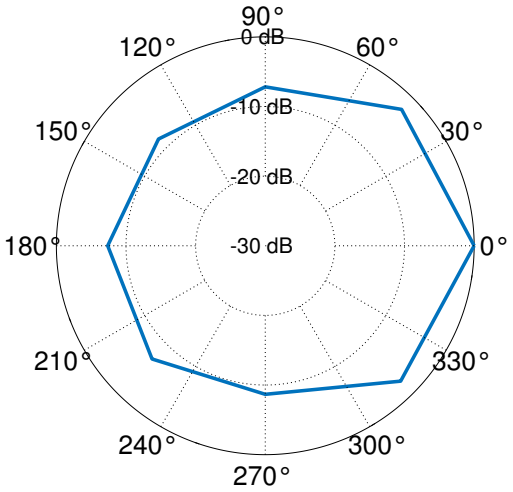
the beamformer a lot.
It is very interesting to see that while most beamformers seem to be able to reduce the signal energy about $-10\,\mathrm{dB}$, the 12 microphone array and the MNS beamformer for 4 microphones and $\delta = 10\,\mathrm{mm}$ have almost omnidirectional patterns.



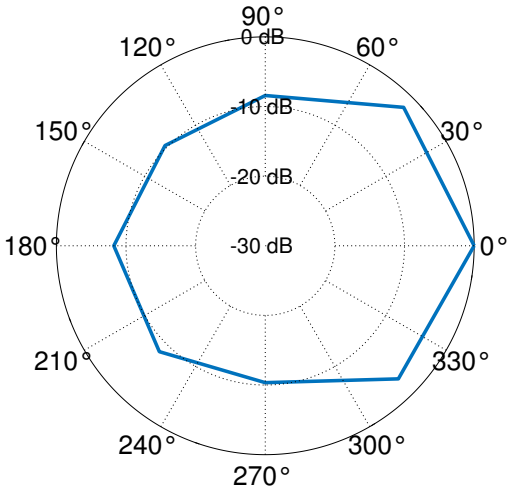
(a) CVX designed beamformer for $M = 12$ and $\delta = 10\,\text{mm}$ with bitmask



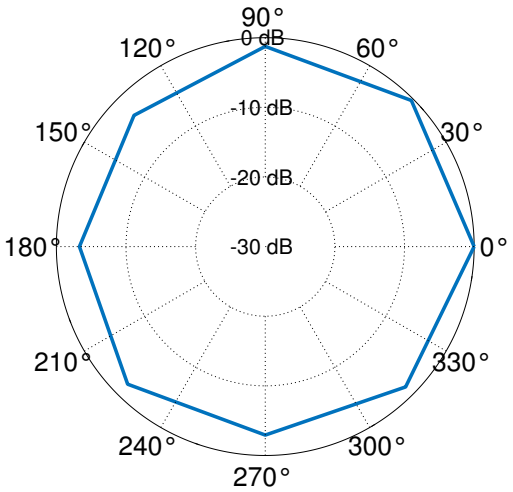
(b) CVX designed beamformer for $M = 6$ and $\delta = 20\,\text{mm}$ with bitmask



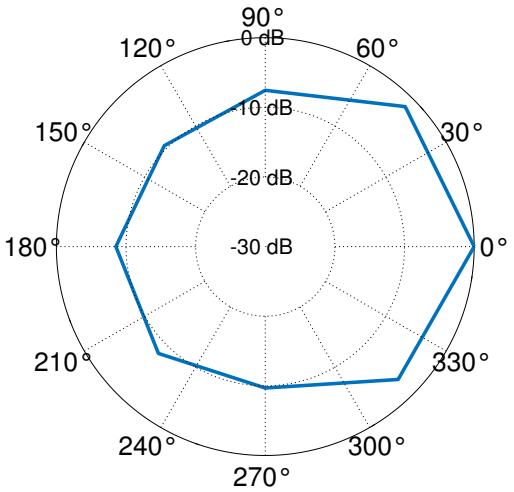
(c) CVX designed beamformer for $M = 12$ and $\delta = 10\,\text{mm}$ with DMA pattern



(d) CVX designed beamformer for $M = 6$ and $\delta = 20\,\text{mm}$ with DMA pattern



(e) MNS beamformer for $M = 4$ and $\delta = 10\,\text{mm}$



(f) MNS beamformer for $M = 6$ and $\delta = 20\,\text{mm}$

Figure 6.1: Energy ratio for different fixed beamformer solutions for the real world scenario

6.2 Localisation

The performance of the SSL was evaluated like proposed in [1] by calculating the localisation error mean and the deviation

$$\begin{aligned} \epsilon &= \frac{1}{N}\sum_{i=1}^{N}(\hat{\theta}_i - \theta_i) \\ \sigma_\epsilon &= \sqrt{\frac{\sum_{i=0}^{N}(\hat{\theta}_i - \theta_i)^2}{N^2}} \end{aligned} \tag{6.1}$$

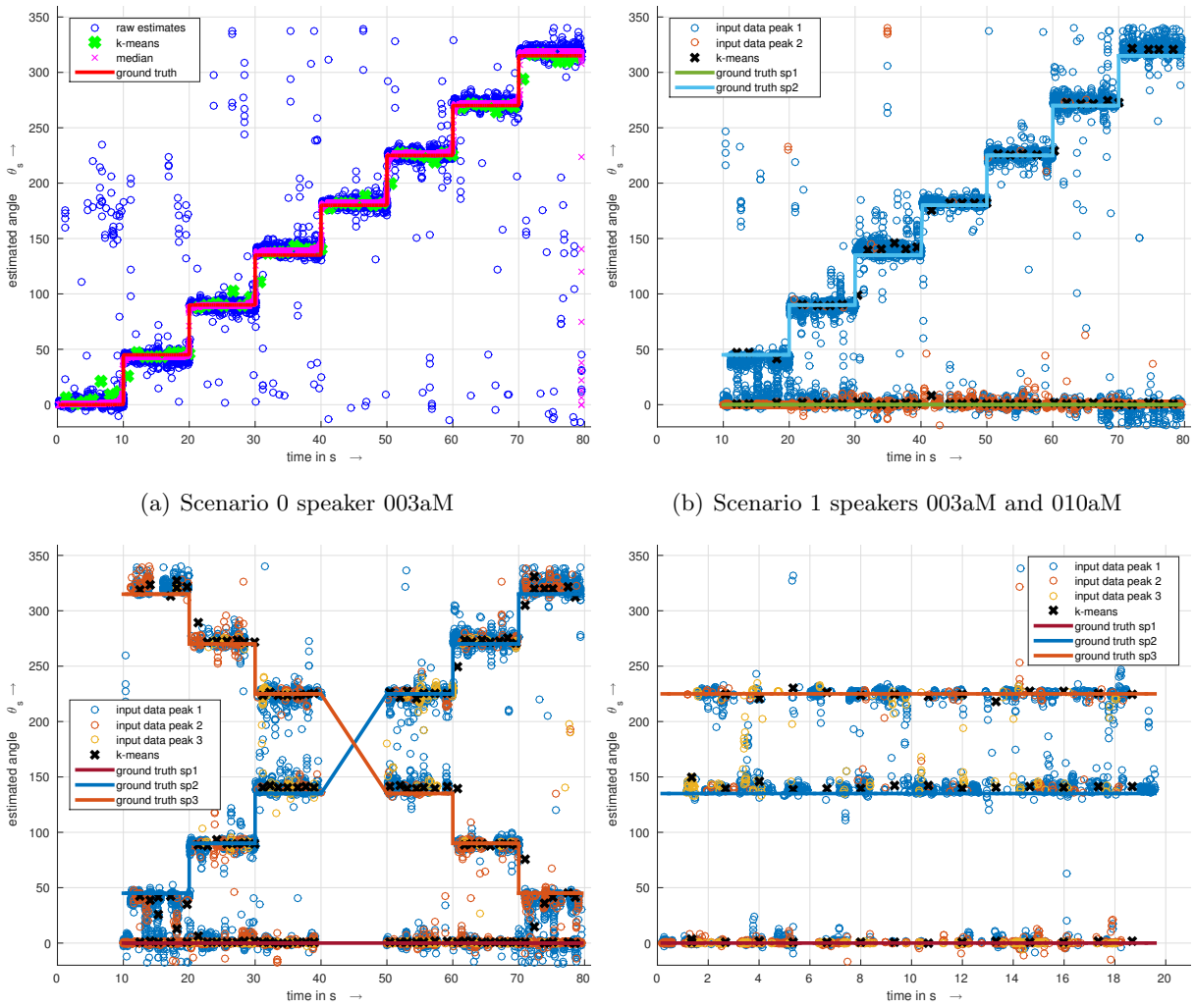
where $\hat{\theta}_i$ is the estimated angle and θ_i is the ground truth. The localisation was evaluated for SRP-PHAT and MUSIC for all four scenarios mentioned in Section 5.3.2. In Fig. 6.2 and Fig. 6.3 two examples of the results can be seen. For the evaluation of the error only parts of the recordings with a constant amount of speakers were taken. In the first 10 seconds of secenario 1 and 2 and in the middle of scenario 2 the number of target position changes. This is either because the number of speakers changes or they are overlapping. To track these situations more sophisticated algorithms that use data association to keep track of which datapoint belongs to which track would be needed for a meaningful error estimation. Since that is out of the scope of this thesis those problematic situations were left out of the error evaluation. Also to make detection of the speakers at $\theta_s = 0°$ easier the results of the localisation were shifted 20° to get a distinctive track for the speaker at $\theta_s = 0°$.

In Fig. 6.2(a) the results for localising a single speaker can be seen. For this case the raw estimates were smoothed with a median filter and a blockwise k-means algorithm. The processing blocksize for this was $N_{proc} = 500$. The median filter yields one result for every input sample while the k-means clustering results in one final estimate every block. So the k-means updates every

$$\begin{aligned} t_{update} &= \frac{N_{proc} \cdot framesize}{fs} \\ t_{update} &= \frac{500 \cdot 256}{48000\ \text{Hz}} = 2.66\ \text{s} \end{aligned} \tag{6.2}$$

The resulting detections could be less since the k-means was constrained to only yield results for clusters that contain more than 10 data points in scenario 1 and 2. If there were clusters found with less values the result was set to NaN so the time between detections could be more than t_{update}. For the scenarios with more than one speaker like in Fig. 6.2(b), Fig. 6.2(c) and Fig. 6.2(d) the smoothing with a median filter is not possible without data association, so these scenarios were only evaluated with k-means clustering.
To calculate the estimation error and deviation for the multi-target scenarios the distance to all ground truths was calculated and the shortest distance was taken for the error calculation.

In both Fig. 6.2 and Fig. 6.3 we can see that the localisation works very well for both MUSIC and SRP-PHAT. Nevertheless it is very suprising that the SPR-PHAT seems to have less erroneous detections than MUSIC. Especially for the single target situation in 6.2(a) and Fig. 6.3(a) SRP-PHAT seems to work better. But this is quite deceptive since most of the raw data points for MUSIC are still at the ground truth even if the deviation is larger than for SRP-PHAT.
What is not suprising is that the separation in the multi target scenarios is better with MUSIC. This can be seen in Fig. 6.2(d) and Fig. 6.3(d) where the amount of false detections between the sources seems to be less with MUSIC. Since MUSIC has a lot sharper peak in its detection function than SRP-PHAT this is not suprising. Section 4.4.3
In the next chapters the results of the error mean and deviation for all scenarios and microphone



(a) Scenario 0 speaker 003aM

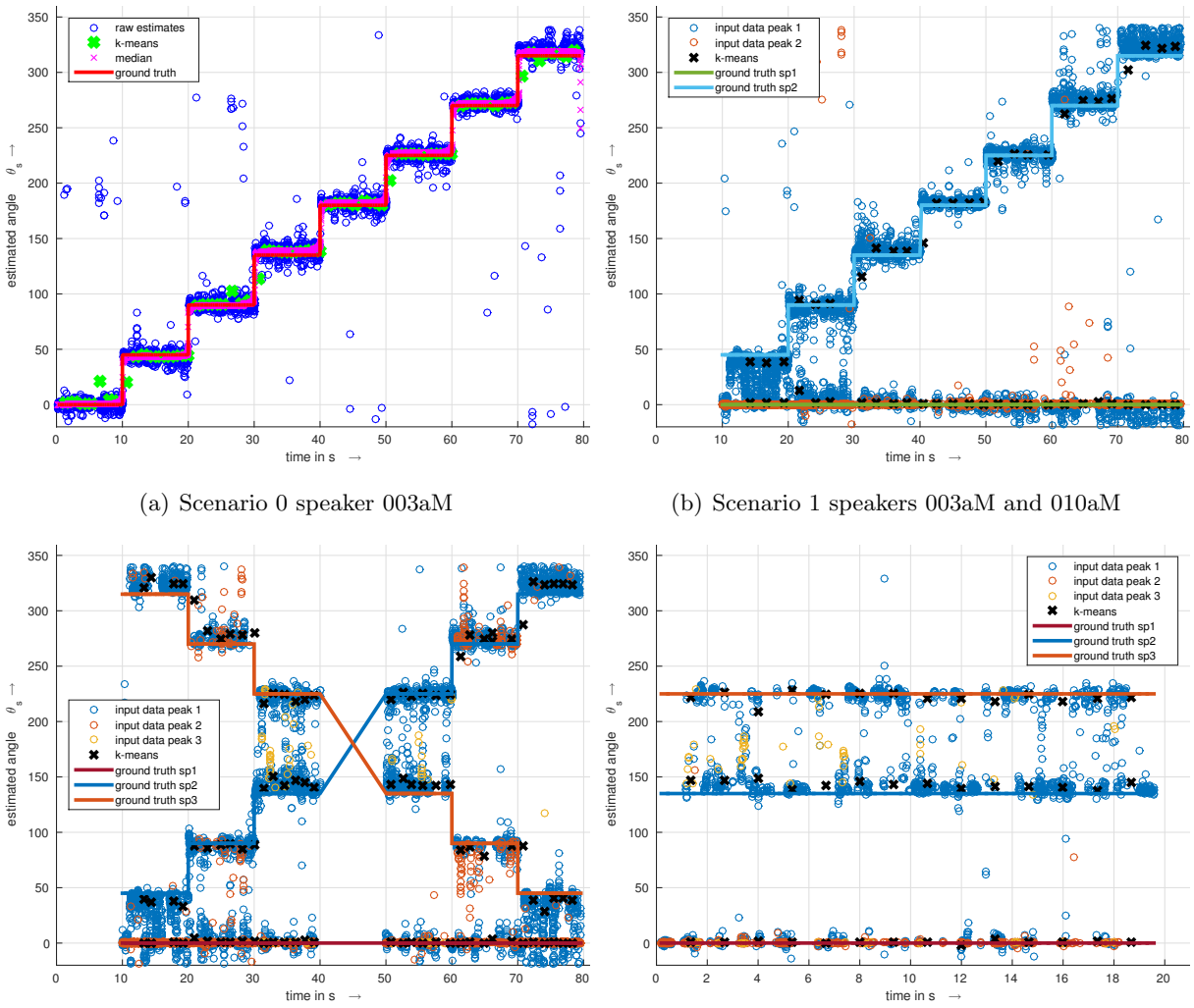
(b) Scenario 1 speakers 003aM and 010aM

(c) Scenario 2 speakers 003aM, 010aM and 009aM

(d) Scenario 3 speakers 003aM, 010aM and 009aM

Figure 6.2: Localisation scenarios for MUSIC with an array with $M = 6$ microphones and $\delta = 20\,\text{mm}$

arrays are shown. The errors and deviations were averaged over all different speakers and in the multi-speaker case also over all speakers.



(a) Scenario 0 speaker 003aM

(b) Scenario 1 speakers 003aM and 010aM

(c) Scenario 2 speakers 003aM, 010aM and 009aM

(d) Scenario 3 speakers 003aM, 010aM and 009aM

Figure 6.3: Localisation scenarios for SRP with an array with $M = 6$ microphones and $\delta = 20\,\text{mm}$

6.2.1 Four Microphones, 10 mm

	scenario 0					
	MUSIC			SRP		
	raw data	k-means	median	raw data	k-means	median
ϵ	3.7	-0.6	1.2	1.1	-2.9	1.2
σ_ϵ	45.9	15.9	13.7	13.3	14.6	9.0

Table 6.1: Localisation errors for 4 microphones with $\delta = 10\,\text{mm}$, scenario 0

	scenario 1		scenario 2		scenario 3	
	k-means		k-means		k-means	
	MUSIC	SRP	MUSIC	SRP	MUSIC	SRP
ϵ	2.3	-1.9	0.1	-1.4	4.0	3.9
σ_ϵ	15.1	12.0	17.6	18.8	16.3	27.2

Table 6.2: Localisation errors for 4 microphones with $\delta = 10\,\mathrm{mm}$, scenario 1,2 and 3

6.2.2 Six Microphones, 10 mm

	scenario 0					
	MUSIC			SRP		
	raw data	k-means	median	raw data	k-means	median
ϵ	-1.4	-5.4	-2.1	-1.7	-6.2	-1.9
σ_ϵ	29.4	13.9	10.9	11.4	13.3	4.5

Table 6.3: Localisation errors for 6 microphones with $\delta = 10\,\mathrm{mm}$, scenario 0

	scenario 1		scenario 2		scenario 3	
	k-means		k-means		k-means	
	MUSIC	SRP	MUSIC	SRP	MUSIC	SRP
ϵ	-2.8	-2.9	-1.7	-3.6	-1.2	-2.0
σ_ϵ	9.8	8.4	12.4	17.1	9.5	22.4

Table 6.4: Localisation errors for 6 microphones with $\delta = 10\,\mathrm{mm}$, scenario 1,2 and 3

6.2.3 Twelve Microphones, 10 mm

	scenario 0					
	MUSIC			SRP		
	raw data	k-means	median	raw data	k-means	median
ϵ	-5.5	-9.9	-1.9	-2.9	-7.3	-1.2
σ_ϵ	39.1	21.7	19.8	32.8	19.5	17.4

Table 6.5: Localisation errors for 12 microphones with $\delta = 10\,\mathrm{mm}$, scenario 0

	scenario 1		scenario 2		scenario 3	
	k-means		k-means		k-means	
	MUSIC	SRP	MUSIC	SRP	MUSIC	SRP
ϵ	-1.3	-1.4	-1.0	-1.4	-7.3	-5.3
σ_ϵ	14.3	11.3	20.6	19.3	13.1	9.5

Table 6.6: Localisation errors for 12 microphones with $\delta = 10\,\mathrm{mm}$, scenario 1,2 and 3

6.2.4 Four Microphones, 20 mm

	scenario 0					
	MUSIC			SRP		
	raw data	k-means	median	raw data	k-means	median
ϵ	0.3	-3.9	0.3	1.2	-3.7	0.7
σ_ϵ	31.8	14.3	12.1	17.6	16.2	5.8

Table 6.7: Localisation errors for 4 microphones with $\delta = 20\,\text{mm}$, scenario 0

	scenario 1		scenario 2		scenario 3	
	k-means		k-means		k-means	
	MUSIC	SRP	MUSIC	SRP	MUSIC	SRP
ϵ	-0.6	-0.9	0.3	-0.8	1.1	0.0
σ_ϵ	8.1	8,0	10.2	16.6	9.4	15.0

Table 6.8: Localisation errors for 4 microphones with $\delta = 20\,\text{mm}$, scenario 1,2 and 3

6.2.5 Six Microphones, 20 mm

	scenario 0					
	MUSIC			SRP		
	raw data	k-means	median	raw data	k-means	median
ϵ	1.9	-2.4	1.2	1.9	-2.8	1.4
σ_ϵ	28.3	13.9	10.4	20.6	15.4	7.2

Table 6.9: Localisation errors for 6 microphones with $\delta = 20\,\text{mm}$, scenario 0

	scenario 1		scenario 2		scenario 3	
	k-means		k-means		k-means	
	MUSIC	SRP	MUSIC	SRP	MUSIC	SRP
ϵ	-0.7	0.4	-0.7	0.5	1.9	2.8
σ_ϵ	8.6	5.9	14.5	11.3	7.7	8.3

Table 6.10: Localisation errors for 6 microphones with $\delta = 20\,\text{mm}$, scenario 1,2 and 3

6.2.6 Discussion

As can be seen in the results in the previous tables the overall mean error for all arrays is very low. Even in the error calculations with the raw data for scenario 1 the mean error is always lower than $\epsilon = 6°$. Only the computed deviation is a lot higher for the raw data compared to results of the smoothed data points.
Comparing the errors of the median filter with k-means it is noticable that the error of the median is always lower than that of the k-means. Because of the large processing window of the median and the simple example with one speaker the median is able to completely supress all erroneous measurements and yields a very stable output.
In generel arrays with more microphones and larger sensor distances should work better. This can be seen in the variance of the raw data errors for scenario 1. The array with 12 microphones makes a difference here. Looking at the data it is likely that the array was not positioned optimally when recording and so a systematic error was introduced. Another problem could be that with the high amount of microphones the fabrication differences of the various devices

caused too much difference in the channels. Looking at the detailed results of the localisation it seems that the array was positioned a little bit off the y-axis and this caused a systematic error.

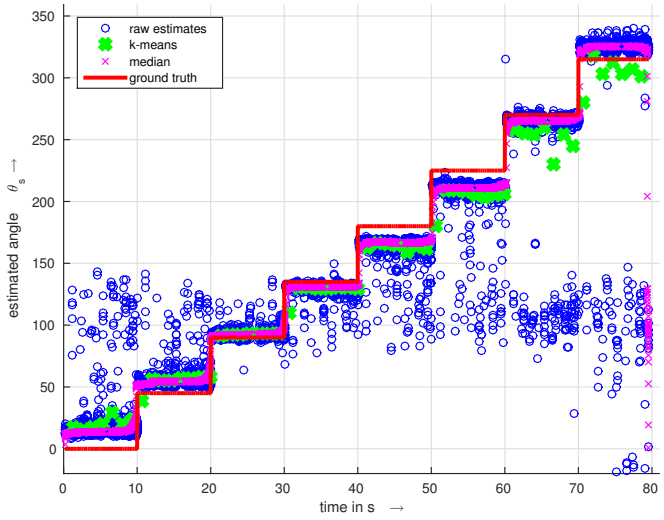


Figure 6.4: Localisation results for 12 microphone array and a single speaker

In Fig. 6.4 one result of the 12 microphone array can be seen. The localisation estimate is fine for $\theta_s = 90°$ and $\theta_s = 270°$ so the array was probably not centered properly. The concentrated erroneous detections that seem to be around 100° are only at the recordings of the 12 microphone array and seem to cause the high variance of the errors.

7 Conclusion

In this thesis algorithms for beamforming with circular differential microphone arrays have been investigated. Many of the design algorithms have severe problems with white noise amplification in the low frequency regions. To mitigate this problem and to potentially find a solution that can optimise beampatterns not only in the two-dimensional plane of the microphone array, a solution utilising convex optimisation was proposed. Using the CVX toolbox two different design methods were proposed and evaluated. One method was trying to optimise the beampattern in a way to reach the pattern of an ideal frequency independent DMA pattern, the other one using a binary mask to compute the filter weights of the microphone filters.
Results showed that the beamformers designed with CVX mimic the closed form solutions a lot in the considered frequency range. While it is possible to constrain the white noise gain over the whole frequency range it is not certain that the optimisation reaches a viable solution for higher order designs. Even for low order designs the constraint on the white noise gain can dangerously influence the directivity pattern to a point where the resulting beampattern is almost omnidirectional in low frequency ranges. For the convex optimisation to work with all desirable constraints more microphones are needed. But even there the given constraints sacrifice the frequency invariance of the beampattern to reach the solution.
The optimisation for three dimensions yielded results very similar to the closed form solutions. To ensure the position of the desired minima of the beampattern additional constraints were introduced. This however lead to problems that were rarely feasible. To improve on this only the first 15° of the elevation were optimised.
While the simulated results of the convex optimisation for the fixed beamformers were quite similar to the MNS solution the noise suppressing capabilities with the recorded data seems to be better than the closed form beamformers. Further the white noise gain can be set to a fixed level compared to the closed form solutions, there the best perfoming solution is still the MNS beamformer using many microphones.

The possibility to steer the main direction of the beamformers to every direction of a microphone without changing the beampattern made it necessary to first localize the speaker of interest.
For that TDE-Interpolation, SRP-PHAT and MUSIC were considered. Because of the dependance of the TDE-Interpolation on the microphone distance this algorithm was not further investigated. SRP-PHAT and MUSIC were both tested with the recorded data and suprisingly SRP-PHAT seemed to even outperform MUSIC in terms of erroneous measurements. To average the framewise localisation results k-means clustering was used and in the simple case of a single

speaker also a median filter was applied to smooth the data. After averaging the error mean and variance of the localisation were both supprisingly low considering the small microphone distances. So it should easily be possible to use the results of the localisation for beamsteering. The employed VAD has proven to work very well with real data, despite its simplicity. The behaviour of the VAD to cut off low energy signal parts even if there is speech in them is no problem for the localizing task. Since the SSL works better on the frames with high signal energy it is better to suppress some speech frames and average later than introducing a lot of faulty detections into the system. Considering the results of the measurements the selected algorithms are suitable to be used in a compact table mounted speech enhancement system but there is still work needed in terms of source tracking and target selection to steer the beamformer in a direction, keep it there and supress other disturbing sources.

7.1 Outlook

To finish the proposed system and get a whole speech enhancement system there is still work to be done in the field of tracking. The localised datapoints have to be associated to a speaker so that destinctive tracks of targets can be separated. After this one target can be specified to be the main speaker and other sources can either be suppressed or saved to separate channels. The proposed convex solution for designing beamformers can be improved by introducing frequency dependent constraints and by looking further into the optimisation for three dimensions. The fact that there is always a strong sensitivity to the 90° elevation is problematic in terms of early reflections from the ceiling and certainly degrades the energy reduction in reverberant environments. Also the output energy minimalisation of the adaptive beamformer should be improved if there is a distinctive minimum at the elevation of the speaker, since the speaker is rarely in the same plane as the microphone array in a realistic scenarios. Three-dimensional source localisation could be applied and beampatterns matching the speaker elevation can be pre-computed and used.
Also an evaluation in terms of objective measures like Perceptual Evaluation of Speech Quality (PESQ) or overall perceptual score (OPS) of the overall system is still to be done.
Finally the implementation into a real-time system can be considered since all of the proposed algorithms were chosen to work well together in a block processing environment. The accuracy of the localisation certainly has to be reduced for this but considering the performance of the SSL this should still suffice for steering the beamformer to every microphone direction.

Bibliography

[1] I. J. Tashev, *Sound Capture and Processing: Practical Approaches*. Wiley Publishing, 2009.

[2] MATLAB, http://www.mathworks.com/.

[3] J. Benesty, C. Jingdong, and I. Cohen, *Design of Circular Differential Microphone Arrays (Springer Topics in Signal Processing)*, 2015th ed. Springer, 1 2015.

[4] H. Cakmak, "A basic polar plot tool in dB linear scale," http://de.mathworks.com/matlabcentral/fileexchange/26476-a-basic-polar-plot-tool-in-db-linear-scale, 25 Jan 2010.

[5] G. W. Elko, *Acoustic Signal Processing for Telecommunication*. Boston, MA: Springer US, 2000, ch. Superdirectional Microphone Arrays, pp. 181–237. [Online]. Available: http://dx.doi.org/10.1007/978-1-4419-8644-3_10

[6] H. Cox, R. Zeskind, and T. Kooij, "Practical supergain," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, no. 3, pp. 393–398, Jun 1986.

[7] E. Mabande, A. Schad, and W. Kellermann, "Design of robust superdirective beamformers as a convex optimization problem," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, April 2009, pp. 77–80.

[8] H. Pessentheiner, G. Kubin, and H. Romsdorfer, "Improving beamforming for distant speech recognition in reverberant environments using a genetic algorithm for planar array synthesis," in *Proceedings of Speech Communication; 10. ITG Symposium*, 2012, pp. 1–4.

[9] G. W. Elko and A.-T. N. Pong, "A simple adaptive first-order differential microphone," in *IEEE ASSP Workshop on Applications of Signal Processing to Audio and Acoustics, 1995.*, Oct, pp. 169–172.

[10] M. Elmar, "Differential microphone arrays," Master's thesis, Graz University of Technology, Austria, 2013.

[11] H. Pessentheiner, T. Pichler, and M. Hagmüller, "Amisco: The Austrian German multi-sensor corpus," Portorož, Slovenia, 05/2016 2016. [Online]. Available: http://lrec2016.lrec-conf.org/

[12] G. W. Elko and J. Meyer, "Second-order differential adaptive microphone array," in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, April 2009, pp. 73–76.

[13] C. Knapp and G. Carter, "The generalized correlation method for estimation of time delay," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 24, no. 4, pp. 320–327, Aug 1976.

[14] S. T. Birchfield and D. K. Gillmor, "Acoustic source direction by hemisphere sampling," in *IEEE International Conference on Acoustics, Speech, and Signal Processing, 2001.*, vol. 5, 2001, pp. 3053–3056 vol.5.

[15] A. K. Nandi, "On the subsample time delay estimation of narrowband ultrasonic echoes," *IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control*, vol. 42, no. 6, pp. 993–1001, Nov 1995.

[16] J. (10584), "allcomb(varargin)," http://de.mathworks.com/matlabcentral/fileexchange/10064-allcomb-varargin-, 20 Feb 2006 (Updated 15 Feb 2016).

[17] N. Yoder, "peakfinder," https://www.mathworks.com/matlabcentral/fileexchange/25500-peakfinder-x0--sel--thresh--extrema--includeendpoints--interpolate-, 06 Oct 2009 (Updated 14 Dec 2015).

[18] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Transactions on Antennas and Propagation*, vol. 34, no. 3, pp. 276–280, Mar 1986.

[19] X. Zhang, E. Song, J. Huang, H. Liu, Y. Wang, B. Li, and X. Yuan, "Acoustic source localization via subspace based method using small aperture mems arrays," *Journal of Sensors*, vol. 2014, 2014.

[20] C. T. Ishi, O. Chatot, H. Ishiguro, and N. Hagita, "Evaluation of a music-based real-time sound localization of multiple sound sources in real noisy environments," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct 2009, pp. 2027–2032.

[21] Audacity, http://www.audacityteam.org/.

[22] ARDOUR, https://ardour.org/.

[23] B. Schuppler, M. Hagmüller, J. A. Morales-Cordovilla, and H. Pessentheiner, "Grass: The graz corpus of read and spontaneous speech," in *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. Reykjavik, Iceland: European Language Resources Association (ELRA), 2014, pp. 1465–1470. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2014/pdf/394_Paper.pdf

A
Speakers used for Recordings

rec. No.	scenario 0	scenario 1	scenario 2	scenario 3
1	021aF	021aF, 018aF	021aF, 018aF, 004aM	021aF, 018aF, 004aM
2	016aF	016aF, 005aM	016aF, 005aM, 014aF	016aF, 005aM, 014aF
3	012aF	012aF, 015aF	012aF, 015aF, 007aM	012aF, 015aF, 007aM
4	003aM	003aM, 010aM	003aM, 010aM, 009aM	003aM, 010aM, 009aM
5	020aF	020aF, 011aF	020aF, 011aF, 002aM	020aF, 011aF, 002aM

Table A.1: Scenarios for playback with speakers

B
Abbreviations

DMA	Differential Microphone Array
CDMA	Circular Differential Microphone Array
UCA	Uniform Circular Array
LDMA	Linear Differential Microphone Array
SNR	Signal to Noise Ratio
MNS	Minimum-Norm Solution
WNG	White Noise Gain
ACDMA	Adaptive Circular Differential Microphone Array
NLMS	Normalized Least Mean Square
SSL	Sound Source Localisation
FFT	Fast Fourier Transformation
VAD	Voice Activity Detection
RMS	Root Mean Square
PHAT	Phase Transform
GCC	Generalized Cross-Correlation
SRP	Steered-Response Power
TDE	Time Delay Estimation
MUSIC	Multiple Signal Classification
NOS	Number Of Sources
CPR	Cocktail Party Room

C Symbols

r	radial coordinate
M	number of microphones
ψ_m	angle of microphone positions
θ	azimuth
m	microphone index
τ_m	delay between microphone m and the coordinate origin
c	speed of sound
$\mathbf{d}(\omega, \theta)$	steering vector
f	frequency variable
ω	angular frequency
$H_m(\omega)$	mth filter element
$X_m(\omega)$	mth input signal element
$S_m(\omega)$	source signal in frequency domain
$Y(\omega)$	beamformer output in frequency domain
$V_m(\omega)$	mth additive noise
$\mathcal{B}_\mathcal{N}$	ideal beampattern for DMA of order N
$a_{N,n}$	design coefficients index n for order N DMA
θ_s	angle of the beamformer mainlobe
$\mathcal{G}_{\mathbf{wn}}$	white noise gain
$\mathbf{h}(\omega)$	filter vector
$\mathbf{v}(\omega)$	noise signal vector
$\mathbf{x}(\omega, \theta)$	signal vector
$\Gamma_\mathbf{v}(\omega)$	pseudo coherence matrix for diffuse noise
δ_{ij}	distance between microphone i and j
$\mathcal{G}_{\mathbf{dn}}$	directivity factor
$\mathcal{D}$	directivity index
N	order of fixed beamformer
N'	number of linear equations for MNS
M'	number of symmetry constraint equations for MNS
$\boldsymbol{A(\omega, \theta)}$	constraint matrix
$\boldsymbol{D}(\omega, \boldsymbol{\theta})$	matrix of steering vectors
$\theta_{N,n}$	desired null n for MNS of order N
$\boldsymbol{C}$	matrix with symmetry constraints
$\boldsymbol{c}_{M,m'}$	vector with symmetry constraint m' for M microphones

$\boldsymbol{b}$	vector with design coefficients for MNS
$\boldsymbol{\beta}$	vector with design coefficients for MNS
$\beta_{N,n}$	design coefficient n for MNS order N
$\boldsymbol{\theta}$	vector with design coefficients for MNS
θ	design coefficient n for MNS order N
ϵ_r	regularisation factor for superdirective beamformer
$\boldsymbol{I}_M$	unity matrix of size M
$\boldsymbol{h}_{max}(\omega)$	transfer function of superdirective beamformer
$\boldsymbol{h}_{\epsilon_r}(\omega)$	transfer function of robust superdirective beamformer
$\boldsymbol{h}_0(\omega)$	transfer function at $\epsilon = 0$
$\boldsymbol{h}_\infty(\omega)$	transfer function at $\epsilon = infty$r
$\boldsymbol{i}_1$	vector with solutions for superdirective beamformer length M + 1
$\boldsymbol{i}_{M,1}$	vector with solutions for superdirective beamformer length M
$\mathcal{B}_{Ch,N}(\theta)$	frequency independent beampattern order N
$b_{N,n}$	design coefficient for frequency independent beampattern
$T_n(\cos\theta)$	chebychev polynomials of fist kind
$\boldsymbol{\Psi}_{N+1}$	matrix of linear equations for Jacobi-Anger solution
$\mathbf{h'}(\omega)$	resulting vector from Jacobi-Anger solution
$\mathbf{H}'_n(\omega)$	resulting vector from Jacobi-Anger solution for one channel
$b^*_{N+1}(\omega)$	vector containing design coefficients
$\overline{\omega}$	scaled angular frequency
$J_n(\overline{\omega})$	nth order Bessel function of the first kind
$J'_n(\overline{\omega})$	variation of nth order Bessel function
$\mathcal{M}$	number of transfer functions resulting from Jacobi-Anger solution
$\mathbf{d_{3d}}(\omega, \theta, \gamma)$	three-dimensianl steering vector
γ	angle of elevation
$\mathbf{w}$	weights/transfer function to optimize
$\boldsymbol{G}(\omega)$	matrix containing steering vectors for desired beampattern
$\boldsymbol{\mathcal{B}_{des}}$	desired beampattern
$\mathbf{d_{3d},}_m(\omega)$	steering vector for microphone Ψ_m
N_θ	number of angles to optimize for azimuth
N_γ	number of angles to optimize for elevation
$\boldsymbol{V}(\omega)$	matrix containing steering vectors for desired minima of beampattern
$C_f(\omega)$	beampattern of the forward facing cardioid
$C_b(\omega)$	beampattern of the backward facing cardioid
β	vector with adaptive weights for the beamformer
μ	step-size for NLMS
Δ	regularization parameter for NLMS
$L^{(t)}_{min}$	estimated noise floor for frame t
$L^{(t)}$	estimated signal level for frame t
T	frame duration
τ_{up}, τ_{down}	time constants for tracking the noise floor
K	number of frequency bins
W_k	frequency weights
$Y^{(t)}_k$	frequency bins of input signal for frame t
$V^{(t)}$	voice activity flag
T_{up}, T_{down}	thresholds for speech and noise
ρ	distance from microphone pair to speaker
τ_D	time delay between two microphones
$\boldsymbol{R_{12}}$	generalized cross-correlation between microphone 1 and 2
$\boldsymbol{\Psi}$	weighting function for generalized cross-correlation

$\Psi_{PHAT}(f)$	PHAT frequency weights
$\boldsymbol{G_{12}}$	cross-correlation between microphone 1 and 2
$\boldsymbol{X}_1, \boldsymbol{X}_2$	input signal vectors of microphone pair
$h_i(\theta, \gamma)$	hypothesis vectors for TDE interpolation
$P_{BF}(\theta)$	localisation function for SRP-PHAT for one frequency bin
$\boldsymbol{S}, \boldsymbol{S}_x$	averaged cross-power matrix for one frequency bin
$\boldsymbol{X}$	buffered input signals for one frequency bin
P_{SSL}	broadband localisation function
F_j	jth source signal
λ	eigenvalues of $\boldsymbol{S}_x$
Λ	diagonal matrix of eigenvalues
$\boldsymbol{U_S}$	matrix of signal eigenvectors
$\boldsymbol{U_N}$	matrix of noise eigenvectors
$\boldsymbol{\Phi_j}$	eigenvectors of $\boldsymbol{S}_x$
P_{mus}	MUSIC localisation function
$L_{A,eq}$	equivalent sound level with A-weighting
ϵ	localisation error mean
σ_ϵ	localisation error deviation
$\hat{\theta}_i$	estimated angle
N_{proc}	number of samples used for averaging
$hopsize$	hopsize used for localisation
t_{update}	update rate of k-means output