

Universidade Federal do Rio de Janeiro

Escola Politécnica

Departamento de Eletrônica e de Computação

**Separação Cega de Misturas Convolutivas no Domínio do  
Tempo Utilizando Clusterização**

Autor:

---

Augusto Proença da Silva

Orientador:

---

Prof. Julio Cesar Boscher Torres, D. Sc.

Co-Orientador:

---

Prof. Diego Barreto Haddad, M. Sc.

Examinador:

---

Prof<sup>a</sup>. Mariane Rembold Petraglia, Ph.D.

DEL

Junho de 2009

## **Dedicatória**

Este trabalho é dedicado a Ramine Magalhães Ferreira, o amor da minha vida, aquela que sempre esteve ao meu lado em todos os momentos, minha fortaleza. Sem ela, eu não teria chegado até aqui.

Saber lidar com as diferenças entre nós é a chave para nosso sucesso.

## **Agradecimento**

Agradeço, em primeiro lugar, à minha família. Ela foi e sempre será a base de toda minha jornada neste mundo.

À minha mãe Mônica, aquela que sempre esteve ao meu lado em todas as horas e ocasiões, apesar de nossas constantes divergências sobre os estudos.

Ao meu pai Luiz Augusto, que seja onde ele estiver, sei que estará feliz por estar realizando este feito.

À minha avó Lindomar, minha mãe em dobro, que desde pequeno cuidou de mim e não me deixou faltar nada.

Ao meu avô Paulo, por todas as caronas e tanques de gasolina dados durante todo o longo caminho até aqui.

Aos meus avós Waldyr e Lucinda, por sempre acreditarem no meu potencial.

À minha futura esposa Ramine, que sempre que precisei esteve ao meu lado, me incentivando e não me deixando desistir.

Ao meu orientador Julio, por ter me guiado de forma tão dedicada e cuidadosa, sempre me fornecendo idéias e sugestões valiosas para a elaboração deste projeto.

Ao meu co-orientador Diego, por todas as inúmeras ajudas em programação e pelo vasto conhecimento da teoria aqui apresentada.

Por último, a todos meus amigos que estiveram presentes nesta longa caminhada até aqui.

## Resumo

O presente trabalho tem como objetivo a implementação de um algoritmo visando realizar a separação cega de misturas convolutivas no domínio do tempo, a fim de fornecer uma solução eficaz para o problema *cocktail party*, como é amplamente conhecido o problema de separação cega de fontes (BSS, do inglês *Blind Source Separation*). Para tal, o algoritmo se baseia no princípio de componentes independentes (ICA – *Independent Component Analysis*) e na teoria de clusterização.

Neste trabalho são apresentados os fundamentos teóricos associados a implementação do algoritmo, como não-gaussianidade e independência estatística.

A eficácia do algoritmo implementado é verificada por meio de simulações realizadas em diferentes cenários.

Palavras-Chave: BSS, ICA, Clusterização, Misturas Convolutivas.

## **Abstract**

The aim of this work is the implementation of an algorithm to perform the blind separation of convolutive mixture using a time-domain approach, in order to provide an efficient solution to the cocktail party problem, as is widely known the Blind Source Separation problem. The algorithm was implemented based on the Independent Component principles and Clusterization theory to achieve its goal.

This work contains the necessary theory related to the algorithm implementation, such as non-gaussianity and statistical independence.

The efficacy of the implemented method is attested by simulations conducted in different scenarios.

Keywords: BSS, ICA, Clusterization, Convolutive Mixtures.

## Siglas

BSS:	<i>Blind Source Separation</i>
EFICA:	<i>Extend Fast ICA</i>
HOS:	<i>Higher Order Statistics</i>
ICA:	<i>Independent Component Analysis</i>
INCA:	<i>Independent Component Analysis</i>
MIMO:	<i>Multiple-Input and Multiple-Output</i>
MISO:	<i>Multiple-Input and Single-Output</i>
NS:	<i>Non-Stationarity</i>
PADS:	<i>Processamento Análogo e Digital de Sinais</i>
PDF:	<i>Probability Density Function</i>
RAM:	<i>Random Access Memory</i>
SAR:	<i>Signal to Artifact Ratio</i>
SCA:	<i>Sparse Component Analysis</i>
SDR:	<i>Signal to Distortion Ratio</i>
SIR:	<i>Signal to Interference Ratio</i>
SNR:	<i>Signal to Noise Ratio</i>
SOS:	<i>Second Order Statistics</i>
STF:	<i>Space-Time-Frequency</i>
UFRJ:	<i>Universidade Federal do Rio de Janeiro</i>

# Sumário

1. Introdução.....	1
1.1. Tema.....	1
1.2. Motivação.....	1
1.3. Objetivos.....	3
1.4. Metodologia.....	3
1.5. Organização.....	4
2. Separação Cega de Fontes.....	5
2.1. Abordagens ao Problema de BSS.....	6
2.1.1. Estatísticas de Ordem Superior (HOS – <i>Higher Order Statistics</i> ).....	6
2.1.2. Estatísticas de Segunda Ordem (SOS – <i>Second Order Statistics</i> ).....	6
2.1.3. Não-Estacionaridade (NS – <i>Non-Stationarity</i> ).....	7
2.1.4. Relações Espaço-Tempo-Frequência (STF – <i>Space-Time-Frequency</i> ).....	7
2.2. ICA – <i>Independent Component Analysis</i> .....	7
2.2.1. Restrições.....	9
2.2.2. Ambiguidades.....	10
2.3. Medidas de Não-Gaussianidade.....	10
2.3.1. Curtose.....	10
2.3.2. Entropia.....	12
2.3.3. Informação Mútua.....	12
2.3.4. Negentropia.....	13
2.4. Funções de Contraste.....	14
2.5. Outras abordagens para BSS:.....	15
2.5.1. Bayesiana.....	15
2.5.2. Análise de Componentes Esparsos.....	16
3. Separação Cega de Fontes para Misturas Convolutivas.....	17
3.1. Domínio do tempo.....	17
3.1.1. <i>Feedforward</i> .....	17
3.1.2. <i>Feedback</i> .....	18
3.2. Domínio da frequência.....	19
3.3. Banco de Filtros.....	19

3.4. Algoritmo Proposto .....	20
4. Resultados.....	24
4.1. Análise 1: Definição do tamanho ideal do bloco para separação.....	26
4.1.1. Teste 1: Bloco com 2000 amostras.....	27
4.1.2. Teste 2: Bloco com 3000 amostras.....	28
4.1.3. Teste 3: Bloco com 4000 amostras.....	29
4.1.4. Teste 4: Bloco com 5000 amostras.....	30
4.2. Análise 2: Eficiência do algoritmo frente à complexidade da mistura.....	31
4.2.1. Teste 1: Mistura de complexidade 2.....	32
4.2.2. Teste 2: Mistura de complexidade 3.....	34
4.2.3. Teste 3: Mistura de complexidade 5.....	36
4.2.3. Teste 3: Mistura de complexidade 5.....	36
4.2.4. Teste 4: Mistura de complexidade 8.....	38
4.2.5. Teste 5: Mistura de complexidade 10.....	40
4.5.6. Teste 6: Mistura de complexidade 15.....	42
4.2.7. Teste 7: Mistura de complexidade 20.....	44
5. Conclusão .....	46
5.1. Sugestões .....	46
6. Referências Bibliográficas.....	48



## Sumário de Tabelas

Tabela 4.1 - Análise 1/Teste 1: resultados para bloco de 2000 amostras .....	27
Tabela 4.2 - Análise 1/Teste 2: resultados para bloco de 3000 amostras .....	28
Tabela 4.3 - Análise 1/Teste 3: resultados para bloco de 4000 amostras .....	29
Tabela 4.4 - Análise 1/Teste 4: resultados para bloco de 5000 amostras .....	30
Tabela 4.5 - Análise 2/Teste 1: resultados ( $M = 2 / L = 2$ ) .....	32
Tabela 4.6 - Análise 2/Teste 2: resultados ( $M = 3 / L = 3$ ) .....	34
Tabela 4.7 - Análise 2/Teste 3: resultados ( $M = 5 / L = 5$ ) .....	36
Tabela 4.8 - Análise 2/Teste 4: resultados ( $M = 8 / L = 8$ ) .....	38
Tabela 4.9 - Análise 2/Teste 5: resultados ( $M = 10 / L = 10$ ) .....	40
Tabela 4.10 - Análise 2/Teste 6: resultados ( $M = 15 / L = 15$ ) .....	43
Tabela 4.11 - Análise 2/Teste 7: resultados ( $M = 20 / L = 20$ ) .....	45

## Sumário de Figuras

Figura 1.1 - Cocktail Party para 3 fontes e 3 sensores .....	2
Figura 2.1 - Modelo BSS .....	5
Figura 2.2 - Formas Gerais da Curtose.....	11
Figura 3.1 - Rede <i>feedforward</i> 2x2 para ICA de misturas convolutivas .....	18
Figura 3.2 - Rede <i>feedback</i> 2x2 para ICA de misturas convolutivas .....	18
Figura 4.1 - Análise 1/Teste 1: fontes e estimativas para bloco de 2000 amostras .....	27
Figura 4.2 - Análise 1/Teste 2: fontes e estimativas para bloco de 3000 amostras .....	28
Figura 4.3 - Análise 1/Teste 3: fontes e estimativas para bloco de 4000 amostras .....	29
Figura 4.4 - Análise 1/Teste 4: fontes e estimativas para bloco de 5000 amostras .....	30
Figura 4.5 - Análise 2/Teste 1: fontes e estimativas ( $M = 2 / L = 2$ ) .....	32
Figura 4.6 - Análise 2/Teste 3: fontes e estimativas ( $M = 3 / L = 3$ ) .....	34
Figura 4.7 - Análise 2/Teste 3: fontes e estimativas ( $M = 5 / L = 5$ ) .....	36
Figura 4.8 - Análise 2/Teste 4: fontes e estimativas ( $M = 8 / L = 8$ ) .....	38
Figura 4.9 - Análise 2/Teste 5: fontes e estimativas ( $M = 10 / L = 10$ ) .....	40
Figura 4.10 - Análise 2/Teste 6: fontes e estimativas ( $M = 15 / L = 15$ ) .....	42
Figura 4.11 - Análise 2/Teste 7: fontes e estimativas ( $M = 20 / L = 20$ ) .....	44

# 1. Introdução

Ao assistirmos um filme policial, onde investigadores são capazes de identificar um assassino isolando uma pequena fala em meio a inúmeras fontes gravadas durante o crime, nos questionamos se aquilo realmente é possível. Deixando o *glamour* dos filmes de lado, tal processo de separação (no caso, áudio), já é possível nos dias de hoje e vem recebendo grande atenção por parte da comunidade científica, devido à gama de aplicações dessa teoria, tais quais o tratamento de sinais de áudio/vídeo [1-4], sistemas de comunicações digitais [5-8], engenharia biomédica [9-12] e análises financeiras [13,14].

## 1.1. Tema

O problema de separação cega de fontes ou simplesmente BSS (do inglês, *Blind Source Separation*) consiste em, dadas algumas misturas de sinais, recuperar cada uma das fontes individuais que compõem as mesmas. O termo “cega” se deve ao fato de não ser necessário nenhum conhecimento prévio sobre as características dos sinais que compõem a mistura com a qual se está trabalhando, tampouco acerca das funções de transferência do ambiente envolvidas.

Uma hipótese comum acerca dos sinais a separar é a de que estes são estatisticamente independentes entre si; trata-se de uma hipótese estatisticamente forte [15], a qual costuma ser suficientemente atendida na prática. O sucesso dos métodos de separação cega de fontes revela-nos que esta hipótese não é problemática. Mesmo que o fosse, as técnicas BSS almejam maximizar a independência das saídas (estimativas), o que pode ser obtido mesmo que haja certa dependência entre as entradas (misturas) [16].

## 1.2. Motivação

O marco inicial da teoria de separação cega de fontes é a técnica proposta por Herault, Jutten e Ans em 1985 [17] denominada por seus desenvolvedores de Análise de Componentes Independentes (originalmente INCA, e posteriormente ICA – *Independent Component Analysis*). Esse trabalho pioneiro visava separar sinais neurais

assumindo apenas que estes eram não-gaussianos e estatisticamente independentes entre si [18,19].

Nos dias atuais, a ICA é um dos métodos mais difundidos no contexto de separação cega de fontes, porém a aplicação desta técnica não está restrita à solução deste problema, sendo também aplicada em outros campos na área de processamento de sinais, tais como desconvolução cega [20] e extração de parâmetros em sinais digitais [21].

O exemplo clássico de separação cega de fontes é o problema popularmente conhecido como *Cocktail Party*. Este foi primeiramente enunciado em 1953 por Cherry [22], no contexto da capacidade humana de reconhecimento de mensagens de voz. Atualmente, existem diversas tentativas de modelar este problema. [23-25]

Para entendermos o problema *Cocktail Party*, consideremos o cenário a seguir: em um ambiente qualquer existem três interlocutores (fontes) conversando simultaneamente, sendo que cada um deles gera um sinal de áudio distinto. No mesmo ambiente, há três microfones (sensores) responsáveis por captar três misturas diferentes dos sinais gerados pelos interlocutores. De posse das misturas, queremos extrair as fontes de sinal originais sem possuímos nenhum conhecimento prévio das propriedades das fontes nem do processo de mistura.

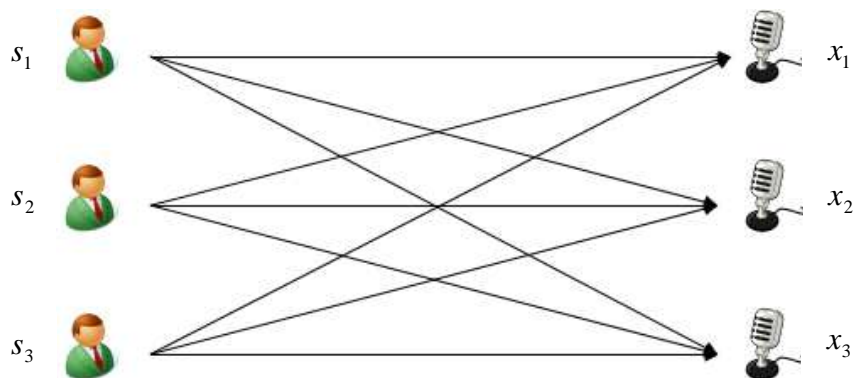


Figura 1.1 - Cocktail Party para 3 fontes e 3 sensores

Matematicamente, sejam  $x_1(t)$ ,  $x_2(t)$  e  $x_3(t)$  os sinais das misturas gravados pelos microfones em função do tempo e,  $s_1(t)$ ,  $s_2(t)$  e  $s_3(t)$  os sinais das fontes independentes. Considerando que as misturas são instantâneas, estas podem ser

consideradas combinações lineares das fontes e representadas pelo sistema linear abaixo:

$$\begin{aligned}x_1(t) &= a_{11} \cdot s_1(t) + a_{12} \cdot s_2(t) + a_{13} \cdot s_3(t) \\x_2(t) &= a_{21} \cdot s_1(t) + a_{22} \cdot s_2(t) + a_{23} \cdot s_3(t) \\x_3(t) &= a_{31} \cdot s_1(t) + a_{32} \cdot s_2(t) + a_{33} \cdot s_3(t)\end{aligned}\tag{1.1}$$

ou alternativamente, na forma matricial [26]

$$\mathbf{x} = \mathbf{A} \cdot \mathbf{s} .\tag{1.2}$$

Assim, conhecendo-se a matriz  $\mathbf{A}$  pode-se obter as fontes originais, por meio de uma mera inversão matricial, a qual só é possível quando o número de sensores não é superado pelo número de fontes.

Entretanto, as misturas encontradas no mundo real não são instantâneas, sendo geradas por meio de atrasos, convoluções e/ou combinações das fontes. A separação de tais misturas, denominadas convolutivas, é a motivação deste trabalho.

### 1.3. Objetivos

O projeto aqui apresentado tem por finalidade a implementação de um método para realizar a separação cega de misturas convolutivas através de uma abordagem no domínio no tempo. Toda a teoria apresentada tem por base a teoria de Análise de Componentes Independentes, ou simplesmente ICA (*Independent Component Analysis*).

### 1.4. Metodologia

O algoritmo implementado, bem como os modelos apresentados no projeto são validados por meio de experimentos práticos utilizando o software Matlab<sup>1</sup> desenvolvido pela MathWorks rodando em um Pentium Dual Core Duo 2.16GHz com 2Gb de memória RAM e sistema operacional Windows XP SP3<sup>2</sup>.

---

<sup>1</sup> Software com licença cedida pelo laboratório PADS (UFRJ)

<sup>2</sup> Software com licença cedida pelo laboratório PADS (UFRJ)

As fontes utilizadas neste projeto são arquivos de áudio do tipo “.wav” amostrados em 16kHz. As misturas convolutivas analisadas são obtidas combinando filtros aleatórios sintéticos com os sinais das fontes.

A eficiência do algoritmo implementado é avaliada por meio de algumas medidas padrão em processamento de sinais [27] como: SNR (*Signal to Noise Ratio*), SIR (*Signal to Interference Ratio*), SAR (*Signal to Artifact Ratio*) e SDR (*Signal to Distortion Ratio*). Os valores obtidos serão comparados com o resultado das mesmas medidas obtidas para outro algoritmo [28] que utiliza uma abordagem diferente da utilizada neste projeto.

## **1.5. Organização**

O presente capítulo visa apresentar uma breve introdução sobre o assunto principal deste projeto, BSS, bem como a motivação para a realização do mesmo. Além disso, são apresentados o escopo e a metodologia utilizada no desenvolvimento do mesmo projeto.

O Capítulo 2 enuncia uma introdução teórica a respeito do tema de separação cega de fontes, tratando de diversos aspectos relevantes no estudo do tema, como gaussianidade e independência estatística. É apresentado também, um breve resumo sobre as principais abordagens para a solução do problema de BSS para misturas instantâneas, focando no método de ICA. Este capítulo traz também um resumo dos principais modelos aplicados para misturas convolutivas, enunciando as vantagens e desvantagens de cada modelo.

O problema de BSS para misturas convolutivas é o tema do Capítulo 3. Nele há um resumo dos principais modelos aplicados para este tipo de misturas, enunciando as vantagens e desvantagens de cada modelo. O algoritmo implementado, bem como sua base teórica, também é mostrado neste capítulo. Trata-se de um método para separação cega de misturas convolutivas utilizando ICA no domínio do tempo.

No Capítulo 4, encontram-se os resultados dos testes realizados com o algoritmo implementado, bem como os resultados obtidos para outro algoritmo de BSS, visando realizar uma análise comparativa de desempenho entre os mesmos.

Por último, no Capítulo 5 são apresentadas as conclusões e sugestões de trabalho futuro com base nos resultados obtidos.

## 2. Separação Cega de Fontes

A Separação Cega de Fontes tem como objetivo estimar componentes individuais mutuamente independentes por meio da observação de sinais obtidos por sensores. Tal tarefa é indispensável quando se trabalha com fontes que encontram-se misturadas através de um modelo desconhecido e apenas misturas destas fontes de interesse estão disponíveis nos sensores para observação. Esta técnica é denominada cega, pois a estimativa é realizada sem nenhum conhecimento prévio das fontes originais, nem do modelo utilizado na mistura das mesmas.

A falta de informação prévia das fontes não deve ser entendida como algo negativo para o modelo, visto que ela é a grande vantagem neste caso, uma vez que o torna uma ferramenta versátil na exploração da diversidade espacial gerada pelo número de sensores utilizados.

O modelo matemático básico para o problema de BSS é dado por

$$\mathbf{x}(t) = \mathbf{A} \cdot \mathbf{s}(t), \quad (2.1)$$

onde

$\mathbf{x}(t) = [x_1(t), x_2(t), \dots, x_m(t)]^T$  : vetor das misturas observadas;

$\mathbf{s}(t) = [s_1(t), s_2(t), \dots, s_n(t)]^T$  : vetor das fontes independentes;

$\mathbf{A}$  : matriz não-singular de mistura com dimensão  $(m \times n)$ .

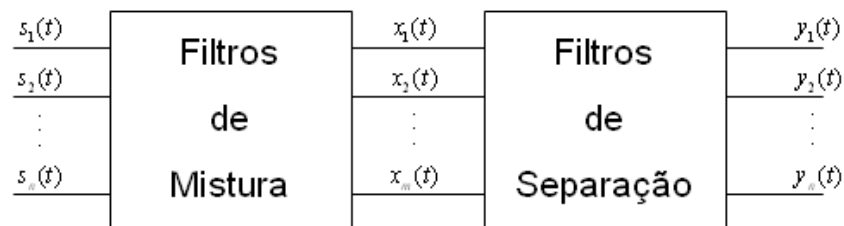


Figura 2.1 - Modelo BSS

A solução do problema de BSS está atrelada a algumas definições, como:

- A mistura ser linear ou não-linear;

- O processo de mistura ser variante ou invariante no tempo;
- A operação de mistura ser convolutiva ou instantânea;
- Os sensores serem imunes, ou não, a ruído;
- Relação entre o número de fontes ( $n$ ) e o número de misturas ( $m$ ).

A relação entre o número de misturas ( $m$ ) e o número de fontes ( $n$ ) caracteriza o sistema analisado, de forma que:

- $m > n$  : Sistema sobre-determinado;
- $m < n$  : Sistema sub-determinado;
- $m = n$  : Sistema determinado.

## **2.1. Abordagens ao Problema de BSS**

Atualmente, existem diversos algoritmos sendo desenvolvidos para solucionar o problema de BSS, porém os princípios utilizados por tais métodos podem ser agrupados a partir de quatro abordagens fundamentais em que se baseiam [29], como descritas nos itens 2.1.1. a 2.1.4.

### **2.1.1. Estatísticas de Ordem Superior (HOS – *Higher Order Statistics*)**

É fundamentada no uso de alguma medida estatística de ordem superior (implícita ou explícita) de independência das fontes que compõem a mistura, como, por exemplo, a curtose. Tais medidas podem ser relacionadas a não-gaussianidade e/ou esparsidade das fontes [31].

### **2.1.2. Estatísticas de Segunda Ordem (SOS – *Second Order Statistics*)**

Caso as fontes sejam espacialmente descorrelacionadas, condições menos restritivas quanto à independência das fontes podem ser aplicadas ao modelo. Neste caso, pode-se utilizar apenas estatísticas de segunda ordem (correlação) para estimar as matrizes de mistura e as fontes desejadas, tornando a solução do problema menos complexa do ponto de vista computacional [31-33].



### 2.1.3. Não-Estacionaridade (NS – *Non-Stationarity*)

Outra abordagem utilizada na solução do problema de BSS explora as propriedades não-estacionárias das fontes estimadas em conjunto com estatísticas de segunda ordem [33-35]. Basicamente, os métodos baseados nessa abordagem trabalham com o fato das variâncias das fontes não serem constantes ao longo do tempo [36].

### 2.1.4. Relações Espaço-Tempo-Frequência (STF – *Space-Time-Frequency*)

A quarta abordagem leva em consideração diversas propriedades dos sinais de interesse, geralmente envolvendo o trinômio espaço-tempo-frequência, como a coerência tempo-espacial do sinal [38].

Paralelamente às abordagens acima descritas, outros métodos de separação cega de fontes vêm sendo desenvolvidos a partir da combinação das abordagens fundamentais anteriormente mostradas (HOS, SOS, NS e STF) na tentativa de separar ou extrair fontes com diversas propriedades estatísticas. Outro importante objetivo desses novos algoritmos é a minimização da influência do ruído ou de outras interferências indesejadas nos sinais de interesse.

Uma das técnicas de separação mais difundida nos estudos de BSS é a Análise de Componentes Independentes (ICA). Esse é o método utilizado no algoritmo implementado neste projeto.

## 2.2. ICA – *Independent Component Analysis*

Análise de Componentes Independentes é uma técnica aplicada na separação cega de fontes que se baseia no uso de estatísticas de ordem superior para estimar cada uma das fontes por meio da observação de diversas misturas geradas a partir destas fontes.

Assumindo que sejam observadas  $n$  misturas lineares  $\mathbf{x}_1(t), \mathbf{x}_2(t), \dots, \mathbf{x}_m(t)$  que possuem informações referentes a  $m$  fontes independentes  $s_1(t), s_2(t), \dots, s_n(t)$ , tem-se:

$$\mathbf{x}_i(t) = \sum_{j=1}^n a_{ij} \cdot s_j(t). \quad (2.2)$$

No modelo geral de ICA, a componente do tempo não é levada em consideração,

uma vez que se tratam cada mistura  $x_i$  e cada fonte  $s_j$  como sendo variáveis aleatórias, ao invés de considerá-las como sinais propriamente ditos.

Utilizando uma notação matricial, podemos escrever o modelo básico de ICA como

$$\mathbf{x} = \mathbf{A} \cdot \mathbf{s} , \quad (2.3)$$

onde

$\mathbf{x}$  : vetor contendo os elementos da mistura;

$\mathbf{s}$  : vetor contendo os sinais fontes;

$\mathbf{A}$  : matriz de mistura.

Normalmente, o ruído é tratado como uma fonte independente no modelo básico de ICA. Entretanto, este modelo pode ser expandido assumindo que estamos lidando com um ruído aditivo, como mostrado abaixo:

$$\mathbf{x} = \mathbf{A} \cdot \mathbf{s} + \mathbf{v} , \quad (2.4)$$

onde  $\mathbf{v}$  é um ruído. Neste caso, a solução do problema pode se tornar mais complexa. Tal modelo não será contemplado no presente trabalho.

Uma vez que as componentes independentes não são observadas diretamente, estas são chamadas de variáveis latentes. Os coeficientes  $a_{ij}$  que compõem a matriz de mistura  $\mathbf{A}$  também são desconhecidos. Neste modelo, apenas as variáveis aleatórias  $x_i$  são conhecidas, sendo os coeficientes  $a_{ij}$  e as componentes independentes  $s_j$  estimadas a partir de  $x_i$ .

O modelo acima é dito modelo generativo, pois este descreve como as misturas observadas são geradas a partir de um processo de mistura das fontes  $s_j$ .

No contexto de BSS, existem casos onde o modelo linear básico não é suficiente para solucionar o problema, logo faz-se necessário algumas definições adicionais a fim de alcançarmos uma solução válida para o problema. Uma delas é assumir que  $\mathbf{x}$  é uma função genérica e possivelmente não linear das fontes,

$$\mathbf{x} = f(\mathbf{s}). \quad (2.5)$$

Neste projeto será considerado apenas o modelo linear, generativo e determinado. Todas as variáveis de mistura serão consideradas como tendo média zero, sem perda da generalidade. Na prática, a condição acima nem sempre é verdadeira, logo pode-se trabalhar com a variável  $\mathbf{x}$ , derivada da mistura original  $\mathbf{x}'$ , cuja média foi extraída:

$$\mathbf{x} = \mathbf{x}' - E\{\mathbf{x}'\}. \quad (2.6)$$

Caso seja necessário recuperar a média das componentes independentes encontradas, isso poderá ser feito através da adição dos resultados encontrados com a média das componentes independentes originais dada por

$$E\{\mathbf{s}'\} = A^{-1} \cdot E\{\mathbf{x}'\}. \quad (2.7)$$

Uma das conseqüências desta definição é que as componentes independentes também terão média zero.

### 2.2.1. Restrições

Visando garantir a convergência do modelo de ICA, algumas considerações necessitam ser feitas [16].

- O número de misturas observadas  $m$  deve ser maior ou igual ao número de componentes independentes  $n$ , logo  $m \geq n$ ;
- As fontes devem ser estatisticamente independentes entre si. A independência estatística de misturas é definida em termos das densidades de probabilidade das mesmas, ou seja, a densidade conjunta pode ser fatorada em um produto das densidades marginais:  $p(\mathbf{s}) = \prod_{j=1}^n p[s_j(\mathbf{t})]$ ; (2.8)
- As componentes independentes devem possuir distribuições de probabilidade não-gaussianas. Na prática, apenas uma componente gaussiana é permitida no modelo, uma vez que qualquer combinação linear de uma variável gaussiana também apresenta distribuição gaussiana [39];

- A matriz de mistura deve ser quadrada [40].

### 2.2.2. Ambiguidades

A utilização deste modelo de ICA gera algumas ambiguidades:

- Uma vez que o vetor  $s$  e a matriz  $A$  são desconhecidos para o modelo básico de ICA, não é possível determinar a ordem das componentes independentes, ou seja, a fonte  $s_j$  pode não corresponder ao sinal observado  $x_j$ ;
- A variância das componentes independentes não pode ser determinada, pois os coeficientes  $a_{ij}$  e as fontes  $s_j$  são desconhecidos, logo pode-se multiplicar por um escalar  $\alpha$  qualquer uma das fontes e reverter esta operação em seguida, dividindo a coluna correspondente da matriz  $A$  pelo mesmo escalar  $\alpha$ ;
- Por convenção, assume-se que as componentes independentes possuem variância unitária. Entretanto, o modelo ainda apresenta uma ambiguidade de sinal (reversão de fase) associada às fontes estimadas, apesar de tal fato não afetar o modelo;

## 2.3. Medidas de Não-Gaussianidade

Para que o modelo de ICA possa ser aplicado na separação de uma mistura, deve-se assumir que as fontes originais não possuam função densidade de probabilidade (pdf – *Probability Density Function*) gaussiana [15].

Em ordem a garantir que a condição acima seja satisfeita, existem diversas medidas estatísticas de gaussianidade que podem ser aplicadas ao modelo.

### 2.3.1. Curtose

A curtose é um parâmetro que descreve a forma de uma função densidade de probabilidade. Ela também pode ser usada como medida de não-gaussianidade de uma variável aleatória, visto que uma distribuição gaussiana possui curtose normalizada igual a zero [41-42].

A curtose normalizada de uma variável aleatória  $y$  pode ser definida matematicamente como

$$kurt(y) = \frac{E\{y^4\}}{E\{y^2\}^2} - 3 \quad , \quad (2.9)$$

onde  $E\{y^4\}$  é o momento de quarta ordem e  $E\{y^2\}$  é a variância.

Assim, temos que:

- $kurt(y) = 0$ : variável tem a mesma função densidade de probabilidade que a distribuição normal. Chamam-se estas funções de mesocúrticas;
- $kurt(y) > 0$ : a distribuição em questão é mais alta e concentrada que a distribuição normal. Estas funções densidade de probabilidade são leptocúrticas;
- $kurt(y) < 0$ : a função de distribuição é mais achatada que a distribuição normal. Este tipo de função é conhecido como platicúrtica.

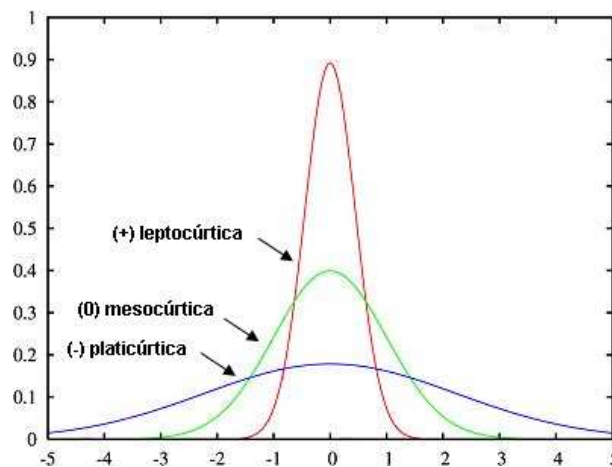


Figura 2.2 - Formas Gerais da Curtose

Na prática, quanto maior o valor da curtose, maior será a variância devido aos desvios atípicos da distribuição.

Devido ao baixo custo computacional e à simplicidade da teoria, a curtose é uma medida amplamente utilizada na estimação de não-gaussianidade nos modelos de ICA.

Uma das desvantagens desse método deve-se ao fato de que esta medida é

sensível a *outliers*, ou seja, o valor da curtose é bastante influenciado por valores na cauda da distribuição [43].

### 2.3.2. Entropia

A entropia de uma variável aleatória é definida como o grau de informação que uma observação desta variável fornece. Esta medida está associada à medida de incerteza desta variável, ou seja, quanto mais aleatória e desestruturada for a variável, maior será sua entropia [44].

Uma vez que a entropia de uma variável aleatória contínua é infinita, podemos definir esta em função da entropia diferencial. Ela é matematicamente definida por [45]

$$H(x) = -E\{\ln[p_x(x)]\}, \quad (2.10)$$

Outra medida bastante utilizada é a entropia condicional. Ela nos fornece o quanto de incerteza na variável  $x$  há após a observação de outra variável  $y$ . Logo:

$$H(x|y) = -E\{\ln[p_{x|y}(x|y)]\}, \quad (2.11)$$

onde

$x, y$  : variáveis aleatórias;

$p_{x|y}(x|y)$  : pdf de  $x$  condicionada a  $y$ .

Com base na teoria da informação, é sabido que uma variável gaussiana possui maior entropia se comparada a outra variável aleatória não-gaussiana de mesma variância. Logo, a entropia pode ser usada como uma medida de não-gaussianidade para variáveis aleatórias.

### 2.3.3. Informação Mútua

A informação mútua de duas variáveis aleatórias  $x$  e  $y$  representa a informação contida em  $x$  após a observação de  $y$ . Essa grandeza é definida como

$$I(x, y) = H(x) - H(x|y). \quad (2.12)$$

Alternativamente, podemos definir a informação mútua utilizando a função conhecida como divergência de Kullback-Leibler [44]

$$I(x, y) = \mathcal{D}(p_{x,y}(x, y) | p_x(x) \cdot p_y(y)), \quad (2.13)$$

onde

$$\mathcal{D}(p_{x,y}(x, y) | p_x(x) \cdot p_y(y)) = \int p_{x,y}(x, y) \cdot \log \frac{p_{x,y}(x, y)}{p_x(x) \cdot p_y(y)} dx dy.$$

A divergência de Kullback-Leibler pode ser interpretada como a distância entre as densidades de probabilidade das variáveis. No entanto, ela não corresponde a uma distância real, uma vez que não é simétrica.

Determinar a informação mútua entre variáveis aleatórias é de grande importância para a ICA, pois esta é sempre não-negativa e só será nula quando as variáveis (no caso,  $x$  e  $y$ ) forem independentes. Assim sendo, podemos utilizá-la como referência para avaliar a independência estatística de variáveis aleatórias.

Quando lidamos com problemas envolvendo separação cega de fontes, temos que maximizar a informação mútua das fontes é equivalente a maximizar a entropia das mesmas.

#### 2.3.4. Negentropia

Negentropia é uma abreviação para “Entropia Negativa” (do inglês, *Negentropy* – *Negative Entropy*). Para uma variável aleatória  $x$ , ela é assim definida:

$$J(x) = H(x_{gauss}) - H(x) \quad (2.14)$$

onde

$x_{gauss}$  : variável aleatória gaussiana com mesma variância que a variável  $x$ .

A negentropia é uma ótima medida de não-gaussianidade, porém possui alto custo computacional, uma vez que se faz necessário o conhecimento (ou ao menos uma estimativa) da função densidade de probabilidade da variável aleatória [46].

Uma solução para este problema acima é estimar a pdf através de aproximações, sendo as mais comuns baseadas na expansão de Edgeworth e na expansão de Gram-Charlier [47]. Dessa forma, é possível medir a não-gaussianidade de uma variável aleatória  $x$  a partir da aproximação abaixo [26]:

$$J(x) \approx \frac{1}{12} E\{x^3\}^2 + \frac{1}{48} kurt(x^2). \quad (2.15)$$

Devido ao fato desta aproximação utilizar a curtose como parâmetro, tal medida é também sensível a *outliers*.

Alem da aproximação mostrada, existem outras abordagens mais complexas para o cálculo da negentropia, como a proposta por Hyvärinen [26] conhecida como método da máxima entropia.

## 2.4. Funções de Contraste

Função de contraste é a definição dada ao conjunto de funções utilizadas como critérios de otimização, algumas das quais atingem seu mínimo somente quando a separação total das fontes é atingida [15].

Uma função  $f$  para ser considerada como uma função contraste para uma variável aleatória  $y$  deve atender às seguintes condições:

1.  $f(\mathbf{p}_y)$  é invariante a permutações, logo:

$$f(\mathbf{p}_{P,y}) = f(\mathbf{p}_y), \text{ para qualquer matriz de permutação } \mathbf{P};$$

2.  $f(\mathbf{p}_y)$  é invariante a mudanças de escala, então:

$$f(\mathbf{p}_{D,y}) = f(\mathbf{p}_y), \text{ para qualquer matriz diagonal } \mathbf{D};$$

3. Se  $y$  possui componentes independentes, então:

$$f(\mathbf{p}_{W,y}) \leq f(\mathbf{p}_y), \text{ para qualquer matriz inversível } \mathbf{W};$$



4. A igualdade na condição 3 deve ser respeitada se, e somente se, a matriz  $W$  for uma matriz de permutação generalizada, ou seja:

$W = P.D$ , onde  $P$  é uma matriz de permutação e  $D$  é uma matriz diagonal inversível.

Para uma função ser considerada uma função de contraste, basta que as condições 1, 2 e 3 sejam verdadeiras. Porém no contexto de BSS, a condição 4 deve ser respeitada de forma que uma solução correta seja encontrada. Uma função de contraste que satisfaça a última condição é definida como discriminante.

## 2.5. Outras abordagens para BSS:

### 2.5.1. Bayesiana

A abordagem bayesiana no contexto de BSS propõe uma possível solução para dois problemas bastante comuns quando se lida com separação cega de fontes via ICA em ambientes reais [48]:

- Garantir que as fontes que compõe as misturas sejam estatisticamente independentes entre si;
- A dificuldade de incorporar informações *a priori* ao modelo de ICA.

Essa abordagem consiste em obter estimativas das fontes e da matriz de mistura a partir da maximização da função densidade de probabilidade *a posteriori* -  $p(\mathbf{A}, \mathbf{s} | \mathbf{x})$  - dada por [49]

$$p(\mathbf{A}, \mathbf{s} | \mathbf{x}) \propto P(\mathbf{x} | \mathbf{A}, \mathbf{s}) \cdot P(\mathbf{A}) \cdot P(\mathbf{s}),$$

(2.16)

onde

$P(\mathbf{x} | \mathbf{A}, \mathbf{s})$ : função de verossimilhança dos dados;

$P(\mathbf{A})$ : função densidade de probabilidade da matriz de mistura;

$P(\mathbf{s})$ : função densidade de probabilidade das fontes.

Outra vantagem desta abordagem é a capacidade de incorporar o ruído ao modelo de separação, bem como, a sua aplicação na separação de misturas sub-determinadas [50,51].

### **2.5.2. Análise de Componentes Esparsos**

A Análise de Componentes Esparsos (SCA – *Sparse Component Analysis*) é uma ferramenta para solução de problemas em BSS, especialmente quando lidamos com sistemas sub-determinados ( $m < n$ ) [52,53].

Essa técnica baseia-se na hipótese da esparsidade das fontes, isto é, assume-se que na maior parte do tempo, as fontes assumem valores próximos a zero [54]. Esta restrição é típica em sinais de voz e instrumentos musicais.

Assumindo que, em determinados intervalos de tempo, apenas uma das fontes está ativa, pode-se estimar a coluna da matriz de mistura para a fonte ativa neste determinado instante, visto que esta possui a mesma direção do vetor de misturas. Esse processo pode ser implementado utilizando técnicas de clusterização [55].

Os métodos mostrados anteriormente foram desenvolvidos para serem aplicados diretamente a misturas instantâneas. Entretanto, a maioria das misturas no mundo real são convolutivas, logo se faz necessário estender o conceito de BSS como será visto no próximo capítulo.

### 3. Separação Cega de Fontes para Misturas Convolutivas

Conforme visto no Capítulo 2, a análise de componentes independentes (ICA) é a ferramenta principal para solucionarmos o problema de separação cega de fontes, uma vez que assumimos que as fontes em questão são independentes entre si. Nos problemas que lidam com misturas instantâneas, os algoritmos de ICA podem ser aplicados diretamente para a separação de tais misturas. Entretanto, a maioria das misturas no mundo real são convolutivas, sendo misturadas por meio de atrasos de propagação e reverberações no ambiente. Logo, visando obter uma solução para o problema de BSS para este tipo de misturas, faz-se necessário estender o conceito de BSS e ICA.

Existem três abordagens fundamentais para quando as misturas das quais se pretende extrair as fontes são convolutivas [29]. Estas são compostas por algoritmos que utilizam o domínio do tempo, domínio da frequência ou banco de filtros para realizar a separação cega das fontes de interesse.

#### 3.1. Domínio do tempo

Neste caso, a ICA é aplicada diretamente às misturas convolutivas [56]. De forma a obter as estimativas das fontes independentes a partir da observação das misturas, consideremos dois tipos de rede no domínio do tempo: a *feedforward* e a *feedback*, como descrito a seguir.

##### 3.1.1. Feedforward

Uma das arquiteturas utilizadas na representação de redes no domínio do tempo é a arquitetura feedforward. Ela pode ser assim representada:

$$s_i(k) = \sum_{j=1}^m \sum_{\tau=0}^L w_{ij}(\tau) \cdot x_j(k - \tau) \quad (3.1)$$

onde

$w_{ij}$ : filtros adaptativos;

$s_i$ : fontes independentes;

$x_j$ : misturas observadas.

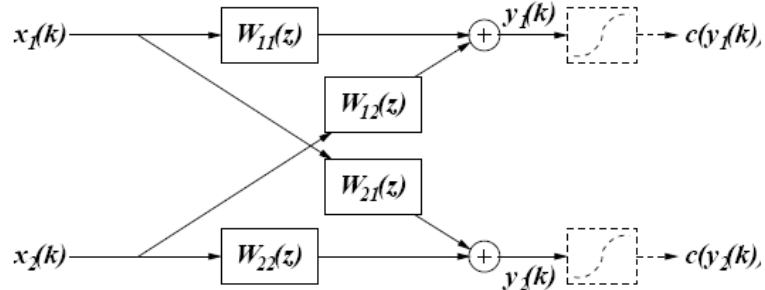


Figura 3.1 - Rede *feedforward* 2x2 para ICA de misturas convolutivas

### 3.1.2. Feedback

Uma rede do tipo *feedback* para a separação cega de fontes no domínio do tempo pode ser assim definida:

$$s_i(k) = \sum_{\tau=0}^L w_{ij}(\tau) \cdot x_j(k - \tau) + \sum_{j=1}^m \sum_{\tau=1}^L w_{ij}(\tau) \cdot s_j(k - \tau). \quad (3.2)$$

A arquitetura *feedback* é constituída de três coeficientes para os filtros distintos: *zero-delay* nos filtros diretos  $w_{ii}(0)$ , outro peso nos filtros diretos  $w_{ii}(k)$ , para  $k \neq 0$ , e outro coeficiente nos filtros cruzados  $w_{ij}(k)$ , para  $i \neq j$ .

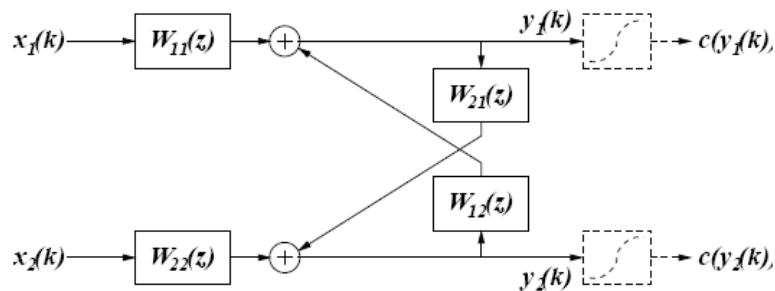


Figura 3.2 - Rede *feedback* 2x2 para ICA de misturas convolutivas

A vantagem do sistema baseado na arquitetura *feedforward* é que este pode utilizar um sistema para inversão mais genérico, visto que este utiliza uma aproximação para o algoritmo de ICA de fase-mínima.

O resultado deste método é satisfatório, desde que o algoritmo convirja, porém seu custo computacional é alto.

### **3.2. Domínio da frequência**

A proposta de um modelo no domínio da frequência visa ser uma alternativa para o alto custo computacional existente no modelo baseado no domínio do tempo. No método para BSS no domínio da frequência, primeiramente as misturas convolutivas são convertidas para o domínio da frequência e, em seguida, o algoritmo de ICA é aplicado para cada faixa de frequência amostrada, que agora podem ser consideradas uma mistura instantânea [57].

A solução do problema de BSS para misturas convolutivas baseada no domínio da frequência é um tanto simples do ponto de vista computacional, porém os problemas de ambiguidade de escala e de permutação, que não são significantes quando se lida com misturas instantâneas, são de grande importância, neste caso. Isso deve-se ao fato de que, ao aplicarmos o algoritmo de ICA para cada faixa de frequência separadamente, a ordem e a escala de cada um dos sinais obtidos são aleatórias. Logo, ao retornarmos ao domínio do tempo, componentes de uma mesma faixa de frequência podem não ser oriundas da mesma fonte ou podem não possuir a escala ideal [58]. Além deste problema, a convolução no domínio do tempo só equivale a produto no domínio da frequência quando é circular; na prática, porém, as convoluções são lineares. Isso significa que o número de raios a empregar deve ser bem maior do que o comprimento dos filtros de mistura, de forma a podermos aproximar uma convolução linear por uma circular.

Atualmente, existem diversas abordagens para solucionar o problema da ambiguidade de escala e permutação [59,60], porém estas não serão tema de estudo deste projeto.

### **3.3. Banco de Filtros**

Essa abordagem utiliza-se de ambos os domínios, tempo e frequência, para realizar a separação das fontes [61,62]. Nela, os coeficientes dos filtros são atualizados no domínio da frequência, ao passo que as funções não-lineares utilizadas como

medidas para a independência dos sinais são aplicadas no domínio do tempo. Este método elimina o problema de permutação, porém há um aumento considerável no tempo de execução, visto que é necessária a troca de domínios durante a execução do algoritmo.

### 3.4. Algoritmo Proposto

O algoritmo implementado neste projeto visa solucionar o problema de separação cega de fontes de áudio convolutivas utilizando para tal fim um modelo estendido de ICA no domínio do tempo [63,64].

Como visto anteriormente, uma aplicação típica do problema de interesse é o problema *Cocktail Party*, onde o sinal proveniente de cada indivíduo deve ser extraído a partir de um conjunto de misturas obtidas em um mesmo ambiente e que contém tais sinais de interesse.

O processo de mistura pode ser descrito como o sistema MIMO (*Multiple-Input and Multiple-Output*) descrito abaixo:

$$\mathbf{x}_i(k) = \sum_{j=1}^d \sum_{\tau=0}^{M_{ij}} h_{ij}(\tau) s_j(k - \tau), \quad (3.3)$$

onde

$\mathbf{x}_1(k), \mathbf{x}_2(k), \dots, \mathbf{x}_m(k)$  : misturas das fontes;

$s_1(k), s_2(k), \dots, s_d(k)$  : fontes independentes;

$h_{ij}$  : resposta das fontes ao impulso;

$M_{ij}$  : comprimento dos filtros de mistura.

Uma vez de posse do número de amostras disponíveis, o objetivo da separação cega torna-se encontrar o filtro FIR MIMO inverso da Eq. (3.3) visando obter os sinais originais da melhor maneira possível. As estimativas das fontes são dadas por:

$$\hat{s}_i(\mathbf{k}) = \sum_{j=1}^m \sum_{\tau=0}^L w_{ij}(\tau) \cdot x_j(\mathbf{k} - \tau), \quad (3.4)$$

onde

$w_{ij}(\mathbf{k})$ :  $n = 0, 1, \dots, L$ ;

$L$ : tamanho do filtro de separação.

O algoritmo implementado é esquematizado em 4 etapas, descritas a seguir, usando como exemplo duas fontes e duas observações (misturas), sem perda de generalização.

1. Aplica-se um algoritmo de ICA ao subespaço  $m.L$  dado por:

$$\mathbf{x}(\mathbf{k}) = [\mathbf{x}_1(\mathbf{k}), \dots, \mathbf{x}_1(\mathbf{k} - L + 1), \mathbf{x}_2(\mathbf{k}), \dots, \mathbf{x}_2(\mathbf{k} - L + 1)]^T, \quad (3.5)$$

onde  $\mathbf{x}_1(\mathbf{k})$  e  $\mathbf{x}_2(\mathbf{k})$  são as amostras no tempo dos sinais observados.

O resultado deste processo são componentes independentes referentes ao  $m.L$  filtros MISO de comprimento  $L$ , selecionados de forma que os sinais de saída sejam independentes.

2. As componentes são agrupadas através de um algoritmo de clusterização, uma vez que é assumido que as componentes de saída obtidas são versões filtradas das fontes originais.
3. Utilizando um processo de reconstrução nas componentes de cada *cluster*, obtemos as respostas individuais.
4. Finalmente, aplica-se um filtro *delay-and-sum beamformer* nas respostas a fim de obterem-se as estimativas das fontes originais.

### **Etapa 1: Decomposição utilizando ICA**

Em geral, as componentes independentes obtidas utilizando o algoritmo de ICA possuem suas interferências espaço-temporal mútuas canceladas o máximo possível.

Considerando um caso ideal, tais componentes são cópias filtradas das fontes independentes  $s_1(k), s_2(k), \dots, s_m(k)$ .

O algoritmo de ICA utilizado neste projeto foi o EFICA (*Extended Fast ICA*) [65].

## Etapa 2: Clusterização

A idéia principal do algoritmo implementado é considerar cada componente independente, encontrada na decomposição via ICA, como uma versão filtrada de uma das  $m$  fontes originais. Assim, cada componente independente obtida poderá ser agrupada em  $m$  *clusters*. Diante desta idéia, devemos definir uma medida para avaliar a similaridade entre as componentes encontradas.

Seja  $c_j(k)$  a  $j$ -ésima componente onde,  $j = 1, \dots, m.L$ . A medida de similaridade entre as  $i$ -ésima e  $j$ -ésima componentes é definida por

$$D_{ij} = \hat{E}[P_i \cdot c_j(k)]^2 + \hat{E}[P_j \cdot c_i(k)]^2, \quad (3.6)$$

onde

$P_i$ : projetor do subespaço  $c_i(k-L+1), \dots, c_i(k+L-1)$ ;

$\hat{E}[\ ]$ : é operador de média.

O operador projeção é dado por:

$$P_i = I - C_i(C_i^T \cdot C_i)^{-1} \cdot C_i^T, \quad (3.7)$$

onde

$C_i$ : matriz contendo as versões atrasadas de  $c_i$ ;

A matriz  $D_{ij}$  é utilizada como referência para um *cluster* hierárquico com uma abordagem baseada na média, ou seja, a cada iteração, o número de *clusters* é reduzido de um através da fusão dos dois *clusters* com a menor distância mútua. Tal distância é



obtida por meio da média das distâncias entre as componentes individuais de cada *cluster*.

O algoritmo é encerrado quando se atinge o estado onde o número de *clusters* é igual ao número de fontes independentes ( $m$ ).

### Etapa 3: Reconstrução

A etapa de reconstrução visa transformar as componentes de cada *cluster* em respostas. Assumindo que qualquer componente (em qualquer *cluster*) pode contribuir na reconstrução de quaisquer fontes, temos

$$\mathbf{x}^i = \mathbf{W}^{-1} \text{diag}[\lambda_1^i, \dots, \lambda_{m.L}^i], \quad (3.8)$$

onde  $\lambda_l^i$  é peso para a reconstrução da  $i$ -ésima fonte utilizando a  $l$ -ésima componente, com  $i = 1, \dots, d$  e  $l = 1, \dots, m.L$ .

Uma vez que a estrutura do vetor  $\mathbf{x}_i^j$  é similar ao de  $\mathbf{x}$ , podemos obter as respostas a partir de  $\mathbf{x}_i^j$ :

$$\hat{s}_i^j(k) = \sum_{\tau=1}^{L+1} \hat{x}_{(i-1),(L+1)+\tau}^j(k + \tau - 1). \quad (3.9)$$

No projeto aqui implementado, utilizamos como referência para o peso das componentes, uma lógica binária. Logo,  $\lambda$  foi definido como 0 ou 1, ou seja, a componente  $\lambda$  do *cluster* contribui ou não para o sinal de saída correspondente.

## 4. Resultados

O presente capítulo trata da apresentação e análise dos resultados obtidos para o algoritmo implementado. Como forma de avaliar a performance do mesmo em relação às demais implementações existentes, também serão apresentados os resultados para um método de separação cega de fontes para misturas convolutivas utilizando o domínio do frequência.

A eficiência do algoritmo foi realizada através da comparação das fontes originais e suas respectivas estimativas [66,67]. Tal análise leva em consideração as diversas distorções que podem ocorrer com os sinais estimados, como interferências, artefatos e ruído.

O presente trabalho utilizou abordagem proposta em [28] que consiste em projeções numéricas dos sinais estimados em subespaços, onde cada subespaço é composto por cópias atrasadas das fontes originais. Cada estimativa da fonte  $\hat{s}_i^j(k)$  pode ser decomposta em

$$\hat{s}_i^j(k) = s_{target}(k) + e_{interf}(k) + e_{noise}(k) + e_{artif}(k), \quad (4.1)$$

onde

$s_{target}(k)$ : sinal original com as distorções permitidas;

$e_{interf}(k)$ : erro de interferência;

$e_{noise}(k)$ : erro de ruído;

$e_{artif}(k)$ : erro de artefato.

A avaliação do desempenho do algoritmo implementado foi feita a partir de três medidas padrão no campo de separação cega de fontes: SAR, SIR, SDR.

Neste projeto, nenhum erro de ruído foi considerado, portanto as relações de distorção são dadas da seguinte forma:

- SDR (*Signal to Distortion Ratio*):

$$SDR = 10 \cdot \log_{10} \frac{\|s_{target}\|^2}{\|e_{interf} + e_{noise} + e_{artif}\|^2} \quad (4.2)$$

- SIR (*Signal to Interference Ratio*):

$$SIR = 10 \cdot \log_{10} \frac{\|s_{target}\|^2}{\|e_{interf}\|^2} \quad (4.3)$$

- SAR (*Signal to Artifact Ratio*):

$$SAR = 10 \cdot \log_{10} \frac{\|s_{target} + e_{interf} + e_{noise}\|^2}{\|e_{artif}\|^2} \quad (4.4)$$

A medida SIR expressa o montante de interferência presente nas estimativas do método. A medida SDR significa o quão distorcidas as fontes encontram-se nas saídas. Retirando-se as interferências e distorções, há a inserção de artefatos nas estimativas, particularmente nas técnicas de separação no domínio da frequência. O total de artefatos é expresso na medida SAR.

Uma vez que a performance do algoritmo implementado depende do tempo de execução deste e a proposição do mesmo é utilizar um reduzido conjunto de amostras das fontes de misturas para realizar a separação das fontes originais e, em seguida, propagar tais resultados para o restante do sinal, primeiramente avaliamos o tamanho ideal do bloco a ser tomado.

Para realizarmos tal definição, além das medidas padrões acima enunciadas, levamos também em consideração o tempo gasto na execução do algoritmo no Matlab.

Visando padronizar as medidas obtidas, todas as fontes analisadas foram amostradas a 16kHz compostas por vozes masculinas e femininas. O algoritmo foi executado em um Pentium Dual Core Duo 2.6GHz com 2Gb de memória RAM.

As misturas analisadas foram obtidas por meio da convolução de duas das fontes utilizadas com filtros aleatórios gerados pelo Matlab. Os valores destes filtros estão disponíveis para todos os testes aqui apresentados.

#### **4.1. Análise 1: Definição do tamanho ideal do bloco para separação**

Definir o tamanho do bloco de separação a ser utilizado pelo algoritmo é de extrema importância quando lidamos com métodos baseados no domínio do tempo, como o aqui apresentado.

Para definirmos o tamanho ideal do bloco a ser empregado nas demais análises, utilizamos, para a convolução das fontes, filtros de mistura de comprimento 2 ( $M = 2$ ). Os filtros de separação também possuíam o mesmo comprimento ( $L = 2$ ).

As condições utilizadas em todos os testes desta análise foram:

- $M = 2$
- $L = 2$
- Filtros de mistura

$$h\{1,1\} = [ 0.4282 \quad 0.8956 ]$$

$$h\{2,1\} = [ 0.0403 \quad 0.6771 ]$$

$$h\{1,2\} = [ 0.7310 \quad 0.5779 ]$$

$$h\{2,2\} = [ 0.5689 \quad -0.2556 ]$$

### 4.1.1. Teste 1: Bloco com 2000 amostras

- Samples = 2000

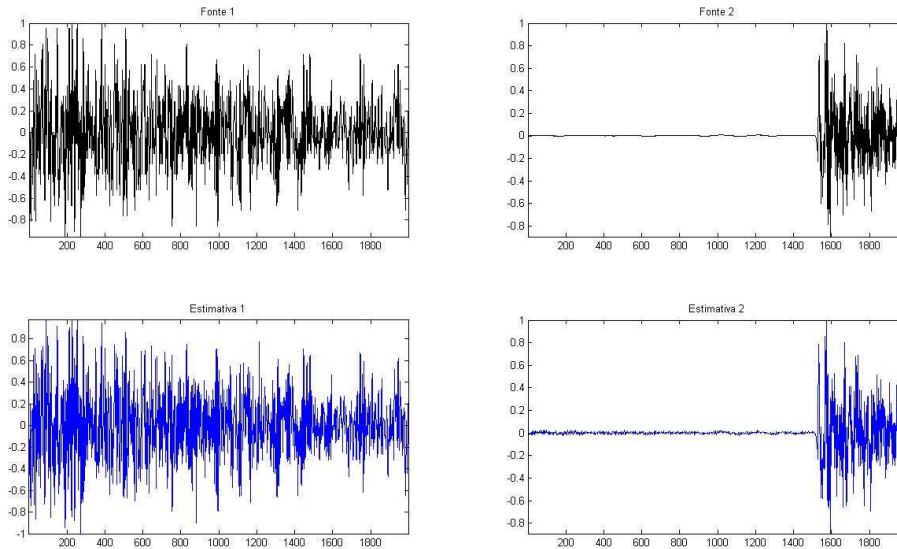


Figura 4.1 – Análise 1/Teste 1: fontes e estimativas para bloco de 2000 amostras

Cluster (1): 1 / 2 / 4

Cluster (2): 3

	SDR	SIR	SAR
Estimativa 1	16.1666	26.2855	16.6213
Estimativa 2	7.8381	15.3353	8.8144

Tabela 4.1 - Análise 1/Teste 1: resultados para bloco de 2000 amostras

Primeiramente, com um bloco com um número pequeno de amostras, vemos na Fig. 4.1 que, visualmente, o resultado é satisfatório. A fonte 2 durante as 1500 primeiras amostras é praticamente nula, porém sua estimativa apresentou um pequeno ruído neste intervalo.

As medidas de comparação das fontes não apresentaram resultados uniformes, uma vez que variaram muito os resultados de cada uma das fontes.

Uma explicação para tal é que uma mistura pode conter mais informação acerca de uma fonte que das demais. Além disso, os filtros de mistura interferem na qualidade da separação, assim como a inicialização da matriz de separação.

#### 4.1.2. Teste 2: Bloco com 3000 amostras

- Samples = 3000

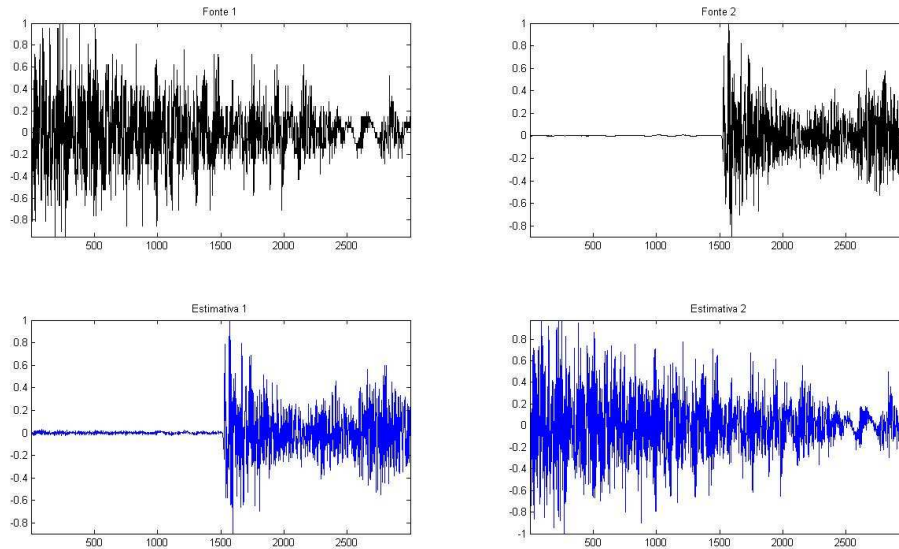


Figura 4.2 - Análise 1/Teste 2: fontes e estimativas para bloco de 3000 amostras

Cluster (1): 1

Cluster (2): 2 / 3 / 4

	<b>SDR</b>	<b>SIR</b>	<b>SAR</b>
<b>Estimativa 1</b>	15.9118	26.3967	16.3286
<b>Estimativa 2</b>	9.0281	17.8754	9.7052

Tabela 4.2 - Análise 1/Teste 2: resultados para bloco de 3000 amostras

Elevando o número de amostras do bloco utilizado, era esperado que a separação cega das fontes apresentasse resultados melhores que o teste anterior, visto que há maior quantidade de informação disponível.

De fato, os resultados obtidos foram mais satisfatórios que para um bloco com 2000 amostras. Entretanto, as medidas de SDR, SIR e SAR em relação as estimativas de cada fonte ainda apresentaram valores um tanto quando discrepantes.

Diferentemente do teste anterior, ocorreu permutação entre as fontes e suas estimativas. Vale lembrar que tal fenômeno é totalmente aleatório.

### 4.1.3. Teste 3: Bloco com 4000 amostras

- Samples = 4000

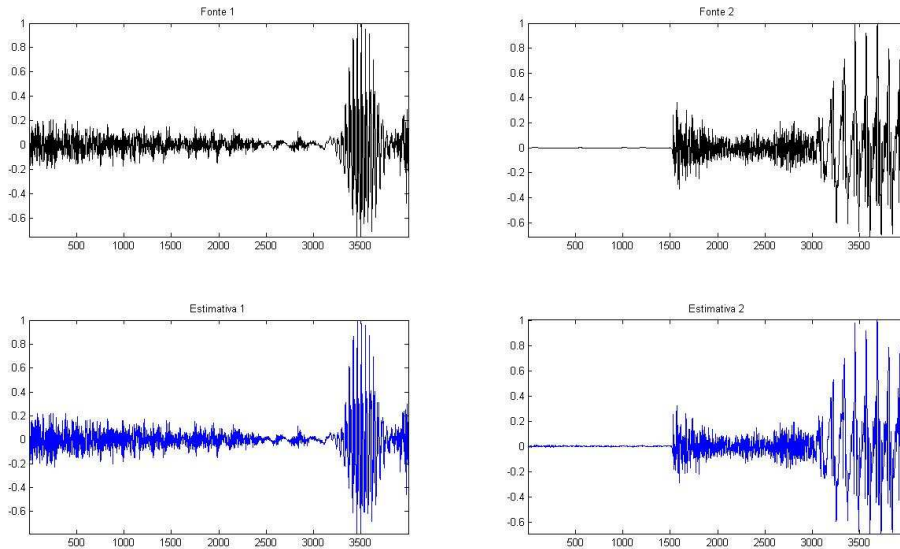


Figura 4.3 - Análise 1/Teste 3: fontes e estimativas para bloco de 4000 amostras

Cluster (1): 1 / 2 / 4

Cluster (2): 3

	<b>SDR</b>	<b>SIR</b>	<b>SAR</b>
<b>Estimativa 1</b>	20.7139	28.6089	21.4893
<b>Estimativa 2</b>	17.7886	26.5533	18.4175

Tabela 4.3 - Análise 1/Teste 3: resultados para bloco de 4000 amostras

Com o tamanho do bloco de amostras igual a 4000, os resultados obtidos foram superiores ao esperado, sendo SIR acima de 20dB, o que é o considerado muito bom pela literatura. As demais medidas apresentaram valores na faixa de 20dB para ambas as fontes, o que também pode ser considerado bom. Visualmente, as estimativas foram bem semelhantes às fontes originais.

A melhora no desempenho do algoritmo se deve ao fato de termos mais amostras no bloco para realizarmos a separação.

#### 4.1.4. Teste 4: Bloco com 5000 amostras

- Samples = 5000

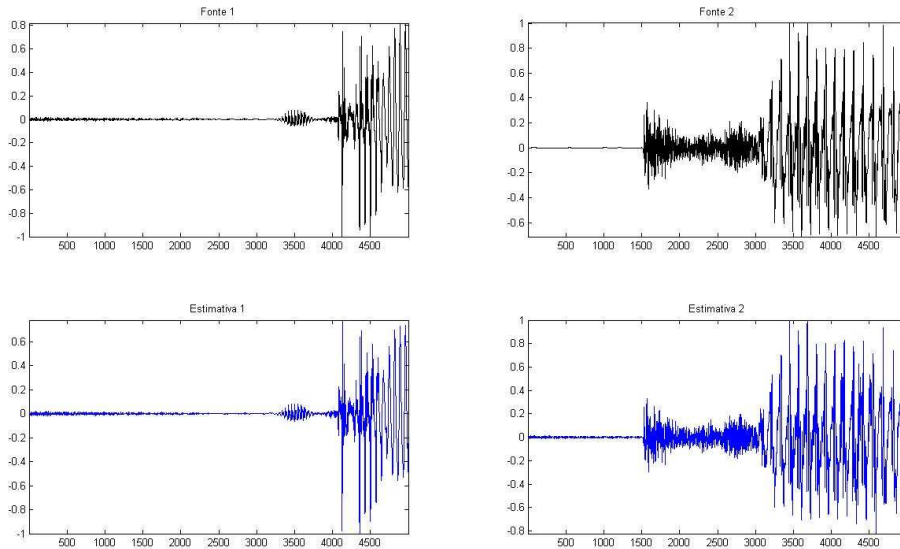


Figura 4.4 - Análise 1/Teste 4: fontes e estimativas para bloco de 5000 amostras

Cluster (1): 1 / 2 / 4

Cluster (2): 3

	SDR	SIR	SAR
Estimativa 1	22.7402	29.5128	23.7702
Estimativa 2	14.2605	16.1008	18.9823

Tabela 4.4 - Análise 1/Teste 4: resultados para bloco de 5000 amostras

Os resultados foram bastante semelhantes aos obtidos com 4000 amostras. Os valor de SDR, SIR e SAR oscilaram bastante entre as estimativas, o que nos leva a crer que para blocos deste tamanho e para estas misturas, o algoritmo não conseguiu realizar a separação de forma eficiente.

O tempo de execução foi muito maior, o que torna inviável a utilização prática do algoritmo com esta configuração.



## **4.2. Análise 2: Eficiência do algoritmo frente à complexidade da mistura**

Diante dos resultados obtidos na análise anterior e tendo em vista uma relação de custo-benefício, definiu-se o tamanho do bloco de amostras para as futuras análises de desempenho do algoritmo como 4000 amostras.

Dando seqüência aos testes de avaliação do algoritmo, comparamos a performance deste com outro também desenvolvido para realizar a separação cega de fontes convolutivas, porém com uma abordagem no domínio da frequência.

Primeiramente, os filtros de mistura foram de tamanho 2, assim como os de separação. Este valor foi aumentado gradativamente e os resultados obtidos são mostrados a seguir.

Nos resultados apresentados nos testes de 1 a 7, o algoritmo 1 (azul) refere-se ao método proposto e o algoritmo 2 (vermelho) refere-se ao método na frequência utilizado para comparação.

### 4.2.1. Teste 1: Mistura de complexidade 2

- $M = 2$
- $L = 2$
- Filtros de mistura:

$$h\{1,1\} = [ -0.4326 \quad -1.6656 ]$$

$$h\{2,1\} = [ -1.1465 \quad 1.1909 ]$$

$$h\{1,2\} = [ 0.1253 \quad 0.2877 ]$$

$$h\{2,2\} = [ 1.1892 \quad -0.0376 ]$$

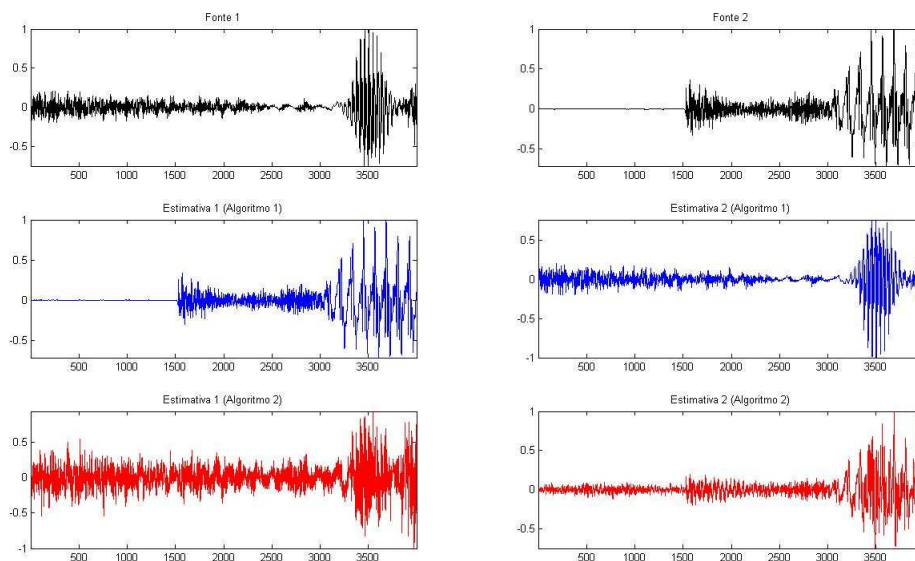


Figura 4.5 - Análise 2/Teste 1: fontes e estimativas ( $M = 2 / L = 2$ )

Cluster (1): 1 / 2 / 4

Cluster (2): 3

Medidas	SDR	SIR	SAR
	Algoritmo 1		
Estimativa 1	16.1210	24.9052	16.7514
Estimativa 2	17.3863	26.1511	18.0161
	Algoritmo 2		
Estimativa 1	4.2093	4.2093	12.2570
Estimativa 2	3.1699	3.4410	16.9727

Tabela 4.5 - Análise 2/Teste 1: resultados ( $M = 2 / L = 2$ )

Visualmente, as fontes e as estimativas obtidas pelo algoritmo proposto são bastante semelhantes. Ocorreu a permutação entre as estimativas (como previamente enunciado). Outro resultado interessante é que ocorreu a inversão de fase para a estimativa da fonte 1 (estimativa 2).

O método na frequência obteve resultados muito abaixo dos esperados, tanto visualmente quanto o resultado medido por meio de SDR, SIR e SAR.

### 4.2.2. Teste 2: Mistura de complexidade 3

- $M = 3$
- $L = 3$
- Filtros de mistura

$$\begin{aligned}
 h\{1,1\} &= [ 0.3312 \quad 0.4557 \quad 0.0528 ] \\
 h\{2,1\} &= [ 0.8603 \quad -0.2706 \quad 1.0250 ] \\
 h\{1,2\} &= [ -0.0652 \quad -1.7503 \quad -0.3189 ] \\
 h\{2,2\} &= [ -0.6977 \quad -0.6218 \quad 0.1558 ]
 \end{aligned}$$

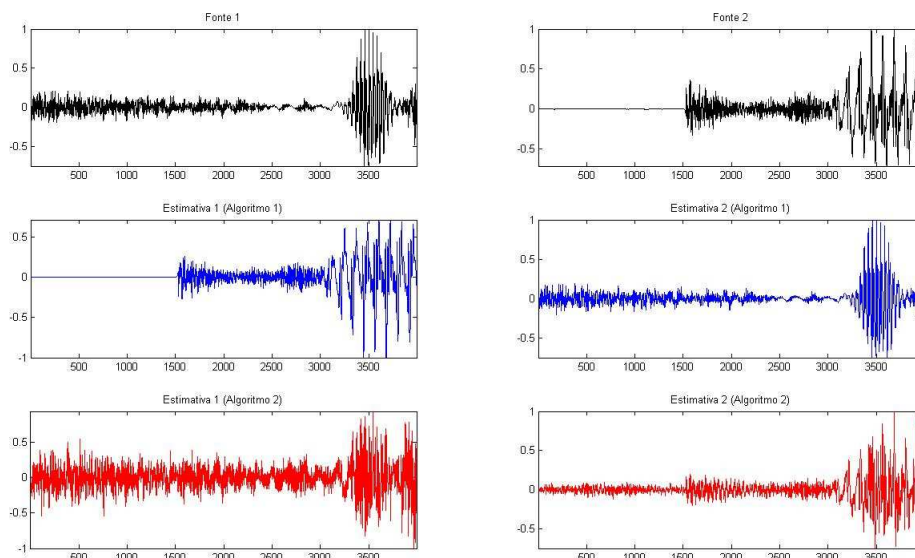


Figura 4.6 - Análise 2/Teste 3: fontes e estimativas ( $M = 3 / L = 3$ )

Cluster (1): 1 / 5 / 3 / 4 / 6

Cluster (2): 2

Medidas	SDR	SIR	SAR
	Algoritmo 1		
Estimativa 1	14.4209	21.9880	15.2843
Estimativa 2	14.5574	23.6891	15.1416
	Algoritmo 2		
Estimativa 1	4.2207	5.2163	12.2482
Estimativa 2	3.1254	3.3962	16.9465

Tabela 4.6 - Análise 2/Teste 2: resultados ( $M = 3 / L = 3$ )

Os resultados obtidos com o algoritmo proposto foram muito bons, ficando acima da média dos demais algoritmos da literatura ( $SIR > 20\text{dB}$ ).

Observando as formas de onda, são bem semelhantes às estimativas e suas respectivas fontes. Mais uma vez, ocorreu a permutação entre as fontes e suas estimativas e também, neste caso, a inversão de fase para a estimativa da fonte 2 (estimativa 1).

Os resultados referentes ao algoritmo 2 ficaram muito aquém dos obtidos utilizando o método proposto para esse tamanho de filtros. Vale ressaltar que os valores obtidos foram bastante próximos dos encontrados para  $M=2$ .

Um fato que merece destaque é a divisão das componentes entre os clusters, uma vez que o cluster (1) possui 5 componentes, quando o cluster (2) apenas 1. Tal divisão assimétrica dos clusters é válida de acordo com o proposto pelo algoritmo.

### 4.2.3. Teste 3: Mistura de complexidade 5

- $M = 5$
- $L = 5$
- Filtros de mistura

$$\begin{aligned}
 h\{1,1\} &= [ 1.0224 \quad -1.8328 \quad 0.8050 \quad 0.3776 \quad -1.4774 ] \\
 h\{2,1\} &= [ -0.3923 \quad -1.5860 \quad 0.8528 \quad 2.4671 \quad 0.0099 ] \\
 h\{1,2\} &= [ 0.7943 \quad -0.0300 \quad -0.7470 \quad 0.4879 \quad -0.0304 ] \\
 h\{2,2\} &= [ 0.6947 \quad -1.1884 \quad -0.0054 \quad 0.6921 \quad -2.6864 ]
 \end{aligned}$$

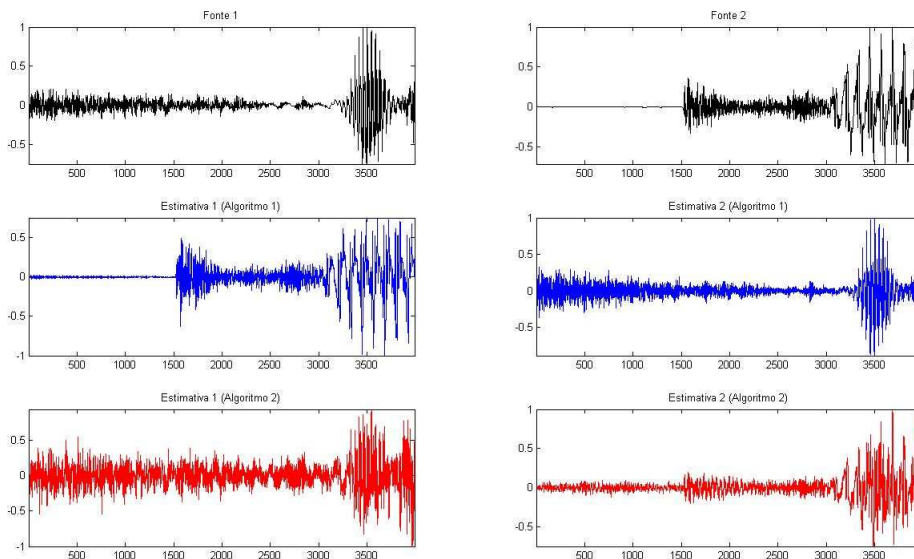


Figura 4.7 - Análise 2/Teste 3: fontes e estimativas ( $M = 5 / L = 5$ )

Cluster (1): 1 / 3 / 9 / 2 / 10 / 4 / 5 / 6

Cluster (2): 7 / 8

Medidas	SDR	SIR	SAR
	Algoritmo 1		
Estimativa 1	7.0840	18.1320	7.5056
Estimativa 2	14.9930	25.4520	15.4148
	Algoritmo 2		
Estimativa 1	4.2654	5.2356	12.3882
Estimativa 2	3.1228	3.3940	16.9397

Tabela 4.7 - Análise 2/Teste 3: resultados ( $M = 5 / L = 5$ )

Visualmente, os sinais proveniente das fontes e suas estimativas parecem bem próximos, com ressalva à amplificação da estimativa 1 (fonte 2). Tal fato é previsto pela abordagem deste trabalho.

Ocorreu a permutação entre as estimativas, bem como a inversão de fase para a estimativa da fonte 1 (estimativa 2).

O algoritmo 2, utilizado como comparação para o algoritmo proposto, apresentou resultados insatisfatórios. Seus valores continuam bem próximos dos obtidos anteriormente, o que indica que este é pouco sensível a variações pequenas de  $M$ .

#### 4.2.4. Teste 4: Mistura de complexidade 8

- $M = 8$
- $L = 8$
- Filtros de Mistura

$$\begin{aligned}
 h\{1,1\} &= [ 1.7714 \quad 0.2426 \quad 0.2374 \quad 0.4851 \quad -0.3086 \quad -1.1836 \quad -0.1157 \quad -2.2207 ] \\
 h\{2,1\} &= [ 0.5178 \quad -0.1019 \quad -1.2002 \quad -1.1209 \quad 0.6537 \quad -1.2232 \quad 0.5980 \quad 0.1930 ] \\
 h\{1,2\} &= [ -0.8506 \quad -0.3156 \quad 0.6612 \quad 0.7830 \quad 0.6827 \quad 0.7135 \quad 0.8487 \quad 0.8778 ] \\
 h\{2,2\} &= [ -0.2288 \quad 1.0324 \quad 0.5256 \quad 1.4577 \quad 1.8859 \quad -0.4944 \quad -0.3187 \quad 0.2838 ]
 \end{aligned}$$

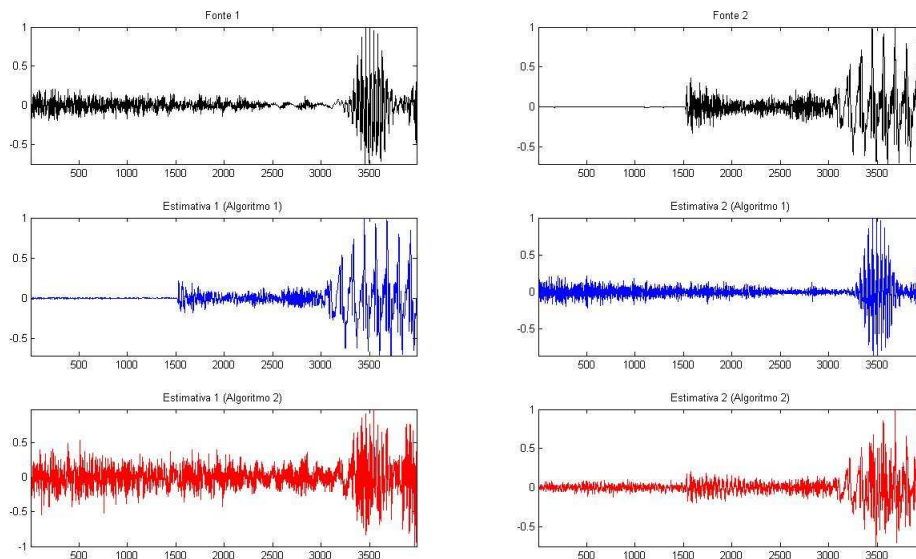


Figura 4.8 - Análise 2/Teste 4: fontes e estimativas ( $M = 8 / L = 8$ )

Cluster (1): 1 / 16 / 5 / 10 / 13 / 6 / 11 / 2 / 12 / 8 / 4 / 15 / 7 / 9

Cluster (2): 3 / 14

Medidas	SDR	SIR	SAR
	Algoritmo 1		
Estimativa 1	5.0037	11.9683	6.2467
Estimativa 2	9.9265	17.1909	10.9117
	Algoritmo 2		
Estimativa 1	4.2893	5.2585	12.4113
Estimativa 2	3.1428	3.4265	16.7588

Tabela 4.8 - Análise 2/Teste 4: resultados ( $M = 8 / L = 8$ )

Para misturas um pouco mais complexas, verificamos que a eficiência do algoritmo proposto decai, porém visualmente, as formas de onda aparentam continuar sendo bastante semelhantes.



Como nos demais cenários, ocorreu a permutação entre as estimativas das fontes. A inversão de fase devido à ambiguidade do modelo, não foi verificada neste caso.

O algoritmo 2 apresentou resultados praticamente inalterados quando comparados com os demais obtidos anteriormente para  $M$  menores.

#### 4.2.5. Teste 5: Mistura de complexidade 10

- $M = 10$
- $L = 10$
- Filtros de Mistura:

$$\begin{aligned}
 h\{1,1\} &= [ \quad 0.3387 \quad 0.1719 \quad -0.4285 \quad -0.8592 \quad -1.3469 \quad -0.4517 \quad 2.1595 \quad -0.8313 \quad 0.9961 \\
 &\quad -0.2947 \quad ] \\
 h\{2,1\} &= [ \quad -1.9915 \quad 1.2703 \quad -0.1224 \quad 1.5030 \quad -1.3336 \quad 0.2463 \quad 0.4433 \quad 0.5466 \quad 0.0546 \\
 &\quad -0.0775 \quad ] \\
 h\{1,2\} &= [ \quad 0.1747 \quad 0.6109 \quad 1.9971 \quad 0.5984 \quad 0.5468 \quad 0.2996 \quad 0.4613 \quad 0.9832 \quad 0.7627 \\
 &\quad 0.0566 \quad ] \\
 h\{2,2\} &= [ \quad 0.8603 \quad -1.0059 \quad -1.6099 \quad 0.6441 \quad -0.1055 \quad -0.021 \quad 1.2104 \quad -0.5117 \quad 0.4761 \\
 &\quad -1.7323 \quad ]
 \end{aligned}$$

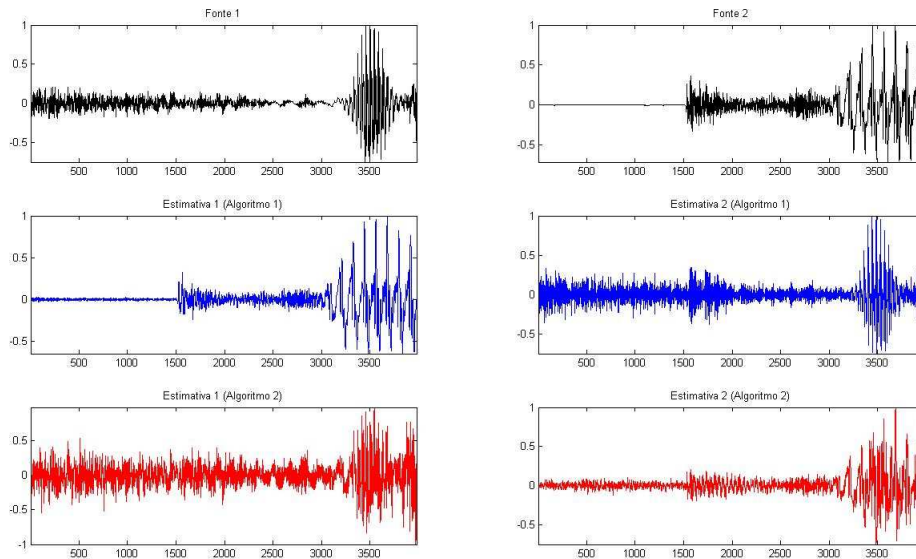


Figura 4.9 - Análise 2/Teste 5: fontes e estimativas ( $M = 10 / L = 10$ )

Cluster (1): 1 / 4 / 14 / 3 / 17 / 5 / 6 / 18 / 9 / 16 / 10 / 20 / 15 / 7 / 11 / 19 / 8

Cluster (2): 2 / 12 / 13

Medidas	SDR	SIR	SAR
	Algoritmo 1		
Estimativa 1	1.6084	8.8458	3.0503
Estimativa 2	6.5613	13.1175	7.8527
	Algoritmo 2		
Estimativa 1	4.3742	5.2938	12.6923
Estimativa 2	3.24834	3.5313	16.8455

Tabela 4.9 - Análise 2/Teste 5: resultados ( $M = 10 / L = 10$ )

Para filtros de mistura de ordem 10, o algoritmo proposto apresentou resultados medianos se comparados ao obtidos para misturas menos complexas (filtros de ordem mais baixa).

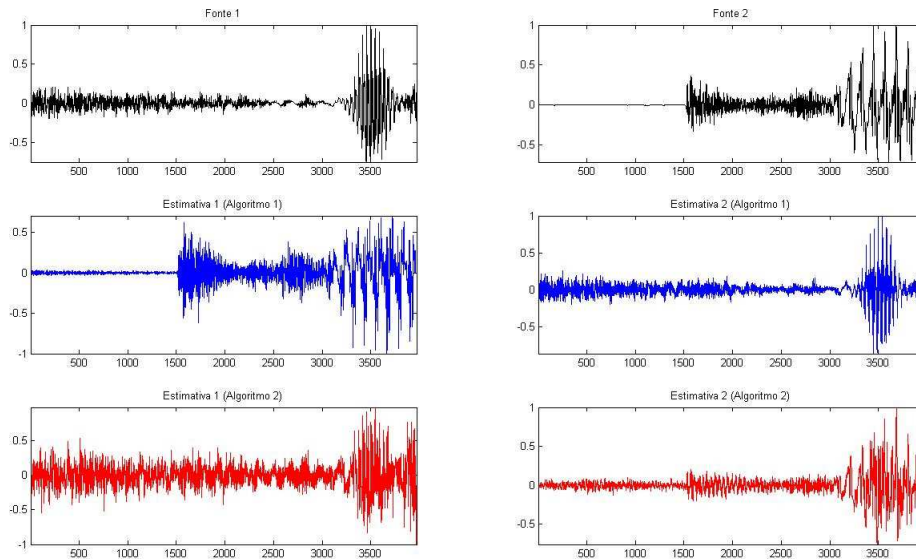
Ainda assim, o algoritmo pode ser considerado bom ( $8\text{dB} < \text{SIR} < 20\text{dB}$ ). De fato, as demais medidas (SDR, SAR) apresentaram valores abaixo do esperado, sendo piores que as obtidas para o algoritmo 2.

O método 2 continuou a apresentar resultados praticamente inalterados, bastante semelhantes aos previamente obtidos.

#### 4.5.6. Teste 6: Mistura de complexidade 15

- $M = 15$
- $L = 15$
- Filtros de mistura

$$\begin{aligned}
 h\{1,1\} &= [ \quad 0.2793 \quad -0.0773 \quad -0.3265 \quad 0.6773 \quad 0.7187 \quad 0.4686 \quad 0.6143 \quad 0.4606 \quad 0.0992 \\
 &\quad -0.2452 \quad -0.3050 \quad 1.8370 \quad 0.4042 \quad 0.0254 \quad 0.1479 \quad ] \\
 h\{2,1\} &= [ \quad 1.0283 \quad -0.7517 \quad 0.5856 \quad 0.4850 \quad -0.4842 \quad -0.5991 \quad 0.2950 \quad -1.7151 \quad 1.4027 \\
 &\quad 0.1625 \quad 1.2894 \quad -0.2174 \quad 1.0966 \quad 1.1517 \quad -0.5025 \quad ] \\
 h\{1,2\} &= [ \quad 0.0750 \quad -0.6112 \quad -0.3958 \quad 0.1295 \quad -1.4362 \quad 1.1932 \quad 0.5690 \quad 0.7685 \quad -0.9422 \\
 &\quad 0.8918 \quad -0.6373 \quad -1.1060 \quad -0.2368 \quad -0.3153 \quad -1.7397 \quad ] \\
 h\{2,2\} &= [ \quad -1.0290 \quad 0.6751 \quad -1.1883 \quad -0.0025 \quad 2.5313 \quad -0.0610 \quad -0.1978 \quad -0.2445 \quad -1.4768 \\
 &\quad 1.2415 \quad -0.0949 \quad 0.2815 \quad -0.3268 \quad 2.0518 \quad -0.5162 \quad ]
 \end{aligned}$$



**Figura 4.10 - Análise 2/Teste 6: fontes e estimativas ( $M = 15 / L = 15$ )**

Cluster (1): 1 / 2 / 8 / 14 / 9 / 18 / 26 / 10 / 21 / 11 / 23 / 25 / 28 / 17 / 5 / 22 / 4 / 7 / 13 /  
19 / 24 / 20 / 12 / 15 / 29 / 30 / 6

Cluster (2): 3 / 16 / 27

Medidas	SDR	SIR	SAR
	Algoritmo 1		
Estimativa 1	5.3274	12.5753	6.4680
Estimativa 2	5.6082	13.6592	6.5311
	Algoritmo 2		
Estimativa 1	4.5677	5.4951	12.8077
Estimativa 2	3.1757	3.4622	16.7396

**Tabela 4.10 - Análise 2/Teste 6: resultados (M = 15 / L = 15)**

Utilizando misturas geradas a partir de filtros de ordem 15, o algoritmo obteve resultados razoáveis, sendo até melhores que os obtidos para filtros de ordem 10.

Observando a forma de onda obtida na saída (estimativas do sinal original), vemos que estas são próximas das formas de onda das fontes. A estimativa 1 (fonte 2), notamos que ocorre a perda da escala de amplitude em relação à fonte original.

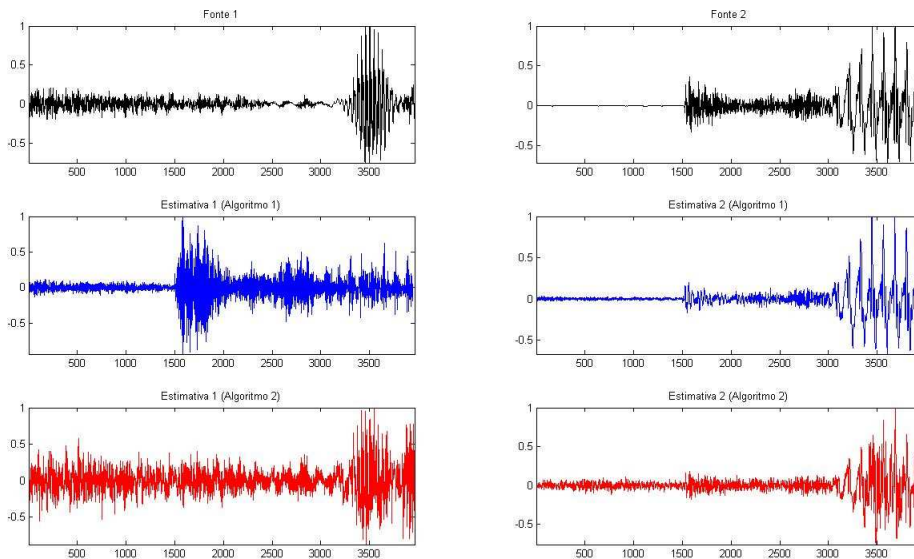
Ocorreu a permutação das estimativas, como na maioria dos testes anteriores.

Os resultados do algoritmo 2 permaneceram praticamente idênticos aos obtidos no anterior.

### 4.2.7. Teste 7: Mistura de complexidade 20

- $M = 20$
- $L = 20$
- Filtros de Mistura

$$\begin{aligned}
 h\{1,1\} &= [ \begin{array}{cccccccccc} 0.2683 & 0.3239 & -0.0479 & -0.8898 & -0.2645 & -0.6166 & 1.8399 & 0.4479 & -0.5194 \\ 0.1452 & 0.085 & 0.9357 & 1.6084 & -1.1459 & 1.0978 & 0.564 & 0.1911 & 1.4763 \\ -1.0857 & -0.7496 & & & & & & & & \end{array} ] \\
 h\{2,1\} &= [ \begin{array}{cccccccccc} -0.2702 & 2.0113 & -0.5186 & 0.2523 & 0.1139 & 0.4776 & -0.1301 & 0.9174 & -0.9515 \\ -0.285 & 0.2721 & 0.1773 & 0.4817 & 0.7228 & -1.8807 & -0.6616 & -1.004 & 0.5739 \\ 0.3975 & 1.8773 & & & & & & & & \end{array} ] \\
 h\{1,2\} &= [ \begin{array}{cccccccccc} -0.1635 & 0.0837 & -0.0575 & 0.3795 & -1.3391 & 1.4023 & 0.3721 & -0.893 & 0.3219 \\ -0.4509 & -0.8856 & -0.5087 & -0.0579 & -0.269 & 0.889 & 1.168 & 0.2604 & 0.1676 \\ -0.9402 & -0.0217 & & & & & & & & \end{array} ] \\
 h\{2,2\} &= [ \begin{array}{cccccccccc} -0.7818 & 0.6098 & -0.0589 & 0.6152 & 0.4382 & -2.0568 & 0.6459 & 1.2993 & 0.0486 \\ 0.3615 & 0.1608 & 0.6509 & 0.5654 & 1.1676 & 0.5137 & -0.4581 & -1.133 & 3.4283 \\ 1.072 & -0.4385 & & & & & & & & \end{array} ]
 \end{aligned}$$



**Figura 4.11 - Análise 2/Teste 7: fontes e estimativas ( $M = 20 / L = 20$ )**

Cluster (1): 1 / 4 / 16 / 21 / 18 / 38 / 25 / 37 / 26 / 2 / 17 / 6 / 22 / 23 / 36 / 5 / 40 / 11 /  
14 / 35 / 8 / 27 / 32 / 12 / 7 / 31 / 15 / 30 / 33 / 10 / 29 / 19

Cluster (2): 3 / 28 / 9 / 13 / 39 / 24 / 34 / 20

Medidas	SDR	SIR	SAR
	Algoritmo 1		
Estimativa 1	-9.7456	-5.3811	-1.2802
Estimativa 2	7.5193	13.7801	8.8697
	Algoritmo 2		
Estimativa 1	5.5152	6.6990	12.5804
Estimativa 2	2.7579	2.9745	17.6604

Tabela 4.11 - Análise 2/Teste 7: resultados (M = 20 / L = 20)

Para filtros de mistura de tamanho 20, o desempenho do algoritmo decaiu muito, ficando aquém do aceitável. Para a estimativa 1 (fonte 1), o resultado foi bastante insatisfatório, tanto numericamente quanto visualmente. Em contra partida, a estimativa 2 (fonte 2) apresentou resultados semelhantes aos obtidos anteriormente com filtros de mistura menos complexos, provavelmente devido à distribuição não-uniforme das componentes pelos *clusters*.

A discrepância entre as estimativas obtidas para cada uma das fontes é explicável visto que trabalhamos com misturas geradas a partir de filtros aleatórios.

Os resultados obtidos para o algoritmo 2 foram mais uniformes, apresentando valores próximos para ambas as estimativas, o que mostra uma separação mais uniforme entre as fontes. Também é verificado que os valores se mantiveram semelhantes aos obtidos com filtros de mistura menores.

Não ocorreu permutação entre as fontes para ambos os algoritmos.

## 5. Conclusão

Os resultados obtidos para o algoritmo implementado foram bastante satisfatórios para misturas pouco complexas ( $M < 8$ ). Como esperado, tal desempenho decaiu com o aumento da complexidade dos filtros de mistura, sendo a última mistura testada ( $M = 20$ ) muito ruim.

Em todos os testes, o algoritmo implementado apresentou resultados superiores ao outro algoritmo testado (baseado no domínio da frequência). Porém, este foi superior para misturas de complexidade elevada.

Cumprir lembrar que as técnicas mais comuns de separação de misturas convolutivas raramente empregam um número tão reduzido de amostras, mesmo para um reduzido comprimento dos filtros de separação.

Apesar de mais eficiente, o algoritmo implementado é extremamente lento para misturas complexas o que torna inviável para o uso em tempo real.

As ambiguidades do modelo de ICA estiveram evidentes ao longo de todos os testes realizados, visto que em diversos momentos ocorreram permutações entre as fontes e suas estimativas, bem como inversão de fase dos sinais e suas estimativas.

Em alguns dos testes, a assimetria em relação ao número de componentes de cada *cluster* se deve a aleatoriedade do modelo ICA e ao fato que uma estimativa pode conter mais informação da mistura do que a outra. Isso não afetou a qualidade das estimativas obtidas.

O tempo gasto na execução do algoritmo é diretamente relacionado ao tamanho escolhido dos filtros de mistura  $L$ , visto que é necessária a utilização de matrizes de tamanho  $m.L$  na execução do mesmo.

### 5.1. Sugestões

Apesar dos resultados obtidos terem sido muito bons, pode-se generalizar o algoritmo para operar com diversas entradas e saídas, não ficando limitado a apenas duas fontes.

Novas medidas de referência para a clusterização podem ser propostas, bem como uma nova abordagem para a realização da lógica de decisão binária implementada. Esta poderia envolver lógica *Fuzzy* ou redes neurais.



O tempo de execução do algoritmo para sinais maiores que o bloco utilizado para a realização da separação pode ser diminuído, uma vez que se obtenham os filtros de separação, reduzindo a geração das estimativas de saída a um processo convolutivo.

O presente trabalho foi desenvolvido para operar via linha de comando, porém uma interface gráfica o tornaria mais interessante e amigável para usuários básicos.

## 6. Referências Bibliográficas

- [1] Mitianoudis N., and Davies, M. E., “Audio source separation: solutions and problems”, *International Journal of Adaptive Control and Signal Processing*, 18:299–314, 2004.
- [2] Asano, F.; Ikeda, S.; Ogawa, M.; Asoh, H.; Kitawaki, N., “A combined approach of array processing independent component analysis for blind separation of acoustic signals”, *IEEE Transactions on Speech and Audio Processing*, Volume 11, Issue 3 Page(s): 204 – 215, May 2003.
- [3] Murata, N., and Ikeda, S., “An on-line algorithm for blind source separation on speech signals”, In *Proceedings of 1998 International Symposium on Nonlinear Theory and its Applications (NOLTA'98)*, pp.923-926, Crans-Montana, Switzerland, Sep. 1998.
- [4] Hoyer P.O., and Hyvärinen, A., “Independent component analysis applied to feature extraction from color and stereo images”, *Network: Computation in Neural Systems*, 11(3):191–210, 2000.
- [5] Feng, M.; Kammeyer, K.-D., “Blind source separation for communication signals using antenna arrays”, *IEEE 1998 International Conference on Universal Personal Communications* Page(s):665 - 669 vol.1, Oct. 1998.
- [6] Zarzoso, V., “Exploiting independence for co-channel interference cancellation and symbol detection in multiuser digital communications,” *Seventh International Symposium on Proceedings of Signal Processing and Its Applications*Page(s): 303 - 306 vol.2, Jul. 2003.
- [7] Gupta, M.; Santhanam, B., “Prior ICA based blind multiuser detection in DS-CDMA systems,” *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers* Page(s): 2155 - 2159 Vol.2, Nov. 2004.
- [8] Cristescu, R., Joutsensalo, J., Karhunen, J., and Oja., E., “A complexity minimization approach for estimating fading channels in CDMA communications”, In *Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation (ICA2000)*, pages 527–532, Helsinki, Finland, Jun. 2000..
- [9] James, C. J. and Gibson, O. J. (2002), “Electromagnetic brain signal analysis using constrained ICA”, *Proceedings of 2nd European Medical and Biological Engineering Conference (EMBEC'02)*, Vienna, Austria, Part I, pp 426-427, Dec. 2002.

- [10] He T., Clifford G., Tarassenko, L.: “Application of independent component analysis in removing artefacts from the electrocardiogram,” *Neural Comput. & Applic.* 15(2): 105-116, 2006.
- [11] Wisbeck, J. O., Barros, A. K., and Ojeda, R. G. “Application of ICA in the Separation of Breathing Artifacts in ECG Signals,” *Proceedings of ICONIP'98*, Kyushu, Japan, Oct. 1998.
- [12] Ungureanu, M., Bigan, C., R. Strungaru, Lazarescum, V., “Independent Component Analysis Applied in Biomedical Signal Processing,” *MEASUREMENT SCIENCE REVIEW*, Bucharest, Romania, Volume 4, Section 2, 2004.
- [13] Back A. D., and Weigend, A. S., “A first application of independent component analysis to extracting structure from stock returns”, *Int. J. on Neural Systems*, 8(4):473–484, 1997.
- [14] Kiviluoto K., and Oja. E., “Independent component analysis for parallel financial timeseries”, In *Proc. Int. Conf. on Neural Information Processing (ICONIP'98)*, volume 2, pages 895–898, Tokyo, Japan, 1998.
- [15] P. Comon, *Independent Component Analysis: A new concept?*, *Signal Processing*, vol. 36, no. 3, pp. 287-314, 1994.
- [16] Cardoso, J.-F., C.N.R.S. e E.N.S.T., “Blind signal separation: statistical principles,” *Proceedings of the IEEE*, VOL. 9, No. 10, PP. 2009-2025, Oct. 1998.
- [17] Héroult, J., Jutten, C., and Ans, B., “Détection de grandeurs primitives dans un message composite par une architecture de calcul neuromimétique en apprentissage non supervisé”, *Actes du Xième colloque GRETSI*, vol. 2, pp. 1017–1022, Nice, France, Mai 1985.
- [18] Jutten, C. and Héroult, J., “Blind Separation of Sources, part I: An adaptive algorithm based on neuromimetic architecture”. *Signal Processing*, 24:1-10, 1991.
- [19] Comon, P., Jutten, C., e Héroult, J., “Blind separation of sources, Part II: Problems Statement”, *Signal Processing*, 24:11–20, 1991.
- [20] Bell, A.J. and Sejnowski, T.J., “An information maximization approach to blind separation and blind deconvolution”, *Neural Computation* 7, pp. 1129-1159, MIT Press, Cambridge MA, 1995.
- [21] Cichocki, A., Amari, S., “Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications”, Wiley, 2003.

- [22] Cherry, E. C., "Some Experiments on the Recognition of Speech, with One and with Two Ears", *The Journal of the Acoustical Society of America*, Vol. 25, Issue 5, pp. 975-979, Sep. 1953.
- [23] Ebata, M., "Spatial unmasking and attention related to the cocktail party problem", *Acoustical Science and Technology*, Vol. 24, No. 5, Special issue on Spatial hearing, pp. 208-219, 2003.
- [24] Lee, T.-W., Ziehe, Orglmeister, A., Sejnowski, R. T., "Combining time-delayed decorrelation and ICA: towards solving the cocktail party problem", *Acoustics, Speech and Signal Processing, Proceedings of the 1998 IEEE International Conference on*, Vol. 2, pp. 1249-1252, Seattle, WA, USA, May 1998.
- [25] Haykin, S. and Chen, Z., "The Cocktail Party Problem", *Neural Computation*, vol. 17, pp. 1875-1902, 2005.
- [26] Hyvärinen, A., Karhunen, J., Oja, E., "Independent Component Analysis," John Wiley and Sons, New York, 2001.
- [27] Vincent, E., Gribonval, R., and Févotte, C., "Performance Measurement in Blind Audio Source Separation", *IEEE Transaction On Audio and Language Processing*, Vol. 14, No. 4, Jul. 2006.
- [28] S. Ikeda and N. Murata. An approach to blind source separation of speech signals. In *Proc. ICANN'98*, pages 1855–1865, 1998.
- [29] S. Choi, A. Cichocki, H.-M. Park, and S.-Y. Lee, "Blind source separation and independent component analysis: A review," 2004. *Neural Information Processing - Letters and Review*, vol. 6, no. 1, pp. 1-57, January 2005.
- [30] Molgedey, L., Schuster, H.G. Separation of a mixture of independent signals using time delayed correlations. *Physical Review Letters*, 72 (23):3634–3637, 1994.
- [31] Nandi, A., editor, "Blind Estimation Using Higher-Order Statistics", Kluwer, 1999.
- [32] S. A. Cruces, L. Castedo, and A. Cichocki. Robust blind source separation algorithms using cumulants. *Neurocomputing*, 49:87–118, December 2002.
- [33] A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, and E. Moulines. A blind source separation technique using second-order statistics. *IEEE Trans. Signal Processing*, 45(2):434–444, February 1997.

- [34] K. Matsuoka, M. Ohya, and M. Kawamoto. A neural net for blind separation of nonstationary signals. *Neural Networks*, 8(3):411–419, 1995.
- [35] S. Choi, A. Cichocki, and S. Amari. Equivariant nonstationary source separation. *Neural Networks*, 15:121–130, 2002.
- [36] S. Choi, A. Cichocki, and A. Belouchrani. Second order nonstationary source separation. *Journal of VLSI Signal Processing*, 32(1–2):93–104, August 2002.
- [37] Molgedey, L. and Schuster, H. G., “Separation of a mixture of independent signals using time delayed correlations”, *Physical Review Letters*, 72, No. 23, pp 3634–3636, 1994.
- [38] A. Belouchrani and M.G. Amin. A new approach for blind source separation using time-frequency distributions. *Proc. SPIE*, 2846:193–203, 1996.
- [39] A. Cichocki and S. Amari. *Adaptive Blind Signal And Image Processing*. John Wiley, New York, 2003. New revised and improved edition.
- [40] Lee, T. W., Lewicki, M.S., Girolami, M., Sejnowski, T.J., "Blind source separation of more sources than mixtures using overcomplete representations", *IEEE Signal Processing Letters*, v. 4, n. 5, 1999.
- [41] Stuart A., e Ord, K., “Kendall’s advanced theory of statistics – Distribution Theory”, Oxford University Press, 6th Edition, Vol. 1, New York, 2006.
- [42] Norman L. J., Kotz, S., Kemp, A. W., “Univariate discrete distributions”, Wiley-Interscience, 3rd Edition, 2005.
- [43] Maronna, R., Martin, D., Yohai, V., “Robust Statistics – Theory and Methods”, John Wiley & Sons, England, 2006.
- [44] Cover, T.M. and Thomas, J. A. “Elements of Information Theory” John Wiley and Sons, New York, 1991.
- [45] Shannon, C. E., “A Mathematical Theory of Communication”, *The Bell System Technical Journal*, Vol. 27, pp. 379–423, 623–656, 1948.
- [46] Hyvärinen, A., "One-unit contrast functions for independent component analysis: A statistical analysis", In: *Proceedings of Neural Networks for Signal Processing VII (Proc. IEEE Workshop on Neural Networks for Signal Processing)*, pp. 388-397, Amelia Island, Florida, 1998.
- [47] Wallace, D. L., “Asymptotic Approximations to Distributions”, *The Annals of Mathematical Statistics*, Vol. 29, No. 3, pp. 635-654, Sep., 1958.

- [48] A. Mohammad-Djafari. A bayesian approach to source separation. In Proc. 19th International Workshop on Bayesian and Maximum Entropy methods (MaxEnt 1999), pages 1–26, Boise, Idaho, USA, 1999.
- [49] A. Papoulis. Probability, Random Variables and Stochastic Processes. McGraw-Hill International, 3 edition, 1991.
- [50] H. Valpola and P. Pajunen. Fast algorithms for bayesian independent component analysis. In Proc. Second International Workshop on Independent Component Analysis and Blind Signal Separation, pages 233–237, Helsinki, Finland 2000.
- [51] Y. Zhanga, X. Shia, and C. H. Chenb. A gaussian mixture model for underdetermined independent component analysis. *Signal Processing*, 86(7):1538–1549, July 2006.
- [52] P. G. Georgiev, F. Theis, and A. Cichocki. Sparse component analysis and blind source separation of underdetermined mixtures. *IEEE Transactions on Neural Networks*, 16(4):992–996, July 2005.
- [53] Y. Li, A. Cichocki, and S.-I. Amari. Sparse component analysis for blind source separation with less sensors than sources. In Proc. 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003), pages 89–94, Kyoto, Japan, 2003.
- [54] P. Bofill and M. Zibulevsky. Underdetermined blind source separation using sparse representations. *Signal Processing*, 81(11):2353–2362, November 2001.
- [55] Z. He and A. Cichocki. Independent Component Analysis and Blind Signal Separation, chapter K-EVD Clustering and Its Applications to Sparse Component Analysis, pages 90–97. Springer, 2006.
- [56] K. Torkkola. Blind separation of convolved sources based on information maximization. In Proc. IEEE Int. Workshop on NNSP, 1996.
- [57] Y. Li, A. Cichocki, S. Amari, S. Shishkin, J. Cao, and F. Gu. Sparse representation and its applications in blind source separation. In Seventeenth Annual Conference on Neural Information Processing Systems (NIPS-2003), Vancouver, December 2003.
- [58] S. Araki, S. Makino, R. Mukai, T. Nishikawa, and H. Saruwatari. Fundamental limitation of frequency domain blind source separation for convolved mixtures of speech. In Proc. Int. Conf. ICA and BSS, pages 132–137, December 2001.

- [59] J. Anemüller and B. Kollmeier. Amplitude modulation decorrelation for convolutive blind source separation. In Proc. Int. Conf. ICA and BSS, pages 215–220, June 2000.
- [60] N. Murata, S. Ikeda, and A. Ziehe. An approach to blind source separation based on temporal structure of speech signals. *Neurocomputing*, 41(1/4):1–24, 2001.
- [61] H.M. Park, S.H. Oh, and S.Y. Lee. A filter bank approach to independent component analysis and its application to adaptive noise cancelling. *Neurocomputing*, 55(3-4):755–759, 2003.
- [62] H.-M. Park, S.-H. Oh, and S.-Y. Lee. An oversampled filter bank approach to independent component analysis for convolved mixtures. In Proc. Joint Int. Conf. Artificial Neural Networks and Neural Information Processing, pages 354–357, June 2003.
- [63] Z. Koldovský and P. Tichavský, "Time-domain Blind Audio Source Separation Using Advanced Component Clustering and Reconstruction", to be presented on The Joint Workshop on Hands-free Speech Communication and Microphone Arrays (HSCMA 2008), May 6-8, Trento, Italy, 2008.
- [64] Z. Koldovský and P. Tichavský, "Time-Domain Blind Audio Source Separation Using Advanced ICA Methods", Proceedings of 8th Annual Conference of the International Speech Communication Association (Interspeech 2007), pp. 846-849, August 2007.
- [65] Z. Koldovský, P. Tichavský, and E. Oja, "Efficient Variant of Algorithm FastICA for Independent Component Analysis Attaining the Cramér-Rao Lower Bound," *IEEE Tr. Neural Networks*, vol. 17, no. 5, pp. 1265- 1277, September 2006.
- [66] Chevalier, P., Albera, L., Comon, P., and Ferreol, A., "Comparative performance analysis of eight blind source separation methods on radiocommunications signals" in: Proc. Intl. Joint Conf. on Neural Networks, Budapest, Hungary, Jul. 2004.
- [67] Schobben, D.W.E., Torkkola, K., and Smaragdis, P., "Evaluation of Blind Signal Separation Methods", First International Workshop on Independent Component Analysis and Blind Signal Separation, Aussois, France, Jan. 1999.