AUDIO SOURCE SEPARATION WITH CONVOLUTIVELY MIXED SIGNALS

by

Doğaç Başaran

B.S. in E.E., Boğaziçi University, 2002

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Electrical and Electronic Engineering
Boğaziçi University
2005

AUDIO SOURCE SEPARATION WITH CONVOLUTIVELY MIXED SIGNALS

APPROVED BY:

Assist. Prof. Kerem Harmancı (Thesis Supervisor)	....
Assoc. Prof. Levent M. Arslan	....
Assist. Prof. Ata Akın	....

DATE OF APPROVAL: 25.10.2005

ACKNOWLEDGEMENTS

I would like to thank to my supervisor Assist. Prof. Kerem Harmanci and my former supervisor Assoc. Prof. Levent M. Arslan for their patience and support during this difficult thesis. I could not complete this thesis without their advises and creative ideas. I would like to thank Assist. Prof. Ata Akın for his gentle attitude and participation in my thesis committee.

I would like to thank my room mates Kemal, Erman, Ismail and Doruk and my dearest friends Kaan,Levent And Onur. They never stop supporting me and make life amusing for me. Having these kinds of friends, I would pass much more difficult problems in my life.

I would like to thank to Prof. Kadri Özçaldıran for his advices and understanding. Whenever I need help in any part of my life, he was there to help me during my whole university life. He encouraged me for following my dreams.

My special thanks go to my family and to Nihan for their endless love and limitless tolerance, they support me in every part of my life. Completing my thesis is even more important for them than it is for me. They never stop believing in me even in worst conditions.

I would like to thank my friends Ebru, Turgay, the legendary Erdem Yörüks and the people in BUSİM.

ABSTRACT

AUDIO SOURCE SEPARATION WITH CONVOLUTIVELY MIXED SIGNALS

In this thesis, we addressed the problem of audio source separation of convolutively mixed signals using microphone arrays. Independent Component Analysis is a major statistical tool for solving this problem. In real room environments, the recordings of audio signals usually involve the signal itself as well as some delayed and amplitude modulated versions of this signal. This is due to reverberation or echo of the room which occurs as a result of reflection of walls, ceiling, ground as well as the furniture inside. Separation of signals that are mixed in these kinds of environments is a challenging problem. There exist both time-domain and frequency-domain solutions to this problem. We mostly focus on frequency domain methods where ICA is performed separately in each frequency bin. Permutation ambiguity which is the basic problem in frequency domain ICA, is also handled with two basic approaches which are, direction of arrival method which is motivated by conventional beamforming theory and interfrequency correlations which is motivated by nonstationarity of speech signals. Conventional and Adaptive Beamforming methods are also implemented here. These methods separate sources by exploiting the physics of the propagation.

ÖZET

BÜYÜK HARFLERLE TEZİN TÜRKÇE ADI

Bu tezde, evrişimli bir şekilde karışmış ses kaynaklarının mikrofon dizileri kullanılarak ayrıştırılması problemini ele aldık. Bağımsız Bileşen Analizi (BBA) bu problemin çözümündeki başlıca istatistiksel yöntemdir. Gerçek oda koşullarında ses sinyallerinin kayıtları genellikle sinyalin kendisini içerdiği gibi, aynı zamanda bu sinyalin gecikmiş ve genliği modüle edilmiş hallerini de içerir. Bu durum, odanın duvar, tavan ve tabanından ve odanın içindeki eşyalardan yansıyan yankılanmalardan kaynaklanır. Bu tür ortamlardaki karışmış sinyallerin ayrımı oldukça zor bir problemdir. Bu probleme hem zaman düzleminde hem de frekans düzleminde çözümler mevcuttur. Biz bu tezde çoğunlukla BBA'nin her frekans bandında ayrı ayrı kullanıldığı frekans düzlemindeki çözümlere odaklandık. Frekans düzleminde BBA'nde karşılaşılan en temel problem olan permütasyon belirsizliğini iki temel yaklaşımla ele aldık: Klasik hüzmeleme teorisinin bir sonucu olarak ortaya çıkan Varış Yönü yöntemi ve durağan olmayan konuşma sinyallerinden ortaya çıkan Frekanslararası Bağıntı yöntemi. Klasik ve Uyarlanabilir Hüzmeleme yöntemleri de bu tezde uygulanmıştır. Bu yöntemler yayılım fiziğinden yararlanarak ses kaynaklarını ayırırlar.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS/ABBREVIATIONS

$x(n)$	Audio Mix Signal
$x_1(n)$	Audio Mix Signal 1
$x_2(n)$	Audio Mix Signal 2
$s(n)$	Source Signal
$\hat{s}(n)$	Source Estimate
$X(f,t)$	Time-Frequency Signal
t_1	Random variable
t_2	Random variable
pdf	Probability Distribution Function
P(t_1)	Probability Distribution Function of t_1
P(t_2)	Probability Distribution Function of t_2
P(t_1,t_2)	Joint Distribution Function of t_1 and t_2
F{.}	Fourier Transform Operator
win()	Windowing Function
t_S	window position
T_s	Number of windows
a_{ij}	Description of a_{ij}
α	Description of α
STFT	Short Time Fourier Transform
DOA	Direction of Approval
DA	Description of abbreviation
ICA	Independent Component Analysis
DFT	Discrete Fourier Transform
FFT	Fast Fourier Transform
FastICA	A fast fixed point algorithm
FD-PSOLA	Frequency Domain Pitch Synchronous Overlap and Add Method
EVD	Eigenvalue decomposition

MIMO	Multiple Input Multiple Output
MVDR	Minimum variance distortionless response
LCMV	Linearly Constrained Minimum Variance Beamforming

1. INTRODUCTION

Blind signal separation (BSS) is an emerging technique of multichannel signal processing and data analysis. The "blind" here means that the mixed components are latent or unobservable, actually only the mixed signals are observable. Assume that you are in a room with lots of people talking simultaneously and you are trying to focus on one of the speakers. This is known as "cocktail party" problem. BSS is motivated by the cocktail party problem in the sense that you are trying to separate one of the sources from others. In practice audio signals recorded with a microphone array are used for BSS. Independent Component Analysis (ICA) which is a linear transformation method, uses statistical independence of the sources to separate them. Nongaussianity of sources is used as a measure of independence in ICA methods. Beamforming is another approach that uses the spatial locations of sources in the physical world, to separate them [15],[14].

Audio Source separation is a BSS problem where the unobserved sources are acoustic signals such as speech and the unobserved mixing system the impulse responses of microphones in a room. In real room environments, the recordings of audio signals not only involve the signal itself but also some delayed and amplitude modulated versions of this signal. This is due to reverberation or echo of the room which occurs as a result of reflection of walls, ceiling, ground as well as the furniture inside. This is called convolutive mixing model which is challenging problem. In this thesis, we concentrated on the frequency-domain ICA methods in order to solve problem of separation of convolutively mixed audio sources [7],[8],[9],[10],[11],[12],[13].

There are several researches based on audio source separation and ICA models. The basic ICA model where the signals are instantaneously mixed which means there is no reverberation is described in [1],[2], [3] and [4].

Methods developed for solutions of basic ICA model can be categorized as gradient methods and fixed point algorithms. A fast fixed point algorithm (FastICA) is developed in [1],[18]. The FastICA algorithm is modified in [19] for the complex valued signals, which is usually the case for the frequency domain ICA. There are several gradient type algorithms which uses different types of nongaussianity measures and combinations of them. Natural gradient is one of the important approaches among gradient methods that is developed by Amari in [1] and [20]. A method that uses information maximization approach [21] and natural gradient is developed in [9]. Maximum Likelihood with infomax principle is used in [22]. Nongaussianity measures are defined in [1],[2],[3] and [21].

For the convolutive source separation problem which is much more complicated than instantaneous mixing, both time domain and frequency domain solutions exist. Although the details of time domain methods can be found in section 2.6.1, it is convenient to highlight some important approaches for time domain here. There are three types of approaches in time domain

- Nonwhiteness approach
- Nonstationary approach
- Nongaussianity approach

In [12] and [13] a broadband algorithm is developed based on nonwhiteness and nonstationarity of signals. In [11], they also add the nongaussianity property in order to use the higher order statistics in their approach.

In the frequency domain ICA, the separation is done at each frequency bin separately which leads to the problems of permutation and scaling which are the inherited ambiguities of ICA. Actually the permutation ambiguity is the difficult problem so that researchers usually focus on that subject. For scaling ambiguity there are 2 robust solutions in [7] and [9].

For solving permutation ambiguity, there are two basic ways, beamforming and interfrequency correlations. Beamforming which is also called Direction of Arrival method is an array processing technique which is used to localize the spatial position of the sources. Beamforming theory is explained in [15] and [14]. DOA arrival approach is discussed in [9] and [10].

A method which is using interfrequency correlations based on temporal structure of speech signals, is developed in [7]. Permutation alignment with neighboring correlations is another method which is discussed in [8].

In [9], a new robust algorithm is developed by using a hybrid algorithm of DOA approach and correlations approach.

Audio source separation is one of the basic applications of independent component analysis which is the main subject of this thesis. Since the noise outside can be considered as an independent component, ICA algorithm can be used to reduce noise or increase SNR, thus can be used for speech enhancement.

In noisy environments, the performance of speech recognition algorithms becomes extremely low. Audio source separation with ICA can also be used for increasing the performance of speech recognition algorithms especially in noisy environments.

ICA is actually developed for dealing with the problems that are closely related to cocktail party problem, but as the interest increased on that subject, new areas discovered that ICA can be used.

ICA algorithms are currently in use for electroencephalographic (EEG) and magnetoencephalographic (MEG) data to separate certain source signals that are artifacts or noise sources not corresponding to brain activity. Another important application area of ICA is feature extraction which is used in data compression and pattern recognition. ICA is also used for data analysis in areas as economics, psychology or social sciences.

In this thesis, we first give the theoretical background of ICA. The definition, mixing models, time and frequency methods, ambiguities and the optimization algorithms are described in this chapter 2. Next chapter the theoretical information on beamforming theory is given involving the conventional and adaptive approaches and the limitations. After that the methodology is discussed in chapter 4. The overlap-add algorithm, FastICA and the methods for solving permutation problem which are DOA approach and correlations approach are given in detail. Then the results of the implemented algorithms are given in chapter 5. In the last chapter, the results are interpreted and some conclusions with thoughts for further research.

2. INDEPENDENT COMPONENT ANALYSIS (ICA)

Several algorithms and methods have been developed to find a suitable linear transformation of multivariate data. Principal Component analysis [1], projection pursuit [24] and [25], factor analysis [23] are some examples of these kind of algorithms. What they all have in common is, these methods define their principle that tells which transformation is optimal according to an optimization criterion. Independent Component Analysis is an emerging method that also searches for a linear transformation of multivariate data.

2.1. Definition

Assume that we observe n linear mixtures $x_1, ..., x_n$ of n independent components

$$x_j = a_{j1}s_1 + a_{j2}s_2 + ... + a_{jn}s_n \quad \text{for all} \quad j \tag{2.1}$$

We assume that each mixture x_j as well as each independent component s_k is a random variable. It is also assumed that both the mixture variables and the independent components have zero mean which may not be the case for some situations.

$$\mathbf{x} = [x_1, x_2, ..., x_n]^T \tag{2.2}$$

$$\mathbf{s} = [s_1, s_2, ..., s_n]^T \tag{2.3}$$

$$\mathbf{A} = (\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_n) = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & \ddots & & \vdots \\ \vdots & & & \\ a_{n1} & \dots & & a_{nn} \end{pmatrix} \tag{2.4}$$

where $\mathbf{x}$ is the random vector whose elements are the mixtures $(x_1, x_2, ..., x_n)$, $\mathbf{s}$ is the random vector whose elements are the $(s_1, s_2, ..., s_n)$, $\mathbf{a}_i$ is a column vector with elements $(a_{i1}, a_{i2}, ..., a_{in})$ and $\mathbf{A}$ is the mixing matrix consists of these vectors $(\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_n)$.

With these definitions ICA model is defined as,

$$\mathbf{x} = \mathbf{A}\mathbf{s} = \sum_{i=1}^{n} \mathbf{a}_i \mathbf{s}_i \tag{2.5}$$

In this model, the independent components are latent variables, which means that they are not observable. The mixing matrix $\mathbf{A}$ is also unknown. The random vector $\mathbf{x}$ is the only observable data in this model, the mixing matrix $\mathbf{A}$ and independent components have to be estimated by using this observable data.

The basic assumption of this model is the components s_i are statistically independent and ICA is a statistical tool to find a linear transformation ,$\mathbf{W}$, so that the estimated components $\hat{s}_i$, are as independent as possible, in the sense of maximizing some function that measures independence,

ICA uses gaussianity as a measure of independence which is motivated by Central Limit Theorem [1]. Basically central limit theorem states that the distribution of a sum of independent random variables tends toward a gaussian distribution, which can be summarized as, sum of two random variables, not necessarily identically distributed, has a distribution that is more gaussian than any of the original random variables.

By using this motivation, ICA finds a linear transformation that makes the estimated components $\hat{s}_i$ as non-gaussian as possible. And maximization of non-gaussianity leads to independence.

$$\hat{\mathbf{s}} = \mathbf{A}^{-1}\mathbf{x} = \mathbf{W}\mathbf{x} = \begin{pmatrix} w_{11} & w_{12} \\ w_{21} & w_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{2.6}$$

ICA method can be expressed in two parts

- *Non-gaussianity measure cost function* which determines the statistical properties like robustness or consistency of ICA method

- *Optimization algorithm* which determines the algorithmic properties like convergence speed or stability

2.2. Non-gaussianity Measures

2.2.1. Kurtosis

Kurtosis is the name of the fourth order cumulant of a random variable. Kurtosis of x, denoted by $\mathrm{kurt}(x)$ is defined as,

$$\mathrm{kurt}(x) = \mathrm{E}\{x^4\} - 3\left[\mathrm{E}\{x^2\}\right]^2 \tag{2.7}$$

where $\mathrm{E}\{.\}$ is the expectation operator and x is a zero mean random variable. If x is normalized so that it has unit variance, $\mathrm{E}\{x^2\} = 1$, then the kurtosis becomes $\mathrm{E}\{x^4\} - 3$ which is simply a version of 4'th order moment. For a zero mean gaussian random variable, the fourth order moment equals to $3\left[\mathrm{E}\{x^2\}\right]^2$, therefore the kurtosis of a gaussian random variable is zero.

Subgaussian random variables, which have flat pdf like uniform distribution, have negative kurtosis. On the other hand, supergaussian random variables, which have sharp pdf with heavy tails like laplacian distribution, have positive kurtosis. That's why the absolute value or the square of kurtosis is used in algorithms as a measure of non-gaussianity.

However kurtosis has some drawbacks as a measure of non-gaussianity that it is very sensitive to outliers in measurements. Since it is using 4^{th} order moment, a single measurement error may mislead the calculation.

2.2.2. Negentropy

Negentropy is based on the information-theoretic quantity of differential entropy. The differential entropy H of a random vector $\mathbf{y}$ with pdf $f(\mathbf{y})$ is,

$$H(\mathbf{y}) = -\int f(\mathbf{y}) \log f(\mathbf{y}) d_{\mathbf{y}} \tag{2.8}$$

Entropy is the basic concept of information theory. Entropy of a random variable is the information you get by observing that random variable. Briefly it states that, the more unpredictable or random the variable is, the larger its entropy. And a gaussian random variable has the largest entropy among random variable with same variance [1].

Negentropy is a normalized version of differential entropy which is zero for gaussian random variable and positive for non-gaussian. Negentropy J is defined as,

$$J(\mathbf{y}) = H(\mathbf{y_{gauss}}) - H(\mathbf{y}) \tag{2.9}$$

where $\mathbf{y_{gauss}}$ is a gaussian random variable.

Negentropy is a statistically well justified and robust estimator of non-gaussianity but the computation is very difficult since it requires to estimate the pdf.

<u>2.2.2.1. Approximations of Negentropy.</u> There are several approximations for negentropy, the classical method is using higher-order moments [1], [5],

$$J(y) \approx \frac{1}{12}\mathrm{E}\{y^3\}^2 + \frac{1}{48} kurt(y)^2 \tag{2.10}$$

However this approximation turns into maximization of kurtosis when the pdf of y is symmetric which is the case most of the time. So it is not robust against outliers.

For this reason, a more robust approximation of negentropy is developed in [6] which is based on maximum-entropy principle,

$$J(y) \approx \sum_{i=1}^{p} k_i \left[\mathrm{E}\{G_i(y)\} - \mathrm{E}\{G_i(\nu)\}\right]^2 \tag{2.11}$$

where k_i are positive constants, ν is zero mean, unit variance gaussian variable and the functions G_i are non-quadratic functions [6]. If only one non-quadratic function is used,

$$J(y) \propto \left[\mathrm{E}\{G(y)\} - \mathrm{E}\{G(\nu)\}\right]^2 \tag{2.12}$$

When the pdf of y is symmetric, this approximation becomes a generalization of (2.10). When the non-quadratic function is chosen as $G(y) = y^4$, then this becomes a kurtosis based approximation as in (2.10).

2.2.3. Minimization of Mutual Information

Mutual information is natural measure of independence between random variables. It is equivalent to Kullback-Liebler divergence between the joint density $f(\mathbf{y})$ and the product of its marginal densities. Using the concept of differential entropy, the mutual information I between m random variables $y_i, i = 1, ..., m$ is defined as,

$$I(y_1, y_2, ..., y_m) = \sum_{i=1}^{m} H(y_i) - H(\mathbf{y}) \tag{2.13}$$

So from (2.13) it can be seen that mutual information is always non-negative and zero if and only if the variables are statistically independent.

For a linear transformation $\mathbf{y} = \mathbf{W}\mathbf{x}$,

$$H(\mathbf{y}) = H(\mathbf{x}) + \log|\det\mathbf{W}| \quad \text{from [1]} \tag{2.14}$$

From (2.13) and (2.14) we have,

$$I(y_1, y_2, ..., y_m) = \sum_{i=1}^{m} H(y_i) - H(\mathbf{x}) - \log|\det\mathbf{W}| \tag{2.15}$$

When y_i is constrained to be uncorrelated and of unit variance, the mutual information becomes,

$$I(y_1, y_2, ..., y_m) = C - \sum_{i} J(y_i) \tag{2.16}$$

As it is seen from the (2.16), there is a fundamental relation between negentropy and mutual information that is minimization of mutual information is equivalent to maximization of negentropy.

2.2.4. Maximum Likelihood Estimation

The log likelihood L as a function of $\mathbf{W} = (\mathbf{w}_1, \mathbf{w}_2, ..., \mathbf{w}_N)^T$ is defined as,

$$L(\mathbf{W}) = \sum_{k=1}^{M} \sum_{i=1}^{N} \log f_i(\mathbf{w}_i^T \mathbf{x}(k)) + M \log|\det\mathbf{W}| \tag{2.17}$$

where f_i are the density functions of source signals, M and N are the number of sensors and number of sources respectively.

When the expectation of (2.17) is taken,

$$\frac{1}{M}\mathrm{E}\{L(\mathbf{W})\} = \sum_{i=1}^{N} \mathrm{E}\{\log f_i(\mathbf{w}_i^T \mathbf{x})\} + \log|\det\mathbf{W}| \tag{2.18}$$

If the f_i are equal to the actual distributions $\mathbf{w}_i^T\mathbf{x}$,then the first term in the (2.18) would be equal to $-\sum_i H(\mathbf{w}_i^T\mathbf{x})$, so likelihood approach would be equal to negative mutual information up to an additive constant.

The difficult part of the maximum likelihood is the estimation of the densities f_i.

The estimation of the densities is a difficult task because in practice we don't know the independent components. Actually these densities need not to be estimated precisely, it is enough to estimate whether they are subgaussian or supergaussian [1].

2.3. Ambiguities of ICA

There are some indeterminacies or ambiguities of ICA method,

- ***Scaling Ambiguity*** is that we can not determine the energies of the independent components. Scaling the estimated independent components by some constants don't change their independence measure. Assume that $\mathbf{W} = \mathbf{A}^{-1}$ which is the exact inverse transformation and $\hat{\mathbf{W}}$ is some scaled version of $\mathbf{W}$. This is illustrated in (2.19) for 2×2 case which is modified version of (2.5),

$$\begin{pmatrix} c_1\hat{s}_1 \\ c_2\hat{s}_2 \end{pmatrix} = \hat{\mathbf{W}}\mathbf{x} = \begin{pmatrix} c_1 w_{11} & c_1 w_{12} \\ c_2 w_{21} & c_2 w_{22} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{2.19}$$

Since multiplying a random variable do not change the type of pdf, the independence criteria between s_1 and s_2 is unchanged

$$f(c_1\hat{s}_1, c_2\hat{s}_2) = f(c_1\hat{s}_1)f(c_2\hat{s}_2) \tag{2.20}$$

Intuitively we can say that multiplying a random variable t_1 with a constant c, does not give more information about another random variable t_2 which is independent from t_1.

- ***Permutation Ambiguity*** is that we can not determine the order of the independent components. Assume that we change the order of the rows of demixing matrix $\mathbf{W}$ in (2.6) where it is exactly equal to the inverse of mixing matrix,

$$\hat{\mathbf{W}} = \begin{pmatrix} w_{21} & w_{22} \\ w_{11} & w_{12} \end{pmatrix} \tag{2.21}$$

then, the resulting estimates become,

$$\hat{\mathbf{s}} = \begin{pmatrix} \hat{s}_2 \\ \hat{s}_1 \end{pmatrix} = \hat{\mathbf{W}}\mathbf{x} = \begin{pmatrix} w_{21} & w_{22} \\ w_{11} & w_{12} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{2.22}$$

This means that the demixing matrix $\hat{\mathbf{W}}$ leads to exactly same independent components but in different order. Since the independence between source estimates do not change, there is no possible way for ICA to distinguish these solutions.

As a result, the demixing matrix $\mathbf{W}$, can be estimated up to a scaling and a permutation which can be illustrated as,

$$\mathbf{H}\,\mathbf{W} = \mathbf{\Lambda}\,\mathbf{P} \tag{2.23}$$

where $\mathbf{\Lambda}$ is a diagonal scaling matrix and $\mathbf{P}$ is a permutation matrix.

2.4. Preprocessing Steps

There are two important preprocessing techniques for ICA which are centering and whitening.

2.4.1. Centering

We usually assume that both the mixtures and independent components have zero mean which simplifies the algorithm. If this assumption is not true for the mixture variables, they can always be made zero mean by centering which is simply subtracting their mean,

$$\mathbf{x}' = \mathbf{x} - \mathrm{E}\{\mathbf{x}\} \tag{2.24}$$

When the mixtures are centered, the estimated components also have zero mean which is shown in (2.25)

$$\mathrm{E}\{\hat{\mathbf{s}}\} = \mathbf{A}^{-1}\mathrm{E}\{\mathbf{x}'\} \tag{2.25}$$

The subtracted mean can be recovered by adding (2.26) to the independent components.

$$\mathbf{m} = \mathbf{A}^{-1}\mathrm{E}\{\mathbf{x}\} \tag{2.26}$$

2.4.2. Whitening

The whiteness of a random vector means that the components of that vector are uncorrelated and have unit variance. Uncorrelatedness which can be shown as,

$$\mathrm{cov}(y_1, y_2) = \mathrm{E}\{y_1 y_2\} - \mathrm{E}\{y_1\}\mathrm{E}\{y_2\} \tag{2.27}$$

is a weaker form of independence that is if two random variables are independent then they are also uncorrelated but uncorrelatedness of two random variable does not imply independence.

Whitening(sphering) is a linear transformation that makes the components of random vector white,

$$\mathbf{z} = \mathbf{V}\mathbf{x} \tag{2.28}$$

where $\mathbf{V}$ is the whitening transformation. Eigenvalue decomposition (EVD) of the covariance matrix is a basic method to obtain whitening transformation.

$$\mathrm{E}\{\mathbf{x}\mathbf{x}^T\} = \mathbf{E}\mathbf{D}\mathbf{E}^T \tag{2.29}$$

where $\mathbf{E}$ is the matrix of eigenvectors and D is the diagonal matrix with the eigenvalues in the main diagonal. The whitening transformation matrix is,

$$\mathbf{V} = \mathbf{E}\mathbf{D}^{-1/2}\mathbf{E}^T \tag{2.30}$$

When you whiten the observed mixtures, according to the model in (2.5) the new mixing matrix becomes $\tilde{\mathbf{A}} = \mathbf{V}\mathbf{A}$ which is orthogonal that is shown,

$$\mathrm{E}\{\mathbf{z}\mathbf{z}^T\} = \tilde{\mathbf{A}}\mathrm{E}\{\mathbf{s}\mathbf{s}^T\}\tilde{\mathbf{A}}^T = \tilde{\mathbf{A}}\tilde{\mathbf{A}}^T = \mathbf{I} \tag{2.31}$$

As a result of the new mixing matrix being orthogonal, whitening transformation restricts our search for demixing matrices into the space of orthogonal matrices which makes whitening a powerful preprocessing step.

2.5. Optimization Algorithms

2.5.1. Gradient Methods

Gradient type optimization algorithms usually have the form of minimizing a cost function $J(\mathbf{W})$ with respect to a parameter matrix $\mathbf{W}$. Gradient Descent is one of the classical approaches of unconstraint optimization problems. In this method, the gradient of $J(\mathbf{W})$ is computed at some initial point and by moving in the direction of negative gradient with distance, the function $J(\mathbf{W})$ is minimized.

The advantage of gradient methods is that they enable fast adaptation in a non-stationary environments. But depending on the choice of the learning rate, the convergence may be too slow or it may totally prevent convergence. Fixed-point iteration algorithms are alternatives to make the learning faster and reliable.

2.5.2. FastICA

FastICA is based on a fixed-point iteration scheme for finding a direction, a unit vector $\mathbf{w}$ such that the projection on that direction $\mathbf{w}^T\mathbf{x}$ maximizes the nongaussianity. Nongaussianity can be measured by kurtosis, negentropy or any of the nongaussianity measures. We used the approximation of negentropy $J(\mathbf{w}^T\mathbf{x})$ given in (2.12) which is developed in [3].

The basic form of FastICA for one unit is,

1. Choose an initial weight vector $\mathbf{w}$, which can be chosen as random.
2. The learning algorithm is

$$\mathbf{w}^+ = \mathrm{E}\{\mathbf{x}g(\mathbf{w}^T\mathbf{x})\} - \mathrm{E}\{g^{'}(\mathbf{w}^T\mathbf{x})\}\mathbf{w} \tag{2.32}$$

   where g and g' are the first and second derivatives of a nonlinear function G respectively.
3. Normalize the weight vector

$$\mathbf{w} = \mathbf{w}^+/\|\mathbf{w}^+\| \tag{2.33}$$

4. Apply from 2 until convergence

The convergence means that the updated weight vector $\mathbf{w}^+$ is in the same direction of old weight vector $\mathbf{w}$, which can be checked by dot product.

To estimate several independent components, the one-unit FastICA has to be used for several units with weight vectors, $\mathbf{w}_1, ..., \mathbf{w}_n$.

The outputs $\mathbf{w}_1\mathbf{x}, ..., \mathbf{w}_n\mathbf{x}$ should be decorrelated in order to prevent these weight vectors from converging the same maxima. Either Deflationary decorrelation algorithm or Symmetric decorrelation algorithm could be used for this purpose.

Deflationary Orthogonalization is a Gram-Schmidt type decorrelation which means the independent components are estimated one by one.

$$\mathbf{w}_{p+1} = \mathbf{w}_{p+1} - \sum_{j=1}^{p} \mathbf{w}_j \mathbf{w}_j^H \mathbf{w}_{p+1} \tag{2.34}$$

$$\mathbf{w}_{p+1} = \frac{\mathbf{w}_{p+1}}{\|\mathbf{w}_{p+1}\|} \tag{2.35}$$

Symmetric Orthogonalization is method to decorrelate all vectors in one step,

$$\mathbf{W} = \mathbf{W}(\mathbf{W^H W})^{\mathbf{-1/2}} \tag{2.36}$$

where $\mathbf{W} = (\mathbf{w_1 ... w_n})$ is the matrix of vectors.

Deflationary decorrelation has an advantage through symmetric decorrelation that it decorrelates from the most non-gaussian to least non-gaussian which is often means from most important to least important.

There are some advantages of FastICA algorithm among other algorithms,

- The convergence is cubic where as the convergence of gradient descent algorithms are linear [3]
- There is no step size parameter to choose
- There is no estimation of pdf of sources, use a nonlinearity to find non-gaussian sources.

2.6. Convolutive Mixing Model

Assume that a source signal is recorded in a real room environment, in that case basic ICA model (2.5) does not hold because of the reverberation of the room. The source signal propagates directly to the sensor as well as by reflecting through walls,ceiling or ground propagates through different paths and reach to the sensor in different angles. So the observed signal consists of an unknown source signal mixed with itself at different time delays and amplitudes. This can be defined by convolutive mixing in which source signals are convolved with impulse responses of the sensor,

$$\begin{aligned} \mathbf{x}(t) &= \mathbf{H}(t) * \mathbf{s}(t), \qquad (2.37) \\ \text{where } \mathbf{H}(t) &= \begin{pmatrix} h_{11}(t) & h_{12}(t) & \cdots & h_{1n}(t) \\ h_{21}(t) & \ddots & & \vdots \\ \vdots & & & \\ h_{n1}(t) & \cdots & & h_{nn}(t) \end{pmatrix} \end{aligned}$$

a more specific definition,

$$x_p(n) = \sum_{q=1}^{P} \sum_{\kappa=0}^{M-1} h_{pq}(\kappa) s_q(n-\kappa) \qquad (2.38)$$

where $h_{qp}(\kappa)$ is the impulse response of the q'th source signal in the p'th sensor. Figure 2.1 shows the linear MIMO system for convolutive ICA.

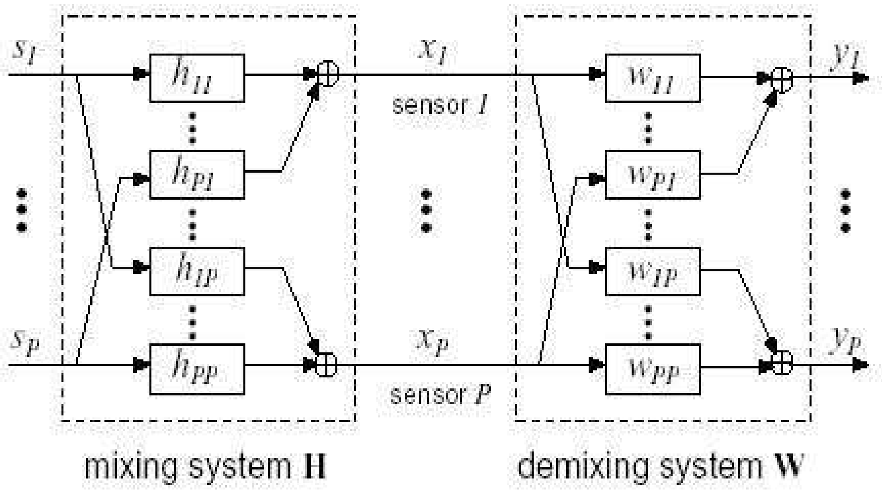


Figure 2.1. Linear MIMO system for ICA

The aim of ICA in this kind of mixing is to find a demixing filter, $w_{pq}(\kappa)$, that makes estimated components as independent as possible.

There are two kinds of algorithms for this problem, time domain methods and frequency domain methods.

2.6.1. Time-Domain Methods

There are three types of approaches for blind source separation problem in time domain,

- Nonwhiteness approach by simultaneous diagonalization of output correlation matrices over multiple time-lags.
- Nonstationarity approach by simultaneous diagonalization of short-time output correlation matrices at different time-lags
- Nongaussianity approach using higher order statistics for ICA

The first two properties are usually based on second order statistics (SOS), higher order statistics is used to exploit nongaussianity.

There are two algorithms Generic SOS and Generic HOS (TRINICON) developed in [12] and [11]. The former one uses second order statistics, the whiteness and nonstationarity properties and the latter one combine all three fundemental approaches for convolutive mixture.

For Generic SOS algorithm a cost function that contains correlation matrices that include several time-lags is used under the assumption of nonstatinarity. For Generic HOS algorithm the cost function is updated so that it includes higher order statistics. For estimation of nongaussianity, the minimization of mutual information is used. And the optimization algorithm is used as natural gradient which is developed by Amari [20].

2.6.2. Frequency Domain Methods

When the observed mixture signals transformed into fourier domain, which is shown for 2×2 case,

$$\mathrm{F}\{x_1\} = \mathrm{F}\{h_{11} * s_1 + h_{12} * s_2\} = X_1(f) = h_{11}(f)s_1(f) + h_{12}(f)s_2(f) \tag{2.39}$$

$$\mathrm{F}\{x_2\} = \mathrm{F}\{h_{21} * s_1 + h_{22} * s_2\} = X_2(f) = h_{21}(f)s_1(f) + h_{22}(f)s_2(f) \tag{2.40}$$

According to (2.39) and (2.40) it can be seen that, the signals are instantaneously mixed at each frequency bin. This property makes working in frequency domain very advantageous because the robust techniques that are developed for instantaneous mixing problem, can be used for this problem.

In frequency domain methods, due to non-stationary of the speech signals, they are segmented into smaller blocks that are assumed to be stationary, by moving windows. This segmentation is done by using overlap-add method which is explained in section 4.1.

After segmenting the audio signal into simple blocks by windowing, DFT of these blocks are taken with zero padding in order to prevent wrap-around effects. This is called Short Time Fourier Transform (STFT). STFT is used when you are not only interested in the frequency content of a signal but also the time information that frequency content is needed.

STFT

$$\hat{\mathbf{x}}(f, t_s) = \sum_t e^{-j2\pi ft}\mathbf{x}(t)\mathrm{win}(t - t_s), \quad t_s = 0, \Delta T, 2\Delta T, ... \tag{2.41}$$

where N is the number of points in discrete fourier transform, t_s is the window position. $\hat{\mathbf{x}}(f, t_s)$ has both time and frequency information of the signal $\mathbf{x}(t)$ which consists of row mixture vectors $(\mathbf{x}_1(t), .., \mathbf{x}_n(t))$.

Since the ICA algorithm is applied to each frequency bin separately, the time information of $\hat{\mathbf{x}}(f, t_s)$ is used for a fixed frequency f_0, that is the data to be solved to find independent components is,

$$\begin{aligned}\text{For } f &= f_0, \\ \hat{\mathbf{x}}(f_0, t_s) &= \hat{\mathbf{x}}(t_s) = \begin{pmatrix} \hat{\mathbf{x}}_1(t_s) \\ \vdots \\ \hat{\mathbf{x}}_n(t_s) \end{pmatrix} = \begin{pmatrix} \hat{x}_1(1) & \hat{x}_1(2) & \cdots & \hat{x}_1(T_s) \\ \hat{x}_2(1) & \ddots & & \vdots \\ \vdots & & & \\ \hat{x}_n(1) & \cdots & & \hat{x}_n(T_s) \end{pmatrix}\end{aligned} \tag{2.42}$$

where T_s is the number of windows.

Briefly we can say that, the source signals are mixed with different mixing matrices at each frequency bin, that's why ICA is applied separately at each frequency bin. But this brings up a new problem with it which are the scaling and permutation ambiguities of ICA.

In basic ICA problem (2.5), the scaling or permutation problem generally does not affect the performance of the algorithm. the independent components could be normalized to cover scaling problem and the order of the independent components is usually not important as long as they are separated.

But in convolutive ICA, these ambiguities become important. Since ICA is applied at each frequency separately, the independent components at each frequency would be scaled and permuted in a random order. It is a challenging work to find the right permutation so that the separated frequency components belong to same source signal. Even when the right permutation is found, each frequency component is scaled with different parameters which causes distortion in the output. The methods for solving scaling and permutation problems in frequency domain are given in sections (4.3) and (4.4), respectively.

3. BEAMFORMING

"Beamforming" is a general signal processing technique used to control the directionality of the reception or transmission of a signal on a transducer array. A receiving type beamformer performs spatial filtering to separate signals that have overlapping frequency content but originate from different spatial locations.

Beamforming applies sampling spatially and takes the advantage of the fact that the distance from the source to each microphone in the array is different, which means that the signal recorded by each microphone will be phase-shifted replicas of each other.

Both time domain and frequency domain approaches exist for beamforming. In time domain beamforming, the spatial filter passes waves propagating from a narrow range of directions or locations by delaying and summing the sensor signals. Frequency domain beamforming implements the calculations entirely in the frequency domain by Fourier Transforming the inputs. The spatiotemporal filter applied and the results are inverse transformed into the time domain. Delay means linear phase shift in frequency domain.

In both approaches the basic assumptions that are used in calculations are,

- The source signals are far enough from the sensor array so that the waves are planar when they reach to the sensor array.
- Direction of propagation of the wave is approximately equal at each sensor.

Beamformers can be classified in two types, data independent(nonadaptive) and statistically optimum(adaptive) according to the choice of weights.

3.1. Conventional Beamforming

Conventional Beamforming is one the data independent methods in beamforming theory. Data independent means that the beamformer weights are designed so that they don't depend on the statistical behaviour of the source signals. Since we are considering the frequency domain beamforming, the beamforming coefficients are calculated for each frequency bin separately.

In conventional beamforming the beamformer coefficients which is a weight vector, $\mathbf{w}$, are chosen as,

$$\mathbf{w}(f,\theta) = [1 \; e^{j2\pi f\tau_2(\theta)} \; e^{j2\pi f\tau_3(\theta)} \; ... \; e^{j2\pi f\tau_N(\theta)}]^H \tag{3.1}$$

$$\tau_i = (i-1)(d/c)\sin\theta \quad \text{for } i = 1, ..., N$$

where d is the sensor spacing, c is the propagation velocity, θ is the arrival angle which is perpendicular to the sensor array.

Actually each of the elements of weight vector ,$\mathbf{w}$, represents the phase shift which is a time delay in time domain, between consecutive sensors with spacing d. It is represented in figure 3.1,

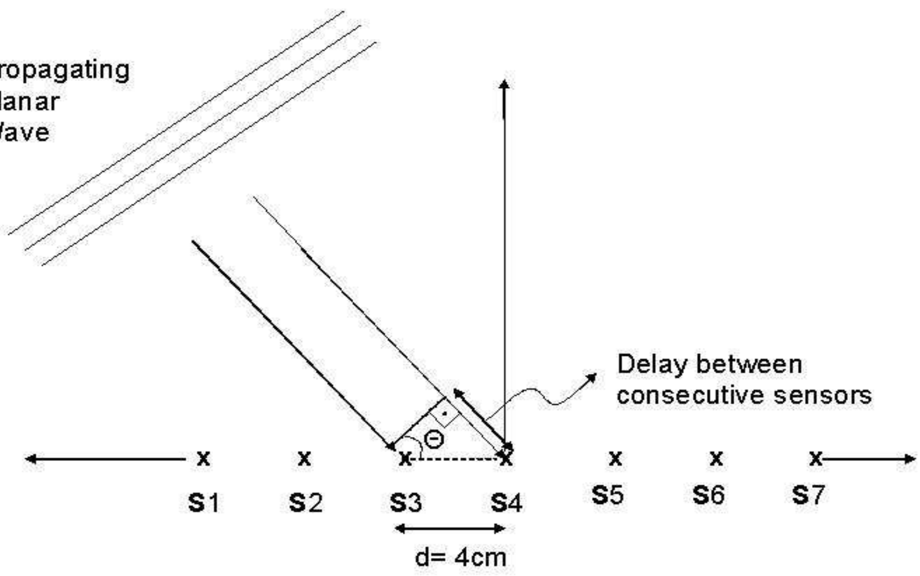


Figure 3.1. The delay between consecutive sensors

As it is seen from the figure 3.1, the actual delay τ_{full} between consecutive sensors is calculated as,

$$\tau_{full} = (d/c)\sin\theta \tag{3.2}$$

3.2. Adaptive Beamfoming

Adaptive beamforming means that the weights are chosen based on the statistics of the data received at the sensors. There are several methods for adaptive beamforming such as Multiple Sidelobe Cancellor, Maximization of Signal to Noise Ratio or Linearly Constrained Minimum Variance Beamforming (LCMV) [14].

The basic idea of LCMV beamforming is to constrain the response of the beamformer so signals from the direction of interest are passed with a specified gain and phase. The weights are chosen to minimize the output power which preserves the desired signal while minimizing contributions to the output due to interfering signals and noise arriving from directions other than the direction of interest.

The response of a beamformer at angle θ and at frequency f is,

$$r(f) = \mathbf{w}^H \mathbf{d}(\theta, f) \tag{3.3}$$

where $\mathbf{d}(\theta, f) = (1, e^{j2\pi f\tau_2(\theta)}, ..., e^{j2\pi f\tau_N(\theta)})^H$. By linearly constraining the weights to satisfy the response $r(f) = g$, where g is complex number, any signal from angle θ and with frequency f has an output response of g. For the special case where $g = 1$, the gain is unity and the phase is zero. this special case is called Minimum variance distortionless response (MVDR). This algorithm is used in our experiments.

Interference of jammer signal with desired signal is optimally reduced by minimizing the expected value of the output power which is,

$$\mathrm{E}\{|y^2|\} = \mathbf{w}^H \mathbf{R_x} \mathbf{w} \;\; \text{where } \mathbf{R_x} = \mathrm{E}\{\mathbf{x}\mathbf{x}^H\} \tag{3.4}$$

So the MVDR problem can be described as,

$$\mathrm{argmin}_{\mathbf{w}}(\mathbf{w}^H \mathbf{R_x} \mathbf{w}) \;\; \text{subject to } \mathbf{w}^H \mathbf{d}(\theta, f) = 1 \tag{3.5}$$

This minimization problem can be solved with Lagrange Multipliers [14], which results in,

$$\mathbf{w} = \frac{\mathbf{R_x}^{-1}\mathbf{d}(\theta, f)}{\mathbf{d}(\theta, f)^H \mathbf{R_x}^{-1}\mathbf{d}(\theta, f)} \tag{3.6}$$

where $\mathbf{w}$ is the beamforming vector.

We used diagonal loading in order to ensure that the $\mathbf{R_x}$ to be invertible.

3.3. Limitations of Beamforming

A beamformer's resolution represents its capability to measure the source's location or its direction accurately so limited resolution means limited ability to determine a plane wave's direction of propagation. Two important kinds of limitations of beamforming are spatial aliasing and angular resolution.

3.3.1. Spatial Aliasing

As we mentioned before, beamforming is sampling the waves in space with array of sensors. The sampling interval is the sensor spacing d. According to Nyquist criteria in the sampling theorem, there should be at least two samples taken from one period of the signal. That means the sampling interval should be smaller than the half of the wave length.

This criteria also stands for spatial sampling [15]. The sensor spacing should be smaller than the half of the wave length of a monochromatic wave with frequency f_o, which can be expressed as,

$$\begin{aligned} \lambda_o &= \frac{c}{f_o}, \\ d &< \frac{\lambda_o}{2} \end{aligned} \tag{3.7}$$

where c is the propagation velocity and λ_o is the wave length. Intuitively this inequality means that the plane wave have to reach consecutive sensors in at most half period time and beyond that range spatial aliasing occurs.

3.3.2. Angular Resolution

The angular resolution means that beamforming methods can calculate the angle of arrival of the planar waves in a certain range. This range is defined in [15] for equally spaced M sensors with spacing d as,

$$PW = \frac{4\sqrt{3}}{k_0 d \cos\theta \sqrt{M^2 - 1}}, \quad \text{where} \;\; k_0 = \frac{2\pi f}{c} \tag{3.8}$$

where PW stands for parabolic width which is the main lobe width of the frequency response of the array pattern, k_0 is the wave number [15], c is the speed of propagation of sound and θ is the angle of interest.

Angular resolution is very important when we are trying to separate more than one plane waves propagating in slightly different directions. From (3.8) it can be seen that, the angular resolution depends on the frequency or wavelength of the signal, the angle of arrival, sensor spacing and number of sensors. Because angular resolution is inversely proportional to the frequency, at low frequencies the range of estimation of arrival angle is pretty higher. That's why in low frequencies beamforming usually fails to separate sources from different locations. In order to minimize the angular resolution either sensor spacing can be decreased which limits the range of frequencies for spatial aliasing, or the number of sensors can be increased.

4. METHODOLOGY

In our method, the frequency domain ICA is used to separate convolutively mixed signals. The block diagram is shown in figure 4.1. As shown in the figure there are 4 basic steps in the algorithm. In the first step, due to non-stationarity of speech signals over long periods of time, the audio mixture is segmented into smaller blocks which are about 25 ms for which the block assumed to be stationary. This segmentation is applied by using overlap-add method which is a strong tool in speech applications due to its easiness and computational efficiency.

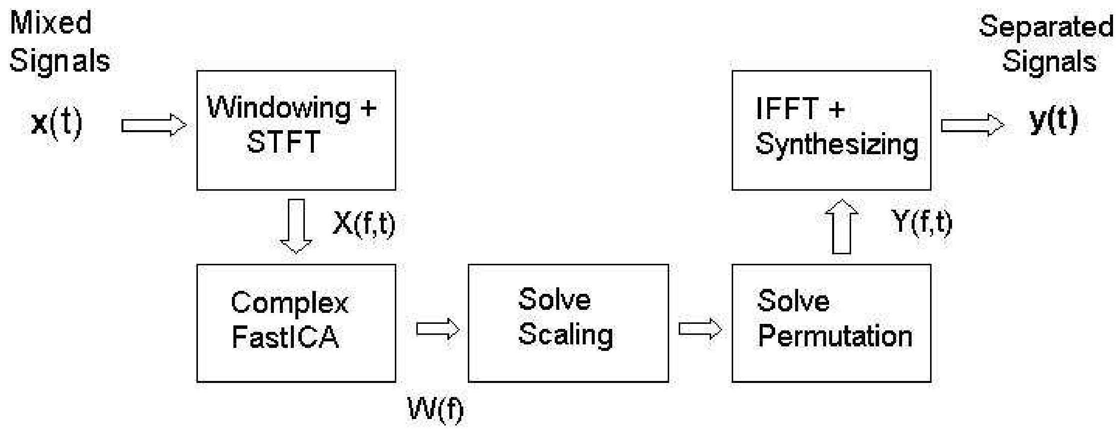


Figure 4.1. Block Diagram of Frequency Domain ICA

4.1. Overlap Add Method

The overlap-add method could be summarized in three steps, decompose the signal into simple components, process each of the components in some useful way, and recombine the processed components into the final signal [17]. This processing would be FFT convolution, filtering in frequency domain etc. In our case this process is finding a linear transformation to obtain independent components in frequency domain.

Windowing functions such as Hanning or Hamming windows, are used in segmentation in order to minimize distortion in frequency domain. We used Hanning window with a length of 25 ms [17]. One of the important properties of Hanning window is with half overlap size, it leads perfect reconstruction. The formula of an M point Hanning window is given by,

$$\mathrm{win}(n) = \frac{1}{2}(1 - \cos\frac{2\pi n}{M-1}) \quad \text{where} \quad 0 \leq n \leq M-1 \tag{4.1}$$

Figure 4.2 illustrates the overlapping Hanning window with a length of 400 samples. The overlap is half length.

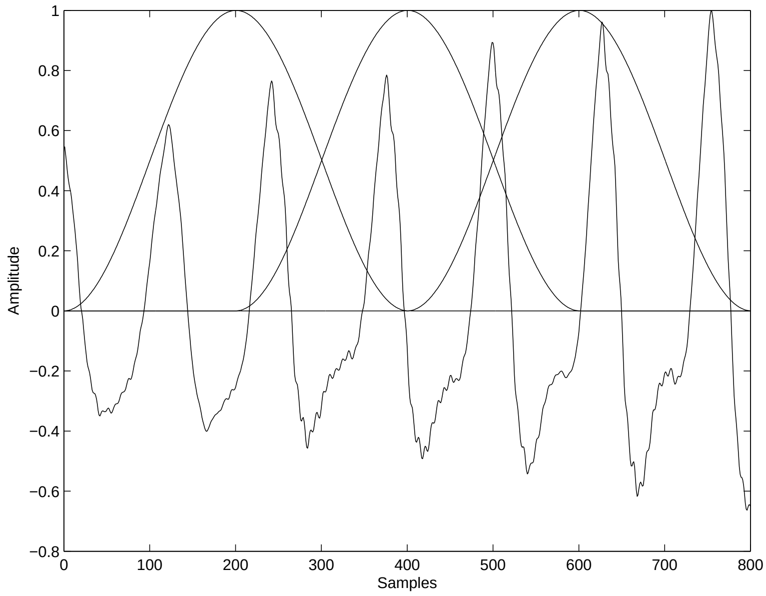


Figure 4.2. Overlapping Hanning window with length=400 samples

There are a lot of versions of Overlap-Add Method [17]. In the one we use, we simply multiply the windowing function(*Hanning window*) with a block of audio mixture by overlapping windows with a skip rate of 12.5 ms. The time information of the blocks are recorded because they are used in the synthesizing part.

$$\hat{x}(t_s) = \sum_t x(t)\mathrm{win}(t - t_s), \quad t_s = 0, \Delta T, 2\Delta T, ... \tag{4.2}$$

After some processing is done over these blocks they are synthesized into a final signal. In our case, we are taking IFFT of these processed signals to transform them into time domain.

Since the time information of the blocks are known from the segmentation step, the resulting blocks are placed according to that information. Since the segmented blocks are overlapping, the resulting blocks also overlap. The overlapping parts of the blocks are added to form the final signal.

After the signals are segmented into blocks and transformed into frequency domain, the data which consists of the time information of each frequency band, is processed in order to have independent components. This separation is done using FastICA algorithm which is modified for complex data.

4.2. Complex FastICA

Methods for separation of convolutively mixed signals in frequency domain involves computations with complex valued signals. That's why FastICA is not appropriate with the version for time domain signals. The FastICA algorithm which is extended to complex valued signals in [19], is described in this section.

It is assumed that the number of independent component variables is the same as the observed linear mixtures, that is n=m. The mixing matrix A is of full rank and it may be complex as well. The preprocessing steps, centering and whitening have to be applied.

A complex random variable may be represented as $y = u + iv$ where u and v are real-valued random variables. The density of y is f(y) = f(u,v). The expectation of y is $E\{\text{y}\} = E\{\text{u}\} + \text{i}\ E\{v\}$. Two complex random variables y_1 and y_2 are uncorrelated if $E\{y_1 y_2^*\} = E\{y_1\}\ E\{y_2^*\}$, where y^* designates the complex conjugate of y. The covariance matrix of a zero-mean complex random vector $\mathbf{y} = (y_1,...,y_n)$ is

$$\mathrm{E}\{\mathbf{y}\mathbf{y}^H\} = \begin{bmatrix} C_{11} & \cdots & C_{1n} \\ \vdots & \ddots & \vdots \\ C_{n1} & \cdots & C_{nn} \end{bmatrix} \tag{4.3}$$

where $C_{jk} = E\{y_j y_k^*\}$ and $\mathbf{y}^H$ stands for the Hermitian of $\mathbf{y}$, that is, $\mathbf{y}$ transposed and conjugated. In this model, all source signals s_j are zero-mean and they have unit variances and uncorrelated real and imaginary parts of equal variances. In general these assumptions imply that s_j must be strictly complex, that is, the imaginary part of s_j may not in general vanish.

The contrast function is chosen as,

$$J_G(\mathbf{w}) = \mathrm{E}\{G(|\mathbf{w}^H\mathbf{x}|^2)\} \tag{4.4}$$

where $G : R^+ \cup \{0\} \to \mathrm{R}$ is a smooth even function, w is an n-dimensional complex weight vector and $E\{|\mathbf{w}^H\mathbf{x}|^2\} = 1$. Finding the extrema of a contrast function is a well defined problem only if the function is real, that's why the contrast function operates on absolute values rather than on complex values.

Maximizing the sum of n one-unit contrast functions, and taking into account the constraint of decorrelation, the optimization problem becomes,

$$\text{maximize} \sum_{j=1}^{n} J_G(\mathbf{w}_j) \text{ with respect to } \mathbf{w}_j, j = 1, ..., n \tag{4.5}$$

$$\text{under the constraint } \mathrm{E}\{(\mathbf{w}_k^H\mathbf{x})(\mathbf{w}_j^H\mathbf{x})^*\} = \delta_{jk} \tag{4.6}$$

where $\delta_{jk} = 1$ for $j = k$ and $\delta_{jk} = 0$ otherwise.

Robustness of the contrast function against outliers is a very important issue. The more slowly G grows as its argument increases, the more robust is the estimator. The choice of G determines the robustness of the estimator. Three different functions are proposed in [19],

$$G_1(y) = \sqrt{a_1 + y}, \quad g_1(y) = \frac{1}{2\sqrt{a_1 + y}} \tag{4.7}$$

$$G_2(y) = \log(a_2 + y), \quad g_2(y) = \frac{1}{a_2 + y} \tag{4.8}$$

$$G_3(y) = \frac{1}{2}y^2, \quad g_3(y) = y \tag{4.9}$$

where a_1 and a_2 are arbitrary constants which are chosen to be $a_1 \approx 0.1$ and $a_2 \approx 0.1$. $G_1(y)$ and $G_2(y)$ give more robust estimators because they grow more slowly then $G_3(y)$. When $G_3(y)$ is used as nonlinear function then the estimator uses kurtosis as a measure of non-gaussianity.

The fixed-point algorithm for one unit is

$$\mathbf{w}^+ = E\{x(\mathbf{w}^H\mathbf{x})^* g(|\mathbf{w}^H\mathbf{x}|^2)\} - E\{g(|\mathbf{w}^H\mathbf{x}|^2 g'(|\mathbf{w}^H\mathbf{x}|^2))\}\mathbf{w} \tag{4.10}$$

$$\mathbf{w}_{new} = \frac{\mathbf{w}^+}{\|\mathbf{w}\|} \tag{4.11}$$

This one unit estimation could be used to estimate all independent components so the whole transformation matrix could be estimated, $\mathbf{s} = \mathbf{W^H x}$. Symmetric orthogonalization is used to decorrelate the separated signals.

4.3. Solving Scaling Ambiguity

The ambiguities of ICA arises as an important problem with this method. Since ICA is applied in each frequency band separately, the resulting demixing matrices have different scalings and permutations. For the scaling problem there are two solutions,

4.3.1. Method 1

In this method, which is described in [7], the decomposition of spectrograms is performed by

$$\hat{\mathbf{v}}_{f_o}(t_s;i) = \mathbf{W}(f_o)^{-1}\mathbf{E}_i\mathbf{W}(f_o)\mathbf{X}(f_o,t_s) = \mathbf{W}(f_o)^{-1}\begin{pmatrix} 0 \\ . \\ . \\ \hat{s}_i(f_o,t_s) \\ . \\ . \\ 0 \end{pmatrix} \tag{4.12}$$

where $\hat{s}_i(f_o,t_s)$ denotes the i'th element (independent component), in the f_o frequency band and at time t_s. $\mathbf{E}_i$ is a diagonal matrix with the i'th diagonal element is one and the rest is zero.

Intuitively, applying $\mathbf{W}(f_o)$ and $\mathbf{W}(f_o)^{-1}$ together, the resulting signal, $\hat{\mathbf{v}}_{f_o}(t_s;i)$, don't have a scaling ambiguity. The elements of the $\hat{\mathbf{v}}_{f_o}(t_s;i)$ are the responses of,

- i'th independent component in the 1'st sensor
- i'th independent component in the 2'nd sensor
- ..
- i'th independent component in the n'th sensor

4.3.2. Method 2

The second method is motivated by Minimum Distortion Principle (MDP) [26]. The goal in this method is to obtain a filtered version of a source $s_{\Pi(i)}(t)$ at each output

$$y_i(t) = \sum_l \alpha_i(l)s_{\Pi(i)}(t-l) \tag{4.13}$$

where $\alpha_i(l)$ is a filter and $\Pi(i)$ is permutation which represent the scaling and permutation ambiguity of ICA respectively. For the scaling ambiguity it is desired to have a delayed version of source signals, not a filtered version. However it is not possible with ICA algorithm unless the source signals $s_i(t)$ are white, which is not the case for speech signals [9]. That's why the filter representing scaling is chosen as

$$\alpha_i(l) = h_{ii}(l) \tag{4.14}$$

by using the MDP [26].

Intuitively, since the impulse responses of the sensors, $h_i(t)$ are unknown, there is no possible way to have more information about source signals, the top goal should the responses of source signals on sensors.

By using (4.14), in frequency domain it can be written,

$$\mathbf{\Lambda}(f)\mathbf{W}(f)\mathbf{H}(f) = \text{diag}[\mathbf{H}(f)] \tag{4.15}$$

where $\mathbf{\Lambda}(f)$ is a diagonal matrix that should satisfy (4.15). When ICA is successfully solved, there should be another diagonal matrix $\mathbf{D}(f)$ that satisfies $\mathbf{W}(f)\mathbf{H}(f) = \mathbf{D}(f)$. So $\mathbf{H}(f)$ can be estimated by $\mathbf{H}(f) = \mathbf{W}^{-1}(f)\mathbf{D}(f)$. By replacing this in the righthand side of (4.15), we get

$$\begin{aligned}
\mathbf{\Lambda}(f)\mathbf{W}(f)\mathbf{H}(f) &= \text{diag}[\mathbf{W}^{-1}(f)\mathbf{D}(f)] \\
&= \text{diag}[\mathbf{W}^{-1}(f)]\mathbf{D}(f) \\
&= \text{diag}[\mathbf{W}^{-1}(f)]\mathbf{W}(f)\mathbf{H}(f)
\end{aligned}$$

Hence we have,

$$\mathbf{\Lambda}(f) = \text{diag}[\mathbf{W}^{-1}(f)] \tag{4.16}$$

Therefore the scaling ambiguity is solved by,

$$\mathbf{W}(f) \leftarrow \mathrm{diag}[\mathbf{W}^{-1}(f)]\mathbf{W}(f) \tag{4.17}$$

4.4. Solving Permutation Ambiguity

Permutation ambiguity is the most important problem when dealing with ICA in frequency domain. Basically the permutation problem is to align each frequency bin so that a separated signal in the time-domain contains frequency components of the same source signal. Direction of Arrival (DOA) approach and interfrequency correlations of signals approach are two basic approaches for this problem. There are also some methods that use both approaches to have a more robust and precise technique [9].

4.4.1. Direction of Arrival (DOA) Approach

DOA approach is motivated by conventional beamforming [14]. In this method, the directions of source signals are estimated and permutations are aligned based on them [9],[10]. Each row of the demixing matrix $\mathbf{W}(f)$ forms spatial nulls in the directions of jammer signals and extracts a target signal in another direction. When the directions are estimated, $\mathbf{\Theta}(f) = [\theta_1(f), ..., \theta_N(f)]^T$ of target signals extracted by every row of $\mathbf{W}(f)$, the permutation matrix $\mathbf{P}(f)$ can be obtained by sorting the angle of directions or clustering $\mathbf{\Theta}(f)$. There are two ways to calculate the direction angle.

4.4.2. Calculating Directivity Patterns

In this method, directions of source signals are found by plotting the directivity pattern of each output $Y_i(f,t)$. In conventional beamforming [14], the frequency response of an impulse response $h_{jk}(t)$ is approximated as

$$H_{jk}(f) = e^{j2\pi f c^{-1} d_j \cos\theta_k} \tag{4.18}$$

where c is the propagation velocity, d_j is the position of sensor x_j and θ_k is the direction of source s_k (the direction orthogonal to the array is 90°). The assumptions for that approximation are that there is no reverberation and the signal is a plane wavefront just as in beamforming theory [15],[14]. The frequency response of separation system impulse response can be written as

$$\begin{aligned} \mathbf{U}(f) &= \mathbf{W}(f)\mathbf{H}(f) && (4.19) \\ U_{ik}(f) &= \sum_{j=1}^{M} W_{ij}(f) H_{jk}(f) && (4.20) \\ &= \sum_{j=1}^{M} W_{ij}(f) e^{j2\pi f c^{-1} d_j \cos\theta_k} && (4.21) \end{aligned}$$

If we use θ_k as a variable θ, the formula becomes,

$$U_i(f,\theta) = \sum_{j=1}^{M} W_{ij}(f) e^{j2\pi f c^{-1} d_j \cos\theta} \tag{4.22}$$

which changes according to the direction θ for fixed frequency band. This is called a directivity pattern.

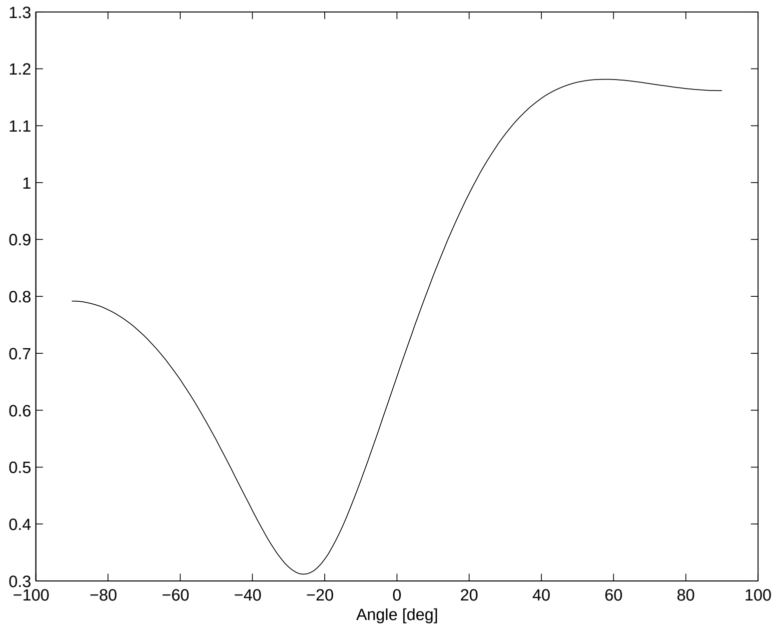


Figure 4.3. Directivity Pattern $U_i(f,\theta)$ for f=3320 Hz with one source approaching from -25^o

4.4.3. Calculating Angle of Arrival directly

The direct calculation of angle of arrival is motivated by the approximation of frequency response of an impulse response (4.18).

From (2.23), it can easily be written that $\mathbf{H} = \mathbf{W}^{-1}\mathbf{P}^{-1}\mathbf{\Lambda}^{-1}$ which means the mixing matrix $\mathbf{H}(f)$ can be estimated up to scaling and permutation ambiguities that is $\mathbf{H}(f)$ columns can be permuted arbitrarily and have arbitrary scaling factors. So an element $H_{jk}(f)$ of the matrix $\mathbf{H}(f)$ may have an arbitrary amplitude which also does not match with the approximation of the mixing system (4.18). That's why the approximation (4.18) should be modified with attenuation A_{jk} (real-valued) and phase modulation $e^{j\varphi_k}$,

$$H_{jk}(f) = A_{jk}e^{j\varphi_k}e^{j2\pi fc^{-1}d_j \cos\theta_k} \tag{4.23}$$

By calculating the ratio between two elements $H_{jk}(f)$ and $H_{j'k}(f)$ which are the elements of the same column of $\mathbf{H}(f)$,

$$\frac{H_{jk}}{H_{j'k}} = \frac{[\mathbf{W}^{-1}\mathbf{P}^{-1}\mathbf{\Lambda}^{-1}]_{jk}}{[\mathbf{W}^{-1}\mathbf{P}^{-1}\mathbf{\Lambda}^{-1}]_{j'k}} = \frac{[\mathbf{W}^{-1}]_{j\Pi(k)}}{[\mathbf{W}^{-1}]_{j'\Pi(k)}} \tag{4.24}$$

From using (4.23) and (4.24) we have,

$$\frac{[\mathbf{W}^{-1}]_{j\Pi(k)}}{[\mathbf{W}^{-1}]_{j'\Pi(k)}} = \frac{A_{jk}}{A_{j'k}e^{j2\pi fc^{-1}(d_j - d_{j'})\cos\theta_k}} \tag{4.25}$$

Then for estimating θ_k arranging with taking the argument of (4.25) leads to,

$$\theta_k = \arccos \frac{\arg(\frac{[\mathbf{W}^{-1}]_{j\Pi(k)}}{[\mathbf{W}^{-1}]_{j'\Pi(k)}})}{2\pi fc^{-1}(d_j - d_{j'})} \tag{4.26}$$

By calculating θ_k for all $k = 1, ..., N$, we can estimate the directions of all source signals and sort them in the way we want to solve permutation problem. This method offers an advantage in computational cost.

4.4.4. The Limitations of DOA approach

The basic limitations in finding permutations with directivity patterns are,

- Direction of arrival can not be well estimated at some frequencies. At high frequencies, spatial aliasing occurs and at low frequencies the phase difference caused by sensor spacing is too small. These are also the basic limitations in beamforming applications.
- When there are more than two sources/sensors, it is difficult to estimate the direction of arrivals.
- For online methods, calculation of null directions from directivity patterns is time consuming.

When the angle of arrival is directly calculated without using directivity pattern but (4.26), sometimes the absolute value of the input of arccos becomes larger than 1, which means θ_k becomes complex and no directions can be estimated. When this happens the formula could be used for another pair j and j'. But for 2×2 case, there is no other possibility to estimate when θ_k becomes complex.

4.5. Correlation Approach

This approach is motivated by the non-stationary property of the speech signals. Based on this property, we may assume that components at different frequencies from the same source signals are under the influence of a similar modulation in amplitude [7]. That is, writing the estimated independent component $\hat{s}_i(w, t_s)$ with amplitude and phase,

$$\hat{s}_i(f, t_s) = a_i(f, t_s)e^{j\phi_i(f,t_s)} \tag{4.27}$$

The amplitude, $a_i(f, t_s)$, changes in time because of non-stationarity. Since $\hat{s}_i(f, t_s)$ and $\hat{s}_j(f, t_s)$ are independent from each other, the correlation of amplitudes which is defined as,

$$\begin{aligned}\mathrm{corr}(a_i(f,t_s),a_j(f,t_s)) &= \frac{1}{T}\sum_{s=1}^{T} a_i(f,t_s)a_j(f,t_s) \\ &\quad -\frac{1}{T}\sum_{s=1}^{T} a_i(f,t_s)\frac{1}{T}\sum_{s=1}^{T} a_j(f,t_s) \\ &= 0,\ i \neq j\end{aligned} \tag{4.28}$$

should vanish if T is sufficiently large. Also the correlation between different frequency components form different source signals should vanish.

$$\mathrm{corr}(a_i(f,t_s),a_j(f',t_s)) = 0,\ i \neq j,\ f \neq f' \tag{4.29}$$

But the correlation between different frequency components from the same source signal should not equal to zero.

$$\mathrm{corr}(a_i(f,t_s),a_i(f',t_s)) \neq 0 \tag{4.30}$$

First step in correlation approach is applying a moving average operator, a smoothing operator, on the envelopes of the separated signals $\hat{s}_i(f,t_s)$. The moving average operator is defined as,

$$\varepsilon\hat{s}_i(f,t_s) = \frac{1}{2M+1}\sum_{t_s'=t_s-M}^{t_s+M} |\hat{s}_i(f,t_s')| \tag{4.31}$$

where M parameter is the filter length. This operator is basically a mean filter which is usually used as a preprocessing before calculation of correlations.

The inner product and norm are defined as,

$$\varepsilon\hat{s}_i(f)\cdot\varepsilon\hat{s}_j(f') = \sum_{t_s}\varepsilon\hat{s}_i(f,t_s)\varepsilon\hat{s}_j(f',t_s), \tag{4.32}$$

$$\|\varepsilon\hat{s}_i(f)\| = \sqrt{\varepsilon\hat{s}_i(f)\cdot\varepsilon\hat{s}_i(f)} \tag{4.33}$$

Similarity of two envelopes is calculated with a similarity measure between i'th independent components in f frequency band and and j'th independent component in f' frequency band,

$$\mathrm{sim}(\varepsilon\hat{s}_i(f), \varepsilon\hat{s}_j(f')) = \sum_{i \neq j} \frac{\varepsilon\hat{s}_i(f) \cdot \varepsilon\hat{s}_j(f')}{\|\varepsilon\hat{s}_i(f)\| \|\varepsilon\hat{s}_j(f')\|} \tag{4.34}$$

Shortly, we align permutations based on interfrequency correlations of the signals. There are two versions of correlation approach.

4.5.1. Neighboring Correlations

This method decides the permutation, Π_f by maximizing the sum of the similarities between neighboring frequencies within a distance δ. There are some versions of this algorithm, in [9], the permutation is calculated with the formula,

$$\Pi_f = \mathrm{argmax}_{\Pi} \sum_{|g-f| \leq \delta} \sum_{i=1}^{N} \mathrm{sim}(\varepsilon\hat{s}_{\Pi_f(i)}(f), \varepsilon\hat{s}_{\Pi_g(i)}(g)) \tag{4.35}$$

where Π_f and Π_g are permutations at frequency bin f and g respectively. This algorithm works with all neighboring frequencies, which may cause misalignments in the permutation. In [9], this kind of correlation approach is used after DOA estimation which is used for pre-sorting of frequencies. Another approach may be,

$$\Pi_f = \mathrm{argmax}_{\Pi} \sum_{(f-g) \leq \delta} \sum_{i=1}^{N} \mathrm{sim}(\varepsilon\hat{s}_{\Pi_f(i)}(f), \varepsilon\hat{s}_{\Pi_g(i)}(g)), \quad f > g \tag{4.36}$$

which uses only past frequencies. When the algorithm works from lower frequencies to higher ones, the past frequencies would be sorted when dealing with a frequency bin f. That's why, this kind of correlation approach would be more successful without pre-sorting. Also by changing the distance parameter, δ to all past frequencies, the algorithm may become more robust.

4.5.2. Correlations between far frequencies based on nonstationarity

This method assumes high correlations of envelopes even between frequencies that are not close neighbors which comes from the fact that each frequency component of a source signal has a similar modulation in amplitude.

The steps in this method,

- Sort frequency bins in order of weakness of similarity between estimated independent components in a frequency bin ,f. The similarity measure in (4.34) can be modified as,

$$\mathrm{sim}(f) = \sum_{i \neq j} \frac{\varepsilon \hat{s}_f(i) \cdot \varepsilon \hat{s}_f(j)}{\|\varepsilon \hat{s}_f(i)\| \|\varepsilon \hat{s}_f(j)\|} \tag{4.37}$$

  so that it calculates the similarity between independent components in one frequency bin, f.

$$\mathrm{sim}(f_1) \leq \mathrm{sim}(f_2) \leq ... \leq \mathrm{sim}(f_N) \tag{4.38}$$

- For frequency bin f_1, the signals $\hat{s}_{f_1}(i)$ are assigned to $\hat{y}_{f_1}(i)$ with the same order,

$$\hat{y}_{f_1}(i) = \hat{s}_{f_1}(i),\ i = 1, ..., n \tag{4.39}$$

- After assigning the first frequency bin, for each f_k frequency bin, find an appropriate permutation ,$\Pi_{f_k}(i)$, which maximizes the similarity between the envelope of f_k and the sum of the envelopes from f_1 through f_{k-1}, that is,

$$\mathrm{argmax}_{\Pi} \sum_{i=1}^{n} \mathrm{sim} \left(\varepsilon \hat{s}_{f_k}(\Pi_{f_k}(i)), \sum_{j=1}^{k-1} \varepsilon \hat{y}_{f_j}(i) \right) \tag{4.40}$$

  which searches all possible permutations that maximizes the similarity.

- When the correct permutation is found, assign it to $\hat{y}_{f_k}(i)$,

$$\hat{y}_{f_k}(i) = \hat{s}_{f_k,\Pi_{f_k}}(i),\ i = 1, ..., n \tag{4.41}$$

- Apply the algorithm for all frequency bins from which has least similar independent components to the highest one.

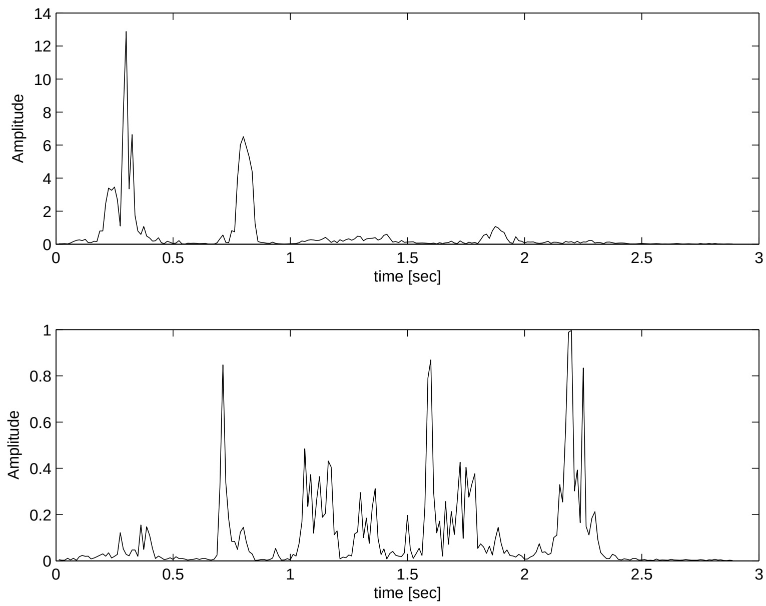


Figure 4.4. The spectrogram of two different frequency bins from same source signal $f_1 = 781$ and $f_2 = 3437$

4.5.3. Limitations of Correlation Approach

In both versions of correlations approach, the algorithms work in a narrow band frequency range but when there is a misalignment in a certain frequency range, it may lead to a complete misalignment of frequencies since the algorithms are using the similarity of independent components between frequency bins. So with a few unseparated frequency bins which are failed by ICA, the correlation may fail easily, which makes the ICA performance extremely important.

Also the assumption that each frequency component of a source signal has a similar modulation in amplitude which leads to have high correlation in between is not satisfied for all pairs of frequencies which is shown in figure 4.4.

4.5.4. A Hybrid Method that uses both DOA and interfrequency correlations approaches

Neither DOA nor correlation approaches may not form a robust system alone to solve the permutation problem. In [9], a new method is presented which uses both of the approaches in a hybrid way.

The DOA approach is robust in a way that a misalignment in a certain frequency does not affect the rest of the frequency bins. But it may not be precise since the approximations (4.18) and (4.23) are based on certain assumptions like there is no reverberation which is our case.

The success of correlations approach is based on successfully estimated independent components so this approach can be precise as long as the signals are well separated by ICA. But not robust since a misalignment at a frequency may cause consecutive misalignments.

This method is using the correlations in two ways, neighboring correlations and Harmonic Structure of Signals.

4.5.4.1. Harmonic Structure of Signals. This method is motivated by the second method in correlations approach [7]. The basic assumption of this method may not be satisfied for all frequency bins but there are strong correlations (similarities) between the envelopes of a fundamental frequency f and its harmonics $2f$,$3f$ and so forth. The similarity between frequency f and its harmonics is calculated as,

$$\mathrm{argmax}_{\Pi} \sum_{g=setOfHarmonics(f)} \sum_{i=1}^{N} \mathrm{sim}\left(\varepsilon\hat{s}_{\Pi(i)}(f), \varepsilon\hat{s}_{\Pi_g(i)}(g)\right) \tag{4.42}$$

where setOfHarmonics(f) is a procedure which produces the harmonics of f.

By using DOA, neighboring correlations and harmonic structure, this method aligns the permutations in 4 steps:

1. Fix the permutations with DOA approach at some frequencies where the confidence is sufficiently high. The confidence measure is that the directions $\Theta_s(f)$ do not differ greatly from the mean directions $\overline{\Theta}_s$,

$$|\Theta_s(f) - \overline{\Theta}_s| < th_\theta \tag{4.43}$$

   where th_θ is the threshold value.

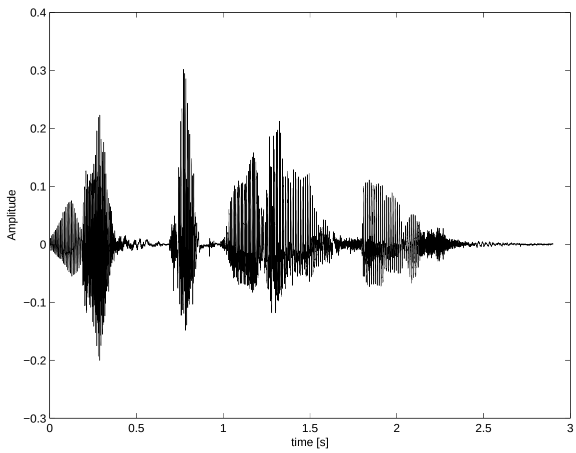
2. Align the permutations by using neighboring correlations without changing the permutations fixed by DOA approach, if the similarity between them is higher than a threshold value, th_{cor}

3. Apply the harmonics method to remaining frequencies which are not fixed either by DOA or by neighboring correlations, align the permutations which have a similarity measure higher than a threshold value, th_{ha}.

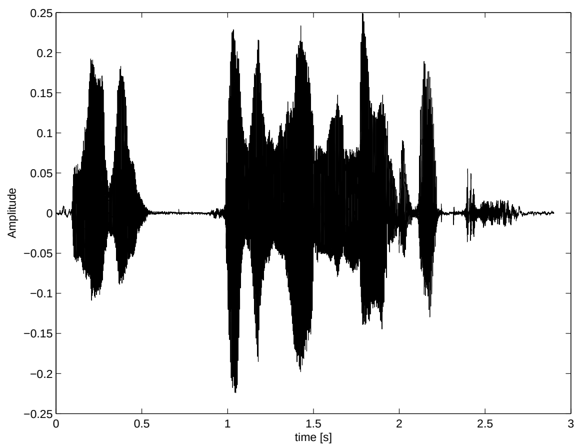
4. Apply neighboring correlations again to remaining frequencies that are not fixed at any step and align them without checking with a threshold value.

5. RESULTS

The experiments are performed to separate speech signals in reverberant conditions. All the experiments are done for 2×2 case. The source signals belong to one male and one female voice which are illustrated in figure 5.1. In the first section the simulation environment such as simulation room is described, then the beamforming results are demonstrated for both conventional and adaptive approaches. After that the results of the time domain ICA method Generic SOS algorithm is shown and in the end frequency domain ICA with different approaches to solve permutation and scaling is given.



(a) Sentence: "Bach çok iyi bir müzisyendir"



(b) Sentence: "Yarın hava yağmurlu olacakmış"

Figure 5.1. Source signals (a) Male voice (b) Female voice

5.1. Simulation Environment

We simulated a room with dimensions 4-4-5 meters which is shown in figure (5.1). The impulse responses are generated by using the image method [16]. In the simulation there are 9 sensors which are uniformly spaced with a sensor spacing of 4cm, the source and the sensor locations are given in table 5.1 The sources are at 27^o and -27^o according to the perpendicular vector to the sensor array. Figure 5.2 shows the simulation room for beamforming experiments.

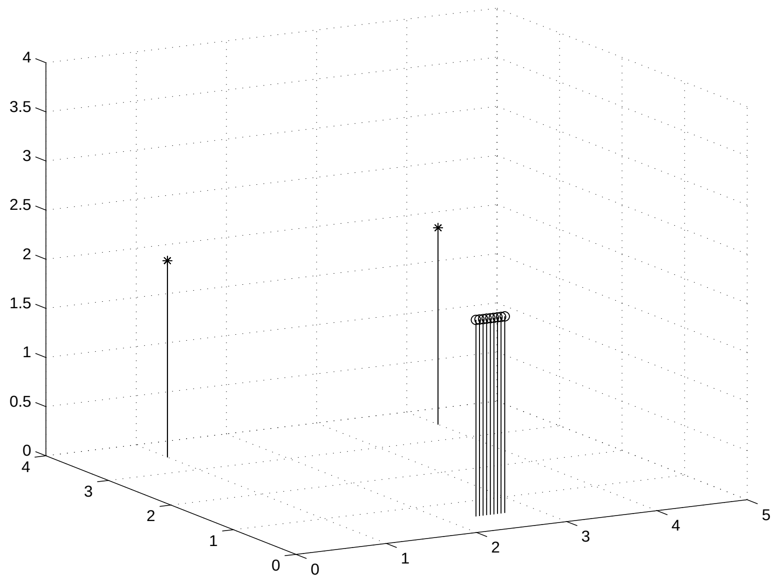


Figure 5.2. Simulation Room for Beamforming

For ICA experiments, the fourth and the fifth sensors are chosen for the 2x2 case which is shown in the figure 5.3. in both figure 5.2 and 5.3 the "$*$" represents the source signal position and the "$\circ$" represents the sensor positions.

The speech signals are recorded at 16kHz and each of them is 4 seconds long. The length of the windows in overlap-add method is chosen as 25ms which leads to 400 samples in 16kHz sampling rate. The length of impulse responses is chosen as 65ms which leads to 1024 samples, they are shown in figure 5.4. The fft size used in calculations is 2048 points.

For the experiments where there is no reverberation but delay, the first peak

Table 5.1. Locations of sensors and sources in the simulation room

	x [mt]	**y [mt]**	**z [mt]**
Source 1	1	3.5	2
Source 2	4	3.5	2

	x [mt]	**y [mt]**	**z [mt]**
Sensor 1	2.34	0.5	2
Sensor 2	2.38	0.5	2
Sensor 3	2.42	0.5	2
Sensor 4	2.46	0.5	2
Sensor 5	2.5	0.5	2
Sensor 6	2.54	0.5	2
Sensor 7	2.58	0.5	2
Sensor 8	2.62	0.5	2
Sensor 9	2.66	0.5	2

sample of the impulse responses are taken and other values are suppressed to be zero.

5.2. Generating Mixed Signals

The mixture signals are generated by two different methods while using simulated impulse responses.

5.2.1. Mixing in Time Domain

This method is simply convolving the speech signals $s_i(t)$ and the impulse responses $h_{ij}(t)$ which is shown in (2.37). Equation (2.37) simplifies to,

$$\begin{pmatrix} x_1(t) \\ x_2(t) \end{pmatrix} = \begin{pmatrix} h_{11}(t) & h_{12}(t) \\ h_{21}(t) & h_{22}(t) \end{pmatrix} * \begin{pmatrix} s_1(t) \\ s_2(t) \end{pmatrix} \tag{5.1}$$

$$\begin{aligned} x_1(t) &= h_{11}(t) * s_1(t) + h_{12}(t) * s_2(t) \\ x_1(t) &= h_{21}(t) * s_1(t) + h_{22}(t) * s_2(t) \end{aligned}$$

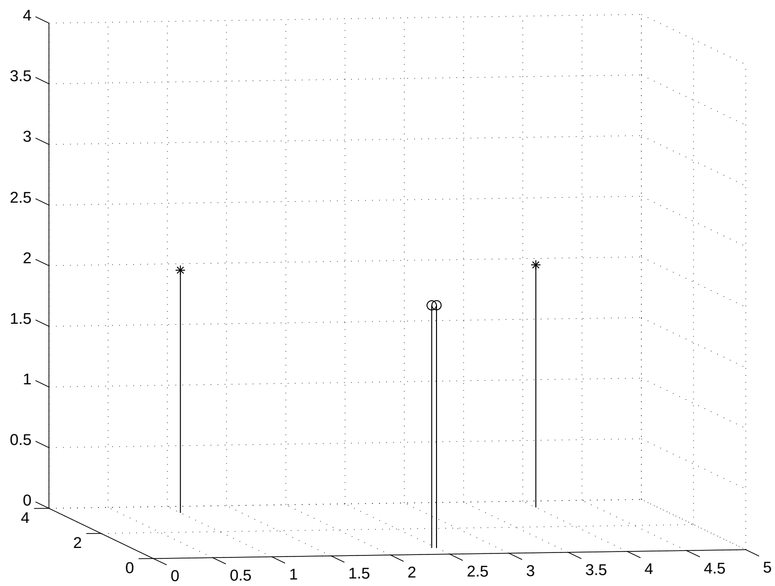


Figure 5.3. Simulation Room for ICA for 2x2 case

5.2.2. Mixing in Frequency Domain

In this method, the source signals and the impulse responses are transformed into fourier domain by STFT. The resulting data which contains time-frequency information of the signals is multiplied and added with the fourier transforms of the impulse responses,

$$\begin{aligned} s_1(t), s_2(t) &\rightarrow \mathbf{STFT} \rightarrow S_1(f,t), S_2(f,t) \\ h_{11}(t), h_{12}(t), h_{21}(t), h_{22}(t) &\rightarrow \mathbf{STFT} \rightarrow H_{11}(f,t), H_{12}(f,t), H_{21}(f,t), H_{22}(f,t) \end{aligned}$$

$$\begin{pmatrix} X_1(f,t) \\ X_2(f,t) \end{pmatrix} = \begin{pmatrix} H_{11}(f,t) & H_{12}(f,t) \\ H_{21}(f,t) & H_{22}(f,t) \end{pmatrix} \begin{pmatrix} S_1(f,t) \\ S_2(f,t) \end{pmatrix} \tag{5.2}$$

The $X_1(f,t)$ and $X_2(f,t)$ are the STFT of the mixed signals which are obtained in frequency domain.

5.2.3. Difference between two methods

In both methods, we obtain the time-frequency signals $X_1(f,t)$ and $X_2(f,t)$. In the first one, we first convolve the signals with impulse responses then take the STFT,

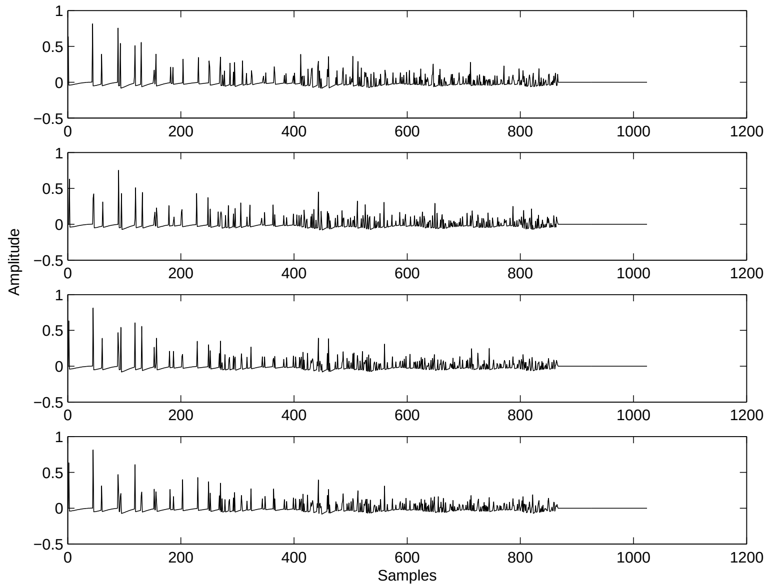


Figure 5.4. Impulse Responses h_{11}, h_{12}, h_{21} and h_{22}

in the second one, we first take the STFT then multiply the impulse responses and signal in frequency domain which also leads to mixed time-frequency signals.

Unfortunately these two methods lead to different signals. The main reason is that the time domain convolution is using circular convolution and the frequency domain method is using linear convolution. In frequency domain ICA methodology, it is assumed that there is linear convolution which causes an ambiguity for real data.

The convolution in time domain is the method that totally describes a real room simulation. The second method is not a true mixing method for real life conditions but it is applied in order to examine the ambiguity that occurs because of circular convolution.

5.3. Beamforming Results

For both approaches of beamforming, the results are shown by plotting the ambiguity functions of separated signals and the sum of frequency energies against the direction of arrival angle which has 5^o resolution.

5.3.1. Conventional Beamforming Results

No reverb Case for conventional beamforming

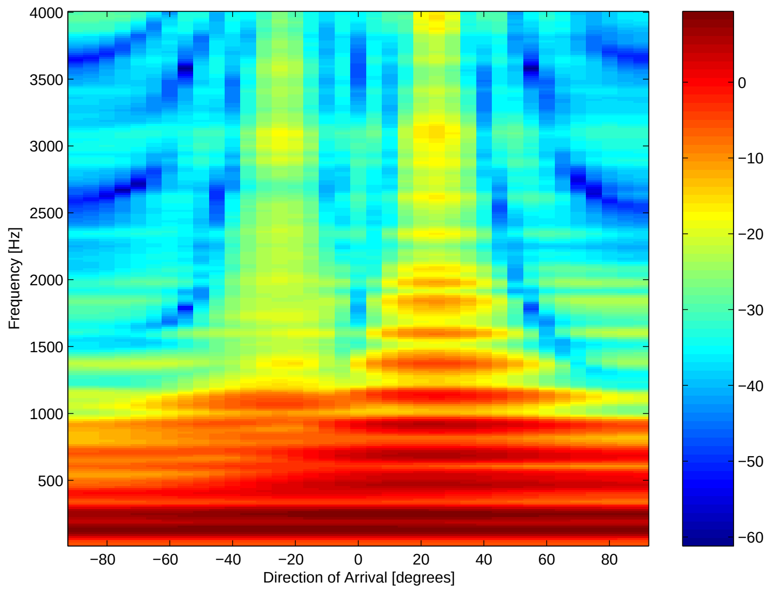


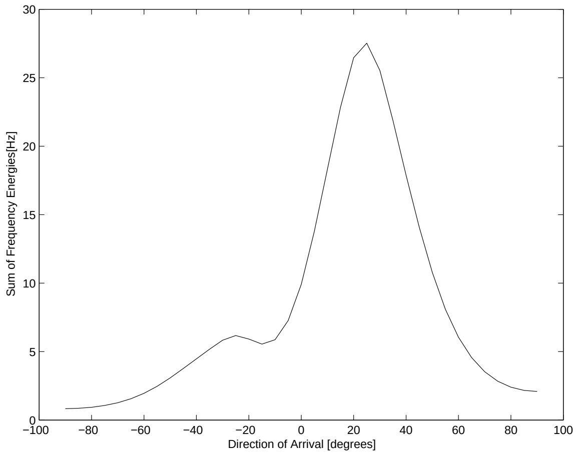
Figure 5.5. Ambiguity function of $\hat{s}_1$ and $\hat{s}_2$ with no reverberation [dB]

When there is no reverberation in the room Conventional Beamforming is able to separate these two signals coming from different locations at around 1kHz range, as it can be seen that there are two maximum energy directions at that range. Due to angular resolution, in the low frequencies below that range, there is no way to separate them because no direction can be estimated.

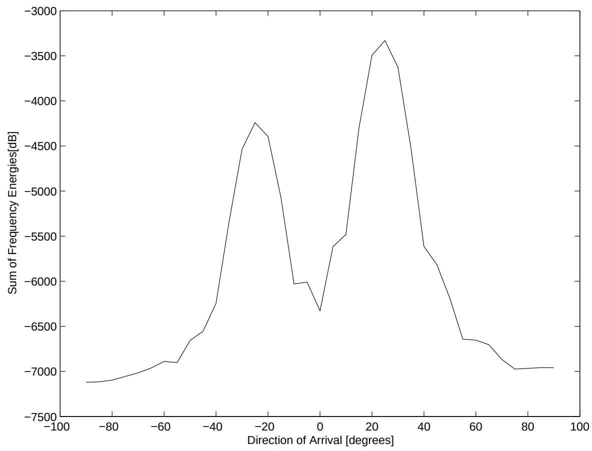
From figure 5.6.a and 5.6.b, it can be seen that, there are two peaks in the plot which corresponds to the locations of sources. Logarithmic sum of frequency energies clarifies the peaks.

Reverb Case for conventional beamforming

The conventional beamforming fails to separate signals in most of the frequency bands in a reverberant condition. A weak separation can be observed around 900Hz, 1100Hz and 1400Hz.



(a)



(b)

Figure 5.6. Sum of energies at each frequency bin according to DOA (a)in [Hz] and (b) in [dB]

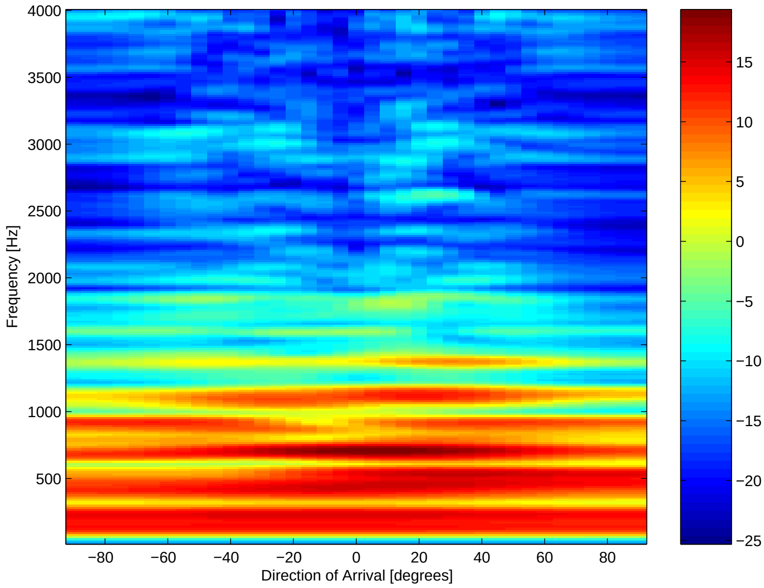


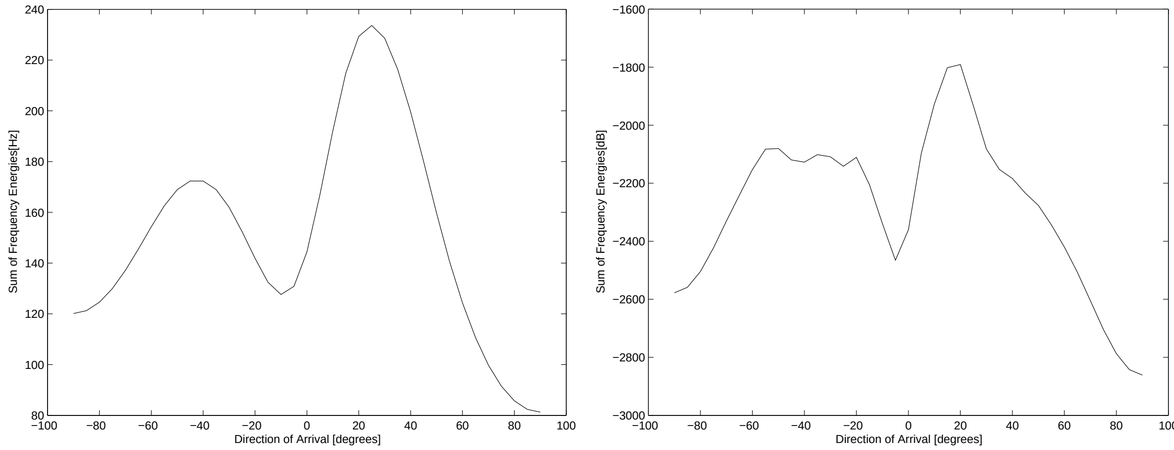
Figure 5.7. Ambiguity function of $\hat{s}_1$ and $\hat{s}_2$ with reverberation [dB]

As it is seen, there are two clear peaks in figures 5.8.a and 5.8.b but the estimated angle is totally wrong for one of the sources.

5.3.2. Adaptive Beamforming Results

No reverb Case for adaptive beamforming

As it can be seen from the figure, there are two clear directions in -25^o and 25^o



(a) (b)

Figure 5.8. Sum of energies at each frequency bin (a)in [Hz] and (b) in [dB]

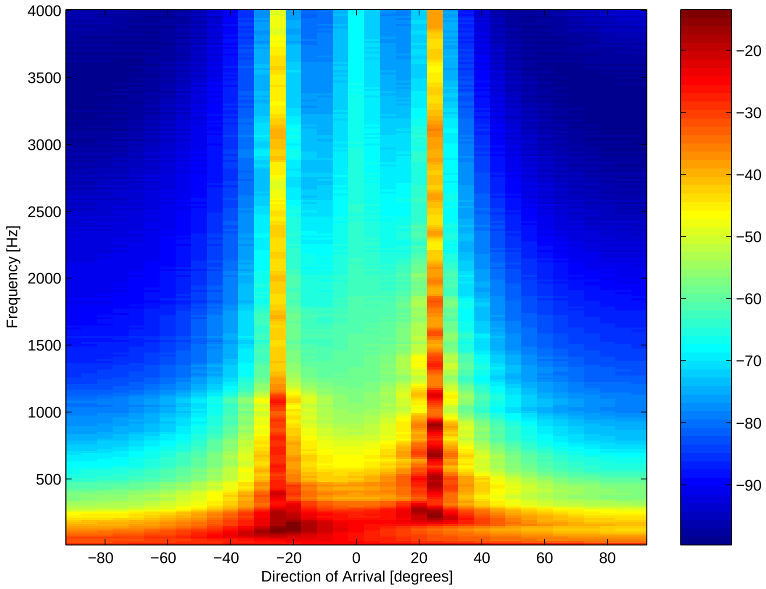
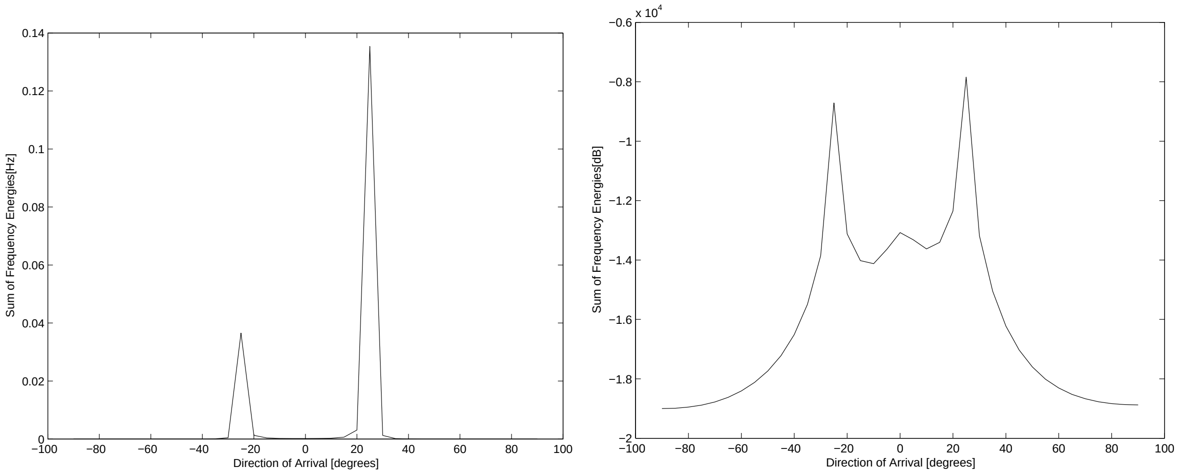


Figure 5.9. Ambiguity function of $\hat{s}_1$ and $\hat{s}_2$ with no reverberation [dB]

which are true angles. But in lower frequencies, below 300 Hz, the separation fails due to the angular resolution.

It can clearly be seen that there are two peaks in the figure 5.10.a which represents the direction of arrival angles for both sources. But although from the sum of logarithms the directions can still be estimated in figure 5.10.b, they are less clear.

Reverb Case for adaptive beamforming



(a) (b)

Figure 5.10. Sum of energies at each frequency bin (a)in [Hz] and (b) in [dB]

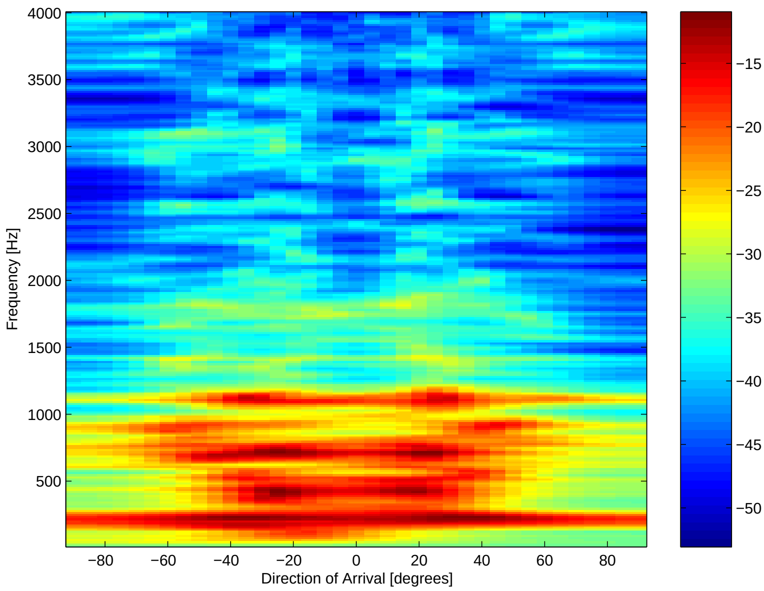
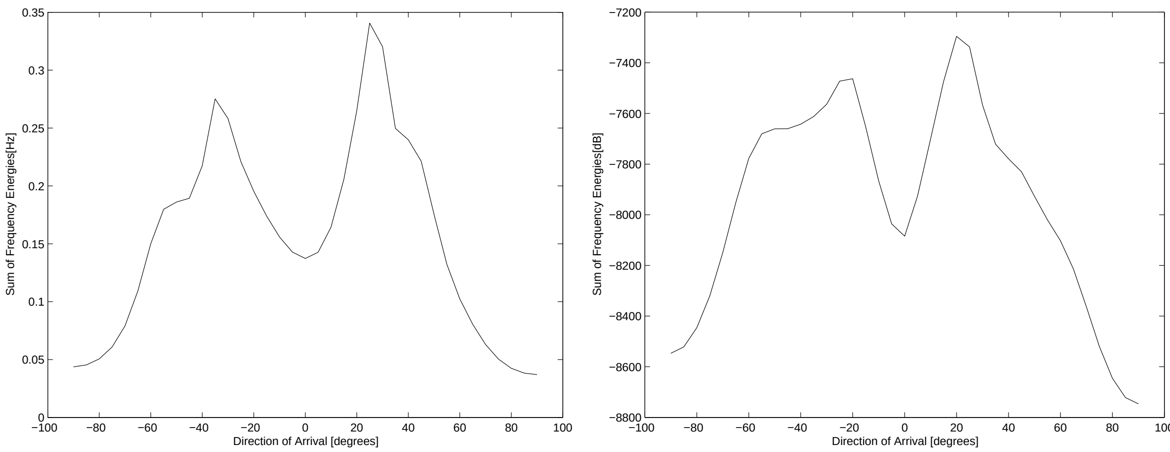


Figure 5.11. Ambiguity function of $\hat{s}_1$ and $\hat{s}_2$ with no reverberation [dB]

From the figure, it can be observed that at some frequencies at around 300 Hz, 500 Hz and 750 Hz, there is more energy for the angles -25^o and 25^o. The results are not clear but well enough to estimate the directions.

It can be seen that one of the directions is truly estimated and other direction miscalculated with around 5^o or 10^o. Summing logarithmic energies also can not clearly estimate the directions actually it is worse then previous case.



(a) (b)

Figure 5.12. Sum of energies at each frequency bin (a)in [Hz] and (b) in [dB]

5.4. ICA Results

5.4.1. Time Domain Method Results (Generic SOS Algorithm)

For the generic SOS algorithm in [12] and [13], it is a parametric algorithm which has 9 parameters to tune. The optimal parameters could not be found for this method therefore the results are not successful.

5.4.2. Frequency Domain ICA Results

Although FastICA algorithm is a robust algorithm for separation even in complex domain, sometimes due to data or the bad mixing matrix it may fail to separate signals. A performance measure is needed to observe the success of separation.

The performance of the algorithm is checked at each frequency bin by calculating the similarity of separated time-frequency signals. If the similarity (4.34) between them is over a predetermined threshold value, then it is decided that ICA failed to separate signals at that frequency bin. The threshold value is chosen as the mean of the calculated similarities.

Sometimes ICA algorithm finds some local minima which is not the desired minimum point that leads to maximum nongaussianity. In that case re-applying ICA algorithm improves the performance. If the stopping point is a global minimum, then re-applying the ICA algorithm won't change the result but if it is a local minimum, then the ICA algorithm may pass this minimum and find the global minimum point. Within the performance check if it is decided that ICA failed, then it is re-applied from the ending point of the first try.

The separation performance for the mixtures generated in time domain with convolution is summarized in table 5.2. The results are given for the environments with no reverberation and vice versa.

Table 5.2. ICA Performance of mixtures generated in time domain

	# of ICA Fails	# of ICA Success	Performance
No Reverb Case	857	1191	%58.1 success
Reverb Case	1285	763	%37.3 success

For the mixtures generated in frequency domain with multiplication, the separation performance is given in table 5.3.

Table 5.3. ICA Performance with mixtures generated in frequency domain

	# of ICA Fails	# of ICA Success	Performance
No Reverb Case	14	2034	%99.3 success
Reverb Case	24	2024	%98.8 success

By applying the FastICA algorithm second time for the failed frequency bins, the separation performance is improved which are shown in table 5.4 and 5.5.

Table 5.4. Improved ICA Performance of mixtures generated in time domain

	# of ICA Fails	# of ICA Success	Performance
No Reverb Case	843	1205	%58.83 success
Reverb Case	1263	785	%38.33 success

Table 5.5. Improved ICA Performance with mixtures generated in frequency domain

	# of ICA Fails	# of ICA Success	Performance
No Reverb Case	2	2046	%99.9 success
Reverb Case	0	2048	%100 success

For mixing type 1, the separation performance is not satisfactory. For no reverberation case, the resulting time domain signals are satisfying in the means of separation of sources. But for reverberation case, it totally fails to separate independent sources. On the other hand, the separation performance of the mixtures generated in frequency domain is quite high in both reverb an non-reverb cases. And it is even improved with re-applying ICA to the failed frequency bins

This case is illustrated for frequency of 195Hz for mixtures generated in frequency domain where there is reverberation. The first figure shows the time-frequency representation of original source signals $S_1(f,t)$ and $S_2(f,t)$ respectively, for frequency 195 Hz.

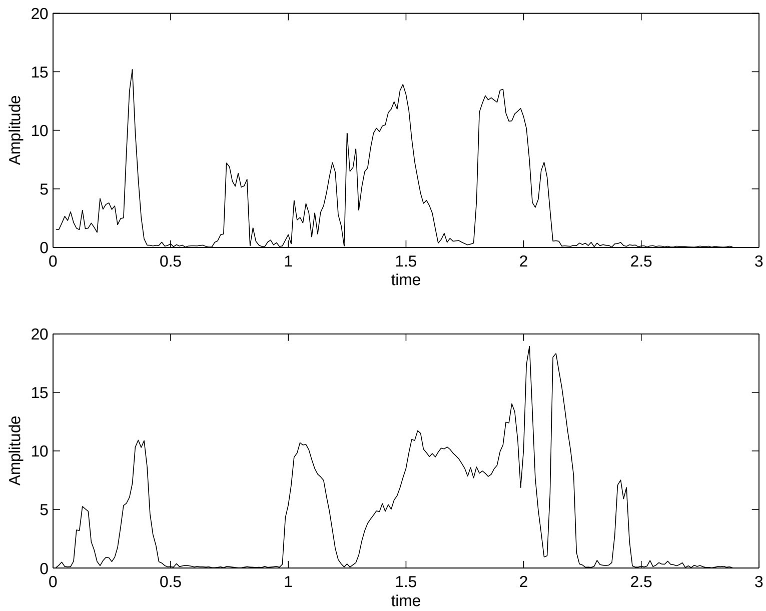


Figure 5.13. Original time-frequency signals $S_1(f,t)$ and $S_2(f,t)$ where $f = 195Hz$

As it is seen from figure 5.14.a, the separation is failed for that frequency bin. It can be seen from the fact that they are pretty much similar to each other or they are not similar with the original signals. When ICA is applied for the second time which is illustrated in figure 5.14.b, the signals are well separated which can also be check by their correlation with the original signals.

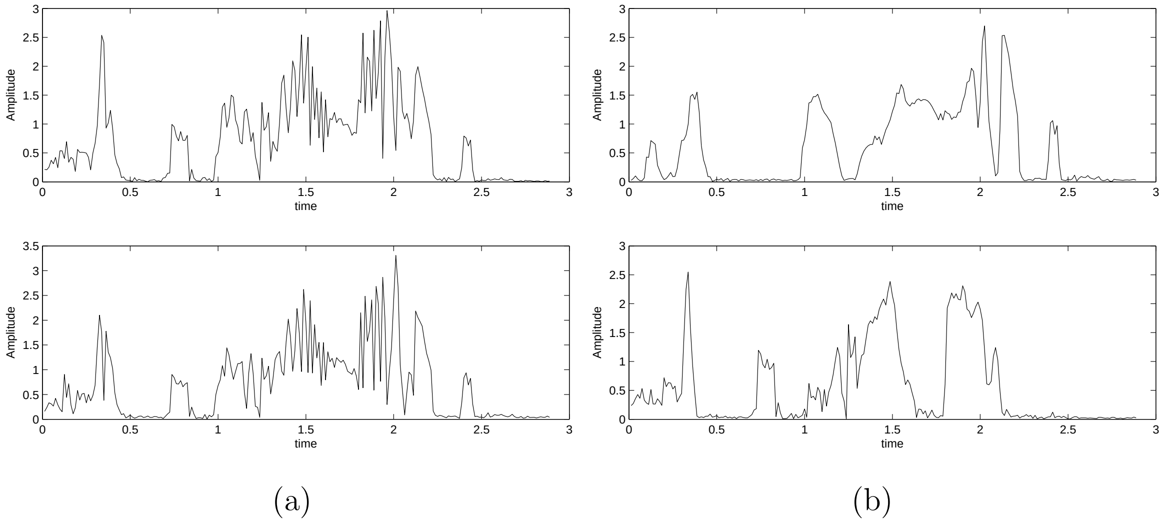


Figure 5.14. Separated signals (a) For ICA applied for once (b) For ICA applied for second time

5.4.3. Permutation Alignment with DOA approach

The sensor spacing is 4cm leads to $4.25kHz$ frequency range according to (3.7), after which there becomes spatial aliasing. This means that the permutations can be estimated by DOA approach up to $4.25kHz$ frequency range. For this reason in [9], the sampling rate of the signals is $8kHz$. But there is a problem with this sampling rate that is the resolution of samples to represent the delay is not enough. For example, assume that there is a source with arrival angle of 30^o, from (3.2) the actual delay occurred between sensors would be 5.810^{-5} seconds. In order to observe this delay in the sensors, this delay should be represented with at least one sample. But for $8kHz$ sampling rate, this delay is represented with $5.810^{-5} \times 8kHz = 0.4706$ samples which is not enough. For sampling rate of $16kHz$ this problem also arises for some angles. In our experiments, the delay between sensors is 1 sample.

<u>5.4.3.1. True Order Calculation.</u> In our method, the exact permutation is calculated by comparing the similarities between separated signals and the original signals in order to find the right permutation. The similarity measure (4.34) is used in calculations. For frequency bins in which the ICA algorithm failed to separate signals, the correct

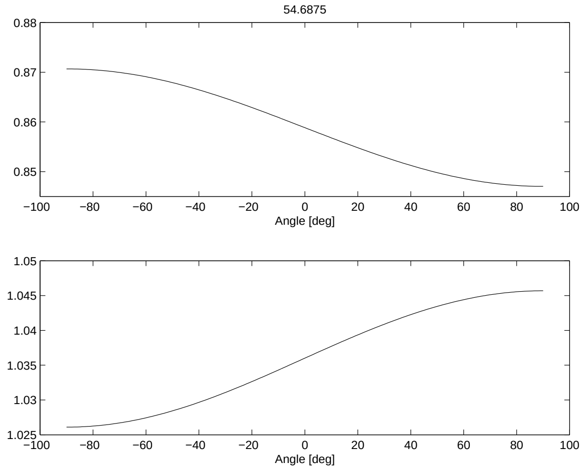
order could not be estimated at that frequency bin. When the signals are not separated they usually look similar to each other and carry the frequency components of both sources, that's why there is no meaning to find a permutation for that frequency bin. The correct order is calculated for all the cases that are "no reverb" and "reverb" cases with mixing type 1 and type 2.

There are two approaches to estimate the arrival angles of the sources as we it is mentioned in sections 4.4.2 and 4.4.3.

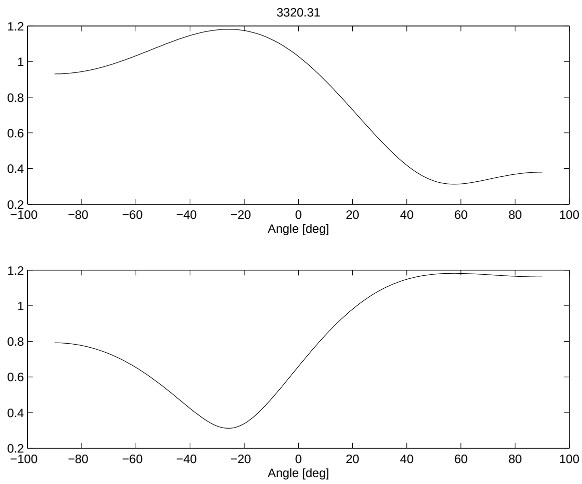
The directivity patterns are calculated for each frequency bin, and the minima of that pattern gives the direction of arrival. When the angles are estimated, the separated frequency components are sorted in increasing order. This method is motivated by conventional beamforming so is has the same limitations with this approach which are angular resolution and spatial aliasing that are mentioned in sections 3.3.1 and 3.3.2. That's why the directions can be estimated in a limited frequency band. The upper boundary of this method is determined by spatial resolution and the lower boundary is determined by angular resolution.

Figure 5.15 shows the directivity patterns for frequency bins 54Hz, 3.3kHz and 6.25kHz. As it can be seen from figure 5.15.a, the directivity patterns are monotone increasing or decreasing functions so the angles could not be estimated from these plots. Without knowing arrival angles, these plots could still be used to estimate whether angle is smaller or higher. For example, the order of the components should be changed at $54Hz$ according to figure 5.15.a. Around 3.5kHz the directivity patterns have clear minima, so that the angles could be estimated and sorted. Figure 5.15.c shows the directivity pattern of $6250Hz$ which is beyond spatial resolution that's why spatial aliasing occurs.

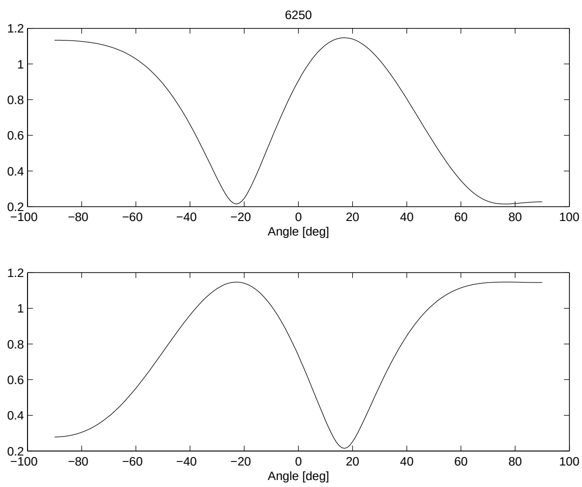
The true orders are used to measure the performance of DOA approach by using directivity patterns which are shown in tables 5.6 and 5.7. The performance is measured up to 512'th frequency bin which leads to 4 kHz. Although DOA approach is weak in low frequencies, the estimated orders are well enough for separation.



(a)



(b)



(c)

Figure 5.15. Directivity patterns for frequencies (a) 54 Hz (b) 3320 Hz (c) 6250 Hz

Table 5.6. DOA Performance with mixtures type 1 and using directivity patterns

	# of DOA Fails	# of DOA Success	Performance
No Reverb Case	2	510	%99.9 success
Reverb Case	164	348	%67.9 success

Table 5.7. DOA Performance with mixtures type 2 and using directivity patterns

	# of DOA Fails	# of DOA Success	Performance
No Reverb Case	1	511	%99.9 success
Reverb Case	185	327	%63.86 success

Confidence of estimated arrival angles can be measured by calculating the difference of the angle with the averaged angles [9]. If this difference smaller than a threshold value such as then this angle is confident.

The problem with this confidence method is that DOA estimation finds right directions in a relatively small frequency band. When the estimation of angles is wrong for the majority of frequencies then the mean of these angles won't give a meaningful measure which means some frequencies would be treated as not confident although they are. Table 5.8 shows the confidence measure of frequencies for mixing method type 2.

Table 5.8. Confidence Measure for mixture type 2

	Confident	Not Confident	Performance
No Reverb Case	251	261	%50.9 confident
Reverb Case	275	237	%53.7 confident

The method of direct calculation of arrival angle with formula (4.26) has a major drawback that sometimes the inside of arccos becomes out of range [-1,1] which leads a complex angle estimation. The reason for this wrong estimation would be the unprecise delay estimation due to low sampling frequency. Increasing sampling rate may lead better solutions. Also miscalculation of demixing matrix by ICA prevent this method to calculate direction of arrival angle. Table 5.9 and 5.10 shows the number of complex and real angles calculated with this method.

Table 5.9. Results for direct calculation of angle with mixture type 1

	# of complex angles	# of real angles
No Reverb Case	145	377
Reverb Case	128	384

Table 5.10. Results for direct calculation of angle with mixture type 2

	# of complex angles	# of real angles
No Reverb Case	140	382
Reverb Case	123	389

There is an alternative way to use this method. Actually there is no need to find the angles. The argument on the upper side of the formula (4.26) is enough to sort the separated components. Since arccos is a monotone decreasing function,

$$\arg(A) > \arg(B) \quad \longrightarrow \quad \arccos(arg(A)) > \arccos(arg(B)) \tag{5.3}$$

So by sorting the arguments $\arg(\frac{[\mathbf{W}^{-1}]_{j\Pi(k)}}{[\mathbf{W}^{-1}]_{j'\Pi(k)}})$, the aim of ordering the separated components can still be accomplished. The performance of the modified algorithm is shown in tables 5.11 and 5.12.

Table 5.11. DOA Performance with mixtures type 1 with direct calculation of angle

	# of DOA Fails	# of DOA Success	Performance
No Reverb Case	2	510	%99.9 success
Reverb Case	163	349	%68.1 success

Table 5.12. DOA Performance with mixtures type 2 with direct calculation of angle

	# of DOA Fails	# of DOA Success	Performance
No Reverb Case	1	511	%99.9 success
Reverb Case	185	327	%63.8 success

5.4.4. Permutation Alignment with correlations approach

Correlation approach, especially neighboring correlations method is very effective when ICA algorithm successfully separates the mixed signals at each frequency bin or at least at a big majority of frequency bins. It does not have an upper bound like DOA approach so all frequency bins could be considered with correlation method. The major drawback of this algorithm is that a single error in a narrow frequency band may lead to complete misalignment of permutations.

The separation of ICA algorithm is pretty much better when the mixing type 2 is used to generate mixed signals. That's why the correlation approach leads to a great success at almost all frequency bins. But due to less successful separation of mixtures generated with mixing type 1, the ordering of the correlation algorithm is not so successful. Table 5.13 and 5.14 shows the performance correlation approach with both mixing types.

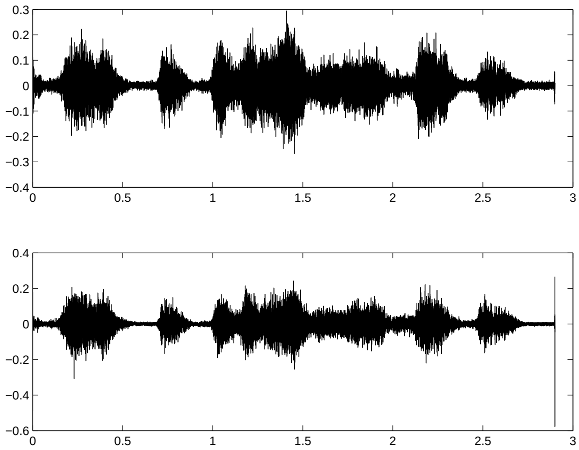
Table 5.13. Correlation Performance with mixtures type 1 with Neighboring Correlations

	# of Correlation Fails	# of Correlation Success	Performance
No Reverb Case	1317	731	%35.7 success
Reverb Case	983	1065	%52 success

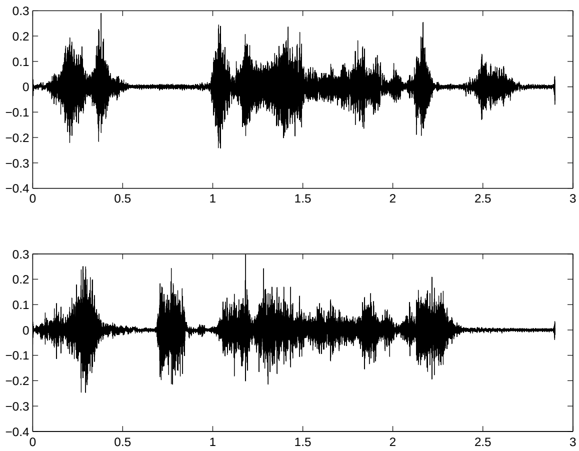
Table 5.14. Correlation Performance with mixtures type 2 with Neighboring Correlations

	# of Correlation Fails	# of Correlation Success	Performance
No Reverb Case	1	2047	%99.9 success
Reverb Case	1	2047	%99.9 success

For the reverberant mixtures, the resulting time domain signals are shown in figure 5.16.a and 5.16.b. As it is seen from figures mixing type 2 leads to a high performance in separating signals, however the performance with mixing type 1 is low.



(a)



(b)

Figure 5.16. The resulting time domain signals (a) For mixing type 1 (b) For mixing type 2

6. CONCLUSIONS

In this thesis, we investigated methods for audio source separation in reverberant environments. Both time domain and frequency domain algorithms are implemented but special care is given to frequency domain methods where the mixing is instantaneous at each frequency bin.

FastICA is a fast and robust algorithm for separation of speech signals. One of the advantages of this algorithm is that there is no tuning parameters for the update rules. The performance of FastICA is much better when the mixtures are generated in frequency domain which proves that it is a robust algorithm for basic ICA model even with the complex numbers. Unfortunately for the first type of mixing, the performance of the algorithm dramatically decreases. An important conclusion may come out the difference of these performances. The theory second mixing method actually forms the basis for the frequency domain separation. Ideally for each window we assume that the window is convolved with impulse response so that multiplying their fourier transforms in frequency domain have to lead to same result. However the first mixing method totally describes the real environment conditions. As a result, we may conclude that the segmentation and STFT of source signals constitute some ambiguity that may prevent frequency domain ICA to separate independent components.

Frequency domain ICA has a very challenging problem of permutation which is one of the ambiguities of ICA. There are two basic approaches for solving permutations namely Direction of Arrival and correlations approach. DOA approach is motivated by conventional beamforming theory. Thus the limitations of angular resolution and spatial resolution are also bounding the range of frequency band that DOA could be estimated. The arrival angles are calculated either by directivity patterns or by directly calculating the angle with (4.26). By observing directivity patterns, one can find the frequency band that angles can be measured confidently. Unfortunately this confident frequency band is narrow eve in 16kHz sampling rate. A solution to increase size of this band from upper bound is decreasing the spacing of sensors but this leads to increase the angular resolution so that the lower bound is also increased.

Beyond these direct calculation of arrival angle has another problem that sometimes the calculated angle becomes complex which means that no angle could be estimated. For 2x2 case, actually one does not need to calculate the angles to solve permutation. True order can still be achieved by sorting the argument $\arg(\frac{[\mathbf{W}^{-1}]_{j\Pi(k)}}{[\mathbf{W}^{-1}]_{j'\Pi(k)}})$ in (4.26)

It may be concluded that DOA approach is a limited approach for having a narrow frequency band that the angles confidently precalculated. However for 2x2 case, if the permutations are aligned at each frequency bin by using the angles calculated which are not necessarily confident, the performance is quite good by both directivity patterns and direct calculation of angle methods. But if number sources is more than two, estimating angles from directivity patterns becomes more challenging. Also ordering by angles that are not confident may not lead to right solution anymore.

Correlations approach is another important approach for solving permutations. Basically it measures the similarity of separated time frequency signals for each frequency bin. Neighboring correlations is a quite robust method when ICA problem is successfully solved for a great majority of frequency bins. This method has a high performance for mixing in frequency domain, which is because FastICA performance is also high. The major drawback of this algorithm is that an error in a narrow frequency band may lead to complete misalignment of permutations. On the other hand, there is another correlation method that assumes separated signals from far frequency bins may also have high correlation due to the temporal structure of signals. This assumption is quite optimistic that from figure 4.4, it can be seen that different time frequency signals from far frequency bins but same source signal have a little similarity between each other.

We also implemented conventional and adaptive beamforming (LCMV) methods which are using source localization in order to separate sources. In the experiments where there is no reverberation especially adaptive beamforming has a quite high performance. However for reverberant environments, the performance of both algorithms decreases.

We propose another method for future research, combining ICA and beamforming. In this case beamforming is a preprocessing step for ICA. This method has an important advantage that the number of sensors does not depend on number of sources, thus a relatively high number of sensors can be used for increasing source separation performance. Further the beamforming approach can be used to determine the number of sources. Then each source detected in the beamforming method, is given to ICA tool as a preprocessed input data so that ICA would separate sources which are already partially separated.

REFERENCES

1. Hyvarinen, A., J. Karhunen and E. Oja, *Independent Component Analysis*, John Wiley and Sons, Toronto, 2001.

2. Hyvarinen, A., “Survey on Independent Component Analysis”, Technical Report, Helsinki University of Technology, 1999.

3. Hyvarinen, A., “Independent Component Analysis, A Tutorial”,Technical Report, Helsinki University of Technology, 1999.

4. Comon, P., “Independent component analysis, a new concept?” *Signal Processing*, Vol.36, pp. 287-314, 1994.

5. Cardoso, J. F. and P. Comon ,“Independent Component Analysis,a survey of some algebraic methods”, *In Proceedings ISCAS'96*, Vol. 2, pp. 93-96, 1996.

6. Capdevielle, V., Ch. Serviere and C. K. I. Williams ,“Blind Separation of wide band sources inthe frequency domain”, *In Proceedings ICASSP'95*, Vol. 3, pp. 2080-2083, May 1995.

7. Murata, N., S. Ikeda and A. Ziehe ,“An Approach to Blind Source Separation based on Temporal Structure of Speech Signals”, *Neurocomputing*, Vol. 41, pp. 1-24, October 2001.

8. Anemuller, J. and B. Kolmeier, “Amplitude modulation decorrelation for convolutive blind source separation”, *Proceedings of ICA*, pp. 215-220, June 2000.

9. Sawada, H., R. Mukai, S. Araki and S. Makino, “A Robust and Precise Method for Solving the Permutation Problem of Frequency-Domain Blind Source Separation”, *IEEE transactions on speech and audio processing*, Vol. 12, No. 5, September 2004.

10. Saruwatari, H., S. Kurita, K.Takeda, F. Itakura and K. Shikano “Blind Source

Separation based on Subband ICA and Beamforming", Technical Report, Nagoya University,Japan, 2003.

11. Buchner, H., R. Aichner and W. Kellermann, "Blind Source Separation for convolutive mixtures exploiting nongaussianity, nonwhiteness and nonstationary",*International Workshop on Acoustic Echo and Noise Control, IWAENC2003* Kyoto,Japan, September 2003.

12. Buchner, H., R. Aichner and W. Kellermann, "A Generalization of Blind Source Separation ALgorithms for Convolutive Mixtures based on Second order Statistics",*IEEE transactions on speech and audio processing* Vol .13, No. 1, January 2005.

13. Buchner, H., R. Aichner, F. Yan and W. Kellermann, "Real Time Convolutive Blind Source Separation based on Broadband approach",Technical Report University of Erlangen-Nuremberg, 2005.

14. Van Veen, B. and K. Buckley, "Beamforming: A Versatile Approach to Spatial Filtering",*IEEE ASSP Mag.*, pp. 2-24, 1988 .

15. Johnson, D. H., D. E. Dudgeon, *Array Signal Proceesing:Concepts and Techniques*, Prentice Hall, Upper Saddle River, New Jersey, 1993.

16. Allen, J. and D. Berkley, "Image Method for efficiently simulating small-room acoustics",Technical Report, Bell Laboratuaries, 1978 .

17. Proakis, J. G., D. G. Manolakis, *Digital Signal Processing*, Prentice Hall, Upper Saddle River, New Jersey, 1996.

18. Hyvarinen A., "The Fixed-Point algorithm and Maximum Likelihood Estimation for Independent Component Analysis",Technical Report, Helsinki University of Technology, 1999.

19. Hyvarinen A., "A Fast Fixed-Point algorithm for Independent Component Analy-

sis of complex valued signals",Technical Report, Helsinki University of Technology, 1999.

20. Amari S., "Natural Gradient works efficiently in learning",Technical Report, Saitama, Japan 1999 .

21. Bell, A. and T. Sejnowski ,"An information-maximizaton approach to blind separation and blind deconvolution", *Neural Computation*, Vol. 7, No. 6, pp. 1129-1159, 1995.

22. Cardoso, J. F. ,"Infomax and maximum likelihood for blind source separation. based on information maximization.", to appear in *IEEE signal processing letters.*

23. Attias, H. ,"Independent Factor Analysis", *Neural Computation.*

24. Friedman, J. H., J. W. Tukey ,"A Projection Pursuit Algorithm for exploratory daha analysis", *IEEE transactions for computers*, c-23(9), pp. 881-890, 1974.

25. Friedman, J. H. ,"Exploratory projection pursuit", *J. of the American Statistical Association*, Vol. 82, pp. 249-266, 1987.

26. Matsuoka, K. and S. Nakashima, "Minimal Distortion Principle for blind source separation", *Proceedings of ICA*, pp. 722-727, December 2001.