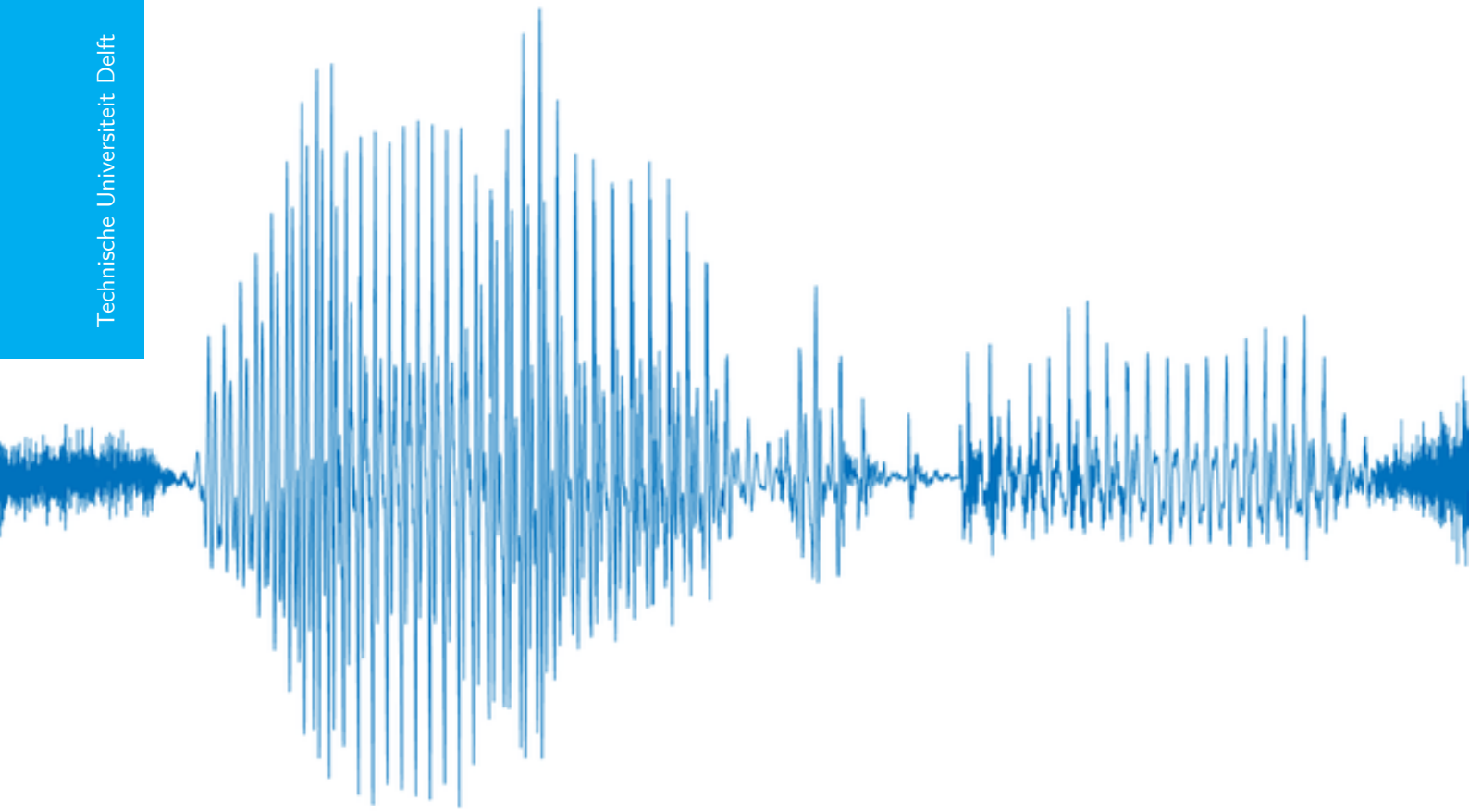
The cocktail party problem

GSVD-beamformers for speech signals in reverberant environments

D.J.S. Hulsinga

Technische Universiteit Delft



TU Delft Delft University of Technology

Challenge the future

THE COCKTAIL PARTY PROBLEM

GSVD-BEAMFORMERS FOR SPEECH SIGNALS IN REVERBERANT ENVIRONMENTS

by

D.J.S. Hulsinga

in partial fulfillment of the requirements for the degree of

Master of Science
in Electrical Engineering

at the Delft University of Technology,
to be defended publicly on Friday the 16^{th}, February 2018 at 10:45 AM.

Thesis committee:	Prof. dr. ir. A.J. van der Veen,	CAS - TU Delft, supervisor
	Dr. ir. R. Heusdens,	CAS - TU Delft
	Dr. ir. J.H. Weber,	DIAM - TU Delft

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft Delft University of Technology

ABSTRACT

Hearing aids as a form of audio preprocessing is increasingly common in everyday life. The goal of this thesis is to implement a blind approach to the cocktail party problem and challenge some of the regular assumptions made in literature. We approach the problem as wideband FD-BSS.

From this field of research, the common assumption of contineous activity is dropped. Instead a number of users detection is implemented as a preprocessing step and ensure the appropriate number of demixing vectors for each time frequency bin. The validity of the standard mixing model used for STFT's is challenged by looking at the response of a linear array. Source separation is achieved by demixing vectors based on the GSVD, derived in a model-based approach. While most permutation solvers offer an a posteriori solution for all users, we looked at finding local solutions for a single user. Combining this with the user identification called the alignment step, we conclude that the permutation problem can be reduced to selecting a demixing vector for each discrete time-frequency instance.

The correlation coefficient proves to be a sufficient metric to couple reconstructions to the original data as it selects most of the active time-frequency bins. In the far-field case, our approach performs in a comparable but not superior manner. We did find that our method is much more robust against inaccuracies introduced when narrowband channels are assumed but not actually available. This is strongly exemplified by our experiment of a changing DFT-size.

The Frobinius norm was suggested as a measure of distance between the estimate STFT and the orignial signals time frequency domain description but it resulted in counter intuitive results which didn't correspond with other metrics used in this thesis. It is expected that there are effects induced by changing the size of the STFT which are not accounted for.

Our demixing vectors achieve comparable inteligibility, measured by STOI, as the compared techniques and it is more robust against smaller sample sizes than the theoretically SINR optimal MVDR.

PREFACE

To finally complete my degree here at the TU Delft gets to be one of the most astounding achievements of my life for a long time. While studying here, the realisation that this far is further than most people have come, has kept me from falling into doubt and with this milestone that reassurance has grown significantly.

In this small number of words that I have space for here, it is challenging to describe how much I owe to the people around me. Coming to think of it, it might still be challenging if I filled this page, its margins and then some. First and foremost I want to thank my parents for supporting my endeavour, even through periods when I was unwilling and undeserving of their help. To complete a university degree wasn't as obvious for me as it was for other I have met here, yet I am here. Also, I want to thank Casey for being my listening ear and my best friend, all the while also being the most amazing girlfriend I could have ever whished for. You lighten my every day, even if I haven't always shown it.

I want to thank Alle-Jan for being my supervisor and guiding me through the process of writing a work like this, providing me with new insights along the way. Finally, I extend my gratitude to all of the CAS department for sharing time, ideas or simply a drink with me in the past time.

Now that I expect to leave the world of academics behind me, I am fulfilled by the idea that I have worked on research from which I believe that you, the reader, can learn - as I have learned from others here. Have fun.

D.J.S. Hulsinga
Delft, February 2018

Contents

1

INTRODUCTION

1.1. THE COCKTAIL PARTY PROBLEM

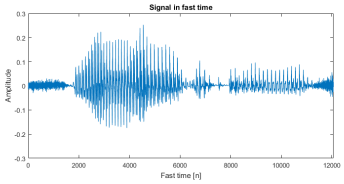
When attending social gatherings, able-hearing people are very adept at understanding their conversation partner(s), regardless of conversations being held or loud music being played in their immediate surroundings. This capability is still not adequately available in modern recording arrays as these tend to simply amplify most of the received data. The desired speech signal is mixed in there somewhere, though degraded because of propagation through a channel and the addition of interferers and noise. To improve the intelligibility of such a degraded speech signal is called the cocktail party problem.

It has been estimated that 22% of the population of Europe is considered to be hearing impaired in at least some degree[1]. Apart from the actual sharing of mixed drinks, the cocktail party problem occurs equivalently in other situations. When that is, can be exemplified by three situations which are characterized by the number of transmitters and receivers involved. In the many-to-one situation there are users of hearing aids who converse with multiple people while they are the only recipient. The one-to-many situation is common for public recordings where a single speaker or possibly singer addresses a group. Many-to-many situations are less common and can for example be a conference calls where two meeting rooms are connected digitally. Because poor hearing impacts peoples lives in such a strong way and the fact that it happens to so many people, this problem is very much worth researching.

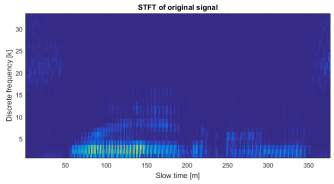
The reasons why this problem is considered so challenging can divided into two categories. First we will explain more about speech as a signal and secondly more about how that propagates in real world environments.

Speech signals differ widely between speakers[2]. Such differences are quantised in suprasegmentals[3]. For example, these aspects can be a speaker's speech rate, stress timing, pause frequency or pause duration. The following three properties of speech signals should be taken into account: First, the fact that speech signals start and stop abruptly at unpredictable times. Secondly, as a speaker does becomes active, this doesn't happen for all frequencies at the same time. Lastly, during the active time, the power in a speech signal is not constant over either time or frequency. All of this contributes to the notion that it is difficult to predict the behaviour of speech signals.

Propagating through real world environments can have a wide variety of effects on acoustic signals. Again,



(a) Signal in the time domain



(b) Signal in the TF-domain

Figure 1.1: Examples of the data used in this thesis.

three of those aspects are highlighted. Other conversations make for activity in the environment which is alike the desired signal, often resulting in an unknown number of interferers. If there are sources that transmit signals which are not at all alike speech, these can be summarized as sources of (coloured) noise. While this does make for a simpler description, this also implies that the noise power and colour can change over time. Finally, speakers and people in their surroundings tend to move about. This implies that time invariant channels, the existence of a dominant line-of-sight and constant reverberation times can not be assumed.

Constructing a complete model of a realistic environment requires quite a complex structure.

1.1.1. SPECIFIC PROBLEM

In this thesis, the cocktail party problem is approached as being a wideband source separation problem. Our goal is to improve the intelligibility of a degraded speech signal by suppressing interferers and reducing noise without having prior knowledge.

Based on the above considerations, only some of the aspects of the real world environment are adopted by this thesis. The assumptions about the signal and channel behaviour are summarized here. The desired signal is a speech signal said to be wideband, non-stationary and to be active at unknown intervals. If it is not further specified, all examples assume two users of which one is the signal and the other the interferer. The noise distributions is assumed additive, white gaussian (AWGN). To emulate the signal propagation through a room, two types of data models are considered: The instantaneous model and the convolutional model. The simulated data is based on a scenario which consist of a linear microphone array.

1.1.2. APPROACH

Wideband signals are most commonly analysed in the time-frequency domain (tf-domain). In literature this approach is called frequency domain blind source separation (FD-BSS) and it consists of two parts. First, an approximation of the data's tf-domain representation is calculated by the short-time Fourier transform (STFT), an example of which can be seen in figure 1.1b. When a source separation beamformer is applied to it, this step results in estimates for each users. However these are spread out over time and frequency, which brings us to the second part: these estimates need to be assigned to either the signal or the interferer. This latter part is called the permutation problem.

This thesis approaches the source separation problem for each of the STFT's frequency channels individually. To solve these problems, we want to extend the work of Mu Zhou [4] which describes a BSS beamformer as interference suppressor and noise canceller. It is based on the generalized singular value decomposition (GSVD) and was initially designed for narrowband radio communication. If the frequency channels in the STFT have a small enough bandwidth, they can be considered narrowband. This allows the technique to be applied to a situation like ours.

The permutation problem is the situation where for a time-frequency instance, two users are detected.

The results of the source separation are in an arbitrary order and when compared to reconstructions from other parts of the STFT, not always in the same order. We are going to discuss solving this indeterminacy in two steps: First there is fitting the right reconstructions together to one estimate, this is often the only part of the permutation problem discussed. Secondly, there is identifying the desired user based on the estimate, sometimes called the alignment step.

Combining these steps, we want to have an estimate of the original time signal as our result.

1.2. STATE OF THE ART

The classification of blind sense separation methods has been covered widely[5]. While new methods have been developed, the taxonomy has not been extended as much. To compare previous work to our approach, we use the outline introduced in section 1.1.2 which states that FD-BSS consists of the steps source separation and permutation solver.

Blind source separation requires an assumption on the structure or statistics of the data. We look at two popular options: Assumptions based on the signal or on the channel properties.

There is a significant interest in beamformer generation based on the signal. Speech signal sparsity is one well known assumption. This states that only one user is active for each time-frequency instance. Based on this assumption, the STFT of the received signal can be divided by a binary mask[6] to reconstruct the different signals. Research has been done for both soft and hard masking [7] [8]. Alternatively, statistical independence between the received signals has been assumed. Solutions can be optimized for this with use of the Kullback-Leibler divergence [9] [10] called ICA.

Alternatively beammformers are based on what is known about the channel. There are TDOA [11] and DOA [12] techniques, both of which assume farfield propagation models. There is currently no other research found where the demixing process is based on the generalized

Now, we look at research surrounding second step in this thesis: Permutation solvers. These solvers can be grouped by which data they compare or by looking at with which metric they compare it. Techniques compare the effect of the channel with a far field assumption[13] or correlation between reconstruction [14]. Instead of solving the permutation problem, IVA[15] elegantly avoids this by reducing the number of permutable items to one per user.

In this overview we see an opportunity to research the complete process of solving the cocktail party problem based on blind estimation of the channel subspace.

1.3. OUTLINE OF THE THESIS

In this first chapter the cocktail party problem has been introduced. The unsolved parts of it, the scenario used in this thesis and current state-of-the-art research have been looked at. The second chapter focusses on formalizing the outlined problem in a model-based approach. In the third chapter a derivation of the proposed algorithm is brought forth. It explains what steps it consists of and how these work. It will also state the interfaces for each step. Chapter four compiles a list of parameters and metrics which are available to evaluate the performance of the system. The simulation results are presented in chapter five. The last chapter is reserved for the conclusion and recommendations on further research.

2

DATA MODEL & PROBLEM STATEMENT

To derive the proposed algorithm, a mathematical description of the system is required. It shapes the assumption for later chapters. First, this chapter introduces some of the most common notation used in this and upcoming chapters. After that we define two important data models: the instantaneous model and the convolutional model. Finally, the problem that this work is trying to solve, is made explicit and challenges are adressed.

2.1. BACKGROUND

2.1.1. NOTATION

For real or complex scalars, letters are used ($a, b, \ldots$). A column vector is written as a bold letter ($\boldsymbol{a}, \boldsymbol{b}, \ldots$) and matrices are denoted with a capital bold letter ($\boldsymbol{A}, \boldsymbol{B}, \ldots$). To describe select parts of the structures presented above, a subscript or index is used. The n^{th} element of a column vector $\boldsymbol{a}$ is denoted as a_n and from a row vector $a[n]$. The n^{th} column of a matrix $\boldsymbol{A}$ can be denoted as $\boldsymbol{a}_n$.

For the following explanations, assume that $\boldsymbol{A}$ is $m \times n$ sized and that $\boldsymbol{B}$ is $p \times q$ sized. The operator $|A|_\circ$ returns the element wise absolute value of the matrix as $|\boldsymbol{A}|_{\circ i,j} = |a_{i,j}|$.

The Trace of A is the sum of the diagonal elements as

$$Tr(\boldsymbol{A}) = \sum_{i=1}^{N} a_{(i,i)} \qquad\qquad N = \min(m, n)$$

The Frobenius norm is defined as the sum of all squared elements - exemplified for the matrix A as

$$|A|_F = \sum_{i=1}^{m} \sum_{j=1}^{n} |a_{i,j}|^2$$

2.1.2. TOOLS FROM LINEAR ALGEBRA

This section summarizes two important decompositions: The singular value decomposition (SVD) and the generalized SVD (GSVD). Both are explained by being applied to a complex matrices.

The SVD is a decomposition which exists for each matrix $\boldsymbol{X}$ as

$$\boldsymbol{X} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^H$$

where $\boldsymbol{U}$ and $\boldsymbol{V}$ are unitary matrices which are the basis for the row and column span of $\boldsymbol{X}$ respectively. The diagonal matrix $\boldsymbol{\Sigma}$ contains the singular values σ_i for $i = 1,2,\dots\min(m,n)$ where it is defined that $\sigma_i \geq \sigma_{i+1}$ and $\sigma \geq 0$.

The GSVD exists for a pair of matrices $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ of sizes $m \times n$ and $p \times q$ if and only if $m = p$, $m \geq n$ and $p \geq q$. The decomposition results in

$$\text{GSVD}(\boldsymbol{X}_1, \boldsymbol{X}_1) \Leftrightarrow \begin{cases} \boldsymbol{X}_1 = \boldsymbol{F}\boldsymbol{C}\boldsymbol{U}^H \\ \boldsymbol{X}_2 = \boldsymbol{F}\boldsymbol{S}\boldsymbol{V}^H \end{cases} \tag{2.1}$$

where $\boldsymbol{F}$ is an invertable matrix of size $m \times m$ and where $\boldsymbol{U}$ and $\boldsymbol{V}$ are semi-unitairy matrice of sizes $n \times m$ and $q \times p$ respectively. The matrices $\boldsymbol{C}$ and $\boldsymbol{S}$ are square diagonal matrices which contain together contain the generalized singular values $\frac{c_{i,i}}{s_{i,i}}$. In contrast to tradition, we will normalize the decomposition such that $|f_i|_2 = 1$ for every column of $\boldsymbol{F}$.

2.2. DATA MODEL

Even though speech is generated in continuous time, microphone arrays are limited to measurements at regular intervals. Consider a frequency F_s at which the signal is sampled and define $T_s = F_s^{-1}$ as the corresponding sample period. From now on forward, instead of continuous time values $x(t)$, the discrete time samples $x(n \cdot T_s) = x[n]$ are used. The additive white noise distribution is modelled by an i.i.d. Gaussian process of mean zero and variance σ_n^2.

2.2.1. INSTANTANEOUS DATA MODEL

The received data equals the sum of multiple delayed versions of the transmitted signal. If the time in between receiving these delayed signals is short compared to the inverse bandwidth, the effect can be approximated by only a phase shift. The signals are then assumed to arrive at all receivers at the same time.

These shifts are defined independently for each receiver/transmitter combination and are multiplications by coefficients $a_{i,j}$. If $x[n]$ is the sample taken by the receiver and $s[n]$ is the transmitted symbol, then at any time instance n, the i^{th}-microphone receives

$$x_i[n] = a_{i,j}\, s_j[n] + e_{n,i}$$

from the j^{th}-source where $e_{n,i}$ is the error introduced by noise. From this source, the array consisting of $i = 0,1,\dots R-1$ microphones receives

$$\begin{aligned} \boldsymbol{x}[n] &= \begin{bmatrix} a_{0,j} \\ a_{1,j} \\ \vdots \\ a_{R-1,j} \end{bmatrix} s_j[n] + \boldsymbol{e}_n \\ &= \boldsymbol{a}_j \cdot s_j[n] + \boldsymbol{e}_n \end{aligned}$$

The environment contains $j = 0,1,\dots D-1$ uncorrelated sources and the symbols which each of them transmits can be stacked in a column vector as $\boldsymbol{s}[n] = [s_0[n]\, s_1[n] \dots s_{D-1}[n]]^T$ and the data from the array

can be calculated as

$$\begin{aligned}\boldsymbol{x}[n] &= \left[\boldsymbol{a}_0\, \boldsymbol{a}_1 \ldots \boldsymbol{a}_{D-1}\right] \cdot \boldsymbol{s}[n] + \boldsymbol{e}_n \\ &= \boldsymbol{A}\boldsymbol{s}[n] + \boldsymbol{e}_n\end{aligned}$$

Finally, measurements are taken by the array over a total of N time instances and they are the result of the same number of symbols. Both the signal and data can be combined as $\boldsymbol{X} = [\boldsymbol{x}[0]\, \boldsymbol{x}[1] \ldots \boldsymbol{x}[N-1]]$ and $\boldsymbol{S} = [\boldsymbol{s}[0]\, \boldsymbol{s}[1] \ldots \boldsymbol{s}[N-1]]$ respectively. The former can be calculated from the latter as

$$\boldsymbol{X} = \boldsymbol{A}\boldsymbol{S} + \boldsymbol{E}_n$$

where $\boldsymbol{A}$, called the channel matrix, is of size $R \times D$, $\boldsymbol{S}$ is of size $D \times N$ and $\boldsymbol{X}, \boldsymbol{E}_n$ are of size $R \times N$.

As stated earlier, this signal model is valid if the time in between receiving signals is short. This delay is defined as the time difference between the element at which the signal arrives first and at which it arrives last. The reasoning behind the approximation - described in appendix A - bases its validity on the time bandwidth product. The conclusion on what delay is considered small is explained here for convenience. If W is the bandwidth in Hz in which the signal is active and τ is the maximum delay that a signal can befall while propagating along the receiver array, its product should satisfy $W\tau \ll 1$ for the signal to be considered narrowband.

In speech processing, all of the spectrum is considered to be used and the Nyquist theorem states that it is possible to reconstruct with certainty a signal with bandwidth W which is equal to $\frac{F_s}{2}$. Often $F_s = 8\text{kHz}$ *(Narrowband)* or $F_s = 16\text{kHz}$ *(Wideband)* is used. In the increasingly common scenario of 16kHz, this means that $\tau \ll 1 \cdot W^{-1} = \frac{2}{F_s} = 0.125$ ms for τ to be considered small. Define the maximum distance across the array as d_{max}, the speed of sound as $v_s = 343$ m/s and if they relate to the delay as $\tau = \frac{d_{max}}{v_s}$ then it can be concluded that $d_{max} = \tau v_s = \frac{2v_s}{F_s} \ll 0.043$ m or equivalently $d_{max} \ll 4.29$ cm to be considered small. From this it can be concluded that the instantaneous data model is not a good representation for how a full band speech signal is received by the type of microphone array which is considered.

2.2.2. CONVOLUTIVE DATA MODEL

A more accurate description of sound waves' traversal through a room is the convolution model. Different from the previous model, the change that a channel induces in a signal is not just described by a shift a but rather by the rooms impulse response vector $\boldsymbol{h}_{i,j}$. The explanation of this data model is structured in a similar way as in the previous section. At any time instance n, define what the i^{th}-microphone receives from the j^{th}-source as

$$x_i[n] = \sum_{l=0}^{N_{60}-1} s_j[n-l]\, h_{i,j}[l] + e_n$$

The memory effect this model introduces allows for arbitrary arrival delays, phase shifts and an overall better description of multipath behaviour. Here, $N_{60} = T_{60}/T_s$ where T_{60} is the length of the channel which is defined as the time after which a signal has decayed by $60\,dB$. This is an arbitrary but common parameter in speech processing.

The behaviour of a narrowband segment of a signal is challenging to describe in the time domain. It is better described how the signal and the data are related when observed in the time-frequency domain. To do so, first we expand the time signal into its respective STFT's and then derive the behaviour of the individual discrete frequency channels.

The STFT of the data $x[n]$ is constructed by cascading local frequency transforms. To define what we mean by local, an additional time index is introduced. The two are used in parallel: The earlier described discrete time instance index n, from here on referred to as the fast time, and the newly introduced time frame index m, from here on referred to as slow time. The frames $m = 0, 1, 2 \dots M-1$ are defined as

$$x_{(m)}[n] = \begin{cases} x[n] & mN_{shift} \leq n < mN_{shift} + K \\ 0 & \text{otherwise} \end{cases}$$

such that K is the number of samples per frame and N_{shift} is the number of samples moved forward in fast time for each step in the slow time. For a time sequence $x[n]$ of fast time length N this means that there are $M = \left\lfloor \frac{N-K}{N_{shift}} \right\rfloor + 1$ steps taken in slow time.

Cutting the received data into segments causes edge effects. To reduce them, a Hanning window function $w_H[n]$ is applied to the data time frame. The discrete Fourier transform (DFT) is used to transform the windowed signal as

$$\begin{aligned} x_{k,m} &= \sum_{l=0}^{K-1} w_H[l] x_{(m)}[mN_{shift} + l] e^{-i2\pi \frac{k \cdot l}{K}} \\ &= \mathscr{F}\left\{ w_H[n] x_{(m)}[n] \right\} \end{aligned}$$

where k is the discrete frequency spectrum index.

Notice that in the time domain the frame index m, if ever lost, can be reconstructed by looking at which part of the signal is zero and which is not. The index can therefore be considered optional and it has been placed between parenthesis to emphasize this. The spectrum of this specific frame however does not contain this information and therefore, if any transformed signals are considered, the index is no longer optional and will not be placed in parenthesis. Hence the notation $x_{k,m} = \mathscr{F}\left\{ x_{(m)}[n] \right\}$.

The resolution of this spectrum estimate can be derived from the sample frequency and the size of the DFT. The Nyquist theorem states that the highest frequency which can be represented with certainty is half the sampling frequency. The total spectral width is going to be represented by K samples, therefore the bandwidth represented by a single frequency sample equals $\frac{F_s}{2K}$. This equation leads to the insight that a larger time frame leads to smaller channels. If the number of DFT points is large enough, the frequency components in the STFT can be considered narrowban - the situation we were looking for.

The complete derivation of the STFT can be found in appendix B. The conclusions which is most relevant is that at slow time index m and in frequency channel k the measurement taken by a single receiver can be modelled as

$$x_{k,m} = h_k s_{k,m} + e_{k,m}$$

This is the scenario which has been discussed in the previous section and for which we can analyse the narrowband channel behaviour in the following way. What a source contributes to al receivers can be written as

$$\boldsymbol{x}_{k,m} = \boldsymbol{h}_k s_{k,m} + \boldsymbol{e}_{k,m}$$

with $\boldsymbol{h}_k$ containing all coefficients of the k^{th} frequency channel.

The contributions of all sources in this specific time frequency instance can be stacked in $\boldsymbol{s}_{k,m}$ and their corresponding channel coefficients, indicated as $\boldsymbol{h}_{k,i}$, are cascaded which leads to

$$\begin{aligned} \boldsymbol{x}_{k,m} &= \left[\boldsymbol{h}_{k,1}\, \boldsymbol{h}_{k,2} \dots \boldsymbol{h}_{k,R-1} \right] \cdot \boldsymbol{s}_{k,m} + \boldsymbol{e}_{k,m} \\ &= \boldsymbol{H}_k \boldsymbol{s}_{k,m} + \boldsymbol{e}_{k,m} \end{aligned}$$

This R-element vector is of great importance and will be referred to as a time-frequency bin or tf-bn: The measurements of all microphones for a given discrete time-discrete frequency combination k, m. Assuming time invariance for the channel, combine the signal symbols and array measurements over slow time as $\boldsymbol{S}_k = \begin{bmatrix}\boldsymbol{s}_{k,0}\, \boldsymbol{s}_{k,1} \dots \boldsymbol{s}_{k,M-1}\end{bmatrix}$ and $\boldsymbol{X}_k = \begin{bmatrix}\boldsymbol{x}_{k,0}\, \boldsymbol{x}_{k,1} \dots \boldsymbol{x}_{k,M-1}\end{bmatrix}$ respectively. The data is then calculated as

$$\boldsymbol{X}_k = \boldsymbol{H}_k \boldsymbol{S}_k + \boldsymbol{E}_k$$

with $\boldsymbol{X}_k$ being a complex $R \times M$ matrix and $\boldsymbol{E}_k$ the error introduced by noise of the same size. The complete received dataset $\mathscr{X}$ is therefore the tensor containing $0, 1, \dots K-1$ channels and is of size $R \times K \times M$.

2.3. PROBLEM STATEMENT

This thesis sets out to implement an algorithm to improve the reconstructed speech' quality and intelligibility. The goal is, given the data tensor $\mathscr{X} \in \mathbb{C}^{R \times K \times M}$ and while the channel impulse response and the original signal are unknown, find a demixing vector $\boldsymbol{w}_{k,m} \in \mathbb{C}^R$ for each time-frequency bin such that

$$\hat{s}_{k,m} = \boldsymbol{w}_{k,m}^H \boldsymbol{x}_{k,m}$$

is an estimate of the original signal at time-frequency instance k, m with noise and interferers suppressed. The local discrete spectrum can be reconstructed from this as

$$\hat{\boldsymbol{s}}_m^{(k)} = [\hat{s}_{0,m}\, \hat{s}_{1,m} \dots \hat{s}_{K-1,m}]^T$$

and from that, find that

$$\hat{s}_{(m)}[n] = \mathscr{F}^{-1}\left\{\hat{\boldsymbol{s}}_m^{(k)}\right\}$$

is the local reconstruction of frame m. The superscript (k) is not intended as an index but as a clarification that the vector contains values from the discrete frequency domain. These estimates are sequenced for the complete estimate to be equal to

$$\hat{s}[n] = \left[\hat{s}_{(0)}[n],\, \hat{s}_{(1)}[n-K], \cdots, \hat{s}_{(M-1)}[n-(m-1)K]\right]$$

If multiple signals are reconstructed at the same time, combine the individual demixing vectors as

$$\boldsymbol{W}_{k,m} = [\boldsymbol{w}_{k,m}^{(1)}\, \boldsymbol{w}_{k,m}^{(2)}]$$

where we call $\boldsymbol{W}_{k,m}$ the beamformer and construct all spectrum estimates at once as

$$\hat{\boldsymbol{s}}_{k,m} = \boldsymbol{W}_{k,m}^H \boldsymbol{x}_{k,m}$$

2.3.1. SIGNAL ACTIVITY & CONTINUITY

Most research into the cocktail party problem contains two conflicting assumptions about the number of sources in each time-frequency bin.

1. A speech signal is so sparse in the time-frequency domain that no two users are active in the same bin.
2. Data from each time-frequency bin will contribute to the reconstruction of a user.

Sparseness, however, is a poorly defined quality and even if one signal and one interferer were 'sparse enough' to not overlap, this says nothing about N interferers. Still, techniques are often generalized and research rarely delves into performance analyse covering what happens when signals do overlap.

Adding the contribution of an 'empty' bin equals adding noise to a users' reconstruction. While near-continuous activity in time is possible, actual conversations tend to have a lot of pauses in them. The reason continuous activity is assumed, is to ensure an equal number of demixing vectors for each bin, often one per user. This simplifies the BSS-problem severely.

Speech is simply too volatile to assume that each tf-bin will contain exactly one user and therefore in this thesis we will not assume this. Instead it is assumed that a signal starts and stops abruptly in a multitude of frequency channels. Where one or multiple users are detected, a demixing matrix should be made available for each user based on the source separation algorithm. This thesis sets out to implement a number-of-users-detection and a signal separating beamformer.

To describe the new assumption of discontinuity completely, the system must be designed in such a way that it can handle a changing number of users. Three situations must be considered:

1. The number of users increases
2. The number of users decreases
3. The number of users stays the same

If the number of users increases, one demixing vector must be assigned to the current user and the leftover demixing vector to a newly introduced user. If the number of detected users decreases because the one user stops while the other is still active, it must be decided which reconstruction is continued and which user we expect to have stopped transmitting. These two situations are primarily important for alignment and user identification.

The last situation which must be considered is when the signal stops and an interferer immediately begins. In this case, the estimated number of users does not change over time. To cover this case, a one to one comparison must be made between sequential beamformers to see if they might belong to the same user. This thesis sets out to find a metric with which to quantify the likeliness between beamformers. It also sets out to design an algorithm which decides what changes in the detected number of users lead to what changes in the distribution of acquired beamformers.

2.3.2. AMBIGUITIES

Based on the data model derived earlier, two commonly known indeterminacies arise: The scaling and permutation problem. While popular approaches to solving exists, both can be considered unsolved. The solutions can best be introduced as the diagonal matrix $\mathbf{\Lambda}_{k,m}$ and the permutation matrix $\mathbf{\Pi}_{k,m}$.

From the model

$$\boldsymbol{x}_{k,m} = \boldsymbol{H}_k \boldsymbol{s}_{k,m} + \boldsymbol{e}_{k,m}$$

it can be seen that the data which is received, is indistinguishable from what would be received from a scenario altered according to any $\mathbf{\Pi}$ and $\mathbf{\Lambda}$ as

$$\begin{aligned}\boldsymbol{x}_{k,m} &= \boldsymbol{H}_k(\mathbf{\Pi}_{k,m}\mathbf{\Lambda}_{k,m}) \cdot (\mathbf{\Lambda}_{k,m}^{-1}\mathbf{\Pi}_{k,m}^{-1})\boldsymbol{s}_{k,m} + \boldsymbol{e}_{k,m} \\ &= \boldsymbol{H}'_k \boldsymbol{s}'_{k,m} + \boldsymbol{e}_{k,m}\end{aligned}$$

Because we want to estimate the signal as

$$\hat{\boldsymbol{s}}_{k,m} = \boldsymbol{W}_{k,m}^H \boldsymbol{x}_{k,m}$$

the mistakes made while trying to remove the channel effects depend on these two ambiguities. We define the error which is the result of our attempt to undo the channel effect as

$$\boldsymbol{W}_{k,m}^{H}\boldsymbol{H}_{k} = \boldsymbol{\Lambda}_{k,m}\boldsymbol{\Pi}_{k,m}$$

where $\boldsymbol{\Lambda}_{k,m}$ and $\boldsymbol{\Pi}_{k,m}$ are both of size $\hat{\boldsymbol{D}} \times \hat{\boldsymbol{D}}$.

The error $\boldsymbol{\Lambda}_{k,m}$ has to be solved for each reconstructed signal individually. Since it is based on scalar multiplication, this leads to an infinite number of matrices $\boldsymbol{\Lambda}_{k,m}$ which could have resulted in the array receiving this data. This thesis does not set out to solve this ambiguity.

The error $\boldsymbol{\Pi}_{k,m}$ is relevant for neighbouring time-frequency bins where this orientation is not the same. This can be exemplified by looking at the situation where the signal and an interferer are both present in frames m and $m+1$. The STFT of the interferer is denoted in the same way as the signal, which is for now $i_{k,m}$. If from bin m we reconstruct

$$\begin{bmatrix} \hat{s}_{k,m} \\ \hat{i}_{k,m} \end{bmatrix} = \boldsymbol{W}_{k,m}^{H}\boldsymbol{x}_{k,m}$$

and we define this outcome as being in the correct order, $\boldsymbol{\Pi}_{k,m} = \boldsymbol{I}$ then there is no guarantee that a new beamformer based on bin $m+1$ would not reconstruct

$$\begin{bmatrix} \hat{i}_{k,m+1} \\ \hat{s}_{k,m+1} \end{bmatrix} = \boldsymbol{W}_{k,m+1}^{H}\boldsymbol{x}_{k,m+1}$$

where it would have an permutation error as

$$\boldsymbol{\Pi}_{k,m+1} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

There is a finite number of permutation matrices, the set of which is denoted as $\mathscr{P}$ and therefore there is a finite number of errors that can be made. Ideally we construct $\Pi_{k,m}^{-1}$. To do so, this thesis sets out to find an algorithm which distributes the demixing vectors resulting from each individual bin among the available users.

2.3.3. USER RECOGNITION

The de-mixing vectors that are looked at for this thesis permute the local spectrum transforms into a single estimate in merely a quantitative way: The local reconstructions which are most alike, are clustered. There is not yet a definitive way to identify an individual user with this algorithm. To complete the process, a description is needed which decides what sequence of de-mixing vectors fits the desired signal. This thesis will not attend to this task to vividly by evaluating the system primarily by using the actual signal to evaluate reconstructions.

3

ALGORITHM

In this chapter, a step by step description is given of the algorithm used to solve the cocktail party problem a bit further. The input of the algorithm is the true data tensor $\mathscr{X}$ of size $R \times K \times M$. This is the collection of individual $\boldsymbol{X}_k$'s. A block scheme of the complete algorithm can be seen in figure 3.1.

3.1. NUMBER OF USERS DETECTION

The goal of this step is to determine what number of individual users is present in the data.

First we repeat the assumption that there are more receivers than sources in the array, $R > D$. Another look at the data model

$$\boldsymbol{X}_k = \boldsymbol{H}_k \boldsymbol{S}_k + \boldsymbol{E}_n$$

reveals that this means that the channel matrix is tall and that in the noiseless case, the data matrix $\boldsymbol{X}_k$ is of low rank. The rank can be found by using the singular value decomposition (SVD)

$$\boldsymbol{X}_k = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^H$$

where the number of singular values which are non-zero is equal to the number of rows in $\boldsymbol{S}_k$. This allows us to conclude that the rank is equal to the number of users.

$$D_k = \text{RANK}(\boldsymbol{X}_k) \qquad \text{for } \boldsymbol{E}_n = 0$$

However the data does contain noise: White, Gaussian, zero-mean noise and because of that the rank will always be full. The Gaussian noise has a covariance matrix $E[\boldsymbol{E}_n \boldsymbol{E}_n^H] = \boldsymbol{R}_n = \sigma_n^2 \boldsymbol{I}$, but because $\boldsymbol{X}_k$ is of

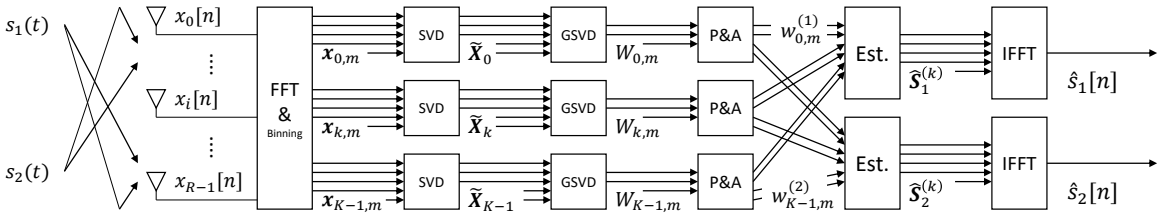


Figure 3.1: Block scheme of the complete process

limited size the assumption that this noise increases each singular value with σ_n is insufficient. To still use the singular values as the basis for our an activity detector, a subspace separation can be applied as

$$\boldsymbol{X}_k = \begin{bmatrix} \boldsymbol{U}_s & \boldsymbol{U}_n \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_s & \\ & \boldsymbol{\Sigma}_n \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_s^H \\ \boldsymbol{V}_n^H \end{bmatrix}$$

where the signal subspace is defined by the threshold $\boldsymbol{\Sigma}_s > \epsilon \boldsymbol{I}$. Here we use the threshold $\epsilon = \sigma_n(\sqrt{2M} + \sqrt{R})$ or slightly larger[4] to detect users. This corresponds to the highest singular value expected from the sample covariance matrix $\boldsymbol{E}_n \boldsymbol{E}_n^H = \hat{\boldsymbol{R}}_n$. From this partition, conclude that if $\boldsymbol{U}_s$ is of size $R \times \hat{D}$ where $\hat{D}$ is the number of detected users.

Earlier we have assumed that signals are not always active. Therefore, to achieve a higher time resolution, it is better to estimate the number of users in smaller windows rather than the complete data matrix $\boldsymbol{X}_k$. The most narrow window would be an individual time-frequency bin: $\boldsymbol{x}_{k,m}$. This is however not possible with the introduced approach because the number of users is smaller than R and if such a window is of size $R \times N_w$ then for $N_w \geq R$, it is certain be large enough. A smaller part of $\boldsymbol{X}_k$ is used to detect users more locally. This segment is called the detection window and defined as

$$\boldsymbol{X}_k[m] = \begin{bmatrix} \boldsymbol{x}_{k,m} \boldsymbol{x}_{k,m+1} \dots \boldsymbol{x}_{k,m+N_w-1} \end{bmatrix}$$

such that the index m of $\boldsymbol{X}_k[m]$ corresponds to the index m of the first bin $\boldsymbol{x}_{k,m}$. The threshold applied in the subspace separation can be altered to be usable by this window as $\epsilon = \sigma_n(\sqrt{2N_w} + \sqrt{R})$.

This section offers two things to the next step in the algorithm. First, windows $\boldsymbol{X}_k[m]$ of size $R \times N_w$ with a corresponding $\hat{D}$. Secondly, the requirement $N_w \geq R$.

3.2. SOURCE SEPARATION

The goal of this step is to find the signal separation beamformer. To achieve this, this thesis extends on the work of [4]. It compares the power of signals appearing in two detection windows, generally referred to as $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$. That work finds a beamformer which reconstructs the column subspaces dominant in $\boldsymbol{X}_1$ and nullifies the column subspaces dominant in $\boldsymbol{X}_2$.

In our situation a part of this method can be used to compare the detection windows as they are described in the previous section. There are multiple choices in our scenario for these detection windows. One might want to rid $\boldsymbol{X}_k[m]$ from interferers also active in $\boldsymbol{X}_k[m+1]$ or compare activity between $\boldsymbol{X}_k[m]$ and $\boldsymbol{X}_{k+1}[m]$. This section will use the former as an example.

To find a basis which both windows share, we apply the GSVD to find

$$\text{GSVD}(\boldsymbol{X}_k[m], \boldsymbol{X}_k[m+1]) \Leftrightarrow \begin{cases} \boldsymbol{X}_k[m] = \boldsymbol{F}\boldsymbol{C}\boldsymbol{U}^H \\ \boldsymbol{X}_k[m+1] = \boldsymbol{F}\boldsymbol{S}\boldsymbol{V}^H \end{cases} \tag{3.1}$$

where the columns of $\boldsymbol{F}$ are normalized left singular vectors for both windows. The matrices $\boldsymbol{C}$ and $\boldsymbol{S}$ are diagonal and their ratios $\frac{c_{i,i}}{s_{i,i}}$ are the generalized eigenvalues.

Based on [16] it can be said that if the generalized singular values are unique, $\boldsymbol{F}$ can be considered a joint diagonalizer which is unique up to a scaling and permutation. This means that for each column in $\boldsymbol{H}_k$ there is a scaled variant in $\boldsymbol{F}$. However $\boldsymbol{F}$ is defined as square and $\boldsymbol{H}$ is not. It still contain subspaces only contributed by $\boldsymbol{E}_{k,m}$. Ideally, this could be removed in a way similar to the number of users detection: Finding ϵ' such that $c_{i,i}, s_{i,i} > \epsilon'$ is the bases of the segmentation $\boldsymbol{F} = [F_s | F_n]$. Sadly enough, no theory was found on the

distribution of generalized eigenvalues in the GSVD. It can be said that the channels' subspace directly exists in $\boldsymbol{F}$ but the noise subspace has to be removed in an other way.

We will remove the noise subspace by rank reduction as shown in the previous section. To prevent counting one user as two, simply because it exists in both windows, apply the SVD to both windows at the same time. The new decomposition

$$\begin{bmatrix} \boldsymbol{X}_k[m] & \boldsymbol{X}_k[m+1] \end{bmatrix} = \begin{bmatrix} \boldsymbol{U}_s & \boldsymbol{U}_n \end{bmatrix} \begin{bmatrix} \boldsymbol{\Sigma}_s & \\ & \boldsymbol{\Sigma}_n \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_s^H \\ \boldsymbol{V}_n^H \end{bmatrix} \qquad \boldsymbol{\Sigma}_s > \epsilon$$

for $\epsilon = \sigma_n(\sqrt{2(2 \cdot N_w)} + \sqrt{R})$. This allows us to perform the rank reduction

$$\tilde{\boldsymbol{X}}_k[m] = \boldsymbol{U}_s^H \boldsymbol{X}_k[m] \qquad \tilde{\boldsymbol{X}}_k[m+1] = \boldsymbol{U}_s^H \boldsymbol{X}_k[m+1]$$

as a preprocessing step to ensure the right number of sources are separated by

$$\text{GSVD}(\tilde{\boldsymbol{X}}_k[m], \tilde{\boldsymbol{X}}_k[m+1]) \Leftrightarrow \begin{cases} \tilde{\boldsymbol{X}}_k[m] = \boldsymbol{F}\boldsymbol{C}\boldsymbol{U}^H \\ \tilde{\boldsymbol{X}}_k[m+1] = \boldsymbol{F}\boldsymbol{S}\boldsymbol{V}^H \end{cases}$$

This ensures there is no common noise subspace in $\boldsymbol{F}$ and the separating beamformer of size $D \times R$ becomes

$$\boldsymbol{W}^H = \boldsymbol{F}^{-1} \boldsymbol{U}_s^H$$

where $\boldsymbol{F}$ is of size $\hat{D} \times \hat{D}$ and $\boldsymbol{U}_s^H$ is of size $\hat{D} \times R$.

Because all windows are compared to a window adjacent to them, this section offers a separating beamformer $\boldsymbol{W}_{k,m}$ for each detection window except the last. This step offers the beamformer $\boldsymbol{W}_{k,m}$ and the rank-reduced data matrices $\tilde{\boldsymbol{X}}_k[m]$ to the next step in the algorithm.

3.3. PERMUTATION & ALIGNMENT

This thesis proposes using correlation ρ as a likeliness measure between two demixing vector. The de-mixing vectors can come from both adjacent time windows by comparing $\boldsymbol{w}_{k,m}$ and $\boldsymbol{w}_{k,m+1}$ and from neighbouring frequency bins by comparing $\boldsymbol{w}_{k,m}$ and $\boldsymbol{w}_{k+1,m}$. To simplify notation for this explanation consider two demixing vectors in a general notation as $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ of size R and the data they are derived from as $\boldsymbol{X}$ of size $R \times N$.

3.3.1. LIKELINESS MEASURE

Two approaches to this likeliness measure are: Comparing demixing vectors of $\boldsymbol{w}_i \in \mathbb{C}^R$ as

$$\rho_f(\boldsymbol{w}_1, \boldsymbol{w}_2) = \frac{|\boldsymbol{w}_1^H \boldsymbol{w}_2|}{|\boldsymbol{w}_1||\boldsymbol{w}_2|} \tag{3.2}$$

and comparing the reconstruction $\hat{\boldsymbol{s}}_i = \boldsymbol{w}_i^H \boldsymbol{X} \in \mathbb{C}^{1 \times N}$ made by the demixing vector

$$\rho_r(\boldsymbol{w}_1, \boldsymbol{w}_2, \boldsymbol{X}) = \frac{|\hat{\boldsymbol{s}}_1 \hat{\boldsymbol{s}}_2^H|}{|\hat{\boldsymbol{s}}_1||\hat{\boldsymbol{s}}_2|} = \frac{|\boldsymbol{w}_1^H \boldsymbol{X}\boldsymbol{X}^H \boldsymbol{w}_2|}{|\boldsymbol{w}_1^H \boldsymbol{X}||\boldsymbol{w}_2^H \boldsymbol{X}|} \tag{3.3}$$

Both are commutative mappings $\rho_f : \mathbb{C}^{R \times R} \mapsto [0,1]$, respectively $\rho_r : \mathbb{C}^{N \times N} \mapsto [0,1]$.

Based on the amount of microphones, the number of multiplications to compare just the filter coefficients can be significantly less than the number required to compare the reconstruction. A consequence, however is that for overdetermined problems $R > D$, wildly varying filters can lead to the same reconstruction.

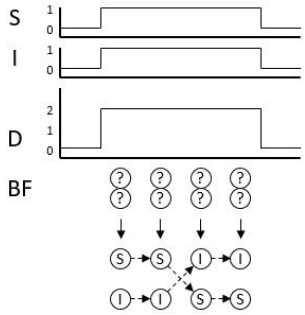


Figure 3.2: An example of how beamformers might be permuted.

Reconstructions by the demixing vectors, based on the same signals inherently should have a likewise result. This allows the system to compare different beamformers. These could be as instances of the same algorithm or to compare outcomes from different algorithms. GSVD-beamformers however explicitly contain the information required to differentiate between active channels. This advantage could be lost and permutation can become difficult if signal reconstructions are to similar[17]. Using the true signal $\boldsymbol{s}[n]$ in this comparison allows for easy evaluation of the algorithm.

Since we are dealing with an overdetermined system, we will use ρ_r as our likeliness measure.

3.3.2. PERMUTATION

The above metric is sufficient to compare adjacent bins or windows if they both expect and have the same number of beamformers. In this section we look at the situation where two adjacent detection windows contain the same number of active users.

Two beamforming matrices are generated according to the GSVD-algorithm. For simplicity they are referred to as $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$, are of equal size and the columns of $\boldsymbol{W}_2$ are randomly permuted. If the correct permutation is written as $\boldsymbol{W}_2^*$, the solution to this mix-up is the permutation matrix which solves $\boldsymbol{W}_2^* \boldsymbol{\Pi}^* = \boldsymbol{W}_2$.

Define that the beamformers are of size $R \times D$ and are a cascade of de-mixing vectors as

$$\boldsymbol{W}_i = \left[\boldsymbol{w}_{i,0}, \boldsymbol{w}_{i,1}, \ldots \boldsymbol{w}_{i,D-1}\right]$$

We construct the matrix $\boldsymbol{C}$ of size $D \times D$ which contain all correlations between the columns of the first and second beamformer as

$$c_{i,j} = \rho_r(w_{1,i}, w_{2,j})$$

where we leave out $\boldsymbol{X}_k$ for convenience. Also denoted as $\boldsymbol{C}(\boldsymbol{W}_1, \boldsymbol{W}_2)$, the order of the columns of $\boldsymbol{W}_1$ is preserved in the row order of $\boldsymbol{C}$ and the column order of $\boldsymbol{W}_2$ is preserved in the column order of $\boldsymbol{C}$. If $\boldsymbol{W}_2$ was ordered correctly, the de-mixing vectors pointing towards the same user would find each other on the diagonal of $\boldsymbol{C}$. Finding the optimal permutation is also equivalent to finding the maximum correlation sum as shown in appendix C. There it is shown that it can be concluded that

$$Tr(\boldsymbol{C}(\boldsymbol{W}_1, \boldsymbol{W}_2^*)) > Tr(\boldsymbol{C}(\boldsymbol{W}_1, \boldsymbol{W}_2))$$

Equivalently the optimal permutation matrix $\boldsymbol{\Pi}^*$ becomes

$$\boldsymbol{\Pi}^* = arg \max_{\boldsymbol{\Pi} \in \mathscr{P}} Tr(\boldsymbol{C}\boldsymbol{\Pi})$$

This optimization problem can be solved iteratively by a greedy algorithm: For each row $i = 0, 1 \ldots D-1$, permute the largest value on the i^{th} place, equivalent to placing them on the diagonal. An example of how beamformers might be permuted can be seen in figure 3.2.

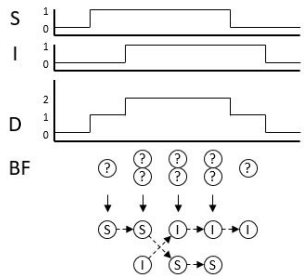


Figure 3.3: The changing number of users and how they could be permuted.

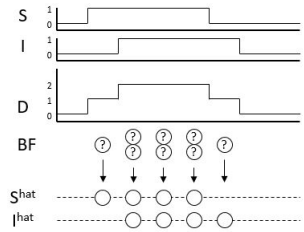


Figure 3.4: The result of alignment: Identified users and assigned beamformers.

3.3.3. CHANGING NUMBER OF USERS

If the two adjacent detection windows do not have the same number of expected users, one user that was previously assigned a de-mixing vector should be considered inactive from then on forward. A comparable situation arises when the number of expected users increases. The system should consider adding an additional speaker to listen to. The de-mixing vector initially left out during the optimization process can be considered as the first de-mixing vector for that user. Both of these decisions can still be made with the approach explained above. The only difference is that the smaller set of beamformers is complemented with a zero vector before comparison and whomever is assigned to it can be considered the exception case. These situations do not alter the system a lot. An example of how a changing number of users might influence the permutation can be seen in figure 3.3.

What does seriously alter our case is when we consider that one user can stop and an other can start with zero detection windows in between. In that case the number of expected users stays the same while the system should conclude that the most recent beamformer does not contain a vector belonging to the previously tracked user. This requires a lower bound on correlation between alike demixing vectors.

If two consecutive beamformers can be considered to be perturbed variants of each other, up to what difference can it be concluded that they reconstruct the same source $\mathcal{H}_1$ or not $\mathcal{H}_0$? For this decision introduce a threshold γ which completes the hypothesis testing problem which answers

$$\mathcal{H}_0 : \rho_r(w_1, w_2) < \gamma$$
$$\mathcal{H}_1 : \rho_r(w_1, w_2) > \gamma$$

and which defines when two de-mixing vectors can be considered to be perturbed variants of each other.

3.3.4. ALIGNMENT

Placing the correct sequence of beamformers with the right user is the final step of our algorithm. As stated earlier, this will be done by comparing the reconstructions received from the source separation step and the original signal. To find the demixing vector that best reconstruct this and is also still eligible after the one to

one comparison, our selection is reduced to

$$\boldsymbol{w}_{k,m}^{*} = arg \max_{\boldsymbol{w}_{k,m} \in \boldsymbol{W}_{k,m}} \rho_r(\boldsymbol{w}_{k,m}, \boldsymbol{X}_k, \boldsymbol{S}_k) \quad s.t.\, \rho_r > \gamma$$

which can be done for each of the approaches.

This step adds the requirement that we have the original signal available.

3.4. RECONSTRUCTION

As we stated in section 2.3, each tf-bin $\boldsymbol{x}_{k,m}$ which has had a demixing vector assigned result in the estimates

$$\hat{s}_{k,m} = \boldsymbol{w}_{k,m}^{H} \boldsymbol{x}_{k,m}$$

which are combined in the k-direction into the local spectrum as

$$\hat{\boldsymbol{s}}_m^{(k)} = \begin{bmatrix} \hat{s}_{0,m} & \hat{s}_{1,m} & \dots & \hat{s}_{K-1,m} \end{bmatrix}^T$$

and those are cascaded into the STFT as as

$$\hat{\boldsymbol{S}}^{(k)} = \begin{bmatrix} \hat{\boldsymbol{s}}_0^{(k)} & \hat{\boldsymbol{s}}_1^{(k)} & \dots & \hat{\boldsymbol{s}}_{M-1}^{(k)} \end{bmatrix}$$

which we called the STFT before.

To obtain the time estimate for this user, apply the inverse Fourier transform defined as

$$\hat{s}_{(m)}[n + (m-1)N_{shift}] = \mathscr{F}^{-1}\left\{\hat{\boldsymbol{s}}_m^{(k)}\right\} = \sum_{k=0}^{K-1} \hat{s}_{k,m} e^{-j2\pi \frac{kn}{K-1}}$$

to each of the local spectra $m = 0, 1, \dots M-1$ which are sequenced to fit

$$\hat{s}[n] = \begin{bmatrix} \hat{s}_{(0)}[n] & \hat{s}_{(1)}[n + N_{shift}] & \dots & \hat{s}_{M-1}[n + (M-1)N_{shift}] \end{bmatrix}$$

3.5. OUTPUT POWER

As described in the problem description, a sound solution for this step would be to find the matrix $\boldsymbol{\Lambda}^{-1}$. This problem is generally still unsolved, even though popular approaches exist[18]. Our implementation in speech processing offers a solution. Since the final receiver of a reconstructed signal is a person, the signal can be scaled at the last possible moment to be played at what is called the Most Comfortable Loudness (MCL)[19] which is broadly researched for both preference and intelligibility. [20].

4

SCENARIO & PERFORMANCE METRICS

The goal of this chapter is to summarize the previous two chapters and explain the scenario as a whole. Also it describes how the quality of the system we have designed is measured.

Where previously the system was divided into a data model and the algorithm, this chapter will reshape that. While the unavoidable perturbations affecting the signal are all found in the data model, not everything from that chapter is written in stone: The DFT-size and the number of microphones for example. To emphasize this, what decisions can be made in tuning the system are grouped together. This new division of Environment versus System is visualized in figure 4.1.

In the Scenario section we explain which of the parameters of the system are used as variables. These will be the inputs to our simulation for which we want to see how they influence the algorithm's performance. In the Performance Metric section, it is explained which points in the system are looked at and with what metrics we define that performance.

4.1. SCENARIO

Goal of this section is to introduce all parameters relevant to our simulation. Those for which we chose a single value are the constants and those for which we scan a range of values are the variables. A summary of both can be found in table 4.1.

4.1.1. SYSTEM

The goal of this subsection is to introduce the variables which have to do with the processing of the data. They are variables which we control and therefore which we can use to tune the system.

TIME FRAME SIZE

The implementation of the first transform which the system performs, the STFT, is mostly covered in appendix B. What is left as a parameter is the size of the time frame K. This decision influences the length of the DFT, therefore the number of frequency channels and the resulting number of steps in slow time. The balancing of precision in the spectral and the temporal domain is called the Gabor-limit and it is a specific version of the uncertainty principle. In literature this frame size is also referred to as the NFFT.

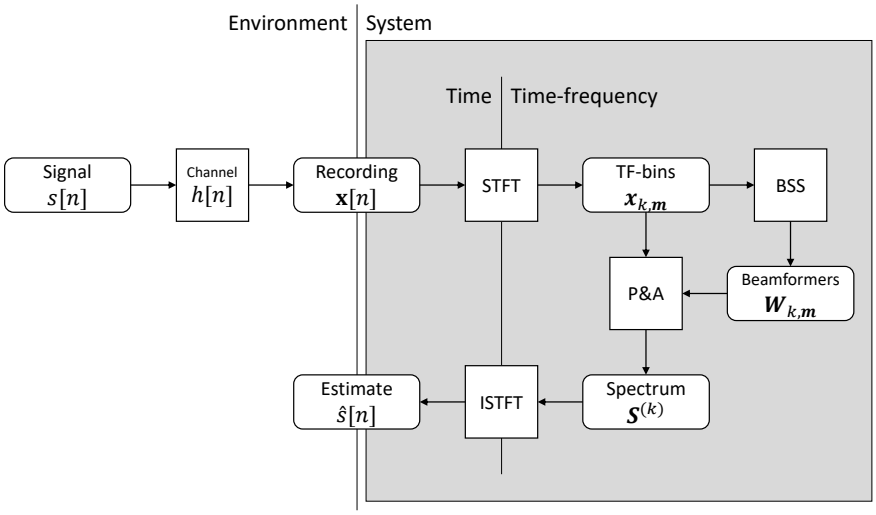


Figure 4.1: Visualisation of the complete scenario

Table 4.1: Values and ranges

(a) Constants

Constants	Value
Room size	$4 \times 3 \times 2\,(x,y,z)\,meter$
Speed of sound	$343\,m/s$
Sampling	$16\,kHz$
Source angle	$-10°$
Element distance	$1cm$
Additional delay	$0.5s$
Receivers	10
Window size	10

(b) Variables

Variable	Range	*(Default)* Value
SNR	$[0,15]\,dB$	$10\,dB$
SIR	$[-5,10]\,dB$	$3\,dB$
Angle of separation	$[10°,60°]$	$50°$
Reverberation time	$[100,300]\,ms$	$100\,ms$
DFT-points	$[32,1024]$	256

Because speech consists of harmonics, an NFFT that is too high can result in empty frequency channels in between the signal. It also gives the algorithm a high latency and this makes it less applicable in real-time situations.

Low NFFT makes for an easier hardware implementation since it required less filter-taps. Since the considered frequency is limited, a short time segment is bound to be more stationary. What is unfortunate is that low spectral resolution will make the signal and interferers overlap faster.

Lest not forget that a wide frequency channel results into a low time-bandwidth product of which the importance was explained in appendix A. If the inter element distance is denoted as d and the speed of sound through air as v_{air} then the time bandwidth product from a uniform linear array can be calculated as

$$\begin{aligned} time-bandwidth &= \tau \cdot W \\ &= \frac{d(R-1)}{v_{air}} \cdot \frac{Fs}{\frac{1}{2}\text{NFFT}} \end{aligned}$$

which, with our constants applied becomes $tbw = \frac{0.84}{\text{NFFT}}$. This can be used as a rule of thumb for when predicting the required number of DFT-points.

A literature standard is to use the time frames of 10ms wide. In this width we can expect each frame to

contain at most one phoneme and that should give a fairly stationary signal. At 16 kHz sampling rate, this boils down to 160 samples. Adding the 50% overlap on each side makes for a total of 320 samples. We use 256 as our default NFFT as it is close to 320 but also a known hardware standard. For exactly the same reason, the upper limit we explore is 1024. Based on emperical evidence we offer 32 as the lower limit

4.1.2. ENVIRONMENT

Goal of this section is to introduce the variables which have to do with data generation. They are the variables which we do not control. We simply see how the system performs under these circumstances.

POWER RATIOS

Well known descriptions for signal processing environments, the signal-to-noise ratio (SNR) and the signal-to-interferer ratio (SIR) require some additional explanation in the field of audio processing. For the non-stationary and non-continuous case which speech as a signal presents, their definitions are different than common in literature.

First, a clairevoyant detector defines the activity regions $\mathscr{A}$ as those samples in the fast time in which the signal is active. The ideal time signal is cut into blocks of a small number of samples and if the energy in this block surpasses a certain threshold, all samples in that block are marked as active. This way, the presence of the speech signal is assumed to be precisely known. This ideal detector allows us to negate the effects of faulty detectors and focus on the algorithm at hand. The signal power is therefore calculated as

$$P_s = \sum_{n \in \mathscr{A}} |s_1[n]|^2$$

The interference power is defined by that part of the interferer which exists in the active regions as

$$P_i = \sum_{n \in \mathscr{A}} |s_2[n]|^2$$

To simulate an environment with a certain SIR, the interferer is scaled in such a way that the part which overlaps with the signal provides that power. If the current $P_{i,c}$ interferer power is known and if the desired interferer power is calculated as

$$P_{i,d} = P_s \cdot 10^{\frac{-SIR}{10}}$$

then the interferer can be scaled by $\frac{P_{i,d}}{P_{i,c}}$. To simulate an environment with a certain SNR, noise with the correct power is generated according to

$$P_n = P_s \cdot 10^{\frac{-SNR}{10}}$$

CHANNEL BEHAVIOUR

The constants applicable to the uniform array are the speed of sound, the element distance and the amount of microphones. It is primarily characterizes by the angular distance between the DOA of the source and that of the interferer. An illustration of this can be found in figure 4.2. The smaller the angular distance, the more alike the channel vectors $\boldsymbol{a}(\theta_s)$ and $\boldsymbol{a}(\theta_i)$ become, the more difficult to distinguish the users based on their channel use. Experimenting with the default values of the system has shown that $\Delta\theta = 10°$ is problematically close and that $\Delta\theta = 60°$ offers little challenge to the system. The default value $\Delta\theta = 50°$ should give a good separation, based on empirical evidence.

The convolutional model introduces an additional variable: The reverberation time T_{60}. Rooms typically have reverberation times of $100ms$ up to $300ms$. The response depends on the locations of the transmittors

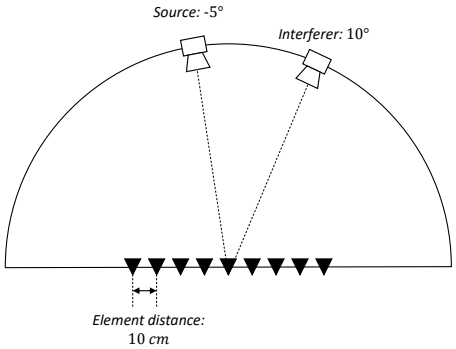


Figure 4.2: Layout of the uniform linear array in 2D as it receives in the far field.

and receivers in comparison to both oneanother and the room. Our realisation and an example of the resulting room impulse response can be seen in figure 4.3. We are choosing a small default value which allows us to see the effects of other parameters more clearly. The lower bound of $100ms$ is used as a default value.

4.2. PERFORMANCE METRICS

The goal of this section is to introduce what combinations of system blocks we look at and explain the metrics that are used to measure their performance. There are three combinations of blocks considered and we compare the effect of the input power ratios for all three metrics.

4.2.1. CORRELATION COEFFICIENT

The performance of the BSS-block is described by the correlation coefficient ρ. To measure this performance actively, a form which uses information that is known to the system, is required. The separation performance of a single demixing vector $\boldsymbol{w}_{k,m}$ can best be evaluated by applying it to all of the data available in the k^{th} channel and compare that to the signal. This is however not possible in real time and it is more realistic to measure performance based the local estimate and apply the demixing vector to the current detection window. To conclude, for each demixing vector $\boldsymbol{w}_{k,m}$, calculate the local estimate $\hat{\boldsymbol{s}}_{(m)} = \boldsymbol{w}_{k,m}^H \boldsymbol{X}_k[m]$ and correlate that to the signal originally in that detection window $\boldsymbol{S}_k[m]$

While evaluating the system, average correlation coefficient is known. We will use that to tune the the threshold. To start out with, $\gamma = 0.5$ is chosen based on empirical results. As stated in section 3.3.4, our procedure is to find

$$\boldsymbol{w}_{k,m}^* = arg \max_{\boldsymbol{w}_{k,m} \in \boldsymbol{W}_{k,m}} \rho_r(\boldsymbol{w}_{k,m}, \boldsymbol{X}) \quad s.t.\ \rho_r > \gamma$$

which can be solved for each k, m combination.

4.2.2. SPECTRUM RECONSTRUCTION

The performance of our algorithm as a whole can be measured by looking at the difference between the signal estimate and the clean signal in the time-frequency domain. By looking at figure 4.1 it can be seen that by doing so, we look at the performance of the FD-BSS-step and P&A-step combined. Because the estimate still contains a random phase shift, only the amplitude envelops are compared: The absolute value of the STFT is defined as

$$|\boldsymbol{S}^{(k)}|_\circ = \left| [\boldsymbol{s}_0^{(k)} \boldsymbol{s}_1^{(k)} \dots \boldsymbol{s}_{M-1}^{(k)}] \right|_\circ$$

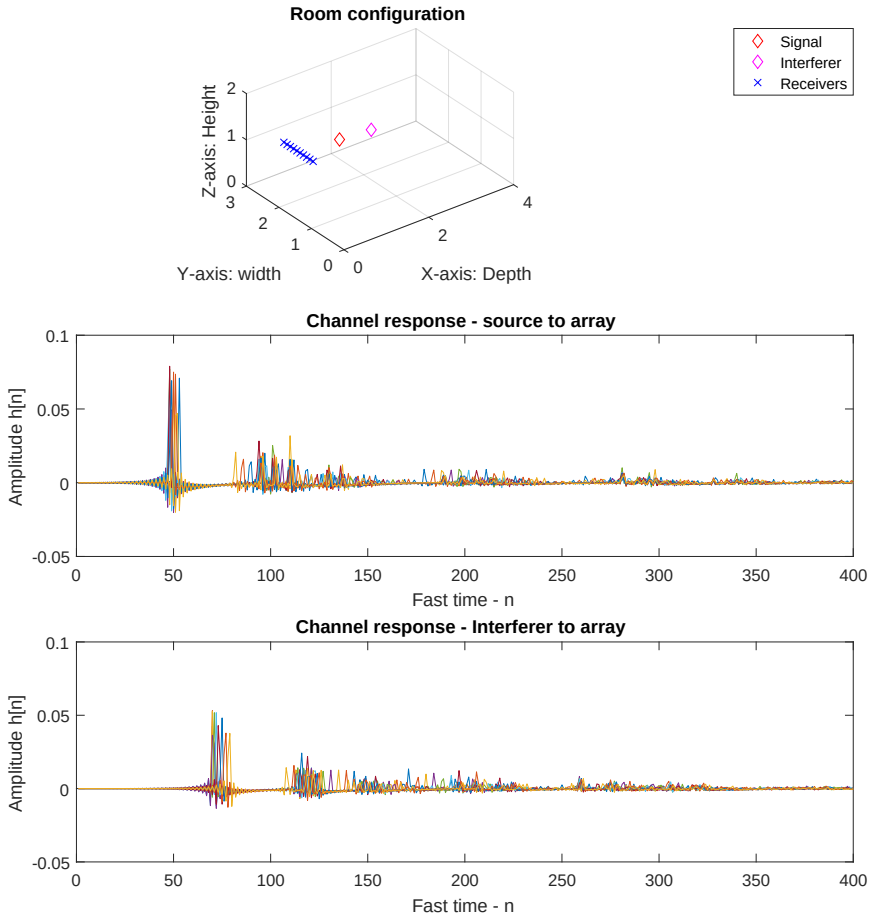


Figure 4.3: Top: Layout of the uniform linear array in 3D as it receives in the near field. Centre/Bottom: The channel response vectors as generated by the rir-generator.[21]

Table 4.2: The considered variable/metric combinations

	SNR	SIR	$\Delta\theta$	T_{60}	$NFFT$
Beamformer permutation	×	×	×		
STFT reconstruction	×	×		×	
Time estimate	×	×			×

It must be stated that this metric is only available in system evaluation: It requires clairvoyant information normally not available to the system. The error measure is defined as

$$e_F = \left|\left||\boldsymbol{S}^{(k)}| - |\hat{\boldsymbol{S}}^{(k)}|\right|\right|_F$$

where $|\cdot|_F$ indicates the use of the Frobenius norm and $|\cdot|_\circ$ the element wise absolute value.

This metric is also used to compare performance of our own algorithm to other beamformers. This is done in two different ways: Unmasked and masked. The unmasked approach offers the complete data tensor to compare to. This is fair in the way that is receives the same information as the complete algorithm. The masked approach offers only the tf-bins with one or more users to compare to. This is done to compare the performance of our separating beamformer.

Next to the power ratio's, this error is also evaluated by varying over the reverberation time of the room.

4.2.3. TIME SIGNAL ESTIMATE

As is known, the tensor used as the input for our algorithm is of size $R \times K \times M$ as a direct result of the design choices made on the system side. Altering these does make using the previous metrics challenging: The number of beamformers changes and the spectrogram becomes of a different size. Predicting what effect this has on the non-linear metrics to distinguish that from real performance issues is not often looked at. To compare tuning parameters which influence the size of the data, we look at the time domain reconstruction.

Ideally, the deterministic signal could be compared with the estimate through some algebraic distance. Sadly enough we can not because small phase-shifts in periodical signals generate large differences in the amplitude envelope. Our specific implementation in speech processing offers a solution. To determine the performance of the complete system, including (I)FFT and user identification, we suggest STOI[22] as an inteligibility measure.

5

SIMULATION

This chapter consists of three sections corresponding to the earlier explain performance metrics. Each section also explains how it is an adaptation from the previous simulation, starting with the technique initially proposed by Mu Zhou[4].

5.1. SINGLE FREQUENCY BIN

This section introduces speech instead of QPSK[1] as the signals of choise to the system. Also, apart from using just the instantaneous model, data are also generated through the convolutional model. First, the systems performance for a single frequency channel is looked at. Examples of what these signals and the data look like are given in figures 5.1 for the instantaneous channel and 5.4 for the convolutional channel. They also show the detection windows which are used in the following steps. Previously we have predicted that if a frequency channel is narrowband that $\boldsymbol{W}_{k,m}^{H}\boldsymbol{H} = \boldsymbol{\Pi}_{k,m}\boldsymbol{\Lambda}_{k,m}$. An indication of how well our beamformer will perform can be seen in figure 5.2. There is a clear diagonal structure in the separation error and the angular response of both demixing vectors is close to zero for the other user. The channel matrix in the frequency domain, however, is only available for the instantaneous model. In figure 5.3 and 5.5 a reconstruction is made with the help of each of the two demixing vectors and this is compared to the original signal. The reconstructions are normalized to the power of the signal to present them together in one image. This has not been done to

[1] Quadrature phase-shift keying(QPSK) is a complex valued alphabet based on the symbols $e^{\frac{i\pi}{2}k}$ for $k \in \{0,1,2,3\}$

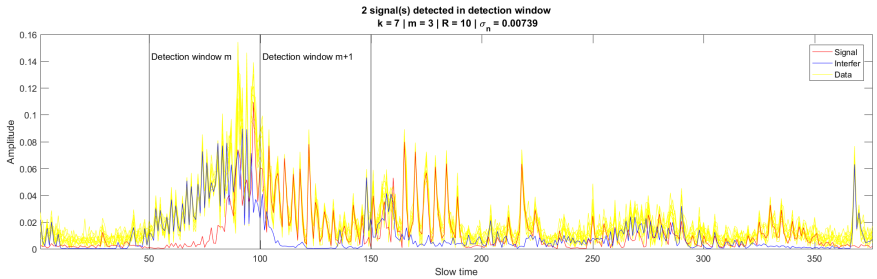


Figure 5.1: Signals and data as an example of a single bin, instantaneous channel simulation.

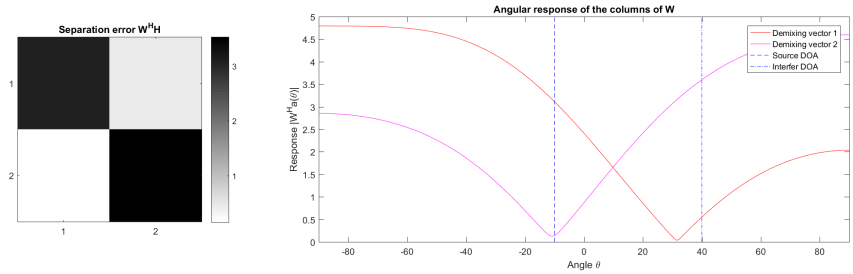


Figure 5.2: Performance visualized by the separation error $\boldsymbol{W}^H\boldsymbol{H}$ and angular response $|\boldsymbol{W}^H a(\theta)|$ for $\theta \in [-90, 90]$. The beamformer is based on the data and windowing as seen in figure 5.1.

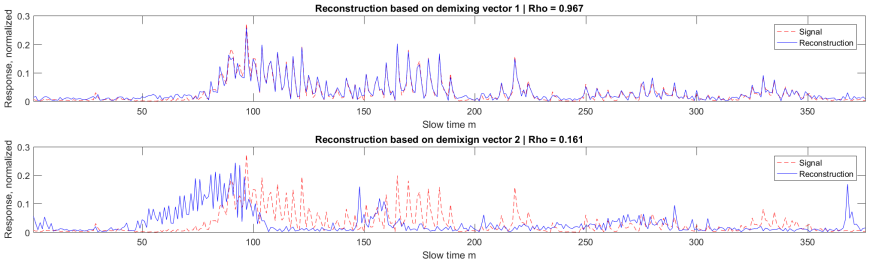


Figure 5.3: Two estimates based on the two demixing vectors resulting from the data and windowing as seen in figure 5.1. Rho calculated is ρ_r

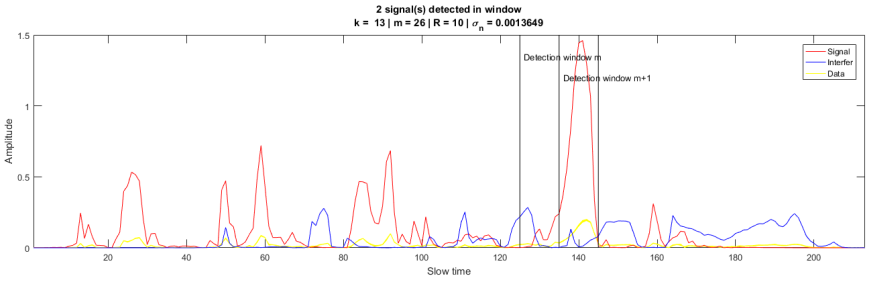


Figure 5.4: Signals and data used as an example of the single bin, convolutional channel simulation.

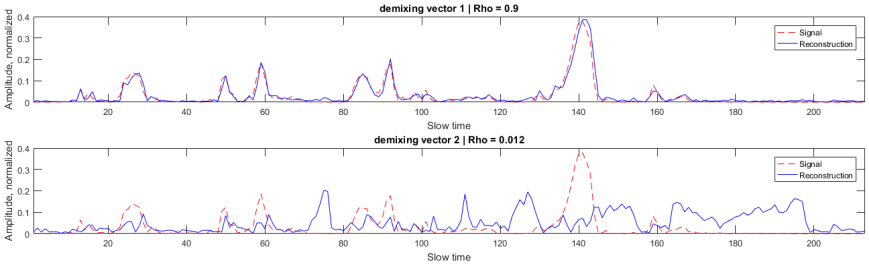
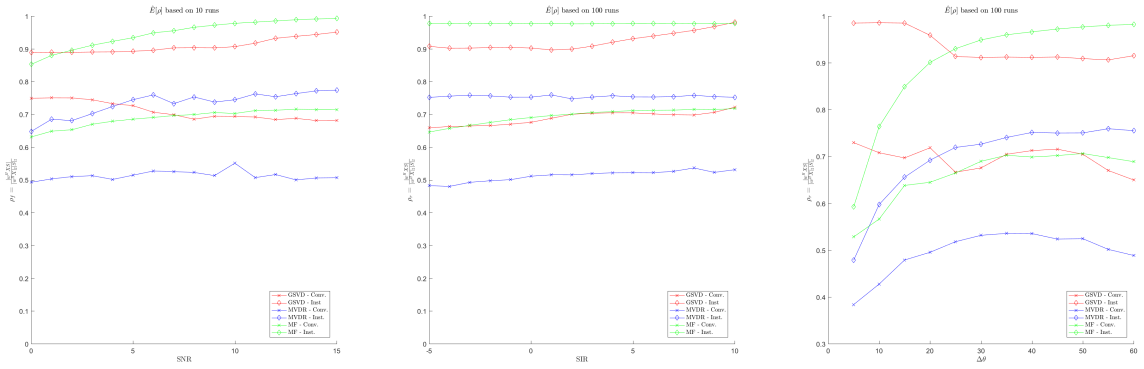
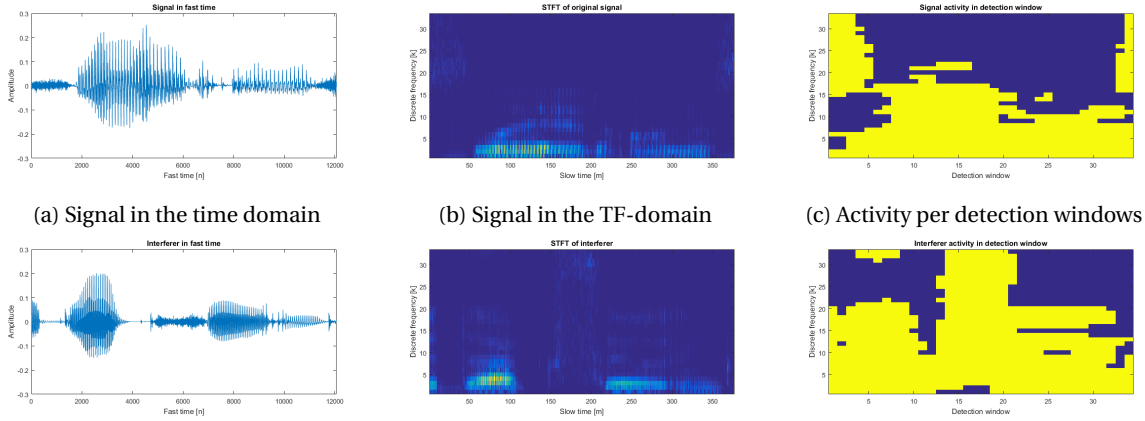


Figure 5.5: Two estimates based on the two demixing vectors resulting from the data and windowing as seen in figure 5.4. Rho calculated is ρ_r



(a) Effect of SNR on coefficient. (b) Effect of SIR on coefficient. (c) Effect of $\Delta\theta$ on coefficient.

Figure 5.6: The results of correlation coefficient experiments.



(a) Signal in the time domain (b) Signal in the TF-domain (c) Activity per detection windows

(d) Interferer in the time domain (e) Signal in the TF-domain (f) Activity per detection windows

Figure 5.7: The signal and interferer used in this simulation

calculate the correlation coefficient. We can see that each of the demixing vectors clearly reconstructs ether the signal or the interferer.

In figure 5.6 the sample mean of the correlation coefficient achieved by our beamformer and two others is compared. The beamformers with which our signal is compared are the MF-beamformer and the MVDR-beamformer[23]. In figure 5.6a and 5.6c an increase of the sample mean of the correlation coefficient can be seen. This is primarily caused by the system detecting less beamformers yet those that are detected reconstruct an estimate highly correlated to the original signal.

5.2. FULL CHANNEL

This section extends our perspective from looking at one channel towards considering all K channels. Because it will be different per channel, we also look at signal activity per detection window. A summary of the two speech signals, both shown in the relevant domains and supplemented with a clairvoyant activity description can be found in figure 5.7. The time-frequency domain representation of the received data can be seen in figure 5.8

The quality of the number of users detection is evaluated only by example and can be seen in figure 5.9.

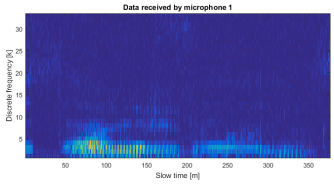
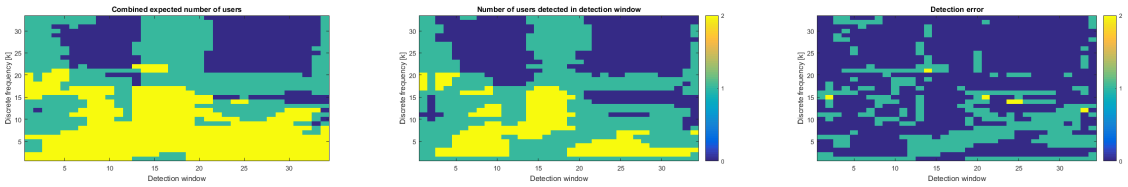


Figure 5.8: TF-domain representation of the data used in this simulation.



(a) Users active in window. (b) Users detected in window. (c) Error between them.

Figure 5.9: Error in the detected number of users based on the known activity as shown in figure 5.7c and figure 5.7f.

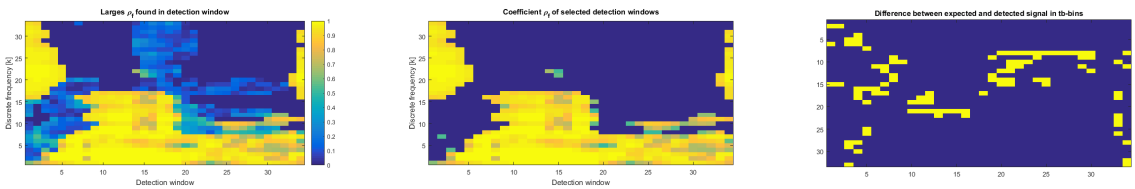
There certainly seem to be users missing in a number of detection windows. In figure 5.10 the three steps in selecting detection windows are shown. First, the largest correlation coefficient when the reconstruction is compared to the original signal and then those coefficients which are larger than the threshold. It can be seen there that there are a lot less detection windows missing. It must be concluded that the errors shown in 5.9c are primarily coming from the case where the interferer is not detected.

Specifically the outcome shown in figure 5.9b is of importance in our next simulation. It is the basis of the difference between what we will call the masked and unmasked comparison. To evaluate the quality of only the separating beamformer, the beamformers that we want to compare it with are only applied if the system detect a user. In that way, the beamformers receive the same information. This we will refer to as the activity mask. To evaluate the quality of both the separating beamformer and the signal detection step, the beamformers for comparison are always applied. It is more fair to offer the same information to both of the approaches. The masked and unmasked cases are both compared to in this and the next (STOI) experiment.

In figure 5.11 the Frobenius norm of the error between the normalized tf-domain representations of the signal and the reconstructions is shown for all considered cases.

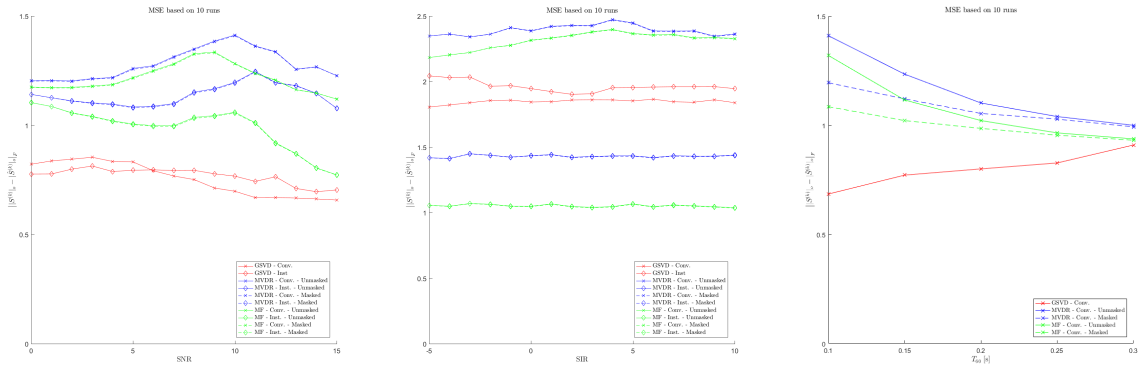
5.3. TIME ESTIMATE

This section looks at the system as a whole by applying the speech intelligibility metric STOI to the reconstructed time signal. This approach ensures that any additional limitations introduced, for example by the



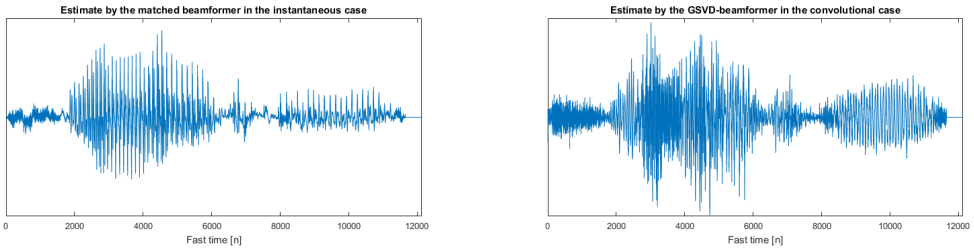
(a) Largest correlation coefficient ρ_r for each bin with detected activity (b) Coefficient ρ_r for selected detection windows (c) Error between selected windows and known activity as seen in 5.7c

Figure 5.10: TF-bins selection through $max(\rho_r|\rho_r > \gamma)$



(a) Effect of noise. (b) The effect of an interferer. (c) The effect of T_{60}.

Figure 5.11: The results of the stft-reconstruction experiment

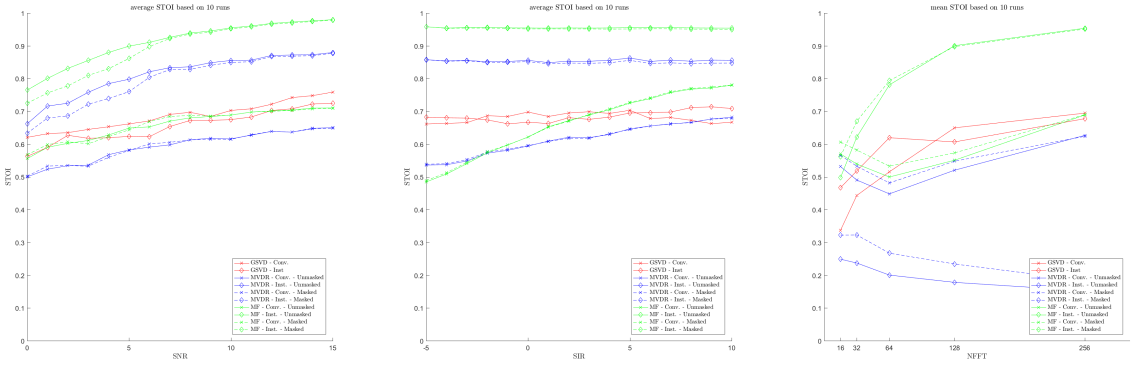


(a) An example of a strong reconstruction. $SNR = 15$ (b) An example of a poor reconstruction. $SNR = 0$

Figure 5.12: Estimates of the original time signal which itself can be seen in 5.7a

(I)STFT, are included and that possibly unheard errors are overlooked. The best time signal reconstruction can be seen in figure 5.12a while the poorest result from the GSVD-based beamformer can be seen in 5.12b.

In figure 5.13 the STOI for our selected beamformers can be found. It is in figure 5.13a that we see the performance of the masked case decrease faster than the unmasked case. This is the result of a decrease in the number of users detected and therefore less beamformers being applied. In figure 5.13b it can be seem that separation by the known beamformers is hardly influenced by the interferer. This can be expected because of the sizeable default difference in angle of arrival. For the convolutional case a strong decrease can be seen for the compared to interferers.



(a) The effect of noise. (b) The effect of an interferer. (c) The effect of the used NFFT-size.

Figure 5.13: The results of the intelligibility experiment

6

CONCLUSION

In this thesis we have applied the known narrowband BSS-technique to a speech signal in two environments.

We have adapted a design rule-of-thumb found in previous work to predict what size of DFT is required. This can be applied if a desired time-bandwidth is know and hardware limitations can be reconsidered. It has been shown that a larger DFT indeed results in more understandable reconstructions.

When enforcing narrowband behaviour by generating data with the instantaneous model, both the known beamformers and our new beamformer result in intelligible signals however we do not yet outperform them. Once the convolutional model is introduced, the new algorithm shows to be more robust to the limitations of te narrowband assumption than the others, even though it does not require prior information.

Leaning on the correlation coefficient as a measure of likeliness between reconstructions has shown to select almost all of the bins where our user is active.

Also, the theoretically SINR-optimal MVDR beamformer performs worse than the matched beamformer in almost all situations and worse than the GSVD-beamformer in most convolutional cases - presumably because of the small number of samples available to the sample autocorrelation matrix.

Using the Frobenius norm as a distance measure between two STFT's has shown counter intuitive results for all beamformers. Both in the situation when one of them was degraded with noise or when it was propagated through multipath channels. No conclusion on the quality of an estimated STFT can be drawn.

6.1. FUTURE RESEARCH

Currently the SVD is used as a preprocessing step to remove the noise space from the data. This can be solved more elegantly if there would be a description of the distribution of the generalized singular values $\boldsymbol{C}$ and $\boldsymbol{S}$. With such a measure, a distinction between $\boldsymbol{F} = [\boldsymbol{F}_s|\boldsymbol{F}_n]$ should come available.

Currently the selection threshold γ was decided upon by looking at the results of the correlation coefficient experiments. If sufficient information on both the speech signal and beamformer is considered, it may lead to a threshold based on a combination of first or second order statistical descriptions. Instead of choosing a functioning threshold, in this way the data could dictate a threshold for itself.

I expect that the definition of the delay in the time-bandwidth product has to be extended beyond what time it takes the line of sight to travel across the array. The delay can be better characterized by including a

power delay profile like a mean excess delay[24].

The fact that our algorithm performs reasonably well with a very small detection window, totalling only 20 samples, makes it more plausible that real time applications are possible. Looking further into this topic, we know user identification is still challenging. To avoid having to do it over and over, some long term memory, for example exponential memory, can be researched. This suggestion also fits the result shown in chapter 5 that stated that maybe a beamformer isn't always found but once it is, it's good. The system should be able to learn from these findings.

The theory surrounding the MF and MVDR-beamformers that has been consulted for their implementation was primarily based on linear arrays. To compare our work to it, all data was based on these forms. The GSVD-beamformer doesn't actually require that limitation and research into different shapes of hardware can lead, for example, to reducing the largest propagation delay while keeping the number of receivers constant.

BIBLIOGRAPHY

[1] B. Shield, *Evaluation of the social and economic costs of hearing impairment,* Hear-it AISBL , 1 (2006).

[2] A. R. Bradlow, G. M. Torretta, and D. B. Pisoni, *Intelligibility of normal speech i: Global and fine-grained acoustic-phonetic talker characteristics,* Speech communication **20**, 255 (1996).

[3] I. Lehiste and N. J. Lass, *Suprasegmental features of speech,* Contemporary issues in experimental phonetics **225**, 239 (1976).

[4] M. Zhou and A.-J. van der Veen, *Blind separation of partially overlapping data packets,* Digital Signal Processing (2017).

[5] M. P. Syskind, J. Larsen, U. Kjems, and L. C. Parra, *A survey of convolutive blind source separation methods,* Springer Handbook on Speech Processing and Speech Communication (2007).

[6] D. Wang, *On ideal binary mask as the computational goal of auditory scene analysis,* Speech separation by humans and machines , 181 (2005).

[7] A. M. Reddy and B. Raj, *Soft mask methods for single-channel speaker separation,* IEEE Transactions on Audio, Speech, and Language Processing **15**, 1766 (2007).

[8] N. Harishkumar and R. Rajavel, *Monaural speech separation system based on optimum soft mask,* in *Computational Intelligence and Computing Research (ICCIC), 2014 IEEE International Conference on* (IEEE, 2014) pp. 1–4.

[9] T. Kim, H. T. Attias, S.-Y. Lee, and T.-W. Lee, *Blind source separation exploiting higher-order frequency dependencies,* IEEE transactions on audio, speech, and language processing **15**, 70 (2007).

[10] A. Sarmiento, I. Durán-Díaz, A. Cichocki, and S. Cruces, *A contrast function based on generalized divergences for solving the permutation problem in convolved speech mixtures,* IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP) **23**, 1713 (2015).

[11] K.-L. Huang and T.-S. Chi, *Tdoa information based vad for robust speech recognition in directional and diffuse noise field,* in *Chinese Spoken Language Processing (ISCSLP), 2012 8th International Symposium on* (IEEE, 2012) pp. 126–130.

[12] M. Zohourian, A. Archer-Boyd, and R. Martin, *Multi-channel speaker localization and separation using a model-based gsc and an inertial measurement unit,* in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on* (IEEE, 2015) pp. 5615–5619.

[13] H. Fukai, *A method to solve the permutation problem in blind source deconvolution for audio signals based on phase linearity estimation,* in *Sensor Array and Multichannel Signal Processing Workshop (SAM), 2016 IEEE* (IEEE, 2016) pp. 1–4.

[14] W. Zhao, Y. Shen, P. Xu, J. Wang, Z. Yuan, Y. Wei, W. Jian, and H. Li, *A new efficient method for permutation and scaling ambiguity of blind source separation signal blocks,* in *Intelligent Control and Information Processing (ICICIP), 2014 Fifth International Conference on* (IEEE, 2014) pp. 23–28.

[15] T. Kim, I. Lee, and T.-W. Lee, *Independent vector analysis: definition and algorithms,* in *Signals, Systems and Computers, 2006. ACSSC'06. Fortieth Asilomar Conference on* (IEEE, 2006) pp. 1393–1396.

[16] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, and E. Moulines, *A blind source separation technique using second-order statistics,* IEEE Transactions on signal processing **45**, 434 (1997).

[17] F. Asano and S. Ikeda, *Evaluation and real-time implementation of blind source separation system using time-delayed decorrelation,* in *Proceedings of the Second International Workshop on ICA and BSS* (2000).

[18] K. Matsuoka, *Minimal distortion principle for blind source separation,* in *SICE 2002. Proceedings of the 41st SICE Annual Conference*, Vol. 4 (IEEE, 2002) pp. 2138–2143.

[19] I. M. Ventry, R. W. Woods, M. Rubin, and W. Hill, *Most comfortable loudness for pure tones, noise, and speech,* The Journal of the Acoustical Society of America **49**, 1805 (1971).

[20] I. Hochberg, *Most comfortable listening for the loudness and intelligibility of speech,* Audiology **14**, 27 (1975).

[21] E. A. Habets, *Room Impulse Response Generator* (2010).

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

[23] A.-j. Van der Veen and G. Leus, *Signal processing for communications,* Made available for TU Delft course ET4147 - Signal processing for communications (2005).

[24] A. Goldsmith, *Wireless communications* (Cambridge university press, 2005).

[25] M. H. Hayes, *Statistical digital signal processing and modeling* (John Wiley & Sons, Inc., 1996).

A

PHASED ARRAY APPROXIMATION

The phased-array data model is based on two the assumption. The first is the far-field approximation which implies that all receivers receive the signal at the same power or equivalently with the same channel loss. A rule of thumb here is that it can be applied from a distance of 10λ and beyond. The second is the narrowband assumption which says that a small difference in time of arrival can be represented by only a phase shift. The validity of this approximation follows from the following derivation, found in [23]

If $s(t)$ is the signal the first microphone receives then let $S(f)$ be its Fourier transform. The delayed signal received by the second microphone includes a delay and a phase shift and is described as

$$s_\tau(t) = s(t-\tau) \cdot e^{j2\pi f_c \tau}$$

The inverse Fourier transform which returns the delayed signal equals

$$s(t-\tau) = \int_{-\infty}^{\infty} S(f) e^{j2\pi f \tau} e^{-j2\pi f t} df$$

Define $W > 0$ as the active bandwidth around the centre frequency f_c, adding the constraint $\frac{W}{2} < f_c$ to include only positive frequencies. It can be seen that the integral contributes

$$s(t-\tau) \approx \int_{f_c - W/2}^{f_c + W/2} S(f) e^{-j2\pi f t} df = s(t)$$

because if within the bandwidth $|f_c - f| < W/2$ the argument $|2\pi f \tau| \ll 1$ then the exponent can be approximated by $e^{j2\pi f \tau} \approx 1$ resulting in

$$s_\tau(t) \approx s(t) e^{j2\pi f_c \tau}$$

Looking at the bandwidth requirement, the worst case condition for this approximation is the boundary of the frequency range $|f_c - f| = \frac{W}{2}$. The argument can then be rewritten as $|2\pi \frac{W}{2} \tau| \ll 1$. The conclusion is that delays between two microphones receiving the same signal can be ignored as long as the approximation $e^{j\pi W \tau} = 1$ holds which is equal to a small time bandwidth product $W\tau \ll 1$.

B

CALCULATING THE STFT

The short-time Fourier transform (STFT) is a discrete approximation of a signals time-frequency spectrum. An STFT is calculated by cascading local frequency transforms. What local implies and how the transform is defined is explained in this appendix. Additionally, we discuss the relationship between the data and the signal in this domain and what improvements can be made on the most elementary STFT to improve spectrum estimation.

B.1. ELEMENTARY

This section derives the simplest STFT and the relationship between signal and data. From the signal $s[n]$, frames are selected in such a way that each frame $m = 0, 1, \dots M-1$ is defined as

$$s_{(m)}[n] = \begin{cases} s[n] & m \cdot K \leq n < (m+1) \cdot K \\ 0 & \text{Otherwise} \end{cases}$$

where K is the as the number of samples in each frame. For these frames, the relationship to the data is known: Applying a discrete convolution with the room impulse response $h[n]$, this frame contributes

$$\tilde{x}_{(m)}[n] = \sum_{l=-\infty}^{\infty} s_{(m)}[n-l]h[l] + e_n$$

to the data. It must be made explicit that this is the data that is the result of the part of the signal generated in window m. The spreading effect of the convolution results in it having $N_f + N_{60} - 1$ samples which are non-zero. The frequency spectrum of this local data can be calculated with the discrete-time Fourier transform (DTFT) which is defined as the infinite sum

$$\tilde{x}_m(\omega) = \sum_{l=-\infty}^{\infty} \tilde{x}_{(m)}[mN_f + l] \cdot e^{-j\omega l}$$

However only finite data is available as a basis to estimate the local frequency spectrum. That is why the cyclic DFT is used as the definition of the Fourier transform $\mathscr{F}$ and it can be calculated as

$$\tilde{x}_m[k] = \sum_{l=0}^{K+N_{60}-2} \tilde{x}_{(m)}[mK + l] \cdot e^{-i2\pi \frac{k \cdot l}{K+N_{60}-1}}$$

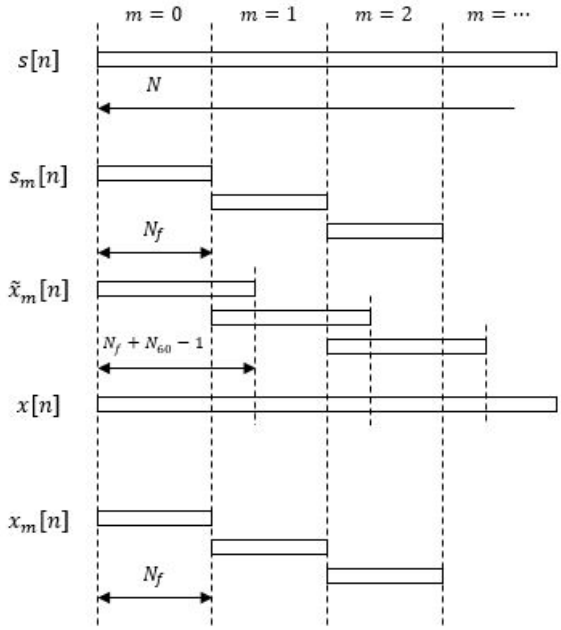


Figure B.1: Visualisation of the segments of $s[n]$ and $x[n]$.

This is equal to the DTFT for $k \in [0, K + N_{60} - 2]$.

Keeping in mind that we want to estimate $s_m[k]$, this data $\tilde{x}_{(m)}[n]$ would be ideal. However it can not simply be recovered since two problems occur. First, N_{60} is unknown and as a channel property, it might not even be constant. Second, the last $N_{60} - 1$ samples of $\tilde{x}_m[n]$ are overlapping with the first $N_{60} - 1$ samples of the data generated by in the next data frame, $\tilde{x}_{m+1}[n]$. Instead, we approximate the data resulting from the signal generated in window m with $x_{(m)}[n]$, the data in window m which is likewise defined as

$$x_{(m)}[n] = \begin{cases} x[n] & m \cdot K \leq n < (m+1) \cdot K \\ 0 & \text{Otherwise} \end{cases}$$

What is known about this data is that it contains the first K samples of $\tilde{x}_{(m)}[n]$ and contains $N_{60} - 1$ overlap in the front, coming from the previous frame $\tilde{x}_{(m-1)}[n]$. The validity of this approximation is based on the ratio between K and N_{60}. The transform of the shorter $x_{(m)}[n]$ is calculated as

$$x_{k,m} = \mathscr{F}\left\{x_{(m)[n]}\right\} \sum_{l=0}^{K-1} x_{(m)}[mK + l] e^{-i2\pi \frac{k \cdot l}{K}}$$

and stacked in a vector $\boldsymbol{x}_m^{(k)} = [x_{0,m}\, x_{1,m} \ldots x_{N_f-1,m}]^T$ it is the local spectrum. The superscript (k) is not intended as an index but as a clarification that the vector contains values from the discrete frequency domain.

The relationship between signal and data in this domain can be concluded from the way it was defined in the time domain and the convolution theorem on Fourier transforms. If we write

$$\boldsymbol{h}^{(k)} = \mathscr{F}\{h[n]\}$$

for the frequency response of the channel and define

$$\boldsymbol{s}_m^{(k)} = \mathscr{F}\left\{s_{(m)}[n]\right\}$$

to be the signal's local spectrum then the data's local spectrum becomes

$$\boldsymbol{x}_m^{(k)} = \boldsymbol{h}^{(k)} \odot \boldsymbol{s}_m^{(k)}$$

or equivalently for the individual elements

$$x_{k,m} = h_k s_{k,m}$$

Either way, as stated before, these local spectra are cascaded the STFT that we were looking for.

$$\boldsymbol{X}^{(K)} = \left[\boldsymbol{x}_0^{(k)} \boldsymbol{x}_1^{(k)} \dots \boldsymbol{x}_{N_t-1}^{(k)}\right]$$

B.2. IMPROVEMENT

In the previous explanation, two assumptions have been made which are uncommon in current state-of-the-art STFT analysis. What these are and how they can be improved upon are explained here.

By simply selecting samples we have unintentionally used the rectangular window function. The properties considered important for a window function are the level of the first sidelobe and the 3dB decay bandwidth. For the rectangular window these are $-13dB$ and $0.89\frac{2\pi}{N}$. An alternative window is the Hanning window. With width N, it is defined as

$$w(n) = \frac{1}{2}\left(1 - \cos\left(\frac{2\pi n}{N-1}\right)\right)$$

for $0 \leq n < N$. It has a level of $-32dB$ at its first side lobe and a 3dB-decay width of $1.44\frac{2\pi}{N}$ [25]. The main lobe is a little broader but the level of the first side lobe is significantly lower.

The second unintentional assumption was not using any overlap. Because of the limited size of the time frames, parts of the signal that are situated at the border of a frame have a high risk of being distorted. They are represented less strongly, especially if a function like the Hanning window is used. Also, any windowing function but the rectangular function descreases the amount of signal energy available. To allow overlap, redefine a time frame as

$$s_{(m)}[n] = \begin{cases} s[n] & mN_{shift} \leq n < mN_{shift} + K \\ 0 & \text{otherwise} \end{cases}$$

where N_{shift} is now the number of samples with which the frame is moved forward in fast time. Often the overlap is described in percentages equivalent to the outcome of $1 - \frac{N_{shift}}{K}$. For a time sequence $x[n]$ of fast time length N this means there are $N_t = \left\lfloor \frac{N-K}{N_{shift}} \right\rfloor + 1$ steps in the slow time.

To ensure a more appropriate spectrum estimation, this thesis uses the Hanning window with a 50% overlap on both sides

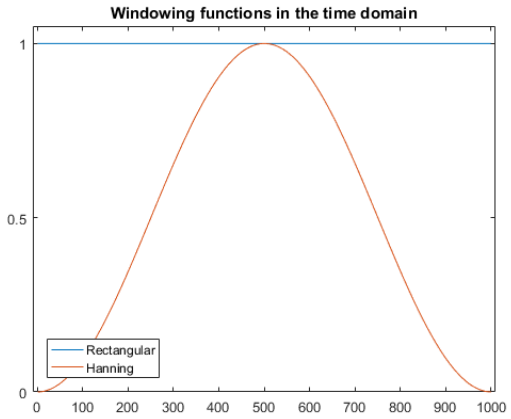


Figure B.2: The rectangular and Hanning windowing function

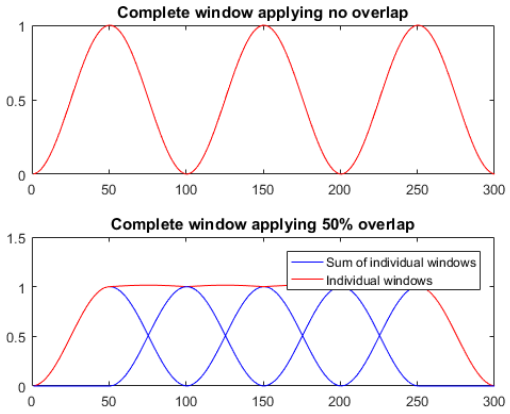


Figure B.3: A visualisation of the constant energy preserving effect by applying overlap.

C

CORRELATION SUM

In this appendix we show that any permutation of column vectors can be reversed if the original order is known. In our case we look at two matrices which are very much alike, yet also perturbed by factors besides the permutation.

Given is the situation that there are two adjacent detection windows which both generate a beamformer containing an equal number of demixing vectors. Let us use a general notation and say that the windows 1 and 2 generate matrices $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ consisting of demixing vectors as

$$\boldsymbol{W}_i = [\boldsymbol{w}_{i,0}, \boldsymbol{w}_{i,1}, \dots \boldsymbol{w}_{i,D-1}]$$

where $\boldsymbol{W}_i$ is of size $R \times D$.

The columns of the beamformers 1 and 2 are said to be *approximately* the same and the order in which they appear in W_1 is considered the correct order. The columns of W_2 are randomly permuted. We denote the correct order as W_2^* and define that the solution to this problem equals finding a permutation matrix that solves

$$\boldsymbol{W}_2^* \boldsymbol{\Pi}^* = \boldsymbol{W}_2$$

or since only the left side is know,

$$\boldsymbol{W}_2^* = \boldsymbol{W}_2 (\boldsymbol{\Pi}^*)^{-1}$$

Before we look for $\boldsymbol{\Pi}^*$, let us repeat the likeliness measure which was introduced in the main body of this thesis. The correlation coefficient

$$\rho(\boldsymbol{w}_{1,i}, \boldsymbol{w}_{2,j}, \boldsymbol{X}) = \frac{|\boldsymbol{w}_{1,i}^H \boldsymbol{X}^H \boldsymbol{X} \boldsymbol{w}_{2,j}|}{|\boldsymbol{w}_{1,i}^H||\boldsymbol{w}_{2,j}|}$$

describes the likeliness between two demixing vectors. The fitness of the complete solution is defined as the sum of the individual likenesses and can be written as

$$\boldsymbol{R}(\boldsymbol{W}_1, \boldsymbol{W}_2, \boldsymbol{X}) = \sum_{i=0}^{D-1} \rho(\boldsymbol{w}_{1,i} \boldsymbol{w}_{2,i}, \boldsymbol{X})$$

Now additionally we introduce a matrix $\boldsymbol{C}$ which contains the likeliness of all $D \times D$ combinations of the demixing vectors of beamformers 1 and 2. Because the elements are defined as

$$c_{i,j} = \rho(\boldsymbol{w}_{1,i}, \boldsymbol{w}_{2,j}, \boldsymbol{X}) \quad \forall i, j = 0, 1, \dots D-1$$

the column order of $\boldsymbol{W}_1$ is preserved in the row order of $\boldsymbol{C}$ and the column order of $\boldsymbol{W}_2$ is preserved in the column order of $\boldsymbol{C}$. The fitness of the current combination $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ can also be described as a function of this matrix and more specifically as

$$\begin{aligned} \mathrm{Tr}(\boldsymbol{C}(\boldsymbol{W}_1, \boldsymbol{W}_2)) &= \sum_{i=0}^{D-1} c_{i,i} \\ &= \sum_{i=0}^{D-1} \rho(\boldsymbol{w}_{1,i} \boldsymbol{w}_{2,i}, \boldsymbol{X}) \\ &= \boldsymbol{R}(\boldsymbol{W}_1, \boldsymbol{W}_2, \boldsymbol{X}) \end{aligned}$$

One final equality is required before the complete optimization problem can be given. As said earlier, the column order of $\boldsymbol{W}_2$ is preserved in $\boldsymbol{C}$. Therefore it can be said that

$$\boldsymbol{C}(\boldsymbol{W}_1, \boldsymbol{W}_2 \Pi) = \boldsymbol{C}(\boldsymbol{W}_1, \boldsymbol{W}_2)\Pi$$

which allows us to describe the previously stated problem as

$$\boldsymbol{\Pi}^* = arg \max_{\boldsymbol{\Pi} \in \mathscr{P}} \mathrm{Tr}(\boldsymbol{C}\boldsymbol{\Pi}^{-1}))$$

or equivalently the permutation matrix $\boldsymbol{\Pi}^*$ places the largest values of $\boldsymbol{C}$ on it's diagonal.

To prove that this optimization problem indeed results into the correct permutation, consider the following: Based on the definition of the correlation coefficient it is known that

$$\rho(\boldsymbol{w}_{1,i}, \boldsymbol{w}^*_{2,i}) > \rho(\boldsymbol{w}_{1,i}, \boldsymbol{w}^*_{2,j}) \quad \forall i \neq j$$

which, while it also holds for our situation, must be extended in three ways before its true consequence is revealed. First, trade in $>$ for $\geq$ and it can be any column which allows us to leave out $\forall i \neq j$. In fact, notice that the column $\boldsymbol{w}^*_{2,j}$ also exists in $\boldsymbol{W}_2$ so the optimality condition of the right half can be left out. Finally, our solutions can never contain just one such mismatch. A single mistake in permutation has two corresponding demixing vector mismatches as a consequence. Therefore we start with the following inequality

$$\rho(\boldsymbol{w}_{1,i}, \boldsymbol{w}^*_{2,i}) + \rho(\boldsymbol{w}_{1,j}, \boldsymbol{w}^*_{2,j}) \geq \rho(\boldsymbol{w}_{1,i}, \boldsymbol{w}_{2,j}) + \rho(\boldsymbol{w}_{1,j}, \boldsymbol{w}_{2,i})$$

which is about one pair of mispermuted demixing vectors. It states that the sum of the correlation coefficients is at its highest if the columns of $\boldsymbol{W}_2$ are permuted correctly. This inequality can be extended to

$$\sum_{i=0}^{D-1} \rho(\boldsymbol{w}_{1,i} \boldsymbol{w}^*_{2,i}) \geq \sum_{i=0}^{D-1} \rho(\boldsymbol{w}_{1,i} \boldsymbol{w}_{2,i})$$

where the left half of the inequality can be recognised as the fitness of the current column order of $\boldsymbol{W}_2$ and therefore also as the trace of $\boldsymbol{C}$.

This is solved a simplistic, two step greedy algorithm. First, check for the current row where the largest value is. Secondly, permute that value to the diagonal. If this is done for all D rows, the combined permutation matrix must equal $\boldsymbol{\Pi}^*$. An example is given in figure C.1.

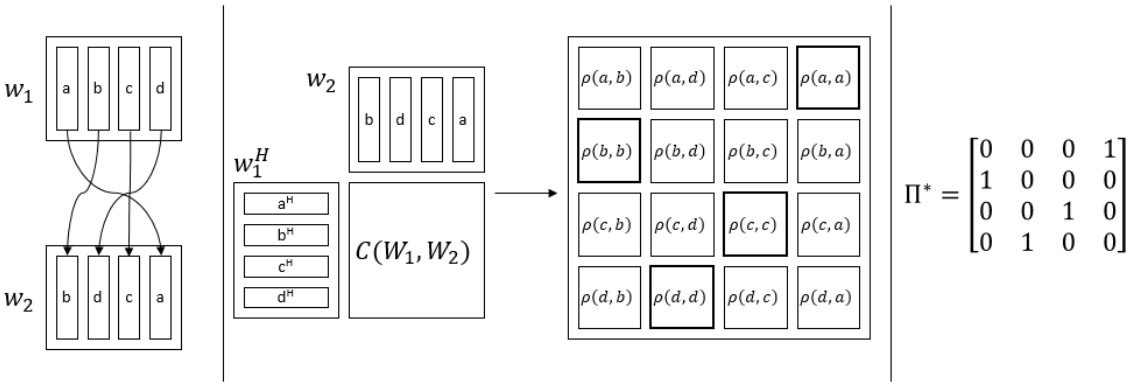


Figure C.1: Left: An unknown permutation. Centre: Matrix $\boldsymbol{C}$ and highlighted the largest values for each row. Right: The corresponding optimal permutation matrix