COMBINING BEAMFORMING AND BLIND SOURCE SEPARATION TO IMPROVE SOURCE SEPARATION PERFORMANCE

By
Aishwarya Sivaramakrishnan
Ruben Johnson Robert Jeremiah

Supervisors from B&K
Andreas Schuhmacher
Karim Haddad

Supervisor from BTH
Benny Sällberg
Acknowledgement

This thesis is the final project work for the Master of Science in Electrical Engineering with emphasis on Signal Processing at the Department of Electrical Engineering, Blekinge Institute of Technology, Karlskrona, Sweden.

This thesis work has been performed at Bruel & Kjær Sound and Vibration Measurements A/S, Copenhagen, Denmark, under the supervision and guidance of Andreas Schuhmacher and Karim Haddad, Researchers at B&K. Without their support and guidance, this thesis would have not been possible. They guided us all through the way and made us explore lot of intricacies in the project. They gave lot of directions to make us propose a very good method. We thank them for guiding us and giving us a unique experience as engineer trainees. This work was also monitored and supervised by Benny Sällberg from our university, who has been very kind to us and gave us a kick and motivation to start this project. He never stopped until we finish the project, constantly encouraged our work and documentation and also very responsive for our future research works. Also, we thank each and everyone in the Innovation team for all their friendly approach during any circumstances.

We would like to thank Blekinge Institute of Technology, Sweden, who gave us a great opportunity to extend our knowledge to a Masters level. Each and every professor at the school were highly dexterous and also composed enough at all levels to clarify our queries and doubts during our academic inspections. We would also like to express our sincere appreciation to Bruel & Kjaer who offered us a great opportunity to put up our theoretical knowledge into a company’s experience.

I, Ruben Johnson would like to thank Monisha Raja and her family for the support they gave me to start my Masters in Sweden. Without them this masters would have just been a dream.

I, Aishwarya, would like to thank SunderRajan Kondulsamy, a person close to heart that motivated me to pursue my Masters in Sweden. He has been a greatest inspiration in my life to achieve lot of strength and conviction in various scenarios. He is a best moral supporter without whom my life in Sweden would not have been so remarkable.

We would like to thank our dear parents and siblings for their most valuable support and encouragement throughout our academic life. We would not have been successful professionals without their love and driving force. We would also like to convey our hearty thanks to our friends in India and Sweden for their assistance and appreciation during each and every move. Finally, we like to thank Rose Fredrikson and her family, who supported us with accommodation during our complete stay in Denmark. Rose and her family treated us as their own kith and kin and we felt like home every day. We learnt different culture and customs during our stay with their family. Finally we thank Sweden for all the experiences we have had for the past two and half years.
Abstract

Beamforming (BF) and Blind Source Separation (BSS) are always two interesting methodologies to witness in order to separate two sources. BSS in frequency domain have been facing a serious issue of permutation ambiguity while performing source separation using Independent Component Analysis (ICA). Permutation Ambiguity is a problem of mismatch of any frequency lines between the sources, so the separation in the time domain cannot exhibit a perfect separation due to the frequency components of other sources present in the time signal of one source. Various methods have been adopted all through the years of research to get rid of this critical issue and no perfect results are produced so far. Beamforming is done with spherical waves where the array is designed according to the corresponding chosen frequency band. The distance between the microphones is always set to be less than half of the source wavelength just in order to avoid aliasing issue. When BF is designed with good resolution, using this beamforming information will give a better insight to work with BSS. The proposed method of combining BF to BSS seems to be a good approach as BF mainly depends on time-difference of arrival information (delay) between the reference microphone to the consecutive microphones. The original delay information is compared to the estimated delays for each frequency lines in order to realign the frequency lines if they see a permutation by ICA. So, there is no possibility of frequency mismatch still existing when the delay information is operated as a major concern. This method is compared with a method called envelope continuity which uses correlation approach between neighboring frequency lines to prove their spectral continuity. This method is also one of the wisest approaches to solve the problems of frequency domain BSS. But failure in aligning one particular frequency may lead to failure of all other successive bins which cannot be avoided. So, BF delay information is used over envelope continuity to separate sources effectively. The performance is measured using Signal to Interference Ratio measurement where Beamforming approach seems to have an improved performance compared to envelope continuity. Simulation results show performance comparison. The algorithm is tested using two speech sources in a non-reverberant environment and the sources are filtered only by delays. We use Short Time Fourier Transform (STFT) for frequency domain transformation. The Then, performance is tested in four different environments and scenarios- changing source spacing, changing microphone spacing, changing the height of source plane from microphone plane and changing the filter lengths of STFT.
Contents

List of Figures

1. Introduction and Motivation ........................................ 1
   1.1 Introduction .................................................. 1
   1.2 Motivation .................................................. 1
   1.3 Research Questions .......................................... 1
   1.4 Hypothesis .................................................. 2

2. Beamforming ....................................................... 3
   2.1 Microphone Array ........................................... 3
   2.2 Planar Wave .................................................. 4
   2.3 Spherical Wave ............................................... 5
   2.4 Delay and Sum Beamforming ................................ 6
   2.5 Aliasing Effect ............................................... 8

3. Blind Source Separation ........................................... 9
   3.1 Independent Component Analysis ............................ 9
   3.2 Instantaneous Mixing ...................................... 9
   3.3 Convolutive Mixing ....................................... 10
   3.4 Ambiguities Inherent to BSS Problems .................... 11
     3.4.1 Permutation Ambiguity ................................ 11
     3.4.2 Scaling Ambiguity .................................... 12
   3.5 Algorithm for ICA .......................................... 12
     3.5.1 Pre-processing of the Data ............................. 12
       3.5.1.1 Centring the Data .................................. 13
       3.5.1.2 Whitening the Data ................................ 13
     3.5.2 Algorithm ............................................... 14
   3.6 Frequency Domain BSS ...................................... 14
   3.7 Problems Introduced by the Frequency Approach ......... 15
     3.7.1 Problem of Conventional Fourier Transform ......... 15
     3.7.2 Use of Contrast Function ............................... 16
     3.7.3 Permutation Ambiguity ................................ 16

4. Existing Methods to Solve Permutation Ambiguity ............ 17
   4.1 Continuity of Spectral Envelope over Frequency .......... 17
     4.1.1 Drawback ............................................... 17
   4.2 Solving Permutation using ICA Based Clustering .......... 18
   4.3 Other Methods to Solve Permutation Ambiguity .......... 19

5. Time Difference of Arrival (TDOA) ............................ 20
   5.1 Combining Technique ....................................... 22
6. Test and Analysis
   6.1 Method
   6.2 Signal to Interference Ratio
   6.3 Comparison of TDOA and Envelope Continuity
   6.4 Initialization Problem in ICA

7. Conclusions and Future Work
List of Figures

2.1. Microphone array with 64(8X8) microphones 4
2.2. Planar Wave 5
2.3. Beampattern for 2 planar waves at -40 and +40 degree 5
2.4 Spherical Wave 6
2.5. Beampattern for 2 spherical waves at -0.3 and 0.75 respectively. 7
2.6. Delay and Sum beamforming with five microphones 7
2.7 Microphone array set up 8
3.1 Convolutive mixing model (a pictorial representation) 10
4.1 Illustration of permutation ambiguity, top two pictures presence of permutation and the below two pictures shows the corrected permutation 18
5.1 TDOA representation 20
6.1 System model 23
6.2 (a) Spectrogram of a female voice (b) Spectrogram of a male voice 24
6.3 SIR improvement for TDOA against Envcont by varying the distance between the microphones 25
6.4 SIR improvement for TDOA against Envcont by varying the distance between the sources 26
6.5 SIR improvement for TDOA against Envcont by varying the distance between the sources and the microphone array 26
6.3 SIR improvement for TDOA against Envcont by varying the window length 27
Chapter 1

Introduction and Motivation

1.1 Introduction

Blind Source Separation is the process of estimating the real emitting source signals from the observed mixed signals from any input channel like microphones. Here, the independence of each signal corresponds to the separation of sources. Similarly, Beamforming is an array signal processing technique which is used to localize sources from different directions. This uses number of linearly arranged microphones. Both these have their own uniqueness and disadvantages as long as they are implemented separately in different applications. Applications like speech enhancement, hearing aids, conference telephony, party gathering, noise-robust speech recognitions and hands-free telecommunication systems requires these processes to have source intelligibility without any irregularities. When BSS is combined with beamforming, it is possible to overcome the drawbacks faced in both the processes. The Beamforming approach is robust since a misalignment at a frequency does not affect other frequencies while in other approaches misalignment may cause consecutive misalignments [7]. Beamforming defines the direction of arrival from Delay and Sum Beamforming algorithm and BSS uses ICA algorithm where both can be made to meet at a common point in order to solve the issue of source separation. This project deals with the point where they both are expected to be combined and the constraints that have to be considered to do this combination.

1.2 Motivation

In a real-time scenario when multiple speakers speak at the same time observed in an array of sensors, the recordings will always contain mixed sources having little or no information about the original sources. It is a very classical and difficult problem to separate them into independent sources. When this problem is investigated in frequency domain, the major issue is Scaling and the Permutation Ambiguity. Scaling ambiguity can be solved simply, but there is no perfect solution to solve permutation ambiguity. So, the issues are being handled by beamforming approach.

1.3 Research Question

1. How can Beamforming approach be employed to solve permutation ambiguity in order to improve the efficiency of BSS?
2. How to examine the performance of beamforming approach over inter-frequency dependence methods?
1.4 Hypothesis

Delay and Sum Beamforming can be implemented in order to solve the issue of permutation ambiguity. Beamformer has the ability to localize the sources using the Time Difference Of Arrival (TDOA) information to the microphone.

Signal to Interference Ratio (SIR) can be calculated and compared in-order to examine the performance of Beamforming approach over Inter Frequency Dependence methods.
Chapter 2

Beamforming

Beamforming is the process of performing spatial filtering i.e., the response of the array of sensors is made sensitive to signals coming from a specific direction while signals from other directions are attenuated [21]. Beamformers combine the signals from spatially separated array-sensors in such a way that the array output emphasizes signals from a certain “look” direction. Thus if a signal is present in the look-direction, the power of the array output signal is high and if there is no signal in the look-direction the array output power is low [6]. Hence, the array can be used to construct beamformers that “look” in all possible directions and the direction that gives the maximum output power can be considered an estimate of the Direction of Arrival (DOA) [3].

Beamforming is an interesting idea for source separation and to localize any form of sources. It can be visualized by color plots showing a beam pattern in the location of radiation of signal. Beamforming can be implemented in various ways, for e.g., time domain and frequency domain depending on choice of parameters, compiling time and the type of signal used. The output of beamforming tells about the number of sources, source positions and the strength of the sources. Currently the methods of beamforming can be divided into two kinds namely frequency domain and time domain methods. Frequency domain method is that by using Fourier Transform the speech signal is divided into different sub-bands and then employing narrowband beamforming method to process them. However these methods have high computational complexity [12].

2.1 Microphone Array

All signals are assumed to be electrical representations of physical quantities. This assumption implies that a sensor has been used to translate a physical quantity (e.g., air pressure level) into an electrical signal so that a certain amount of electrical voltage (or current) in the electrical sensor signal corresponds to a certain amount of a physical quantity. Unless otherwise stated, it is henceforth assumed that a continuous time electrical sensor signal has been sampled with the sampling frequency \( F_s \) (Hz) and that it is being correctly represented by a corresponding discrete-time signal. Microphone array is a collection of multiple microphones in a certain arrangement functioning as a uni-directional input device. This set-up is normally accommodated when a person/no. of people speaks from different directions that cannot have microphone at their desired position. Microphone array, arranged in a specific fashion, spatially locate principle sources and distinguish from each other [1].
Distinguishing sounds based on the spatial location of their source is achieved by filtering and combining the individual microphone signals from the array. To form a microphone array, array definition is a basic requisite to perform source separation. Distance between microphones, size of the array, array shape, source distance at each axes from the array are the parameters that falls under array definition. When array is rightly defined according to the application used, microphone array can produce the best output of source separation.

### 2.2 Planar Wave

Planar wave is a constant frequency wave which propagates from a far distant source to the receivers, say microphone. Planar wave cannot be generated on its own but the waves that is generated from a far-field source approach a receiver in the form of planes. Planar waves have the wave-front that are infinite parallel planes.

The general formula for a plane wave is

\[
f(x, t) = \exp[j(\omega_0 t - k_0 x)]
\]

The figure shows the e.g., of two plane waves arriving at a microphone array of constant spacing. Plane wave reaches the array at a certain distance

\[
t_{mb} = m \frac{d}{c} \sin \theta_b
\]

Where, \(m\) = no. of microphones in an array
\(d\) = distance between microphones
\(c\) = speed of sound in air 340 m/s
\(\theta_b\) = angle of arrival of plane wave
The plane wave produces a beampattern where it contains main lobe and side lobes. The beampattern speaks about the power of the signal where beam experiences a constructive interference in the angle of arrival of the plane wave while the other angles experience a destructive interference and have a less power in the sidelobes. To form a beampattern, we use $10\log_{10}(P/P_{\text{max}})$ where $P$ is the output of the delay and sum beamforming.

2.3 Spherical Wave

In the theory of acoustics, point source produces a spherical wave in an ideal isotropic (uniform) medium such as air. Furthermore, the sound from any radiating surface can be computed as the sum of spherical wave contributions from each point on the surface (including any relevant reflections). Thus, all linear acoustic wave propagation can be seen as a superposition of spherical travelling waves. The below figure shows two point source generating spherical waves to the rectangular array of microphones.
The spherical wave has a different impact on a microphone array. It considers the radius of the wave from the point source to calculate the time delay and the formula is shown below.

\[ t_{mb} = \frac{r_1 - r_i}{c} \]  \rightarrow (2.3)

Initially, the test of spherical wave is done with a linear array and during the later stages, considering the practical applications which needed many number of microphones having uniform and non-uniform arrangement, the test was broaden using rectangular array.

The beampattern for spherical waves (in case of two sources) with a rectangular arrangement is show in the figure 2.5.

### 2.4 Delay and Sum Beamforming

Typically, a beamformer linearly combines the spatially sampled time series from each sensor to obtain a scalar output time series of a signal from a given direction, in the same manner as an FIR filter linearly combines temporally sampled data to select a signal in a given frequency range. [16]. There are different types of beamforming algorithms available such as Delay and Sum, LCMV, Adaptive beamforming and many. For simplicity and easy computation, we have chosen Delay and Sum Beamforming. One of the simplest of beamforming techniques is the Delay and Sum Beamforming (DSB) to localize sources. The delay and sum beamformer is based on the idea where the output signal from each sensor will be the same, except that each value from each sensor will be delayed by a different amount.
The output of each sensor is delayed appropriately and then added together, the response of an array of sensors is made sensitive to signals coming from a specific direction while signals from other directions are attenuated.

\[ y_b(t) = \sum_{m=0}^{M-1} a_m x_m(\omega t - \omega t_{mb}) \]  \rightarrow (2.4)

![Diagram of delay and sum beamforming with five microphones]

The array used for our setup is a rectangular array consisting of 8x8 microphones in linear fashion. It is arranged in x-y plane where source is placed parallel to this array. Distance between the microphones is set to be 0.0425m decided depending on the wavelength of the source signal used. Wavelength can be calculated using this formula \( \lambda = c/f \) where f is the maximum
frequency of the source signal. For speech, the maximum frequency is generally 4 KHz. The length of the array is 0.15m on both the axes. The spacing higher than $(\frac{1}{2})c/f$ will cause spatial aliasing.

![Figure 2.7 Microphone array set up](image)

2.5 Aliasing Effect:

It is possible to define the threshold frequency ($f_{max}$) of an equally spaced array as:

$$f_{max} = \frac{c}{d} \quad \Rightarrow (2.5)$$

where $c$ is the speed of sound and $d$ is the spacing between the microphones. If the sound source frequency exceeds this critical frequency, ghost sources appear in the beampattern. [11]. The ghost sources, numerically generated by the beamforming algorithm, do not correspond to real emitting sources and cause therefore identification errors. The aliasing effects can be seen, for eg, when an array with critical frequency of about 4KHz is used to identify a sound source placed in front of it, where the source has a growing frequency from 2 KHz to 5 KHz. When the sound source frequency is less than the critical value, a main lobe identify the real emitting source, while the typical side lobes decrease. Exceeding the critical frequency, instead, many ghost sources appear. Thus, the standard solution to avoid aliasing errors is the reduction of the microphone spacing $d$.

$$d < \frac{\lambda}{2} \quad \Rightarrow (2.6)$$
Chapter 3

Blind Source Separation

Recently, Blind source separation by Independent Component Analysis (ICA) has received attention because of its potential applications in signal processing such as in speech recognition systems, telecommunications and medical signal processing [4]. The goal of ICA is to recover independent sources given only sensor observations that are unknown linear mixtures of the unobserved independent source signals. In contrast to correlation-based transformations such as Principal Component Analysis (PCA), ICA not only de-correlates the signals (2nd-order statistics) but also reduces higher-order statistical dependencies, attempting to make the signals as independent as possible. The difficult of the source separation depends on the number of sources, the number of microphones and their arrangements, the noise level, the way the source signals are mixed within the environment and on the prior information about the sources, microphones and mixing parameters [9].

3.1 Independent Component Analysis

ICA is one of the methods to separate sources in a set of recordings. As the name indicates, this method separates the mixed recordings into a set of independent components. The recordings and sources are considered as a set of random variables. Therefore here independence has to be taken with its statistical meaning [2]. Two independent variables \(x_1\) and \(x_2\) are statistically independent if and only if their joint probability density function (pdf) is the product of their marginal pdf.

\[
p(x_1, x_2) = p_1(x_1) \cdot p_2(x_2) \quad \rightarrow (3.1)
\]

Therefore ICA aims at getting as close as possible to this equation. If more than two components are involved, say \(n\), the previous equation is extended to the \(n\)-th dimension: the variables are independent if \(p(x_1, \ldots, x_n) = p_1(x_1) \cdots p_n(x_n)\). An ICA algorithm will combine linearly the recordings until such an independence condition for the combinations is reached. ICA theory relies on the assumption that the sources have to be statistically independent. This result is a consequence of the fact that any set of \(n\) linear combinations of \(n\) independent random variables are no longer independent, unless each combination is just proportional to one given source [2].

3.2 Instantaneous Mixing

In this part, only the simple ‘instantaneous’ mixing process is described. Instantaneous means that a given recording from the set of recordings is a function only of the original sources, and not of the time: at a given time, the recorded sample will be a function of the sources at this same
time [4]. The main assumption of the instantaneous mixing process is that the recording is done in the linear combination of the sources with the real coefficients. For \( n \) number of sources and \( m \) number of microphones the equation can be written as,

\[
x_i = \sum_{j=1}^{N} a_{ij} \cdot s_j, \quad 1 \leq j \leq n, 1 \leq i \leq m \quad \Rightarrow (3.2)
\]

The above equation can be written as

\[
X = AS \quad \Rightarrow (3.3)
\]

Where \( S = (s_1, s_2, \ldots, s_n)^T \), \( X = (x_1, x_2, \ldots, x_n)^T \), in this case \( A \) is random full rank matrix.

### 3.3 Convolutive Mixing

The mixtures obtained in the real time recordings are not instantaneous mixtures as assumed in the previously anymore but so-called convolutive mixtures. Each recording is a mixture of filtered versions of the original sources.

![Figure 3.1 Convolutive mixing model (a pictorial representation)](image)

A general blind convolutive mixing case can be easily put into equation as follows,

\[
x_i(t) = \sum_{j=1}^{n} (h_{ij} * s_j)(t) \quad \Rightarrow (3.4)
\]
The function \( h_{i,j} \) is therefore the impulse response of the environment (room, or any other environment where the mixing is performed) from the position of the receiver \( i \) to the source \( j \). The expression \( (h_{i,j} \ast s_j)(t) \) denotes the convolution between the impulse response \( h_{i,j} \) and the signal originated by source \( s_j \) [4]. It is illustrated in the following figure for the case of two sources and two recordings.

In a practical case like the above picture with two sources and two recordings, recording \( x_1 \) for example is the sum of

- source \( s_1 \) filtered by the room's impulse response from the position of source \( s_1 \) to the receiver \( x_1 \).
- source \( s_2 \) filtered by the room's impulse response from the position of source \( s_2 \) to the receiver \( x_1 \).

### 3.4 Ambiguities Inherent to BSS problems

#### 3.4.1 Permutation Ambiguity

Permutation ambiguity is a mismatch of the frequency bins in one separated signal. It is necessary to solve this issue before transforming to time domain else we would hear the same mixed sources as output [4]. The spectral envelope of the estimated signal should not vary and the mismatch should be swapped back in order to maintain the spectral envelope.

The permutation of \( A \) and the inverse permutation on \( s \) will also satisfy the same equation termed as permutation ambiguity.

A permutation matrix \( P \) is a matrix filled with ‘1’ and ‘0’ but contains only one ‘1’ per line per column. For instance, \( P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \), is a permutation matrix, which will permute the second and third line (column wise) of a matrix \( A \) if multiplied to the left.

\[ \{A, s\}, \text{ is a solution} \]

\[ \uparrow \]

\[ \{A \cdot P, P^{-1} \cdot s\}, \text{ is a solution} \]
3.4.2 Scaling Ambiguity

Scaling ambiguity is a simple yet a considerable issue to be also solved. It is also an issue about each frequency line. As the BSS problem is dealt with, in the frequency domain, ICA algorithm for instantaneous mixtures is independently applied for each frequency bin for each mixed signal combination. You have the original signal termed \( S(f,t) \) which remains unknown. But when you take the known mixed signal \( X(f,t) \) and send it through complex ICA for instantaneous mixtures and run it for each frequency bin, we obtain an independent component for each frequency bin named \( Y(f,t) \). This \( Y(f,t) \) expects to be equal to \( S(f,t) \) but it cannot happen in real time.

\[
Y(f,t) = W \ast X(f,t) \quad \Rightarrow \quad (3.5)
\]

where \( W \) is a separation filter and \( X(f,t) \) is a mixed signal. Due to separation matrix’s formation method through ICA algorithm, the output \( Y(f,t) \) consists of Permutation Matrix \( P(f) \) and a diagonal matrix of gains \( D(f) \). This diagonal matrix is nothing but the scaling ambiguity where \( Y(f,t) \) is scaled by different gains at different bins \( f \). Thus \( Y(f,t) \) will be the combination of \( d_{kk}(f) \ast S(f,t) \) where \( d_{kk}(f) \) is the \( k \)-th diagonal component of \( D(f) \). Thus, if we reconstruct \( y(n) \) in the time-domain, it is the FIR filtered version of source signal \( s(n) \).

3.5 Algorithm for ICA

The algorithm that will be described in the following is actually the FastICA algorithm developed by Aapo Hyvärinen at Helsinki University of Technology. Many other algorithms exist, however the simplicity and excellent efficiency of the FastICA algorithm made it a good choice to be studied and used here.

The algorithm can be seen as two independent parts. First, the data have to be preprocessed to fit the assumptions that were used previously: each set of data (recordings) should be of zero mean and of unit variance. Then the optimization part for this preprocessed data. The algorithm will be presented step by step in what follows [2].

3.5.1 Preprocessing of the Data

To perform ICA, there is a common assumption to be followed corresponding to the contrast functions used. The assumption is that the data should have zero mean and unit variance. Now the recordings observed are just a true random variable, it had to be pre-processed to fit into the requirements to do ICA.
The preprocessing mainly comprises of two steps.

- Centering the data
- Whitening the data

### 3.5.1.1 Centering the Data:

It is relatively easy to ensure that all the recordings have zero mean. It is possible to estimate the mean of the recordings, say the vector ‘x’ (raw recording), by taking the average value \( m = E(x) \). Using \( x' = x - m \) will provide a new set of data, \( x' \), having zero mean.

After estimating the un-mixing matrix, the original mean can be restored to the estimates of the sources, ‘y’, by making \( s \approx y' = y + W \cdot m \).

### 3.5.1.2 Whitening the Data

Whitening is a simple linear transformation that provides unit variance recordings. It also makes the recording to be uncorrelated to each others. In the end, when the recordings \( x \) have been whitened, the resulting ‘white’ components \( x_w \) verify the following property:

\[
\text{cov}(x_w) = E(x_w x_w^T) = I \quad \rightarrow (3.6)
\]

Where, \( I \) is the identity matrix of order \( n \) (number of recordings) and \( \text{Cov}(x) \) the covariance matrix of \( x \). Whitening can be easily achieved using a simple eigen-value decomposition (EVD) of the covariance matrix of the recordings.

\[
\text{cov}(x') = E. D. E^T \quad \rightarrow (3.7)
\]

where \( E \) is an orthogonal matrix (so \( E^T = E^{-1} \)) made from the Eigen vectors of \( \text{Cov}(x) \), denoting a change of space basis, and \( D \) a diagonal matrix containing its Eigen values.

So to obtain a whitened data, we follow as below,

\[
x_w = E. D^{-1/2}. E^T x' \quad \rightarrow (3.8)
\]

These two steps, centering and whitening the recordings, provide a new set of data that fits the assumptions made for all the gaussianity measurement of the previous section. Moreover, as he centering and whitening process are linear, it does not affect the search for independent components. Indeed it simplifies it, as less parameter have to be estimated.
3.5.2 Algorithm

After these steps, we get into ICA algorithm to find the independent components. After the data
had been preprocessed by means of centering and whitening, the algorithm has to repeat the
following process for each vector \( w \). [2]

1. Starting the algorithm with a chosen or a random vector \( w_0 \)

2. The iteration process starts here and has to be repeated as long as the convergence
test at its end fails:
   - Compute next step using the following equation
     \[
     w_{n+1} \leftarrow E\{xg(w_n^T x)\} - w_n E\{g'(w_n^T x)\} \quad \Rightarrow (3.9)
     \]
   - Apply basic Gram Schmidt orthogonalization (to force the orthogonality of the
     unmixing matrix): if the \( k \)-th vector is evaluated and it is the \( n+1 \) step of the
     iteration, let us use the notation \( w_{n+1}^{(k)} \) for the vector. Then the orthogonalization
     is achieved through
     \[
     w_{n+1}^{(k)} \leftarrow w_{n+1}^{(k)} - \sum_{i=1}^{k-1} (w_{n+1}^{(k)})^T w^{(i)} \quad \Rightarrow (3.10)
     \]
   - Normalize \( w_{n+1} \): \( w_{n+1} \leftarrow \frac{w_{n+1}}{|w_{n+1}|} \) to force unit variance of the result
   - Test convergence, for instance by evaluating \( w_{n+1} - w_n \).
     If there is any convergence, store the current value \( w_{n+1} \) for \( w^{(k)} \), reset index \( n \)
     to zero, raise index \( k \) and go back to step 1 if there are still components to evaluate
     \( (k < n) \).
     If there is no convergence, continue the iteration process, raise \( n \) and go back to
     step 2.

3. When \( k = n \), all the components have been evaluated. The independent components are
given back by taking each \( (w^{(k)})^T x \), for \( 1 < k < n \). [2]

3.6 Frequency Domain BSS

In a ‘blind source separation’ problem, we know only the recordings \( x_i \), where we do not know
the original sources \( s_i \), and the mixing filters \( h_{i,j} \). Different methods have been developed to
solve this problem. One is the time domain approach and the other is the frequency domain.
Time domain approach is fairly complicated; require strong computation resources and computation time with the existing algorithms whereas frequency domain approach is known to be faster and easy to apprehend and implement. Therefore, for the following, it has been chosen to use a frequency based method.

Equation (time domain equation number) sums up the problem faced here. It is possible to formulate it in the frequency domain. It is well known that the convolution that appears in this equation will be expressed as a simple product in the frequency domain.

\[ X_i(\omega) = \sum_{j=1}^{n} H_{ij}(\omega) \cdot S_j(\omega) \]  \( \Rightarrow (3.11) \)

With the acoustical application of BSS, signals are generally mixed in a convolutive manner. With the help of short-time Fourier transforms, convolutive mixtures in time domain can be approximated as multiple instantaneous mixtures in the frequency domain. Here, the separation is performed in each frequency bin with a simple instantaneous separation matrix. We employ complex-valued ICA to calculate the separation matrix.

The main idea for solving convolutive ICA problems can be seen as the following simplified steps:

- Transform the recordings into the frequency domain using Fourier transform
- For each frequency, solve the instantaneous mixing case
- Pass the independent components found back to the time domain using inverse Fourier transform

3.7 Problems Introduced by the Frequency Approach

Solving the convolutive case in the frequency domain is very much similar to solving the instantaneous case but we still have some issues to be taken care of.

3.7.1 Problem of conventional Fourier Transform

Algorithms of ICA that operate in the time domain may suffer from a heavy computational load. This problem is significant even for a moderately advanced task such as computing a matrix multiplication between a square matrix and a vector, which is the case in, e.g., the Recursive Least Squares (RLS) algorithm. The rate of convergence for adaptive filters is generally reduced for long filters since the step-size is often inversely proportional to the number of filter taps [1].
When we use the above steps to perform ICA, transforming the recorded signals to Fourier transform will also lead to insufficient information of signals. If one recorded signal is passed in the frequency domain using a conventional Fourier transform, it will result into one new signal where each sample corresponds to one frequency. Hence all the time-related information is lost and the ICA algorithm derived before will be helpless as it relies on averages of contrast functions over time. Thus we transform the recorded signals into time-frequency plane instead of frequency plane. This transformation is called Short-Time Fourier Transform (STFT). A short-time Fourier transform of a signal contains information about its frequency content and the variations of this content over time [2].

3.7.2 Use of Contrast Function

ICA algorithm designed so far in the instantaneous case treat the actual recordings which are real-valued time series. But, the information we obtain from STFT are complex as for regular Fourier Transform. Fast ICA for complex valued time series should have contrast functions that are defined with complex arguments. Therefore the algorithm has to be adapted to be suitable to complex-valued arguments.

3.7.3 Permutation Ambiguity

The most delicate problem and the main issue of BSS in the frequency domain approach is the permutation ambiguity, this ambiguity is induced by the instantaneous ICA algorithm. The independent components evaluated by the algorithm are similar to the sources up to a permutation of the channels and a multiplication by a constant. However in this frequency approach, it can be seen that for each frequency a new ‘instantaneous’ ICA problem has to be solved. So there are as many instantaneous problems to solve as frequency bins. But due to the permutation ambiguity, for different frequencies, the components will be evaluated in a priori different order. In order to be reconstructed and transformed back to the time domain, the random permutations have to be targeted and reversed. This is nowadays a huge problem when dealing with convolutive blind source separation. Several algorithms exist, based on various things such as geometrical properties of the positions of the sources and receivers, properties of the sources signals, or assumptions about the un-mixing filters. However, none of them is able to achieve this task perfectly, regardless of the kind of signals used. It is indicated that spatial information is very valid for solving the source ordering ambiguity inherent in the frequency-domain ICA [8]
Chapter 4

Existing Methods to solve Permutation Ambiguity

The permutation ambiguity becomes an important problem as opposed to the instantaneous mixing case. This is due to the fact that permutation ambiguities are present for every frequency bins. Before passing the estimated independent components back to the time domain, it is necessary to reverse these permutations. Else the resulting component in the time domain would be again a mixture of the sources, with one frequency band belonging to one source, and another belonging to a different source.

Different approaches exist to solve this problem. Basically the permutations can be picked out and frequency band switching is done. The basic idea is to locate the permutations by noticing sudden changes in some properties of the filters or the spectra of the estimated components.

4.1 Continuity of Spectral Envelope over Frequency

This method explained here forces the continuity of the un-mixing filters. Permutations are achieved when the un-mixing filters are less continuous than their permuted versions. It is especially efficient for speech signals or signals which present similar characteristics than speech. It relies on the assumption that, in a time-frequency plane, the temporal envelope of a signal varies slowly across the frequency. This is illustrated by figure 4.1

For one signal, at one frequency $f_k$, the correlation between the STFT of the first component at this frequency $f_k$ and the same component at the next frequency $f_{k+1}$ is compared with the correlation between the first component at $f_k$ and the second component at $f_{k+1}$. If the first component at $f_k$ happens to be more correlated to the second component at $f_{k+1}$ then a permutation is made.

4.1.1 Drawback

The main drawback of this method is that it is comparing correlation only between two neighboring frequencies. If an error is made at one frequency, then the next frequency bin will be compared to the erroneous previous one so it is likely to be wrong too. Hence when an error is made at one frequency, the error propagates to the upper frequencies. In the end results from this method are just isolated permutations.
4.2 Solving Permutation using ICA based Clustering

This method relies on the observation that the estimated un-mixing vectors or their inverse, for a same component, seem to follow a given direction. To achieve this, the problem is regarded as an instantaneous ICA problem: the un-mixing vectors are seen as unknown linear combination of two basis vectors which are actually pointing to these privileged directions (in this case horizontal and vertical directions). The estimated mixing vectors (column vectors) $a_i(f)$ for the two channels ($i = 1; 2$) are arranged in a matrix

$$X = \begin{bmatrix} a_1(f_1), & a_2(f_1), & a_1(f_2), & a_2(f_2), & \ldots, & a_1(f_{\text{max}}), & a_2(f_{\text{max}}) \end{bmatrix} \quad \Rightarrow (4.1)$$

and the following ICA problem is solved using a complex-valued ICA algorithm

$$\tilde{X} = \tilde{A} \cdot \tilde{S} \quad \Rightarrow (4.2)$$

The estimated matrix

$$\tilde{X} = [\tilde{a}_1, \tilde{a}_2] \quad \Rightarrow (4.3)$$

contains the privileged direction of the mixing vectors (basis vectors) and should be normalized column-wise. The matrix $\tilde{S}$ contains the coefficients of the linear combinations to obtain the un-mixing vectors from the basis vectors. Two clusters are built, corresponding to each of the basis vectors. Then the un-mixing vectors are assigned to a cluster by pairs. When one vector is assigned to a cluster, the vector from the other channel at the same frequency should be assigned to the other cluster. In matrix $\tilde{S}$, column vectors are coming by pair, each pair corresponding to a
frequency. The vectors from an odd column correspond to the first channel and the vectors from an even column to the second channel. By examining the coefficients in $\hat{S}$, it is possible to know which vector from the pair should be assigned to which direction. If for one vector from $\hat{S}$ the coefficient associated to $\hat{a}_1$ is larger in modulus than the coefficient associated to $\hat{a}_2$ then this vector should be assigned to the first cluster. If however this is also the case for the other vector of the pair, then they cannot be assigned to the same cluster, as this would not be a permutation anymore. Then the ratios of the coefficients from $\hat{S}$ for these two vectors should be examined, and assign the vector of the pair which has the strongest weight on the direction of basis vector $\hat{a}_1$ to the first cluster.

4.3 Other Methods to Solve Permutation Ambiguity

In [10], presents some underlying principles of different algorithm to solve the permutation ambiguity. In [4] a method is also presented that relies on information from both the un-mixing filters and time-frequency representation of the estimated components. The continuity of the filters is a first preprocessing step that solves partly the permutation ambiguity, providing new filters that present only isolated permutations. These isolated permutations are then detected by assuming a smooth variation of the time-frequency representation when changing the frequency. Sudden changes in energy indicate an isolated permutation.
Chapter 5

Time Difference of Arrival (TDOA)

A convolutive blind source separation system can be viewed as multiple sets of adaptive beamforming, which means the separation filter array for every output can be viewed as a beamformer. Thus a beamforming approach is used to combine with frequency domain convolutive BSS to deal with frequency permutation and the arbitrary scaling problem [16].

The fundamental principle behind (DOA) estimation using microphone arrays is to use the phase information present in signals picked up by microphones that are spatially separated. When the microphones are spatially separated, the acoustic signals arrive at them with time differences [3]. For an array geometry that is known, the time-delays are dependent on the Direction of Arrival (DOA) of the signal. When the sound source is present in the far field, it is sensible to estimate the DOA of the signal as we need to know the direction from which the sound source emerges so that will help in source separation or source position estimation. In the current application, the sources are spherical waves and so it is not necessary to calculate DOA. Instead we go for Time Difference of Arrival (TDOA). By analyzing the directivity patterns formed by a separation matrix, source directions can be estimated and permutations can be aligned [13].

The method we have implemented depends on the Time Difference of Arrival of signals to the microphones. TDOA technique is a beamforming approach which uses the time delay information to solve the permutation ambiguity that ICA suffers from [5].

![Figure 5.1 TDOA representation](image-url)
The time difference of arrival is nothing but the time difference of the signal that is arriving between each microphone with respect to reference microphone [5]. Original time difference of arrival of signal is modeled in such a way that the number of time difference values depends on the number of microphones. If we have M microphones, we can define \( \frac{1}{2}M(M-1) \) TDOA of a source, for each pair of microphones. Thus, let us consider as below,

\[
\eta_{jk} = \tau_{jk} - \tau_{Jk}, j = 1, \ldots, M
\]

\( \rightarrow (5.1) \)

Where, \( r \) is the original time difference of arrival and \( \tau \) is the time delay between source \( k \) to microphone \( j \) where \( J \) is the reference microphone.

This original time delay is used as a reference in order to check with estimated time difference of arrival. The estimated TDOA are calculated here in the frequency domain because they depend on basis vector elements \( a_{ji} (f) \) and so they are frequency dependent. The separation matrix \( W \) which is formed from complex ICA gives an output \( Y \) by the formula \( Y = W*X \). Since the original mixing process remains unknown practically, it is not able to realize the mixing matrix. But, a separation matrix \( W \) produced from ICA can be used to find the estimated mixing matrix \( A \). \( W^T \) gives the estimated mixing matrix understood from the formula \( X = W^T * Y \), where \( Y \) is the independent signal which is expected to be similar to the original source signal at ideal conditions. This inverse value \( A = W^T \) gives the basis vector elements \( a_{ji} \) which is used to determine the estimated time difference of arrival. Each basis vector element \( a_{ji} \) in the matrix \( A \) is used to calculate TDOA as in the formula,

\[
\eta_{ji}(f) = \frac{-\arg\left(\frac{a_{ji}(f)}{a_{j1}(f)}\right)}{2\pi f}
\]

\( \rightarrow (5.2) \)

\( r \) is the time difference of signal with respect to microphone. Here, two different subscripts \( k \) (original source index) and \( i \) (estimated source index) are used for the source index to calculate original TDOA and estimated TDOA because permutation alignment is not done in this stage.

We explain the reason below. With the time delay model in Figure 5.1, the frequency response \( h_{jk} (f) \) can be approximated as

\[
h_{jk} (f) \approx e^{-i2\pi f \tau_{jk}}
\]

\( \rightarrow (5.3) \)

It is also to be understood that,

\[
a_{ji} (f)/a_{ji} (f) = a_{ji} y_i / a_{ji} y_i \approx h_{jk} s_k / h_{jk} s_k = h_{jk} (f) / h_{jk} (f) = e^{-i2\pi f (\tau_{jk} - \tau_{jk})}
\]

\( \rightarrow (5.4) \)
Now, when we take argument for the above result, we obtain the estimated TDOA from (5.2). Here, there are only 2 microphones considered in order to perform BSS.

### 5.1 Combining Technique

TDOA technique is used as the proposed method here to solve the problem of permutation ambiguity. It is a simple robust technique to align the frequency bins of the independent source signals $Y$ according to the time delay information obtained using TDOA formula [5]. When the delay information is taken into consideration, the permutation problem can be almost solved from low to high frequency bins as it is a particular value that can be compared to the original value and hence there will not be any chaos between consecutive frequency lines. The advantage of beamforming over source separation lies in its use of geometric information. Information such as sensor positioning or source location is often readily available and can be used to design responses [10].

The original TDOA of each pair of microphones is calculated to be $\tau_{jk}$ that has been calculated using the original time delay information and is used as a reference to compare with the estimated TDOA for each frequency bins. The estimated TDOA for each frequency bins gives a value that would be far nearer to any one of the original TDOA corresponding to any source. So, they are compared to see whether the estimated TDOA has a nearer value to either TDOA of source 1 or source 2. So, this is done with all frequency bins. Suppose, the TDOA value currently present in $Y_1$ seems to have a value nearer to the original TDOA of source $S_2$, the corresponding frequency bin is permuted so as to group the frequency bin of source 1. If there is no such mismatch with $Y_1$ and $S_2$ or $Y_2$ with $S_1$, then they are left as such without any alignment. This is continued throughout all frequency bins to check whether they are permuted in anyways. Here, interestingly, the corresponding columns of the separation matrix $W$ are also permuted when the rows of independent source signals $y_1$ and $y_2$ are permuted. If this is done to all the frequency bins, then they are scaled back to the time domain in order to physically hear to the estimated sources. Now, we could hear a clearly separated speech very much equal to the original speech.

The output of this approach is seen in the later chapters to compare and witness the improved outputs after implementing TDOA technique.
Chapter 6

Test and Analysis

In this chapter the results of the implemented method described in chapter 5 is discussed. This method is tested on various scenarios of mixed speech signal and analyzed by comparing the results with the previously implemented method (Envelope Continuity). All the test are operated for two recordings and two sources, the implemented algorithm can solve the permutation ambiguity for more than two sources but for easy understanding two source and two recording scenario is considered.

6.1 Method

A system was modelled with 64 microphones (8X8) and 2 sources. Only free-field condition is considered here.

\[ a(i, k) = \frac{1}{r(i, k)} e^{j \frac{2 \pi f r(i, k)}{c}} \]

Figure 6.1 System model

From the microphone array we could easily find the position of the source in space and calculate the time difference of arrival for each microphone. Since just 2 microphones are enough to separate the signals, any 2 adjacent microphones are selected from the array and source separation is performed as explained in chapter 5.

For our case we have considered two speech signals of length 4sec and the separation has been carried out. The spectrogram of both the signals are shown in figure 6.2
figure 6.2(a) is a voice of a female hence some lines are visible in the spectrogram which depends upon the generation of voice of each individual and figure 6.2(b) is a voice of male. These two signals were mixed in the above condition as shown in figure 3.1 and are separated using the implemented algorithm.

6.2 Signal to Interference Ratio

Signal to interference ratio (SIR) is a measure of the level of a desired signal to the level of the interfering signal. It is defined as the ratio of the signal power to the interfered signal.

\[
SIR_{\text{improvement}} = SIR_{\text{output}} - SIR_{\text{input}} \rightarrow (6.1)
\]

6.3 Comparison of TDOA and Envelope Continuity

Various scenarios have been considered as an input to the algorithm, they have been discussed as follows.

Case 1

First the distance between microphones were changed and the performance was recorded. The graph shows the variation of SIR improvement when the distance between the microphones are changed. SIR improvement of the envelope continuity in this case was not able to follow a trajectory while the SIR improvement of the TDOA method is predictable. Envelope continuity method has a great effect on the initialization vector of the Fast-ICA algorithm as discussed above.

In TDOA the SIR improvement decreases when the distance between microphones increase because the delay will increase when the distance between microphones keep increasing. This may certainly lead to decrease in separation performance as the microphones may not receive
sources completely and thus selecting the right source to permute back would be a problem causing the decrease in SIR improvement.

![Figure 6.3 SIR improvement for TDOA against Envcont by varying the distance between the microphones](image)

**Figure 6.3 SIR improvement for TDOA against Envcont by varying the distance between the microphones**

**Case 2**

Secondly the distance between the sources were changed and the performance was recorded, the figure 6.4 shows that when the distance between the sources increase, there is a decrease in SIR improvement. For both envcontinuity and TDOA, initially in the graph, there is an increase in SIR improvement and then it starts decreasing. The initial increase in the SIR improvement is due to the fact that the sources are in a right position for the system to obtain the maximum SIR improvement. Then, the decrease in SIR improvement is due to the fact that the source signals may not reach the recording system properly. The delays are larger and will be almost the same for both the sources so there is a confusion of aligning the delays of each frequency line to its respected source, thereby creating permuted sources again. But apart from this, the initialization vector in the ICA algorithm plays a major role for this orderless trajectory on both the methods. It should have a very good convergence rate of 500 or 1000 depending upon the choice of the initialization vector in order to have a gradual trajectory pattern

**Case 3**

The third scenario is to change the distance between the microphone array plane and the source plane to check the SIR improvement. As in the previous case, for both envelope continuity and
TDOA, there is an initial increase in the SIR improvement in figure 6.5 is due to the right position for the system to obtain the maximum SIR improvement. Then it gets reduced for the same reason we have discussed above in case 2. Initialization vector is the major role for this oscillatory trajectory for both the methods.

Figure 6.4 SIR improvement for TDOA against Envcont by varying the distance between the sources

Figure 6.5 SIR improvement for TDOA against Envcont by varying the distance between the sources and the microphone array
Case 4

The final scenario is to change the window length for the STFT function. Here in figure 6.6, for both envelope continuity and TDOA, we could see the drop in SIR improvement when there is an increase in window length for the STFT function. For a little increase in window length for eg. (256 to 512), there are more number of frequency lines at a closer interval, which can form a smooth curve and thus gives a better performance in SIR. But for the further increase in window length such as 1024 or 2048, it results in decrease in the SIR improvement which is due to the less number of computational time signals available when separating the sources apart from having more number of frequency lines at closer interval. Still, when we compare both the methods, TDOA performs better than envelope continuity which is the advantage of the proposed method.

![Graph showing SIR improvement for TDOA against Envcont by varying the window length](image)

Figure 6.3 SIR improvement for TDOA against Envcont by varying the window length
6.4 Initialization Problem in ICA:

The conventional ICA method inherently has one other significant disadvantage which is due to slow and poor convergence through nonlinear optimization in ICA, particularly when introducing a poor initial setting of the unmixing matrix [8]. Unmixing matrix did not have a qualitative initialization which also causes bad signal to interference ratio at certain conditions. It is predicted that the lots of misalignment of SIR as we reviewed above in different scenarios is mainly due to the fact that the initialization of unmixing matrix is not robust. There are also recent researches going on in order to make a robust initialization. One other way of reducing the misalignment can be increasing the number of iterations that the unmixing matrix runs for each source. This cannot solve the problem completely but can be steady to some extent. For all the above test cases, each value at each condition is run for 15 times and average of 15 is taken to depict every value. There was lot of variations in the values for every condition. It is suggested, going through the reference [8] to find a reasonable solution the author has defined.
Chapter 7

Conclusions and Future work

• In this project, we have reviewed and implemented the approach of combining Beamforming and BSS for convolutive mixtures for a better separation performance.

• Solving permutation ambiguity was the main aim of this project. We proposed TDOA technique to solve this issue. The algorithm is tested using 2 speech mixtures. (male and female voice of 30 sec. audio)

• Results were evaluated in-terms of SIR improvement. Simulation results confirm our expectations and show that TDOA works pretty well than the envelope continuity method in all the conditions.

• When microphone positions are changed, the performance of TDOA is better and highly predictable compared to envelope continuity. Envelope continuity did not follow a predictable decreasing fashion of SIR and it oscillates for every increase in microphone distance.

• Similarly, with the increase in the window length, both TDOA and envelope continuity decreases in its SIR improvement. But, for the highest value of window length, the performance of TDOA is very much better than envelope continuity. SIR is better even at the worst case scenario also, for TDOA.

• We have implemented this technique in free field simulation. The future work would be to implement this method to the real environment.

• Since Fast-ICA has a random initialization vector and has a lots of parameters to consider, more robust algorithms like JADE algorithm can be replaced over Fast-ICA
References

Dissertations


Books


Journal Articles


**Conference Articles**


Technical Reports