



## LIMITAÇÕES DE ALGORITMOS DE SEPARAÇÃO CEGA DE SINAIS DE VOZ EM AMBIENTES REVERBERANTES

Bruna Luisa Ramos Prado Vasques

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientadora: Mariane Rembold Petraglia

Rio de Janeiro  
Junho de 2015

LIMITAÇÕES DE ALGORITMOS DE SEPARAÇÃO CEGA DE SINAIS DE  
VOZ EM AMBIENTES REVERBERANTES

Bruna Luisa Ramos Prado Vasques

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO  
ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE  
ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE  
JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A  
OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA  
ELÉTRICA.

Examinada por:

---

Prof. Mariane Rembold Petraglia, Ph.D.

---

Prof. Carlos Augusto Duque, D.Sc.

---

Prof. Fernando Gil Vianna Resende Junior, Ph.D.

RIO DE JANEIRO, RJ – BRASIL  
JUNHO DE 2015

Vasques, Bruna Luisa Ramos Prado

Limitações de algoritmos de separação cega de sinais de voz em ambientes reverberantes/Bruna Luisa Ramos Prado Vasques. – Rio de Janeiro: UFRJ/COPPE, 2015.

XV, 62 p.: il.; 29,7cm.

Orientadora: Mariane Rembold Petraglia

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2015.

Referências Bibliográficas: p. 59 – 62.

1. Separação Cega de Fontes. 2. Misturas Convolutivas. 3. Filtragem Adaptativa. 4. Estruturas em Subbandas. 5. Sinais de Áudio. I. Petraglia, Mariane Rembold. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

*À minha família e amigos.*

# Agradecimentos

Gostaria de agradecer primeiramente aos meus pais e irmãos pela força e apoio.

Ao meu noivo Igor pelo apoio e compreensão em todos os momentos.

À minha orientadora Mariane Rembold Petraglia pela paciência, compreensão e dedicação.

Aos professores Carlos Augusto Duque e Fernando Gil Vianna Resende Junior que aceitaram participar da banca para a defesa desta dissertação.

Ao Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq pelo apoio financeiro nos primeiros meses de mestrado.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

## LIMITAÇÕES DE ALGORITMOS DE SEPARAÇÃO CEGA DE SINAIS DE VOZ EM AMBIENTES REVERBERANTES

Bruna Luisa Ramos Prado Vasques

Junho/2015

Orientadora: Mariane Rembold Petraglia

Programa: Engenharia Elétrica

Nesta dissertação foram investigadas as limitações de algoritmos de separação cega de fontes em ambientes reverberantes utilizando processamento adaptativo e análises em subbandas.

Para isso, foram empregados os algoritmos adaptativos supervisionados NLMS (*Normalized Least-Mean Square*) e NSAF (*Normalized Subband Adaptive Filter*), para determinar as limitações dos modelos do sistema de separação tanto em banda cheia quanto em subbandas.

Além disso, as características do sinal de voz (não-gaussianidade, não-estacionariedade e espectro colorido) foram exploradas de maneira que os parâmetros do algoritmo de separação cega de fontes TRINICON (*Triple-N ICA for Convolutional Mixtures*), que leva em conta todas as três características, pudessem ser melhor escolhidos.

Por fim, foi feita uma modificação no algoritmo TRINICON de modo a explorar as características dos sinais de voz em cada subbanda de frequência. Com isso, para o caso de misturas determinadas, com duas fontes e dois sensores, e com tempo de reverberação de 100ms foi possível obter uma melhora de aproximadamente 4 dB na SIR (*Signal-to-Interference Ratio*).

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

## LIMITATIONS OF BLIND SOURCE SEPARATION ALGORITHMS IN REVERBERANT ENVIRONMENTS

Bruna Luisa Ramos Prado Vasques

June/2015

Advisor: Mariane Rembold Petraglia

Department: Electrical Engineering

In this thesis, the limitations of blind source separation (BSS) algorithms in reverberant environments were investigated, using adaptive processing and subband analysis.

With this purpose, the adaptive supervised algorithms NLMS (Normalized Least Mean Square) and NSAF (Normalized Subband Adaptive Filter) were employed in order to determine the limitations of the separation system models in both fullband and subbands.

Furthermore, speech signal characteristics (nongaussianity, nonstationarity and nonwhiteness) were explored so that the parameters of the BSS algorithm TRINICON (Triple-N ICA for Convolutional Mixtures), which considers all three characteristics, could be wisely chosen.

Finally, the algorithm TRINICON was modified to explore the speech characteristics in the different frequency subbands. Thus, for determined mixtures, using two sources and two sensors, and reverberation time of 100ms, it was possible to obtain an improvement of approximately 4 dB in SIR (Signal-to-Interference Ratio).

# Sumário

<b>Lista de Figuras</b>	<b>x</b>
<b>Lista de Tabelas</b>	<b>xii</b>
<b>Lista de Símbolos</b>	<b>xiii</b>
<b>Lista de Abreviaturas</b>	<b>xv</b>
<b>1 Introdução</b>	<b>1</b>
1.1 O problema de separação de fontes . . . . .	3
1.2 Tipos de misturas . . . . .	4
1.2.1 Misturas instantâneas . . . . .	4
1.2.2 Misturas convolutivas . . . . .	5
1.3 Ambiguidades na separação . . . . .	5
1.3.1 Ambiguidade da filtragem . . . . .	6
1.3.2 Ambiguidade da permutação . . . . .	6
1.4 Características dos sinais de voz . . . . .	7
1.5 Banco de filtros . . . . .	9
1.5.1 Banco de filtros modulados por cosseno . . . . .	10
1.6 Medidas de avaliação . . . . .	11
<b>2 Separação cega de fontes</b>	<b>13</b>
2.1 Separação cega através de decomposição de autovalor generalizada . . . . .	13
2.1.1 Abordagem para sinais não-estacionários . . . . .	14
2.1.2 Abordagem para sinais não-brancos . . . . .	15
2.1.3 Abordagem para sinais não-gaussianos . . . . .	15
2.2 Abordagem utilizando as três características . . . . .	16
2.2.1 Notação matricial . . . . .	16
2.2.2 Função custo . . . . .	17
2.2.3 Regra de atualização . . . . .	18
2.2.4 Algoritmo TRINICON para PDF laplaciana . . . . .	19
2.2.5 Algoritmo TRINICON para PDF gaussiana . . . . .	19



<b>3</b>	<b>Estudo das características dos sinais de voz</b>	<b>20</b>
3.1	Não-gaussianidade . . . . .	20
3.2	Não-estacionariedade . . . . .	26
3.3	Não-branco . . . . .	33
3.4	Comentários gerais . . . . .	38
<b>4</b>	<b>Separação de fontes com algoritmos adaptativos supervisionados</b>	<b>40</b>
4.1	Sinal desejado no problema de separação cega . . . . .	40
4.2	Algoritmos adaptativos . . . . .	42
4.2.1	Algoritmo NLMS . . . . .	42
4.2.2	Algoritmo NSAF . . . . .	45
4.3	Resultados das simulações . . . . .	50
4.4	Avaliação dos resultados . . . . .	52
<b>5</b>	<b>Exploração das características dos sinais de voz e dos ambientes re-</b>	
	<b>verberantes para melhoria do algoritmo de separação TRINICON</b>	<b>53</b>
5.1	Simulações para os sinais em banda completa . . . . .	54
5.2	Simulações para os sinais em subbandas . . . . .	55
<b>6</b>	<b>Conclusões</b>	<b>58</b>
	<b>Referências Bibliográficas</b>	<b>59</b>

# Lista de Figuras

1.1	Forma de onda de duas pessoas falando simultaneamente . . . . .	1
1.2	Ilustração do problema de separação de fontes . . . . .	2
1.3	Estrutura geral do problema de separação de fontes . . . . .	4
1.4	Ilustração do problema de ambiguidade da permutação . . . . .	6
1.5	PDFs subgaussiana, gaussiana e supergaussiana . . . . .	7
1.6	PDFs de sinais de voz e mistura . . . . .	8
1.7	30 ms de um sinal de voz . . . . .	9
1.8	Bancos de filtros de análise e de síntese . . . . .	9
3.1	Histogramas - Decomposição em 2 subbandas - Voz masculina 1 . . . .	21
3.2	Histogramas - Decomposição em 2 subbandas - Voz masculina 2 . . . .	21
3.3	Histogramas - Decomposição em 2 subbandas - Voz feminina 1 . . . .	21
3.4	Histogramas - Decomposição em 2 subbandas - Voz feminina 2 . . . .	21
3.5	Histogramas - Decomposição em 4 subbandas - Voz masculina 1 . . . .	22
3.6	Histogramas - Decomposição em 4 subbandas - Voz masculina 2 . . . .	22
3.7	Histogramas - Decomposição em 4 subbandas - Voz feminina 1 . . . .	22
3.8	Histogramas - Decomposição em 4 subbandas - Voz feminina 2 . . . .	23
3.9	Histogramas - Decomposição em 8 subbandas - Voz masculina 1 . . . .	23
3.10	Histogramas - Decomposição em 8 subbandas - Voz masculina 2 . . . .	24
3.11	Histogramas - Decomposição em 8 subbandas - Voz feminina 1 . . . .	24
3.12	Histogramas - Decomposição em 8 subbandas - Voz feminina 2 . . . .	25
3.13	Janelas iniciais do algoritmo adaptativo STFT . . . . .	27
3.14	Janelas após a primeira iteração, juntando os segmentos . . . . .	27
3.15	Janelas após a primeira iteração, sem juntar os segmentos . . . . .	28
3.16	Janela fixa . . . . .	28
3.17	Janela variável . . . . .	29
3.18	Histogramas das janelas - 2 subbandas - Voz masculina 1 . . . . .	29
3.19	Histogramas das janelas - 2 subbandas - Voz masculina 2 . . . . .	30
3.20	Histogramas das janelas - 2 subbandas - Voz feminina 1 . . . . .	30
3.21	Histogramas das janelas - 2 subbandas - Voz feminina 2 . . . . .	30
3.22	Histogramas das janelas - 4 subbandas - Voz masculina 1 . . . . .	31

3.23	Histogramas das janelas - 4 subbandas - Voz masculina 2 . . . . .	31
3.24	Histogramas das janelas - 4 subbandas - Voz feminina 1 . . . . .	31
3.25	Histogramas das janelas - 4 subbandas - Voz feminina 2 . . . . .	31
3.26	Histogramas das janelas - 8 subbandas - Voz masculina 1 . . . . .	32
3.27	Histogramas das janelas - 8 subbandas - Voz masculina 2 . . . . .	32
3.28	Histogramas das janelas - 8 subbandas - Voz feminina 1 . . . . .	33
3.29	Histogramas das janelas - 8 subbandas - Voz feminina 2 . . . . .	33
3.30	Matrizes de covariância - 2 subbandas - Voz masculina 1 . . . . .	34
3.31	Matrizes de covariância - 2 subbandas - Voz masculina 2 . . . . .	34
3.32	Matrizes de covariância - 2 subbandas - Voz feminina 1 . . . . .	35
3.33	Matrizes de covariância - 2 subbandas - Voz feminina 2 . . . . .	35
3.34	Matrizes de covariância - 4 subbandas - Voz masculina 1 . . . . .	35
3.35	Matrizes de covariância - 4 subbandas - Voz masculina 2 . . . . .	36
3.36	Matrizes de covariância - 4 subbandas - Voz feminina 1 . . . . .	36
3.37	Matrizes de covariância - 4 subbandas - Voz feminina 2 . . . . .	36
3.38	Matrizes de covariância - 8 subbandas - Voz masculina 1 . . . . .	37
3.39	Matrizes de covariância - 8 subbandas - Voz masculina 2 . . . . .	37
3.40	Matrizes de covariância - 8 subbandas - Voz feminina 1 . . . . .	38
3.41	Matrizes de covariância - 8 subbandas - Voz feminina 2 . . . . .	38
4.1	Algoritmo NLMS . . . . .	43
4.2	Algoritmo NSAF . . . . .	46
4.3	Comparação entre os algoritmos para tempo de reverberação igual a 0.1 s . . . . .	50
4.4	Comparação entre os algoritmos para tempo de reverberação igual a 0.25 s . . . . .	51
4.5	Comparação entre os algoritmos para tempo de reverberação igual a 0.5 s . . . . .	52
5.1	Estrutura empregada para separação em subbandas . . . . .	54
5.2	Comparação entre os algoritmos TRINICON para tempo de rever- beração igual a 0.1 s . . . . .	56
5.3	Comparação entre os algoritmos TRINICON para tempo de rever- beração igual a 0.25 s . . . . .	56
5.4	Comparação entre os algoritmos TRINICON para tempo de rever- beração = 0.5 s . . . . .	57

# Lista de Tabelas

1.1	Tipos de misturas . . . . .	4
1.2	Valores de curtose . . . . .	8
3.1	Valores de curtose para decomposição em 2 subbandas . . . . .	22
3.2	Valores de curtose para decomposição em 4 subbandas . . . . .	23
3.3	Valores de curtose para decomposição em 8 subbandas . . . . .	25
5.1	Parâmetros utilizados para geração do sinal de mistura . . . . .	54
5.2	Valores de SIR para simulação em banda completa com $T_{60} = 0.1$ s .	54
5.3	Valores de SIR para simulação em banda completa com $T_{60} = 0.25$ s .	55
5.4	Valores de SIR para simulação em banda completa com $T_{60} = 0.5$ s .	55
5.5	Configuração do algoritmo TRINICON em subbandas . . . . .	55

# Lista de Símbolos

$D$	Número de atrasos empregados no algoritmo TRINICON, p. 16
$F_i(z)$	$i$ -ésimo filtro de análise, p. 10
$G_i(z)$	$i$ -ésimo filtro de síntese, p. 10
$K_\nu$	Função de Bessel modificada de $\nu$ -ésima ordem, p. 19
$L$	Comprimento do filtro de separação, p. 5
$L_D$	Fator de decimação, p. 10
$L_w$	Comprimento da janela para cálculo da STFT, p. 26
$M$	Comprimento do filtro de mistura, p. 5
$M_D$	Número de subbandas, p. 10
$N$	Tamanho dos blocos do algoritmo TRINICON, p. 16
$P$	Número de sensores, p. 3
$Q$	Número de fontes, p. 3
$T_{60}$	Tempo de reverberação, p. 2
$\beta$	Função peso, p. 18
$\mathbf{H}$	Matriz de mistura, p. 3
$\mathbf{I}$	Matriz identidade, p. 14
$\mathbf{R}_x$	Matriz de covariância, p. 14
$\mathbf{W}$	Matriz de separação, p. 3
$\mathbf{s}(n)$	Vetor das fontes, p. 3
$\mathbf{x}(n)$	Vetor das misturas, p. 3

$\mathbf{x}_p$	Sinal da $p$ -ésima mistura, p. 16
$\mathbf{y}(n)$	Vetor de saída, p. 3
$\mathbf{y}_q$	Sinal da $q$ -ésima saída estimada, p. 16
$\mathcal{J}$	Função custo, p. 17
$\mu$	Fator de convergência, p. 43
$d(n)$	Sinal desejado, p. 43
$d_{i,D}$	Sinal desejado decomposto em subbandas e decimado, p. 46
$e(n)$	Erro, p. 43
$e_{artef}$	Parcela da estimativa que corresponde à presença de artefatos, p. 12
$e_{i,D}$	Erro do sinal decomposto em subbandas e decimado, p. 46
$e_{interf}$	Parcela da estimativa correspondente à interferência das outras fontes, p. 12
$e_{ruído}$	Parcela da estimativa que corresponde a ruídos dos sensores, p. 12
$p_0(n)$	Resposta ao impulso do filtro protótipo, p. 11
$s_{fonte}$	Sinal original da fonte, p. 12
$y_{i,D}$	Sinal de saída estimado decomposto em subbandas e decimado, p. 46

# Lista de Abreviaturas

DCT	<i>Discrete Cosine Transform</i> , p. 10
DOA	<i>Direction Of Arrival</i> , p. 6
FIR	<i>Finite Impulse Response</i> , p. 40
ICA	<i>Independent Component Analysis</i> , p. 15
NLMS	<i>Normalized Least-Mean Square</i> , p. 3
NSAF	<i>Normalized Subband Adaptive Filter</i> , p. 3
PDF	<i>Probability Density Function</i> , p. 7
SAR	<i>Signal to Artifacts Ratio</i> , p. 12
SDR	<i>Signal to Distortion Ratio</i> , p. 11
SIRP	<i>Spherically Invariant Random Process</i> , p. 19
SIR	<i>Signal to Interferences Ratio</i> , p. 11
SNR	<i>Signal to Noise Ratio</i> , p. 11
STFT	<i>Short-Time Fourier Transform</i> , p. 26
TDOA	<i>Time Difference Of Arrival</i> , p. 6
TRINICON	<i>Triple-N ICA for Convolutional Mixtures</i> , p. 2

# Capítulo 1

## Introdução

Quando vamos à uma festa, nossos ouvidos captam diversos sons como a música que está tocando, pessoas conversando e o barulho dos talheres e copos. Mesmo com essa mistura de sons, somos capazes de manter uma conversa porque, sem perceber, nosso cérebro já processa toda essa informação e conseguimos distinguir a voz da pessoa que está falando da música ambiente.

Esse é um exemplo de uma tarefa simples para nós, seres humanos, porém, imagine agora como essas informações chegam a um computador. São sinais completamente misturados como a forma de onda da Figura 1.1.

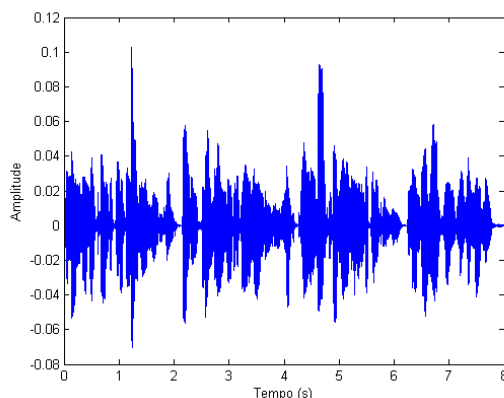


Figura 1.1: Forma de onda de duas pessoas falando simultaneamente

Então, para que um computador seja capaz de realizar a tarefa de separar sinais misturados, diversos algoritmos estão sendo desenvolvidos e nesta dissertação serão abordados métodos de separação cega de fontes.

Os métodos de separação cega de fontes visam recuperar os sinais originais a partir de uma ou mais misturas desses sinais, como ilustrado na Figura 1.2. O termo ‘cega’ significa que não há nenhuma informação prévia sobre as fontes ou sobre a forma em que foi feita a mistura.

A situação descrita anteriormente da festa é conhecida pelo problema da *Cock-*



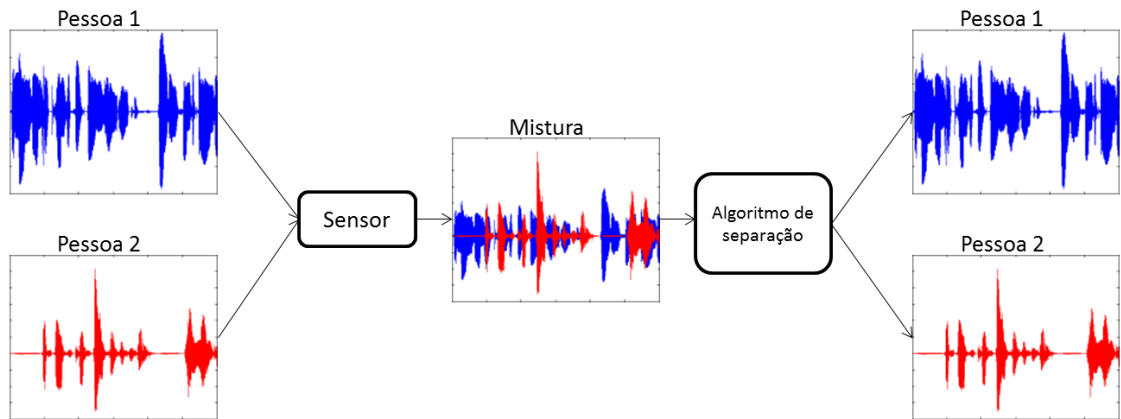


Figura 1.2: Ilustração do problema de separação de fontes

*tail Party*, mas existem inúmeras aplicações para os métodos de separação como na medicina através de estudos de ondas cerebrais [1] e eletrocardiogramas [2], em processamento de áudio para reconhecimento de voz [3], em processamento de imagens [4], entre outros.

Já existem algoritmos de separação cega de fontes que apresentam boa performance quando o número de sensores é igual ao de fontes e as misturas dos sinais são instantâneas, ou seja, compostas pelos sinais originais com diferentes atenuações mas sem nenhum atraso. Porém, quando o ambiente é reverberante esses algoritmos apresentam certas limitações. Um ambiente reverberante gera uma mistura convolutiva, na qual chegam também aos sensores as cópias atenuadas e atrasadas desses sinais que refletiram nas paredes e obstáculos do ambiente.

Nesta dissertação, foi empregado o simulador proposto em [5] e [6] e código-fonte em MATLAB disponível em [7], para simular os sinais recebidos em uma sala reverberante. Com esse simulador é possível configurar a frequência de amostragem, dimensão da sala, tempo de reverberação desejado, posições dos sensores acústicos e das fontes, o ângulo entre as direções de chegada das fontes, entre outros. O parâmetro tempo de reverberação, definido pelo termo  $T_{60}$ , corresponde ao tempo necessário para que o nível de pressão sonora decaia 60 dB.

O objetivo desta dissertação é estudar as limitações de algoritmos de separação cega de fontes em ambientes que possuam alta reverberação, utilizando processamento adaptativo e análises no domínio da frequência.

Para isso, no restante deste capítulo, foram apresentados o problema de separação de fontes, as características dos sinais de voz, o conceito de banco de filtros utilizado para as análises em subbandas e as medidas que podem ser utilizadas para avaliar o desempenho dos algoritmos. No Capítulo 2 foram exploradas diversas abordagens para o problema de separação cega e foi apresentado o algoritmo TRINICON (*Triple-N ICA for Convolutional Mixtures*) utilizado nos experimentos desta dissertação. No

Capítulo 3, foram analisadas diversas características dos sinais de voz para que fosse possível extrair a melhor combinação de parâmetros para os algoritmos testados. No Capítulo 4, foram utilizados dois algoritmos supervisionados adaptativos, o NLMS (*Normalized Least-Mean Square*) e o NSAF (*Normalized Subband Adaptive Filter*), para verificar as limitações do modelo de separação de fontes em banda cheia e em subbandas. No Capítulo 5, os parâmetros identificados no Capítulo 3 foram incorporados aos algoritmos e uma tentativa de melhoria em subbandas foi implementada. Por último, no Capítulo 6, as conclusões relevantes foram apresentadas.

## 1.1 O problema de separação de fontes

Misturas lineares de sinais podem ser definidas de acordo com a equação:

$$\mathbf{x}(n) = \mathbf{H}\mathbf{s}(n) \quad (1.1)$$

em que  $\mathbf{H}$  é a matriz de mistura de dimensões  $P \times Q$  sendo  $Q$  o número de fontes e  $P$  o número de sensores,

$$\mathbf{H} = \begin{bmatrix} h_{11} & h_{12} & \dots & h_{1Q} \\ h_{21} & h_{22} & \dots & h_{2Q} \\ \vdots & \vdots & \ddots & \vdots \\ h_{P1} & h_{P2} & \dots & h_{PQ} \end{bmatrix} \quad (1.2)$$

o vetor  $\mathbf{s}(n)$  é composto por todas as fontes da forma

$$\mathbf{s}(n) = \begin{bmatrix} s_1(n) & s_2(n) & \dots & s_Q(n) \end{bmatrix}^T \quad (1.3)$$

e  $\mathbf{x}(n)$  é o vetor contendo as misturas que chegam aos sensores, dado por

$$\mathbf{x}(n) = \begin{bmatrix} x_1(n) & x_2(n) & \dots & x_P(n) \end{bmatrix}^T \quad (1.4)$$

O problema de separação de fontes consiste em encontrar uma matriz de separação  $\mathbf{W}$  capaz de separar as misturas obtidas através do vetor  $\mathbf{x}(n)$ , ou seja,

$$\mathbf{y}(n) = \mathbf{W}\mathbf{x}(n) \quad (1.5)$$

em que

$$\mathbf{y}(n) = \begin{bmatrix} y_1(n) & y_2(n) & \dots & y_Q(n) \end{bmatrix}^T \quad (1.6)$$

e a matriz de separação  $\mathbf{W}$  de dimensões  $Q \times P$  definida por

$$\mathbf{W} = \begin{bmatrix} w_{11} & w_{12} & \dots & w_{1P} \\ w_{21} & w_{22} & \dots & w_{2P} \\ \vdots & \vdots & \ddots & \vdots \\ w_{Q1} & w_{Q2} & \dots & w_{QP} \end{bmatrix} \quad (1.7)$$

A estrutura geral de um problema de separação cega de fontes é ilustrada na Figura 1.3.

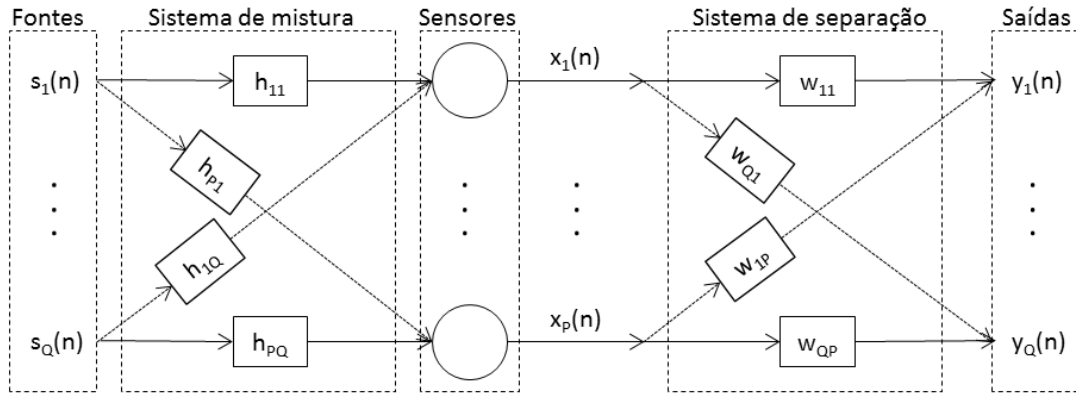


Figura 1.3: Estrutura geral do problema de separação de fontes

## 1.2 Tipos de misturas

As misturas podem ser classificadas como sub-determinadas, determinadas ou sobre-determinadas de acordo com o número de fontes e de sensores, conforme a Tabela 1.1.

Tabela 1.1: Tipos de misturas

Classificação	Relação entre o número de fontes e de sensores
Sobre-determinada	$Q < P$
Determinada	$Q = P$
Sub-determinada	$Q > P$

Além disso, as misturas lineares também podem ser classificadas como instantâneas ou convolutivas dependendo do seu sistema de misturas.

### 1.2.1 Misturas instantâneas

Nas misturas instantâneas, as observações dependem apenas dos valores das fontes no mesmo instante de tempo. Dessa forma, para a  $j$ -ésima fonte e para o  $i$ -ésimo

sensor, os elementos de  $\mathbf{H}$  são definidos por constantes como na equação:

$$[\mathbf{H}]_{ij} = [h_{ij}] \quad (1.8)$$

então, podemos descrever o sinal obtido através de um sensor  $i$  através da equação:

$$x_i(n) = \sum_{j=1}^Q h_{ij}s_j(n) \quad (1.9)$$

e a saída esperada para recuperar o sinal da  $j$ -ésima fonte

$$y_j(n) = \sum_{i=1}^P w_{ji}x_i(n) \quad (1.10)$$

### 1.2.2 Misturas convolutivas

As misturas convolutivas são aquelas compostas das fontes atrasadas devido à reverberação do ambiente. Assim, podemos definir a matriz de mistura conforme a equação:

$$[\mathbf{H}]_{ij} = \left[ \sum_{k=0}^{M-1} h_{ij}(k)z^{-k} \right] \quad (1.11)$$

em que  $M$  é o comprimento do filtro e  $z^{-k}$  é o operador atraso. O sinal da  $i$ -ésima mistura é dado por:

$$x_i(n) = \sum_{j=1}^Q \sum_{k=0}^{M-1} h_{ij}(k)s_j(n-k) \quad (1.12)$$

e a saída esperada para recuperar o sinal da  $j$ -ésima fonte

$$y_j(n) = \sum_{i=1}^P \sum_{k=0}^{L-1} w_{ji}(k)x_i(n-k) \quad (1.13)$$

em que  $L$  é o comprimento do filtro de separação.

## 1.3 Ambiguidades na separação

A solução para o problema de separação cega de fontes não evita que as possíveis ambiguidades descritas abaixo ocorram.

### 1.3.1 Ambiguidade da filtragem

Os sinais das fontes estimados podem estar escalonados, ou seja multiplicados por um fator, e com um atraso constante. Isto ocorre porque os algoritmos não tentam desfazer a reverberação do sinal.

De acordo com [8], a ambiguidade da filtragem pode ser reduzida para apenas uma ambiguidade de escalonamento se o comprimento do filtro de separação for maior ou igual ao comprimento do filtro de separação ótimo dado por:

$$L_{opt} = \frac{(Q - 1)(M - 1) + 1}{P - Q + 1} \quad (1.14)$$

### 1.3.2 Ambiguidade da permutação

A ordem das fontes estimadas não pode ser determinada sem nenhuma informação a priori, ou seja, o sinal de entrada 1 não vai estar necessariamente estimado pelo sinal de saída 1. Na Figura 1.4, por exemplo, o sinal de entrada 1 está sendo estimado pelo sinal de saída 3.

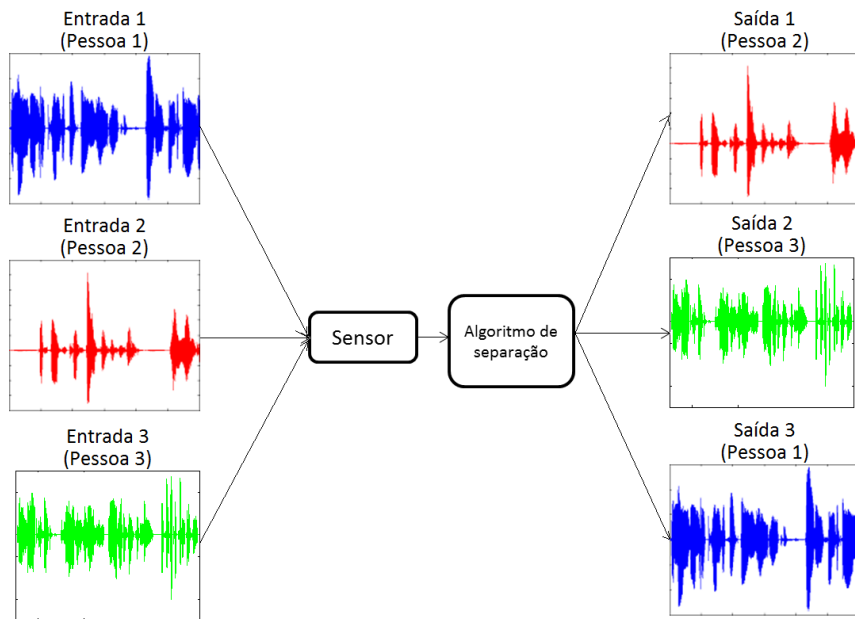


Figura 1.4: Ilustração do problema de ambiguidade da permutação

Esse problema se torna crítico para os algoritmos no domínio da frequência e já existem alguns métodos que resolvem essa ambiguidade utilizando o TDOA (*Time Difference of Arrival*) [9] ou o DOA (*Direction of Arrival*) [10], [11].

## 1.4 Características dos sinais de voz

Nesta dissertação, as características próprias dos sinais de voz serão exploradas para melhor aproveitamento dos algoritmos. Nesta seção, essas características são brevemente expostas e no Capítulo 3 serão mostrados os estudos feitos sobre cada uma.

Quanto à distribuição estatística das amostras do sinal de voz, a sua PDF (*Probability Density Function*) geralmente não é gaussiana. A não-gaussianidade pode ser explorada utilizando estatísticas de ordens elevadas, como por exemplo a curtose, que é uma medida estatística de quarta-ordem. A curtose normalizada, para a variável aleatória  $y$ , é definida por [12]:

$$\text{curtose}(y(n)) = \frac{E\{y(n)^4\}}{(E\{y(n)^2\})^2} - 3 \quad (1.15)$$

em que  $E\{\cdot\}$  representa o valor esperado.

Se a curtose de um sinal  $y(n)$  é próxima de zero, a distribuição de suas amplitudes é aproximadamente gaussiana; se a curtose é positiva, a distribuição é supergaussiana, como observado em sinais de voz; já se a curtose é negativa, a distribuição é subgaussiana. Os três tipos de PDFs estão ilustrados na Figura 1.5.

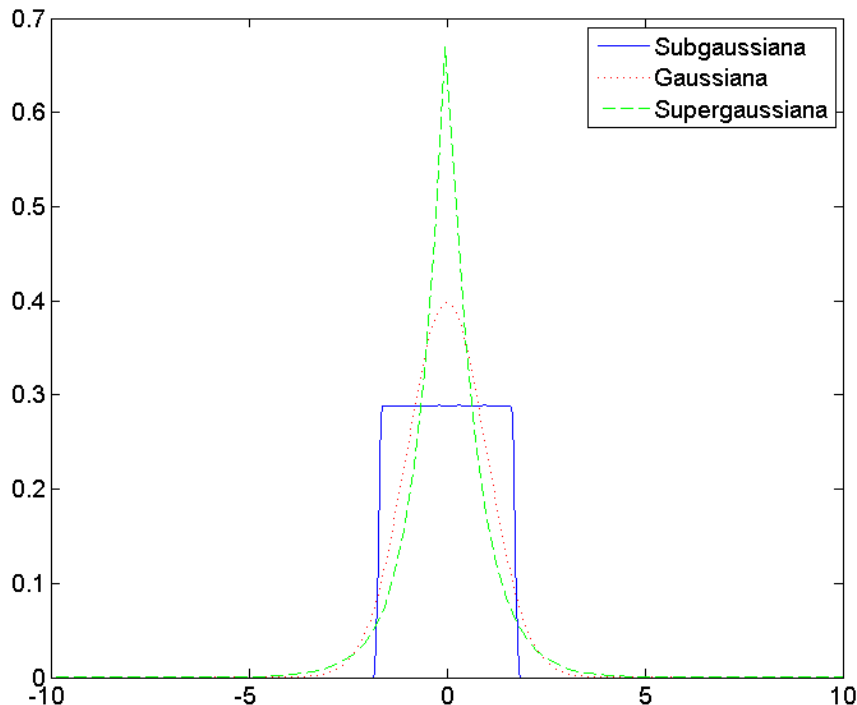


Figura 1.5: PDFs subgaussiana, gaussiana e supergaussiana

Na Tabela 1.2 são mostrados os valores de curtose para as fontes de voz masculina

e feminina antes da mistura e após a mistura dos mesmos sinais com tempos de reverberação diferentes.

Tabela 1.2: Valores de curtose

Sinal	Curtose
Voz masculina	15.845
Voz feminina	3.998
Mistura com reverberação de 100 ms	3.839
Mistura com reverberação de 250 ms	2.602
Mistura com reverberação de 500 ms	2.271

Na Figura 1.6 são mostradas as PDFs das fontes de voz masculina, feminina e da mistura dos dois sinais com reverberação de 500ms.

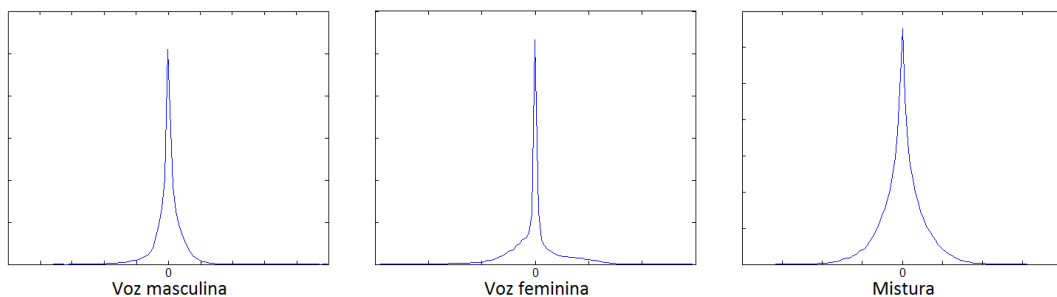


Figura 1.6: PDFs de sinais de voz e mistura

Observando a Tabela 1.2 e a Figura 1.6 percebemos que a curtose diminui quando os sinais são misturados e também quanto maior for o tempo de reverberação.

Quanto à estacionariedade, podemos dizer que o sinal de voz é um sinal não-estacionário, uma vez que suas estatísticas variam ao longo do tempo; porém, em curtos intervalos de tempo e principalmente durante os sons de vogais, pode-se observar uma condição de quase-estacionariedade. Portanto, para explorar essa característica, os sinais de voz podem ser analisados em blocos, normalmente de duração entre 10 e 30 ms. A Figura 1.7, mostra um trecho de 30 ms de um sinal de voz, onde pode-se claramente observar uma certa periodicidade do sinal, representando um trecho do sinal de voz mais bem comportado.

Quanto ao espectro, o sinal de voz apresenta um espectro colorido, pelas próprias características do trato vocal, que é o mecanismo humano que produz a voz.

Os algoritmos de separação cega que exploram cada uma dessas características separadamente serão descritos nas seções 2.1.1, 2.1.2 e 2.1.3. Já na seção 2.2, será apresentado o algoritmo que leva em consideração todas as três características.

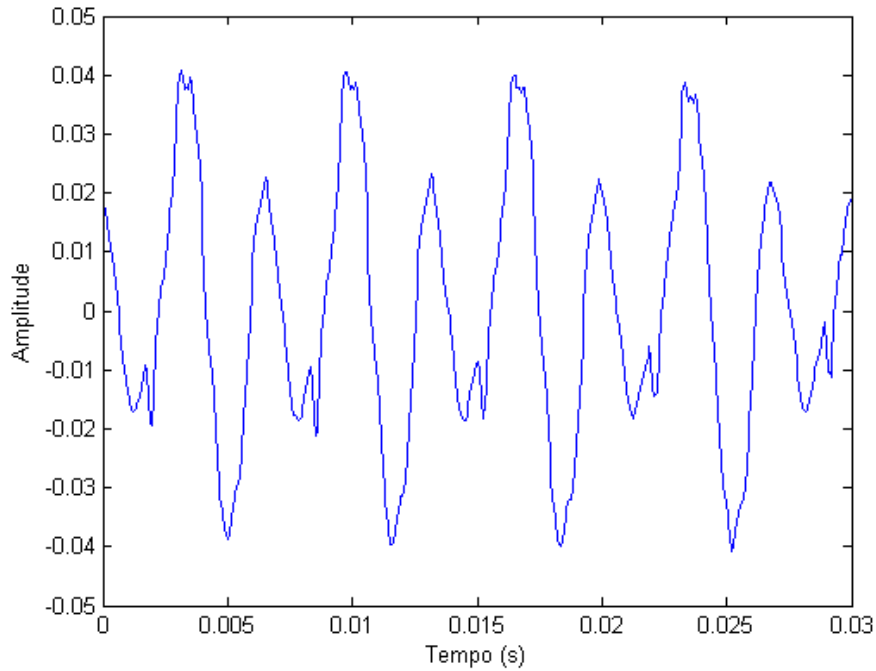


Figura 1.7: 30 ms de um sinal de voz

## 1.5 Banco de filtros

Para algumas análises das características de voz e também nos algoritmos de separação, os sinais foram decompostos em subbandas. Essa decomposição foi realizada utilizando bancos de filtros.

Um banco de filtros é um conjunto de filtros passa-banda o qual pode ser de análise, para decompor o sinal de entrada em um conjunto de sinais cada um ocupando uma parte da banda de frequência original, ou de síntese, para recuperar os vários sinais em subbandas em um sinal de saída.

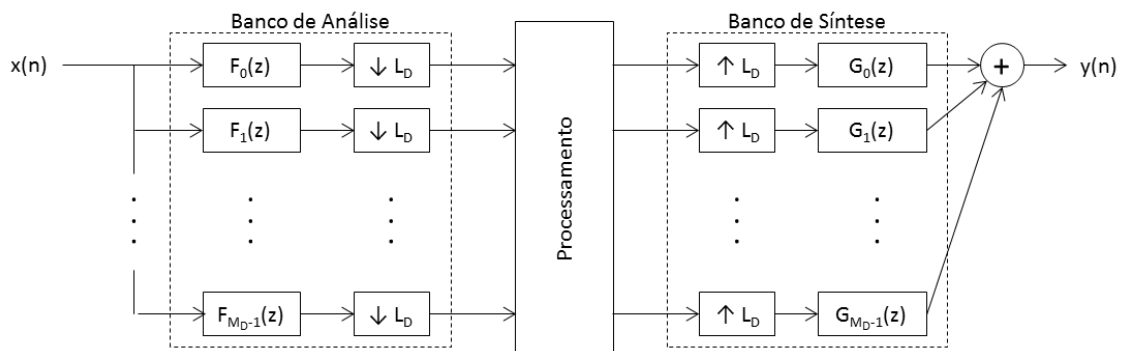


Figura 1.8: Bancos de filtros de análise e de síntese

Na Figura 1.8, o sinal de entrada  $x(n)$  é decomposto através dos filtros de análise  $\{F_0(z), \dots, F_{M_D-1}(z)\}$ , em que  $M_D$  é o número de subbandas, e em seguida o sinal



é decimado por um fator inteiro  $L_D$ . Depois de feito o processamento desejado, os sinais são interpolados pelo mesmo fator  $L_D$  da decimação para que a taxa de amostragem seja a mesma da entrada e então, os sinais são recombinaados através dos filtros de síntese  $\{G_0(z), \dots, G_{M_D-1}(z)\}$ .

O banco de filtros pode ser classificado como uniforme ou não-uniforme. No caso uniforme, os filtros de análise e síntese tem a mesma largura de faixa de passagem. Os bancos de filtros também podem ser maximamente decimados, quando o fator de decimação é igual ao número de subbandas ( $L_D = M_D$ ), ou não maximamente decimados ( $L_D < M_D$ ).

Os filtros de análise  $\{F_0(z), \dots, F_{M_D-1}(z)\}$  e de síntese  $\{G_0(z), \dots, G_{M_D-1}(z)\}$  devem ser escolhidos de forma que o efeito de *aliasing* (sobreposição do espectro) seja cancelado, ou mais ainda, de forma que o sinal  $y(n)$  não possua distorções de amplitude e fase, que são os chamados bancos de filtros de reconstrução perfeita [13].

### 1.5.1 Banco de filtros modulados por cosseno

Existem diversas formas de implementar um banco de filtros [14], porém, nesta seção será descrito apenas o banco de filtros modulados por cosseno, que foi o utilizado na dissertação.

Nesses sistemas, todos os filtros de análise são derivados de um único filtro protótipo. As vantagens desses sistemas são:

- Custo computacional: o custo computacional do processamento do sinal pelos filtros de análise equivale ao de um único filtro mais o da modulação do sinal, que pode ser implementada de forma rápida através da DCT (*Discrete Cosine Transform*). O processamento relativo aos filtros de síntese tem o mesmo custo que o dos filtros de análise.
- Baixo número de parâmetros para otimizar: durante a fase de projeto, apenas os coeficientes do filtro protótipo precisam ser otimizados.

As expressões para as respostas ao impulso dos filtros de análise  $f_k(n)$  e de síntese  $g_k(n)$  são as seguintes[14]:

$$f_k(n) = 2p_0(n)\cos\left(\frac{\pi}{M_D}(k + 0.5)\left(n - \frac{N}{2}\right) + \phi_k\right), \quad 0 \leq k \leq M_D - 1 \quad (1.16)$$

$$g_k(n) = 2p_0(n)\cos\left(\frac{\pi}{M_D}(k + 0.5)\left(n - \frac{N}{2}\right) - \phi_k\right), \quad 0 \leq k \leq M_D - 1 \quad (1.17)$$

em que  $p_0(n)$  é a resposta ao impulso do filtro protótipo, normalmente passa-baixa com frequência de corte  $\pi/2M_D$ , cujos coeficientes devem ser otimizados para que o filtro atinja as condições necessárias para reconstrução perfeita.

## 1.6 Medidas de avaliação

Para avaliar o desempenho dos algoritmos de separação, foram utilizadas as medidas descritas em [15]. Considerando  $y_j$  o sinal estimado da  $j$ -ésima fonte após a separação, é possível decompor  $y_j$  da seguinte forma:

$$y_j = s_{fonte} + e_{interf} + e_{artef} + e_{ruído} \quad (1.18)$$

em que  $s_{fonte}$  é o sinal original da  $j$ -ésima fonte com alguma distorção aceitável,  $e_{interf}$  é a parcela da estimativa correspondente a interferência das outras fontes,  $e_{ruído}$  é a parcela da estimativa que corresponde a ruídos dos sensores e  $e_{artef}$  é a parcela da estimativa que corresponde a presença de artefatos, ou seja, erros que não provém nem de interferências nem do ruído dos sensores.

Um procedimento para obtenção dessas componentes está descrito em [15], o qual requer o conhecimento do sinal da fonte original. Assim, as medidas de avaliação são definidas por:

- Razão Sinal-Interferência (SIR - *Signal to Interferences Ratio*): mede a razão entre as potências dos sinais da fonte e das outras fontes interferentes, em dB.

$$SIR := 10 \log_{10} \frac{\|s_{fonte}\|^2}{\|e_{interf}\|^2} \quad (1.19)$$

- Razão Sinal-Ruído (SNR - *Signal to Noise Ratio*): mede a razão entre as potências dos sinais da fonte e do ruído, em dB.

$$SNR := 10 \log_{10} \frac{\|s_{fonte} + e_{interf}\|^2}{\|e_{ruído}\|^2} \quad (1.20)$$

- Razão Sinal-Distorção (SDR - *Signal to Distortion Ratio*): mede a razão entre as potências dos sinais da fonte e dos sinais correspondente às distorções causadas por transformações não-lineares, janelamento, ruído e outras interferências não relacionadas às outras fontes, em dB.

$$SDR := 10 \log_{10} \frac{\|s_{fonte}\|^2}{\|e_{interf} + e_{ruído} + e_{artef}\|^2} \quad (1.21)$$

- Razão Sinal-Artefatos (SAR - *Signal to Artifacts Ratio*): mede a razão entre

as potências dos sinais da fonte estimados e dos artefatos.

$$SAR := 10 \log_{10} \frac{\|s_{fonte} + e_{interf} + e_{ruído}\|^2}{\|e_{artef}\|^2} \quad (1.22)$$

Em [16], o autor de [15] disponibilizou o código-fonte MATLAB com as funções para o cálculo das medidas descritas.

# Capítulo 2

## Separação cega de fontes

Neste capítulo serão descritas as diferentes abordagens para cada uma das características do sinal de voz separadamente e por último será apresentado o algoritmo TRINICON que considera todas as características.

### 2.1 Separação cega através de decomposição de autovalor generalizada

Em [17], os autores propõe a solução para encontrar a matriz de separação  $\mathbf{W}$  através de autovalores generalizados que simultaneamente diagonalizem as matrizes de covariância das observações e uma matriz adicional simétrica cuja forma depende de suposições particulares para cada abordagem (não-gaussianidade, não-estacionariedade ou não-branco).

Portanto, sendo o vetor  $\mathbf{x}(t)$  definido pelos valores dados por cada sensor no tempo

$$\mathbf{x}(t) = \begin{bmatrix} x_1(t) & x_2(t) & \dots & x_P(t) \end{bmatrix}^T \quad (2.1)$$

e o vetor  $\mathbf{s}(t)$  definido pelos valores dados por cada fonte no tempo

$$\mathbf{s}(t) = \begin{bmatrix} s_1(t) & s_2(t) & \dots & s_P(t) \end{bmatrix}^T \quad (2.2)$$

temos que, para a matriz de mistura  $\mathbf{H}$ :

$$\mathbf{x}(t) = \mathbf{H}\mathbf{s}(t) \quad (2.3)$$

Multiplicando a equação (2.3) à esquerda por  $\mathbf{W}^H$ , temos que

$$\mathbf{W}^H \mathbf{x}(t) = \mathbf{W}^H \mathbf{H}\mathbf{s}(t) \quad (2.4)$$

Portanto, para recuperar os sinais através de  $\mathbf{x}(t)$ , basta encontrar a matriz  $\mathbf{W}^H$  tal que  $\mathbf{W}^H \mathbf{H} = \mathbf{I}$ , em que  $\mathbf{I}$  é a matriz identidade.

Podemos admitir que para sinais de fontes independentes ou decorrelacionadas a média no tempo da matriz de covariância  $\mathbf{R}_x$  é diagonal e é dada por

$$\mathbf{R}_x = \sum_t E[\mathbf{x}(t)\mathbf{x}^H(t)] = \mathbf{H}\mathbf{R}_s\mathbf{H}^H \quad (2.5)$$

Assim como a matriz de covariância, podemos utilizar outras estatísticas cruzadas que tenham a mesma propriedade de diagonalização, formando uma matriz  $\mathbf{Q}_x$  que satisfaz a equação:

$$\mathbf{Q}_x = \mathbf{H}\mathbf{Q}_s\mathbf{H}^H \quad (2.6)$$

sendo  $\mathbf{Q}_s$  a matriz com as estatísticas correspondentes das fontes. Essas duas condições são suficientes para separação de fontes. Multiplicando à direita a equação (2.5) por  $\mathbf{W}$  e a equação (2.6) por  $\mathbf{W}\mathbf{Q}_s^{-1}$  temos que

$$\mathbf{R}_x \mathbf{W} = \mathbf{H}\mathbf{R}_s\mathbf{H}^H \mathbf{W} = \mathbf{H}\mathbf{R}_s \quad (2.7)$$

e

$$\mathbf{Q}_x \mathbf{W}\mathbf{Q}_s^{-1} = \mathbf{H}\mathbf{Q}_s\mathbf{H}^H \mathbf{W}\mathbf{Q}_s^{-1} = \mathbf{H} \quad (2.8)$$

Combinando as equações (2.7) e (2.8), obtemos a seguinte equação de autovalor generalizada

$$\mathbf{R}_x \mathbf{W} = \mathbf{Q}_x \mathbf{W}\mathbf{Q}_s^{-1} \mathbf{R}_s = \mathbf{Q}_x \mathbf{W} \Lambda \quad (2.9)$$

em que

$$\Lambda = \mathbf{Q}_s^{-1} \mathbf{R}_s \quad (2.10)$$

Nas próximas seções, será dada a forma de  $\mathbf{Q}_x$  assumindo cada uma das diferentes características do sinal de voz.

### 2.1.1 Abordagem para sinais não-estacionários

Nesta abordagem, são utilizadas estatísticas de segunda-ordem fornecidas por sinais não-estacionários para encontrar a matriz  $\mathbf{W}$  que torna os sinais independentes.

Assim, a não-estacionariedade dos sinais de voz é explorada através de uma diagonalização simultânea das matrizes de correlação de saída para diferentes instantes de tempo. Ou seja,  $\mathbf{Q}_x = \mathbf{R}_x(t)$ , para qualquer tempo  $t$ .

$$\mathbf{R}_x(t) = E[\mathbf{x}(t)\mathbf{x}^H(t)] = \mathbf{H}E[\mathbf{s}(t)\mathbf{s}^H(t)]\mathbf{H}^H = \mathbf{H}\mathbf{R}_s(t)\mathbf{H}^H \quad (2.11)$$

Essa abordagem foi utilizada nos algoritmos desenvolvidos em [17], [18], [19] e

[20].

### 2.1.2 Abordagem para sinais não-brancos

O espectro colorido dos sinais de voz também é explorado através da diagonalização simultânea das matrizes de correlação de saída, como no caso de sinais não-estacionários, porém para múltiplos atrasos [21].

Assim, a forma de  $\mathbf{Q}$  para esta abordagem será  $\mathbf{Q}_x = \mathbf{R}_x(\tau)$ .

$$\mathbf{R}_x(\tau) = E[\mathbf{x}(t)\mathbf{x}^H(t + \tau)] = \mathbf{H}E[\mathbf{s}(t)\mathbf{s}^H(t + \tau)]\mathbf{H}^H = \mathbf{H}\mathbf{R}_s(\tau)\mathbf{H}^H \quad (2.12)$$

Essa abordagem foi empregada em [17], [21], [22] e [23].

### 2.1.3 Abordagem para sinais não-gaussianos

Os algoritmos que utilizam essa abordagem são derivados do método ICA (*Independent Component Analysis*). As restrições básicas do ICA são:

- As componentes devem ser estatisticamente independentes
- As componentes devem ter distribuição não-gaussiana

A ICA pode ser realizada através de várias funções-objetivo e algoritmos de otimização.

Uma função-objetivo, usualmente empregada corresponde à busca da matriz de separação  $\mathbf{W}$  através da maximização da não-gaussianidade, já que, pelo teorema central, a mistura é mais gaussiana do que as fontes separadas.

A não-gaussianidade pode ser medida através da curtose ou da negentropia [12], e a partir daí são aplicados algoritmos que procuram maximizar essa medida, como o algoritmo gradiente.

A solução pela decomposição do autovalor generalizada é obtida utilizando a matriz  $\mathbf{Q}_x$  dada pelo somatório sobre cumulantes de quarta ordem, ou seja,

$$\begin{aligned} \mathbf{Q}_x &= \sum_k \text{Cumulantes}(s_i, s_j, s_k, s_l) \\ &= E[\mathbf{x}^H \mathbf{x} \mathbf{x} \mathbf{x}^H] - \mathbf{R}_x \text{Trace}(\mathbf{R}_x) - E[\mathbf{x} \mathbf{x}^T] E[\mathbf{x}^* \mathbf{x}^H] - \mathbf{R}_x \mathbf{R}_x \end{aligned} \quad (2.13)$$

Outros algoritmos com funções-objetivo diferentes, por exemplo minimização da informação mútua, estão descritos em [12]. Algoritmos que tratam de misturas convolutivas utilizando ICA podem ser encontrados em [24] e [25].

## 2.2 Abordagem utilizando as três características

O algoritmo que utiliza todas as características citadas anteriormente é o *framework* TRINICON (*Triple-N ICA for convolutive mixtures*) [26].

O algoritmo TRINICON explora a não-gaussianidade através de estatísticas de ordens elevadas utilizadas para formular sua função-custo.

Já o espectro colorido é explorado através da diagonalização simultânea de matrizes de correlação das saídas para múltiplos atrasos, neste caso, será utilizado o parâmetro  $D$  que representa o número de intervalos de tempo considerados, que pode ser no máximo igual ao comprimento do filtro de separação, dado pelo parâmetro  $L$ .

E por fim, a não-estacionariedade que é explorada através da diagonalização simultânea das matrizes de correlação de saída para diferentes instantes de tempo, para isso, será definido o parâmetro  $N$  que representa o tamanho dos blocos de saída e deve ser tal que  $N \geq PD$ , em que  $P$  é o número de sensores.

### 2.2.1 Notação matricial

Para derivar o algoritmo TRINICON, os autores reformularam a convolução linear que fornece os valores do sinal de saída com a seguinte notação matricial:

$$\mathbf{y}_q(m, j) = \sum_{p=1}^P \mathbf{x}_p(m, j) \mathbf{W}_{pq}(m) \quad (2.14)$$

em que  $m$  é o índice do bloco e  $j$  é o deslocamento no tempo dentro do bloco de tamanho  $N$  e varia de 0 a  $N - 1$ .

Os vetores  $\mathbf{x}_p(m, j)$  e  $\mathbf{y}_q(m, j)$  são dados por:

$$\mathbf{x}_p(m, j) = \begin{bmatrix} x_p(mL + j) & \dots & x_p(mL - 2L + 1 + j) \end{bmatrix} \quad (2.15)$$

$$\mathbf{y}_q(m, j) = \begin{bmatrix} y_q(mL + j) & \dots & y_q(mL - D + 1 + j) \end{bmatrix} \quad (2.16)$$

em que  $D$  representa o número de atrasos levados em conta.

Para expressar a convolução linear, a matriz  $\mathbf{W}_{pq}$  de dimensão  $2L \times D$  que contém os coeficientes do filtro de separação em cada coluna, é dada por:

$$\mathbf{W}_{pq}(m) = \begin{bmatrix} w_{pq,0} & 0 & \cdots & 0 \\ w_{pq,1} & w_{pq,0} & \ddots & \vdots \\ \vdots & w_{pq,1} & \ddots & 0 \\ w_{pq,L-1} & \vdots & \ddots & w_{pq,0} \\ 0 & w_{pq,L-1} & \ddots & w_{pq,1} \\ \vdots & & \ddots & \vdots \\ 0 & \cdots & 0 & w_{pq,L-1} \\ 0 & \cdots & 0 & 0 \\ \vdots & \cdots & \vdots & \vdots \\ 0 & \cdots & 0 & 0 \end{bmatrix} \quad (2.17)$$

Para permitir uma notação mais compacta e já considerando o número de fontes igual ao número de sensores, ou seja,  $Q = P$ , as equações (2.14) a (2.17) podem ser reescritas na forma abaixo:

$$\mathbf{y}(m, j) = \mathbf{x}(m, j)\mathbf{W}(m) \quad (2.18)$$

$$\mathbf{x}(m, j) = \begin{bmatrix} \mathbf{x}_1(m, j) & \cdots & \mathbf{x}_P(m, j) \end{bmatrix} \quad (2.19)$$

$$\mathbf{y}(m, j) = \begin{bmatrix} \mathbf{y}_1(m, j) & \cdots & \mathbf{y}_P(m, j) \end{bmatrix} \quad (2.20)$$

$$\mathbf{W}(m) = \begin{bmatrix} \mathbf{W}_{11}(m) & \cdots & \mathbf{W}_{1P}(m) \\ \vdots & \ddots & \vdots \\ \mathbf{W}_{P1}(m) & \cdots & \mathbf{W}_{PP}(m) \end{bmatrix} \quad (2.21)$$

## 2.2.2 Função custo

Baseando-se em uma generalização de informação mútua de Shannon [27], foi definida a seguinte função custo levando em consideração todas as três características do sinal de voz:

$$\mathcal{J} = - \sum_{i=0}^m \beta(i, m) \frac{1}{N} \sum_{j=0}^{N-1} \{ \log(\hat{p}_D(\mathbf{y}_1(i, j)) \cdots \hat{p}_D(\mathbf{y}_P(i, j))) - \log(\hat{p}_{PD}(\mathbf{y}_1(i, j)\Lambda_1, \dots, \mathbf{y}_P(i, j)\Lambda_P)) \} \quad (2.22)$$

em que  $\hat{p}_D(\cdot)$  representa a PDF (*Probability Density Function*) de cada sensor de dimensão  $D$  e  $\hat{p}_{PD}(\cdot)$  representa a PDF conjunta de dimensão  $PD$  para todos os sensores.

Como mencionado anteriormente, o parâmetro  $D$  expressa o tamanho da



memória ou do atraso para modelar o espectro colorido do sinal e deve ser escolhido entre  $1 \leq D \leq L$ . O parâmetro  $N$  representa o tamanho do bloco em que o sinal seria considerado estacionário. Para uma estimação adequada da PDF, devem ser considerados  $N > PD$  instantes de tempo. As matrizes  $\Lambda_p$ , para  $p = 1, \dots, P$ , representam filtros nas saídas dos sinais para melhorar a convergência para sinais não-estacionários, removendo as restrições de magnitude.

A função  $\beta(i, m)$  é uma função peso para permitir implementações *offline*, *online* e *block-online* do algoritmo e deve ser normalizada de acordo com:

$$\sum_{i=0}^{\infty} \beta(i, m) = 1 \quad (2.23)$$

- Implementação *offline*:  $\beta(i, m) = 1/B$ , em que B é o número total de blocos.
- Implementação *online*:

$$\beta(i, m) = \begin{cases} (1 - \lambda)\lambda^{(m-i)} & \text{para } 0 \leq i \leq m \\ 0 & \text{caso contrário} \end{cases} \quad (2.24)$$

em que  $\lambda$  é o fator de esquecimento, geralmente um pouco menor do que 1.

- Implementação *block-online*:  $\beta(i, m) = 1/K$ , em que apenas os últimos K blocos são levados em consideração pelo algoritmo.

### 2.2.3 Regra de atualização

Para minimização da função custo, foi feito o cálculo do gradiente natural [28] de  $\mathcal{J}(m)$  em relação a matriz de separação  $\mathbf{W}(m)$ :

$$\Delta \mathbf{W} \propto \mathbf{W} \mathbf{W}^H \frac{\partial \mathcal{J}}{\partial \mathbf{W}^*} \quad (2.25)$$

então, é obtida a seguinte regra de atualização:

$$\mathbf{W}(m) = \mathbf{W}(m-1) - \mu \Delta \mathbf{W}(m) \quad (2.26)$$

$$\Delta \mathbf{W}(m) = - \sum_{i=0}^m \beta(i, m) \sum_{j=0}^{N-1} \mathbf{W}(i) \mathbf{y}^H(i, j) \Phi(\mathbf{y}(i, j)) - \mathbf{I} \quad (2.27)$$

e a função *score*:

$$\Phi(\mathbf{y}(i, j)) = \begin{bmatrix} \frac{\partial \hat{p}_D(\mathbf{y}_1(i, j))}{\partial \mathbf{y}_1(i, j)} & \dots & \frac{\partial \hat{p}_D(\mathbf{y}_P(i, j))}{\partial \mathbf{y}_P(i, j)} \\ \hat{p}_D(\mathbf{y}_1(i, j)) & \dots & \hat{p}_D(\mathbf{y}_P(i, j)) \end{bmatrix} \quad (2.28)$$

## 2.2.4 Algoritmo TRINICON para PDF laplaciana

Para que fosse possível incluir todas as estatísticas necessárias do sinal foi utilizado o modelo de SIRPs (*Spherically invariant random processes*) [29] que representa com a devida precisão os sinais de voz.

A forma geral da PDF para SIRPs correlacionados de ordem  $D$  é dada, para uma função  $f_D$  devidamente escolhida, por:

$$\hat{p}_D(\mathbf{y}_p(i, j)) = \frac{1}{\sqrt{\pi^D \det(\mathbf{R}_{pp}(i))}} f_D(\mathbf{y}_p(i, j) \mathbf{R}_{pp}^{-1} \mathbf{y}_p^H(i, j)) \quad (2.29)$$

Depois de algumas derivações descritas em [26], é possível chegar a seguinte regra de atualização considerando a função  $f_D(\cdot)$  para PDF laplaciana:

$$\Delta \mathbf{W}(m) = \sum_{i=0}^m \beta(i, m) \mathbf{W}(i) \begin{bmatrix} \mathbf{0} & \tilde{\mathbf{R}}_{12}(i) \mathbf{R}_{22}^{-1}(i) \\ \tilde{\mathbf{R}}_{21}(i) \mathbf{R}_{11}^{-1}(i) & \mathbf{0} \end{bmatrix} \quad (2.30)$$

em que:

$$\tilde{\mathbf{R}}_{pq}(i) = \mathbf{Y}_p^H(i) \tilde{\Lambda}_q(i) \mathbf{Y}_q(i) \quad (2.31)$$

$$\tilde{\Lambda}(i) = -\phi_D(\text{diag}(\mathbf{Y}_q(i) \mathbf{R}_{qq}^{-1}(i) \mathbf{Y}_q^H(i))) \quad (2.32)$$

$$\phi_D(s) = \frac{1}{D - \sqrt{2s} \frac{K_{D/2+1}(\sqrt{2s})}{K_{D/2}(\sqrt{2s})}} \quad (2.33)$$

e  $K_\nu$  representa a função de Bessel modificada de  $\nu$ -ésima ordem.

## 2.2.5 Algoritmo TRINICON para PDF gaussiana

Ao utilizar uma PDF gaussiana, o algoritmo passa a utilizar apenas as características de não-estacionariedade e espectro colorido, tornando o algoritmo mais simples computacionalmente.

Assim, para a seguinte PDF gaussiana:

$$\hat{p}_D(\mathbf{y}_p(i, j)) = \frac{1}{\sqrt{(2\pi)^D \det(\mathbf{R}_{pp}(i))}} e^{\frac{1}{2} \mathbf{y}_p(i, j) \mathbf{R}_{pp}^{-1}(i) \mathbf{y}_p^H(i, j)} \quad (2.34)$$

a regra de atualização para duas fontes e dois sensores é a seguinte:

$$\Delta \mathbf{W}(m) = \sum_{i=0}^m \beta(i, m) \mathbf{W}(i) \begin{bmatrix} \mathbf{0} & \mathbf{R}_{12}(i) \mathbf{R}_{22}^{-1}(i) \\ \mathbf{R}_{21}(i) \mathbf{R}_{11}^{-1}(i) & \mathbf{0} \end{bmatrix} \quad (2.35)$$

em que:

$$\mathbf{R}_{pq}(i) = \mathbf{Y}_p^H(i) \mathbf{Y}_q(i) \quad (2.36)$$

$$\mathbf{Y}_q(i) = \left[ \mathbf{y}_q^T(i, 0) \quad \dots \quad \mathbf{y}_q^T(i, N-1) \right]^T \quad (2.37)$$

# Capítulo 3

## Estudo das características dos sinais de voz

Neste capítulo, cada característica do sinal de voz será investigada e comparada em diferentes subbandas para posteriormente orientar as escolhas dos parâmetros para o algoritmo de separação. Na última seção, os resultados de cada característica serão comentados.

Nas simulações foram utilizados quatro sinais de voz, sendo dois de voz masculina e dois de voz feminina, com duração de 10 segundos e frequência de amostragem 16 kHz. Estes sinais encontram-se disponibilizados em [30].

### 3.1 Não-gaussianidade

Nesta seção, serão apresentadas as distribuições e os valores de curtose de algumas fontes de voz em banda cheia e em subbandas para que seja possível comparar e compreender o comportamento das distribuições do sinal para cada caso.

As Figuras 3.1 a 3.4 apresentam os histogramas correspondente às distribuições das amostras de quatro sinais de voz distintos, duas vozes masculina e outras duas femininas, de 10 segundos de duração, considerando os sinais em banda completa e separados em duas bandas de frequência. Para a decomposição em subbandas empregou-se o banco de filtros modulado por cosseno, descrito na seção 1.5, com filtro protótipo de comprimento 16.

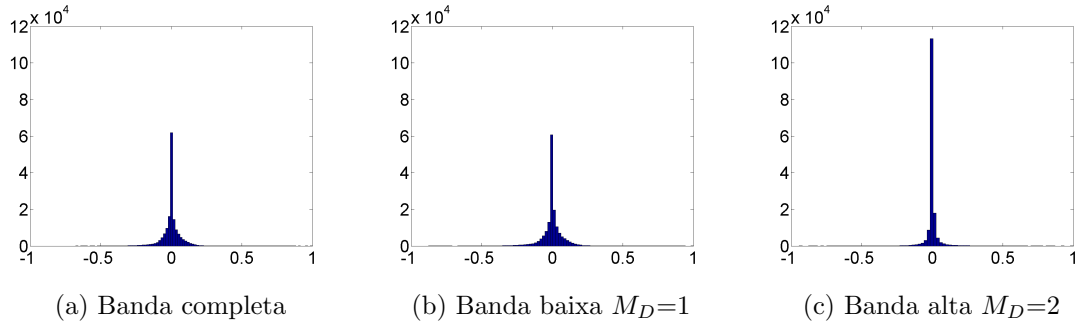


Figura 3.1: Histogramas - Decomposição em 2 subbandas - Voz masculina 1

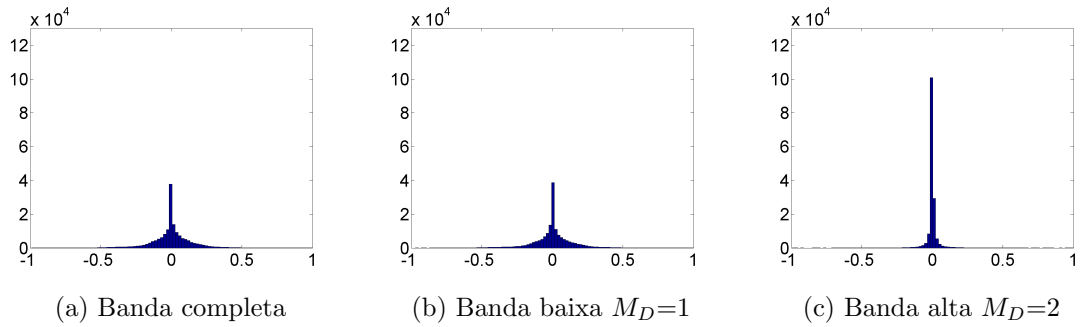


Figura 3.2: Histogramas - Decomposição em 2 subbandas - Voz masculina 2

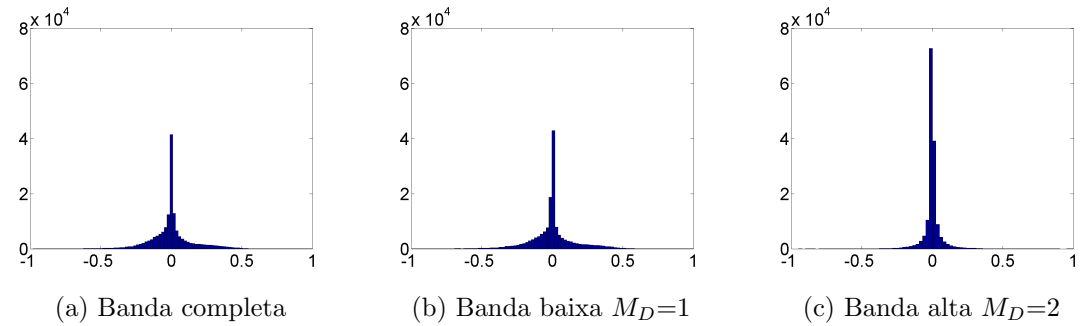


Figura 3.3: Histogramas - Decomposição em 2 subbandas - Voz feminina 1

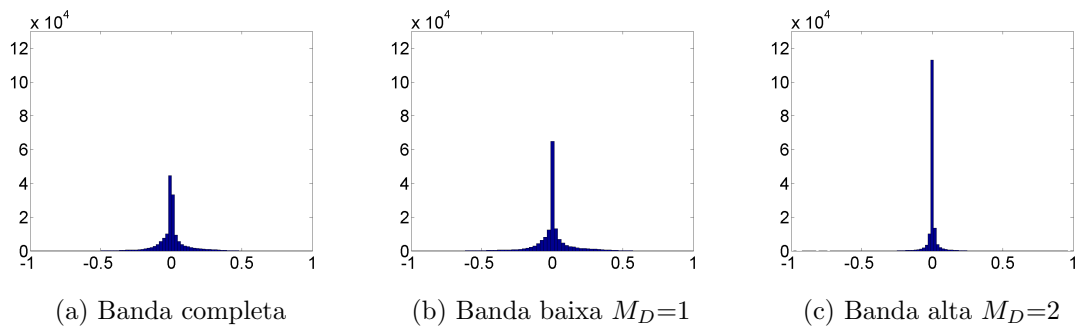


Figura 3.4: Histogramas - Decomposição em 2 subbandas - Voz feminina 2

A Tabela 3.1 apresenta os valores de curtose para os sinais correspondentes a cada histograma, ou seja, os valores para os quatro sinais de voz distintos, tanto em banda cheia quanto decompostos em duas subbandas.

Tabela 3.1: Valores de curtose para decomposição em 2 subbandas

Subbanda	Voz masculina 1	Voz masculina 2	Voz feminina 1	Voz feminina 2
Completa	15.5770	4.2912	3.9659	7.6421
$M_D=1$	15.9385	4.5294	3.9698	7.9923
$M_D=2$	47.1086	61.1777	26.6727	40.0679

As Figuras 3.5 a 3.8 apresentam os histogramas correspondente às amostras dos mesmos quatro sinais de voz citados na seção anterior considerando os sinais separados em quatro bandas de frequência. Para a decomposição em subbandas empregou-se o banco de filtros modulado por cosseno, descrito na seção 1.5, com filtro protótipo de comprimento 32.

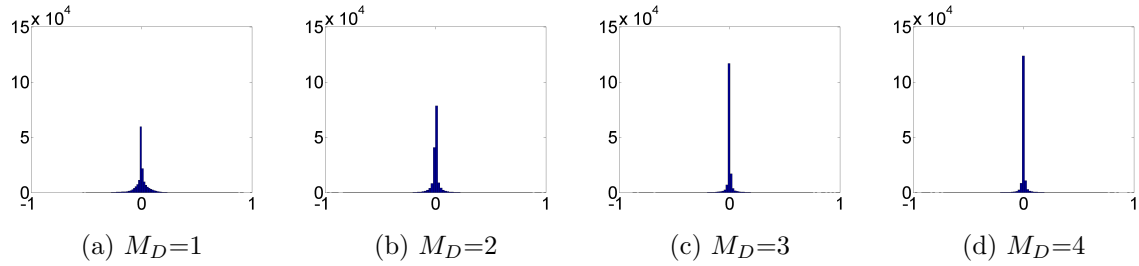


Figura 3.5: Histogramas - Decomposição em 4 subbandas - Voz masculina 1

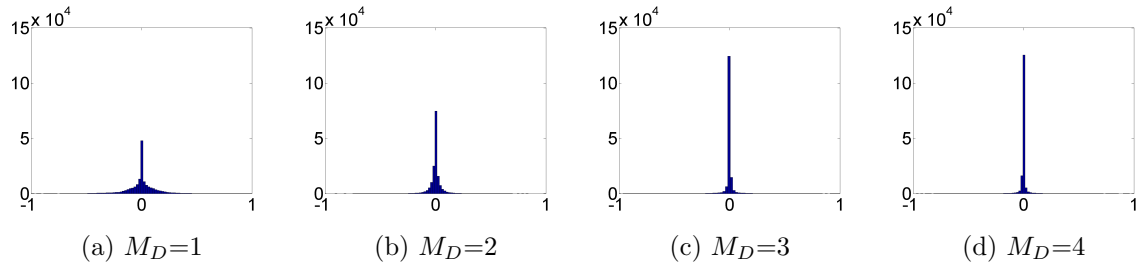


Figura 3.6: Histogramas - Decomposição em 4 subbandas - Voz masculina 2

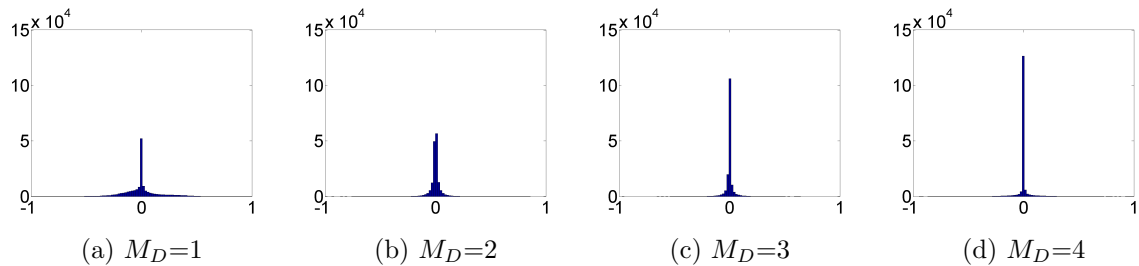


Figura 3.7: Histogramas - Decomposição em 4 subbandas - Voz feminina 1

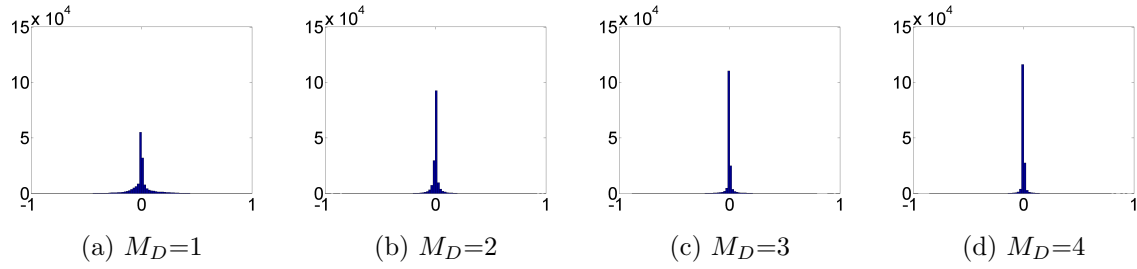


Figura 3.8: Histogramas - Decomposição em 4 subbandas - Voz feminina 2

A Tabela 3.2 apresenta os valores de curtose para os sinais correspondentes a cada histograma, ou seja, os valores para os quatro sinais de voz distintos decompostos em quatro subbandas.

Tabela 3.2: Valores de curtose para decomposição em 4 subbandas

Subbanda	Voz masculina 1	Voz masculina 2	Voz feminina 1	Voz feminina 2
$M_D=1$	16.5251	4.7694	3.7705	7.9865
$M_D=2$	40.8147	22.9327	27.4362	55.6464
$M_D=3$	48.6645	64.3155	30.2464	41.8458
$M_D=4$	46.5691	74.0007	24.3919	94.9726

As Figuras 3.9 a 3.12 apresentam os histogramas correspondente às amostras dos mesmos quatro sinais de voz citados anteriormente considerando os sinais separados em oito bandas de frequência. Para a decomposição em subbandas empregou-se o banco de filtros modulado por cosseno, descrito na seção 1.5, com filtro protótipo de comprimento 64.

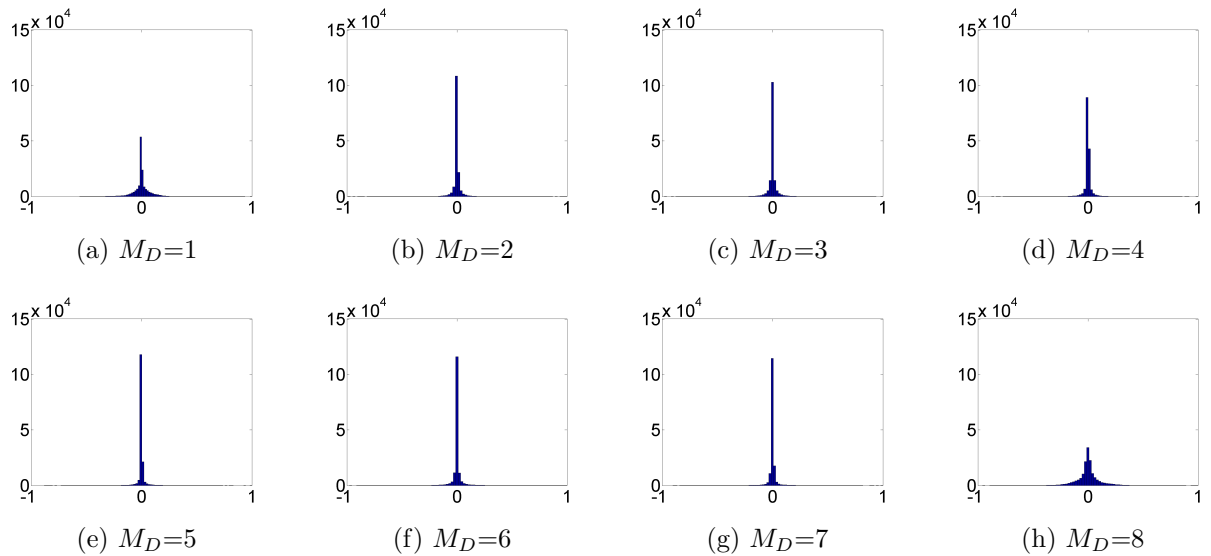


Figura 3.9: Histogramas - Decomposição em 8 subbandas - Voz masculina 1

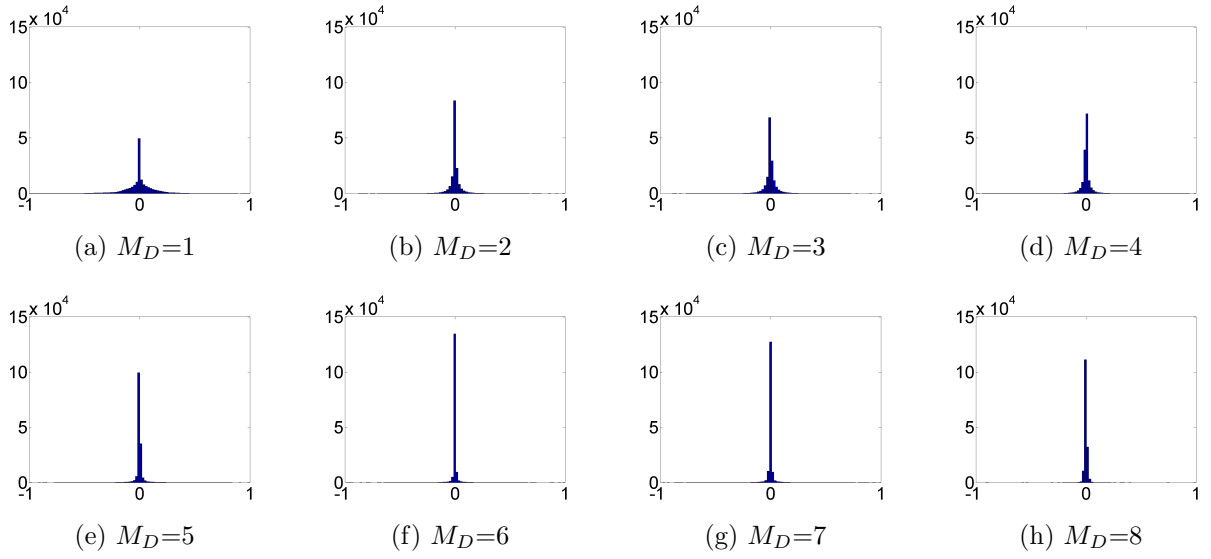


Figura 3.10: Histogramas - Decomposição em 8 subbandas - Voz masculina 2

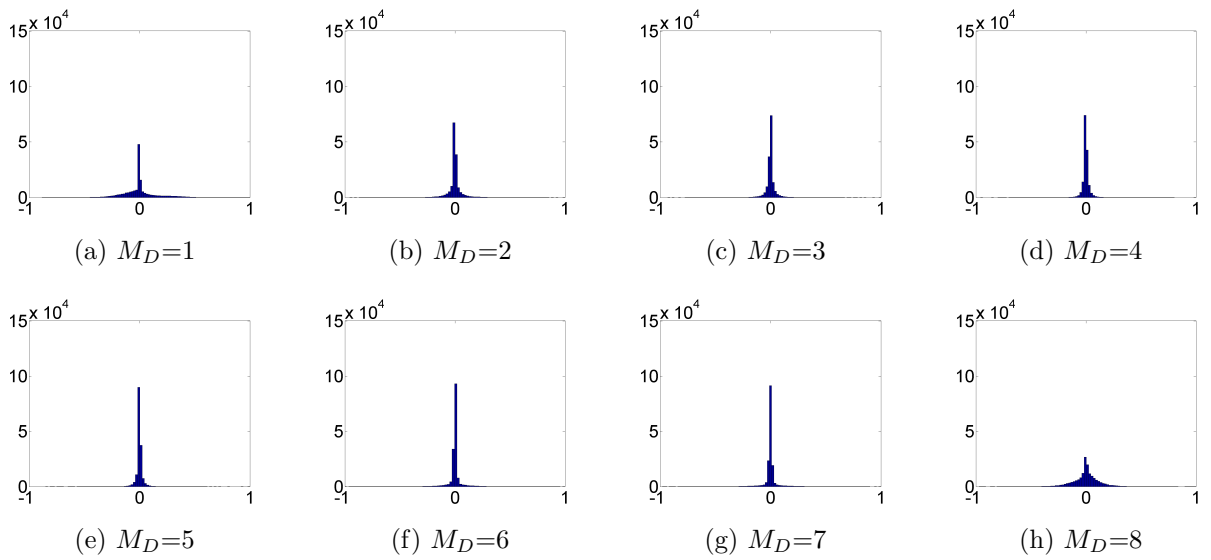


Figura 3.11: Histogramas - Decomposição em 8 subbandas - Voz feminina 1

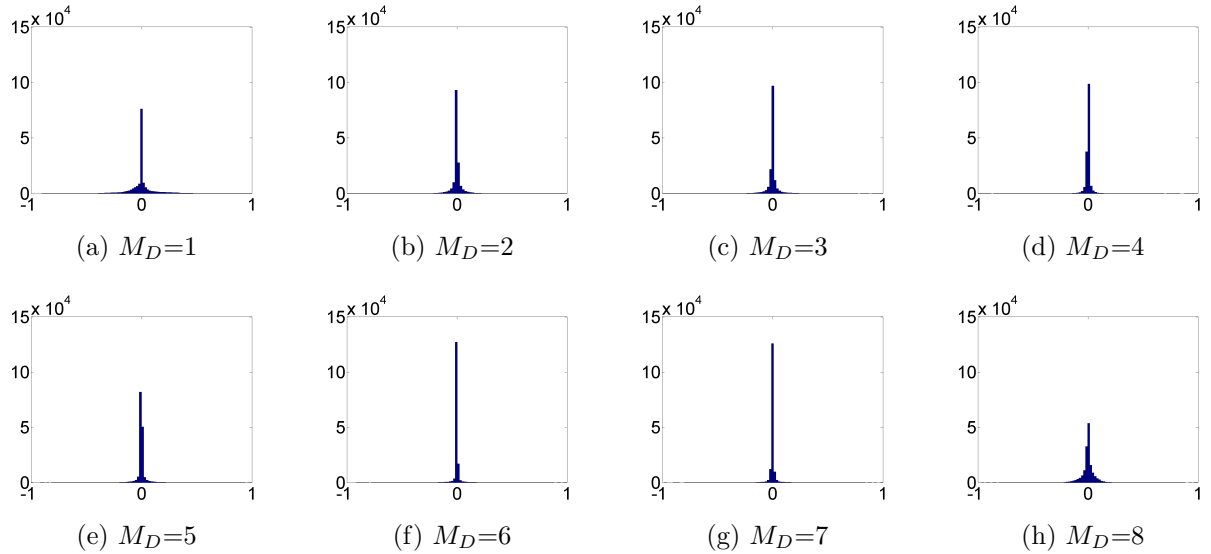


Figura 3.12: Histogramas - Decomposição em 8 subbandas - Voz feminina 2

A Tabela 3.3 apresenta os valores de curtose para os sinais correspondentes a cada histograma, ou seja, os valores para os quatro sinais de voz distintos decompostos em oito subbandas.

Tabela 3.3: Valores de curtose para decomposição em 8 subbandas

Subbanda	Voz masculina 1	Voz masculina 2	Voz feminina 1	Voz feminina 2
$M_D=1$	11.7798	4.7233	3.4350	6.9111
$M_D=2$	77.8060	26.7734	20.1181	62.4879
$M_D=3$	44.7064	21.3658	31.1324	39.5715
$M_D=4$	56.6751	40.5741	45.9364	118.4237
$M_D=5$	57.6723	42.2237	56.8517	40.4292
$M_D=6$	43.7068	94.8310	25.7774	84.3486
$M_D=7$	36.4527	72.5724	19.1640	89.3334
$M_D=8$	10.7702	610.8249	0.8193	23.1679

A característica de não-gaussianidade do sinal de voz fica evidente nos histogramas apresentados.

A decomposição em duas subbandas já nos permite diferenciar as bandas através da curtose. A subbanda com as frequências mais baixas apresentam valor de curtose próximo ao do sinal em banda cheia, enquanto que a subbanda com frequências mais altas, para todos os sinais de voz, apresenta um valor bem mais alto.

Na decomposição em quatro subbandas, percebe-se que apenas o valor de curtose para a subbanda com frequências mais baixas se assemelha ao da curtose do sinal em banda cheia, para as outras três bandas, o valor da curtose aumenta consideravelmente.



Por fim, decompondo o sinal em oito subbandas, nota-se, que, assim como para as decomposições em duas ou quatro subbandas, o valor de curtose para a banda de frequências mais baixas é próximo do valor de curtose para o sinal em banda cheia. Além disso, temos que para os sinais de voz masculina 1, feminina 1 e feminina 2, o valor de curtose para a banda de frequências mais altas também é baixo. No caso do sinal de voz masculina 2, o valor de curtose encontrado, para este caso, foi bem mais alto, o que pode ter sido ocasionado pelo fato de que nesta subbanda haveria muito mais ruído do que sinal propriamente dito.

## 3.2 Não-estacionariedade

Para o estudo da não-estacionariedade, foi utilizado o algoritmo apresentado em [31] que propõe o emprego de uma STFT (*Short-Time Fourier Transform*) adaptativa na melhoria da qualidade de sinais de voz. O objetivo do algoritmo é encontrar o tamanho apropriado da janela a ser utilizada no cálculo da STFT, o qual é alterado de acordo com a variação da concentração tempo-frequência da energia do sinal.

A medida de concentração tempo-frequência utilizada foi a curtose espectral, dada por:

$$C(x_w) = \frac{\sum_k |\sum_n x_w[n] e^{-j2\pi kn/N}|^4}{(\sum_k |\sum_n x_w[n] e^{-j2\pi kn/N}|^2)^2} \quad (3.1)$$

em que  $x_w[n]$  é um segmento do sinal  $x[n]$  para uma determinada janela  $w[n]$  de comprimento  $2L_w$ .

A cada iteração do algoritmo, as medidas de concentração de dois segmentos adjacentes,  $x_l[n] = x[n]w_l[pL_w - n]$  e  $x_r[n] = x[n]w_r[(p+1)L_w - n]$ , começando com  $p = 1$ , são calculadas e comparadas com a medida de concentração  $x_m[n] = x[n]w_m[pL_w - n]$  em que  $w_m$  é uma janela que engloba as duas outras janelas, dada por  $w_m = w_l[pL_w - n] + w_r[(p+1)L_w - n]$ . As janelas  $w_l$ ,  $w_r$  e  $w_m$  estão ilustradas na Figura 3.13, considerando janelas iniciais triangulares.

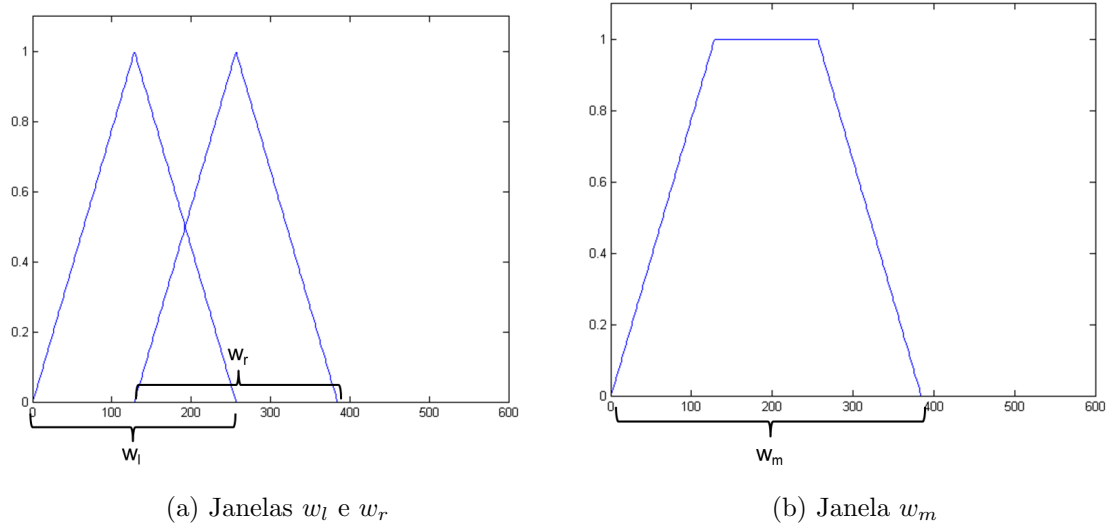


Figura 3.13: Janelas iniciais do algoritmo adaptativo STFT

Ou seja, se a concentração tempo-frequência dos segmentos concatenados, dada por  $C(x_m)$ , for maior do que uma das concentrações dos segmentos individuais dadas por  $C(x_l)$  e  $C(x_r)$ , então a janela  $w_l$  utilizada na próxima iteração será dada por  $w_m$ , como ilustrado na Figura 3.14.

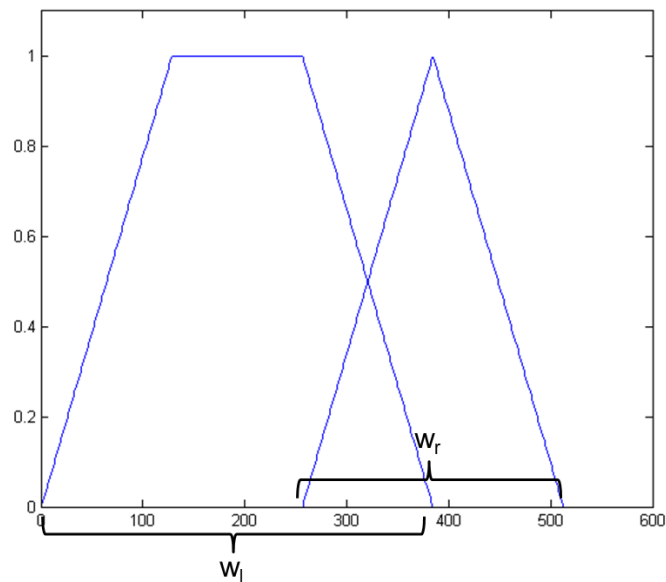


Figura 3.14: Janelas após a primeira iteração, juntando os segmentos

Caso contrário, é definido o tamanho da janela da esquerda, representado em vermelho na Figura 3.15, e o algoritmo segue analisando o restante do sinal.

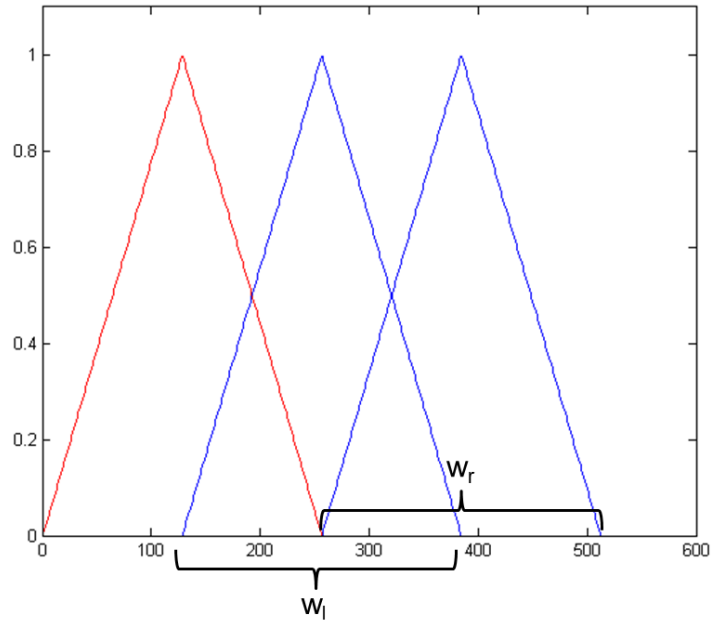


Figura 3.15: Janelas após a primeira iteração, sem juntar os segmentos

Assim, na Figura 3.16 é apresentado o sinal com as janelas fixas de tamanho 512, e na Figura 3.17 as janelas obtidas através do algoritmo proposto em [31].

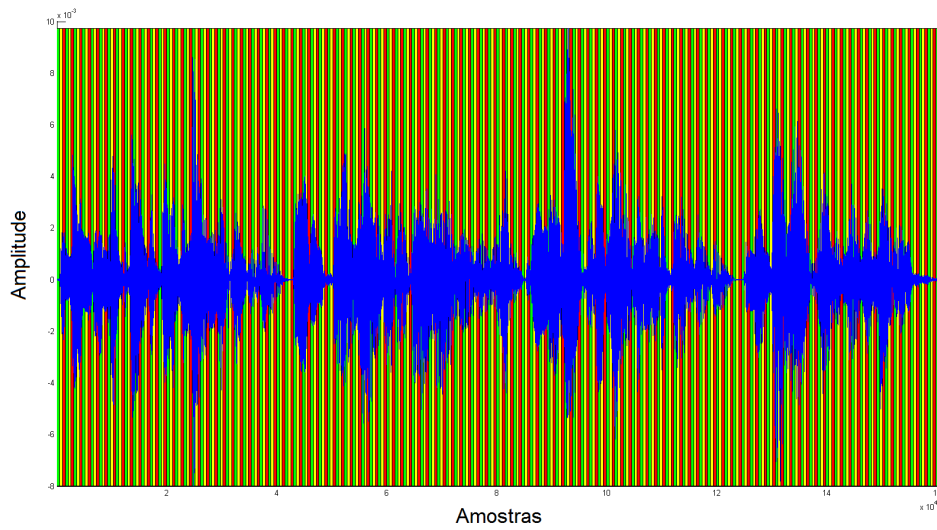


Figura 3.16: Janela fixa

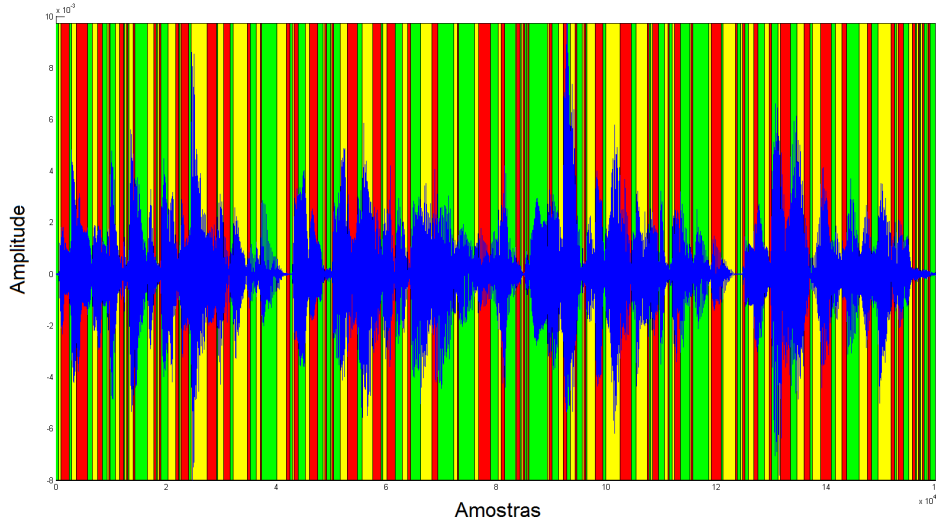


Figura 3.17: Janela variável

Para as diversas fontes de voz e diferentes decomposições em subbandas, foram obtidas as janelas variáveis e o estudo foi feito para os histogramas de valores de janelas. O valor de  $L_w$  utilizado foi 128, o que corresponde a uma janela inicial de 256 amostras equivalendo, para a frequência de amostragem utilizada de 16 kHz, a 16 ms.

As Figuras 3.18 a 3.21 apresentam os histogramas correspondentes dos tamanhos de janelas obtidos com o algoritmo descrito para quatro sinais de voz distintos decompostos em duas subbandas. Para a decomposição em subbandas empregou-se o banco de filtros modulado por cosseno, descrito na Seção 1.5, com filtro protótipo de comprimento 16.

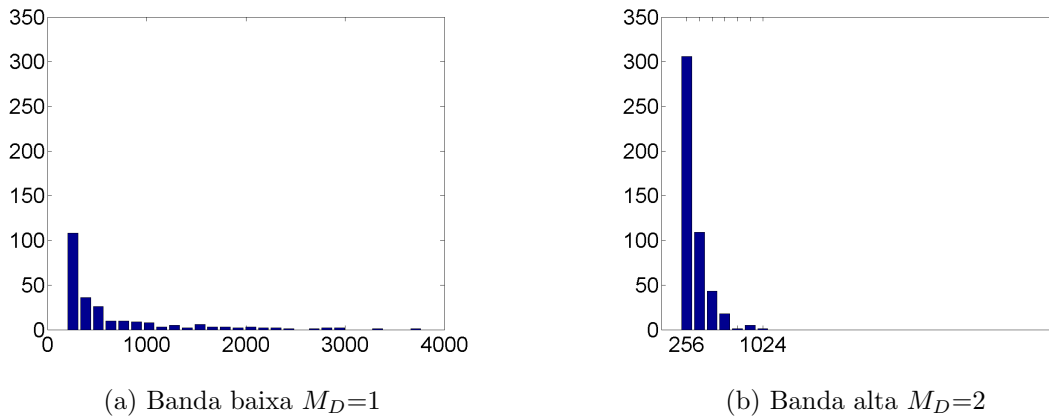


Figura 3.18: Histogramas das janelas - 2 subbandas - Voz masculina 1

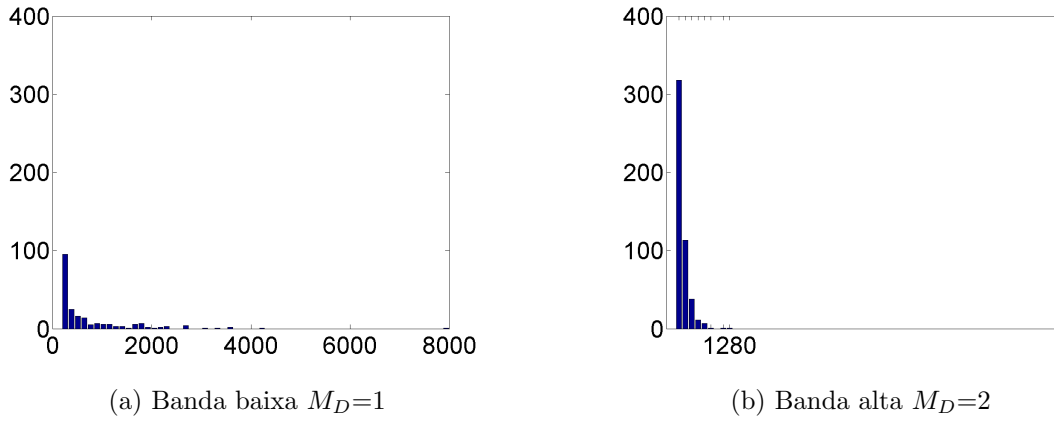


Figura 3.19: Histogramas das janelas - 2 subbandas - Voz masculina 2

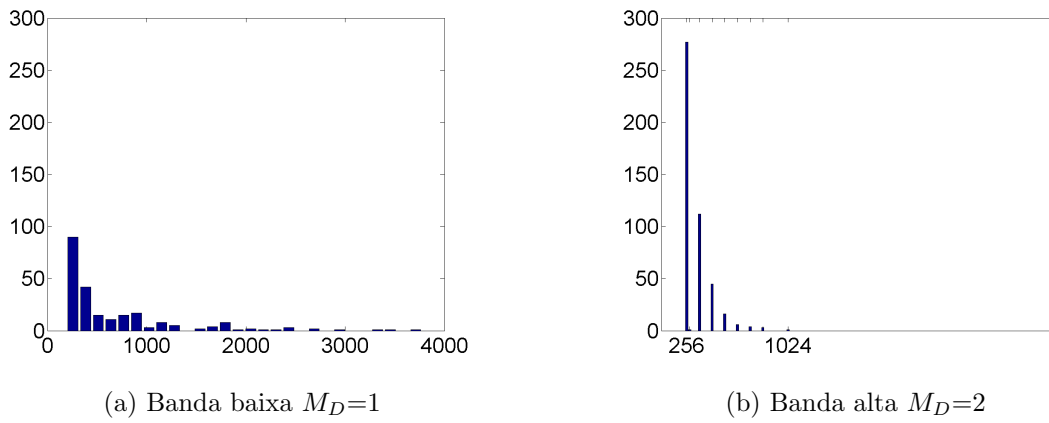


Figura 3.20: Histogramas das janelas - 2 subbandas - Voz feminina 1

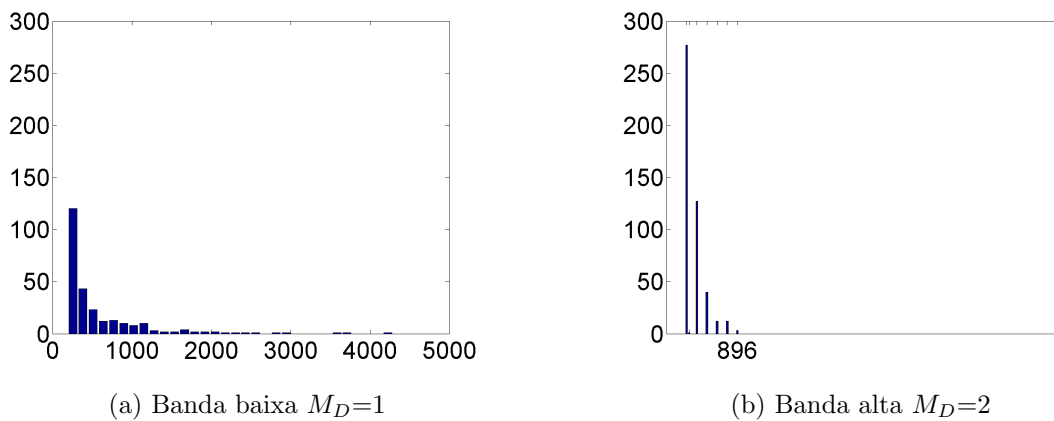


Figura 3.21: Histogramas das janelas - 2 subbandas - Voz feminina 2

As Figuras 3.22 a 3.25 apresentam os histogramas correspondentes aos tamanhos de janelas obtidos com o algoritmo descrito para quatro sinais de voz distintos

decompostos em quatro subbandas. Para a decomposição em subbandas empregou-se o banco de filtros modulado por cosseno, descrito na seção 1.5, com filtro protótipo de comprimento 32.

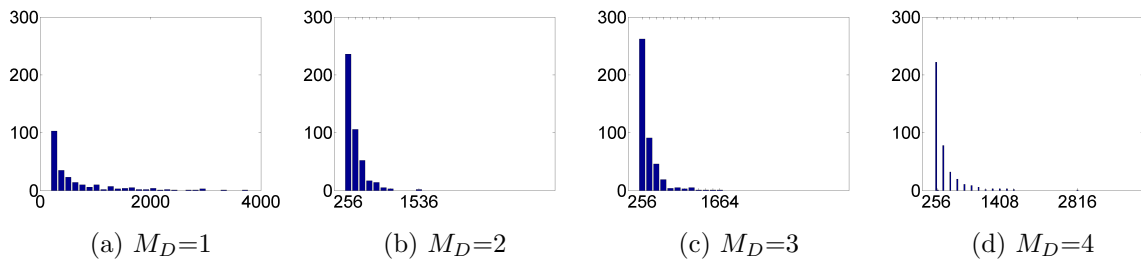


Figura 3.22: Histogramas das janelas - 4 subbandas - Voz masculina 1

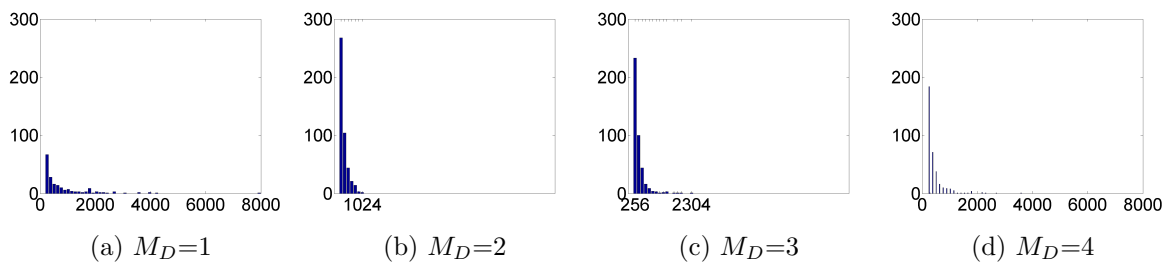


Figura 3.23: Histogramas das janelas - 4 subbandas - Voz masculina 2

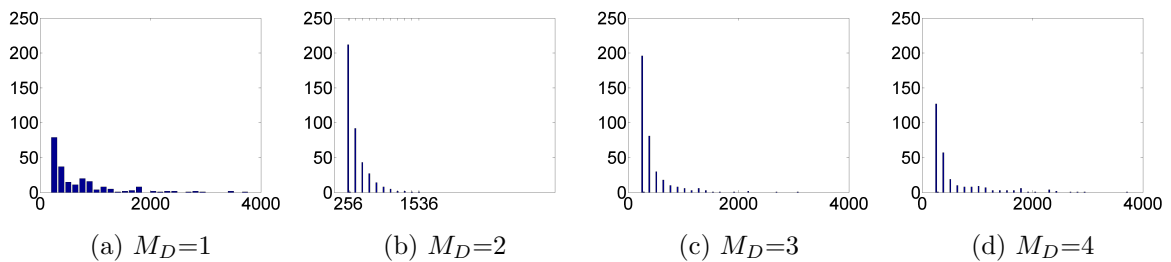


Figura 3.24: Histogramas das janelas - 4 subbandas - Voz feminina 1

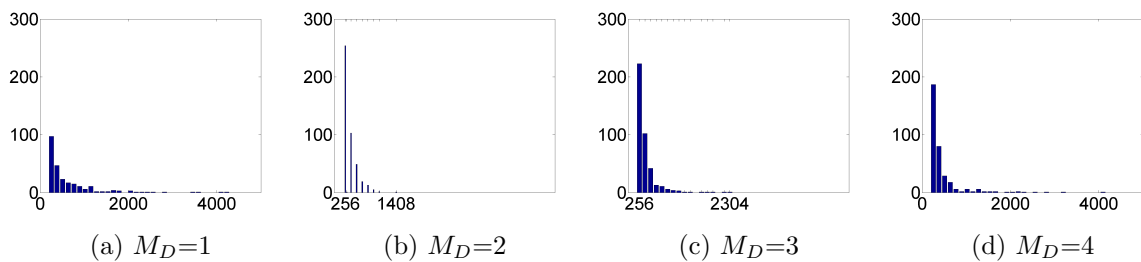


Figura 3.25: Histogramas das janelas - 4 subbandas - Voz feminina 2

As Figuras 3.26 a 3.29 apresentam os histogramas correspondentes aos tamanhos de janelas obtidos com o algoritmo descrito para quatro sinais de voz distintos decompostos em oito subbandas. Para a decomposição em subbandas empregou-se o banco de filtros modulado por cosseno, descrito na seção 1.5, com filtro protótipo de comprimento 64.

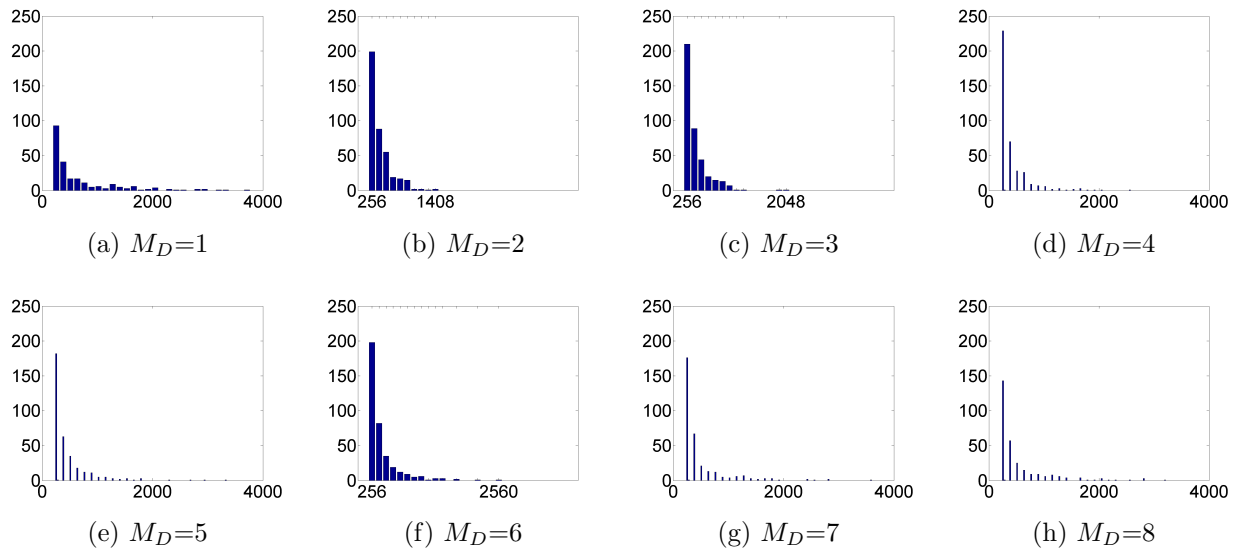


Figura 3.26: Histogramas das janelas - 8 subbandas - Voz masculina 1

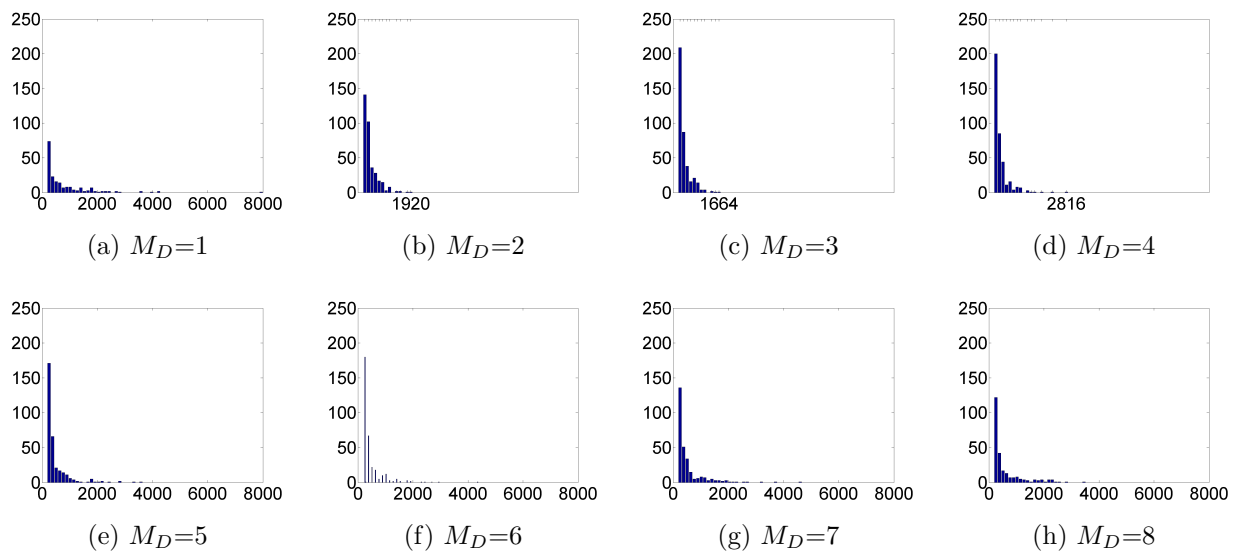


Figura 3.27: Histogramas das janelas - 8 subbandas - Voz masculina 2

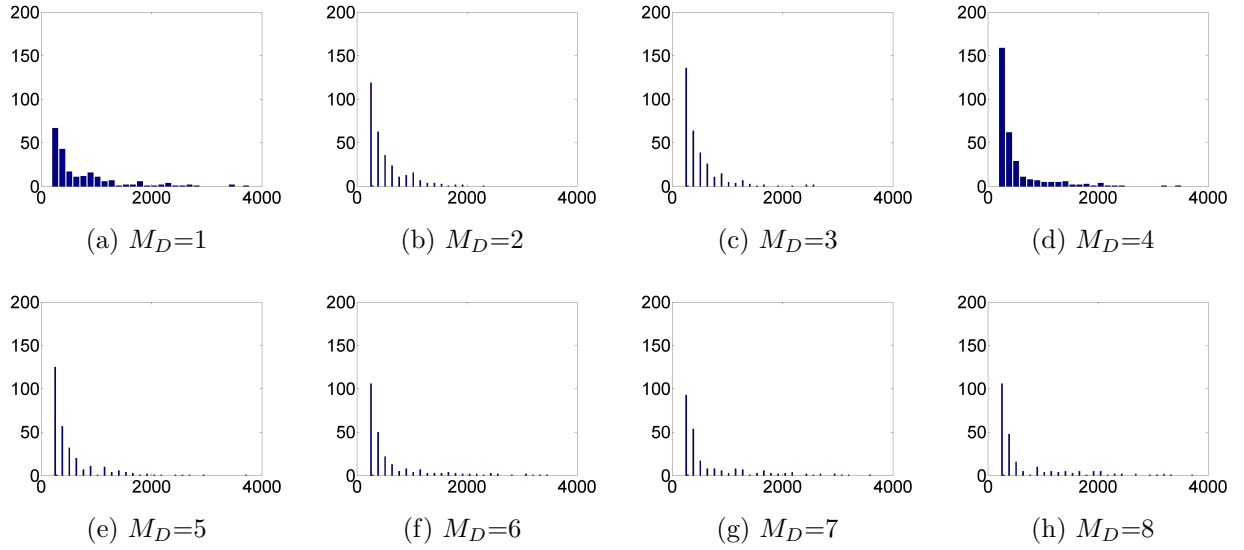


Figura 3.28: Histogramas das janelas - 8 subbandas - Voz feminina 1

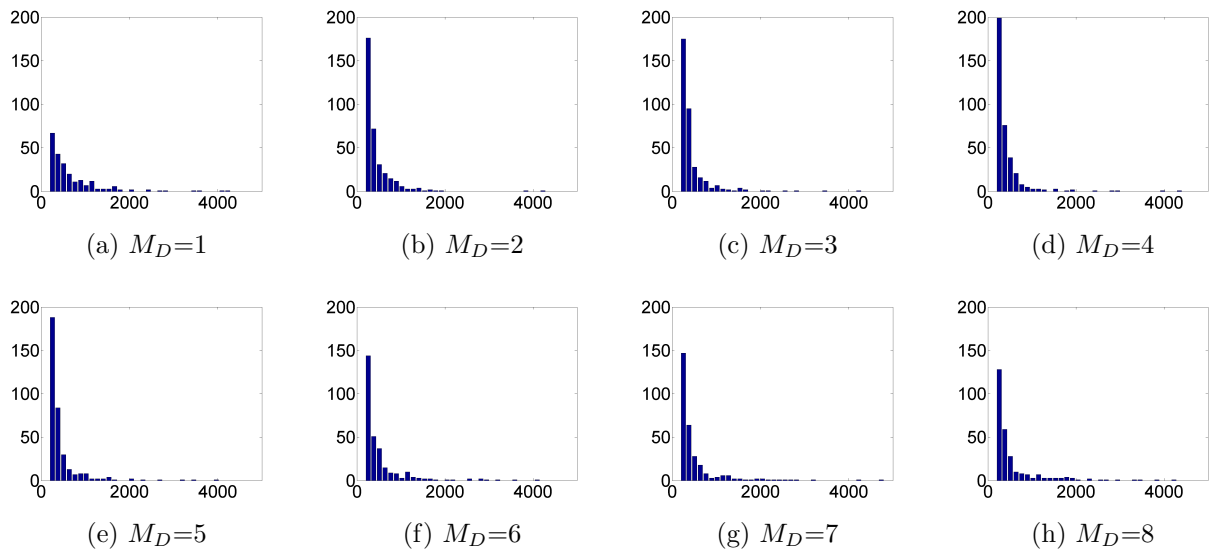


Figura 3.29: Histogramas das janelas - 8 subbandas - Voz feminina 2

Analisando os histogramas gerados, observa-se que os valores do histograma para a subbanda de frequências mais baixas se espalham mais indicando que é possível formar janelas de tamanhos maiores, enquanto que nas frequências mais altas os valores se concentram perto de 256.

### 3.3 Não-branco

A análise do espectro colorido do sinal de voz foi feita através da matriz de covariância normalizada da magnitude das componentes de frequência dos sinais de voz. Os sinais de voz foram amostrados em 16 kHz.



Para a análise dos sinais decompostos em subbandas, o gráfico mostra todas as componentes de frequência. Portanto, algumas áreas, como os quadrados na cor vermelho escuro, não devem ser consideradas na análise pois representam áreas em que o sinal foi filtrado.

As Figuras 3.30 a 3.33 apresentam as matrizes de covariâncias para os quatro sinais de voz distintos decompostos em duas subbandas. Para a decomposição em subbandas empregou-se o banco de filtros modulado por cosseno, descrito na seção 1.5, com filtro protótipo de comprimento 16.

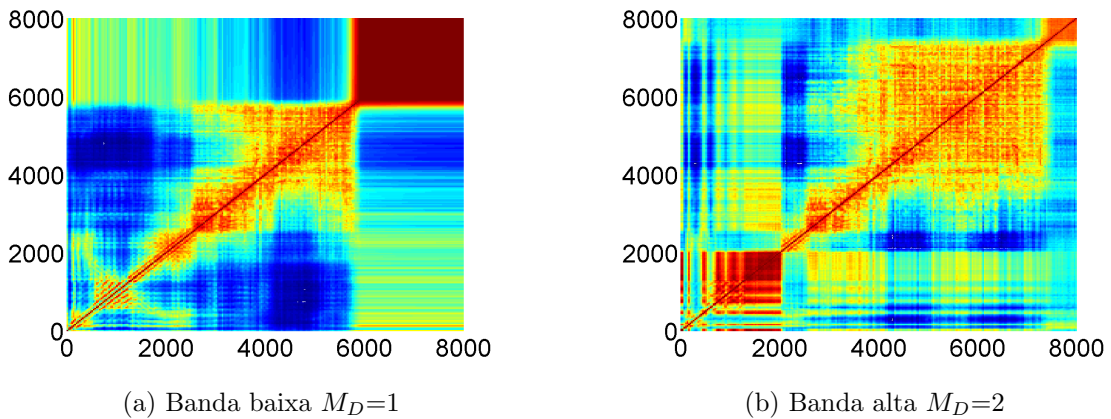


Figura 3.30: Matrizes de covariância - 2 subbandas - Voz masculina 1

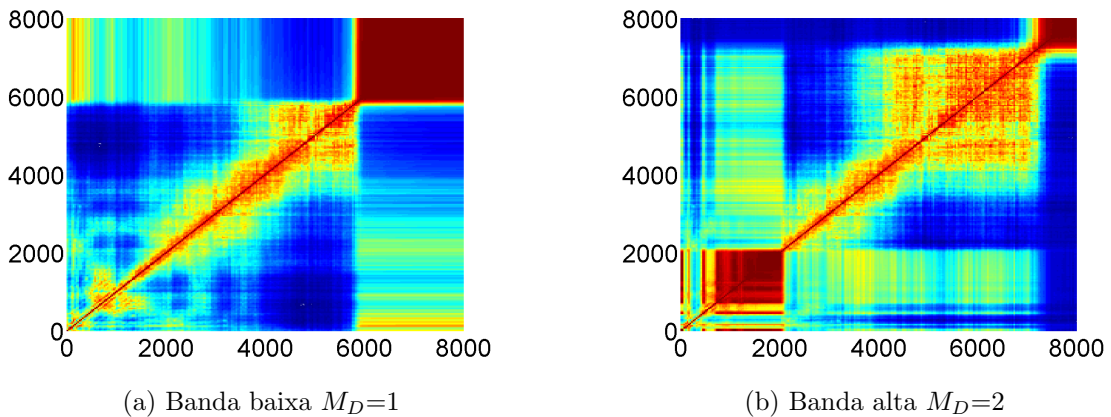


Figura 3.31: Matrizes de covariância - 2 subbandas - Voz masculina 2

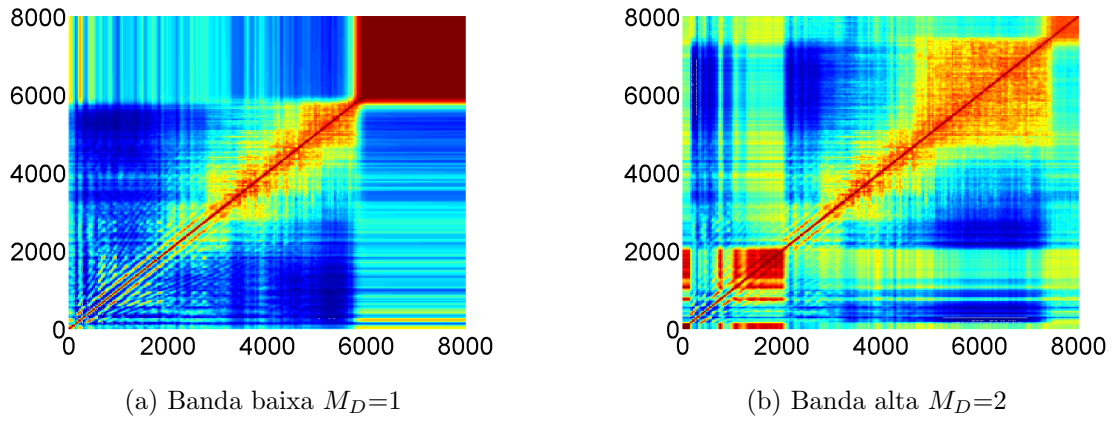


Figura 3.32: Matrizes de covariância - 2 subbandas - Voz feminina 1

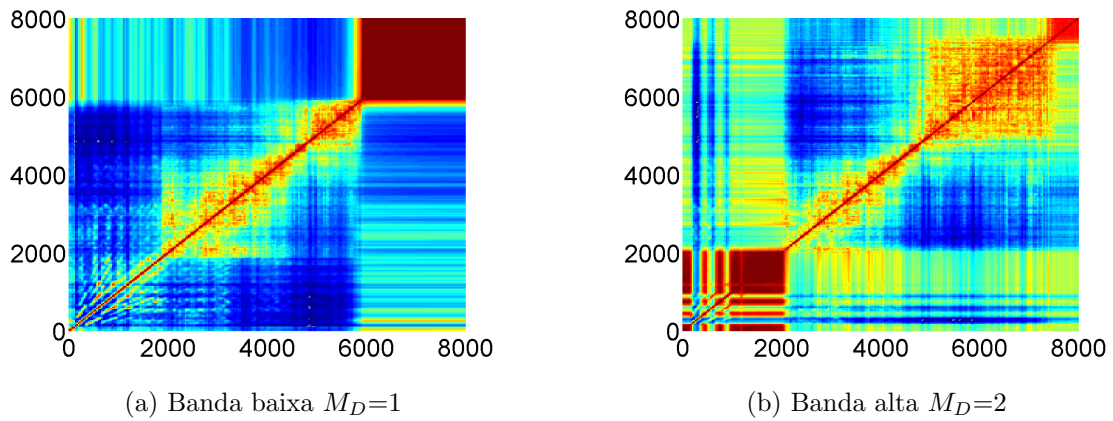


Figura 3.33: Matrizes de covariância - 2 subbandas - Voz feminina 2

As Figuras 3.34 a 3.37 apresentam as matrizes de covariâncias para os quatro sinais de voz distintos decompostos em quatro subbandas. Para a decomposição em subbandas empregou-se o banco de filtros modulado por cosseno, descrito na seção 1.5, com filtro protótipo de comprimento 32.

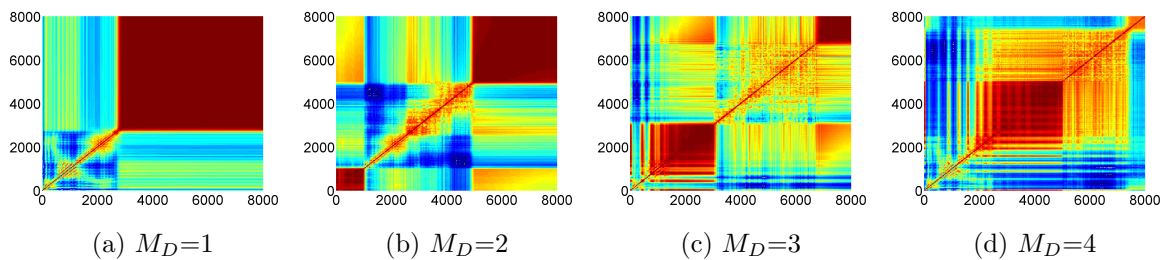


Figura 3.34: Matrizes de covariância - 4 subbandas - Voz masculina 1

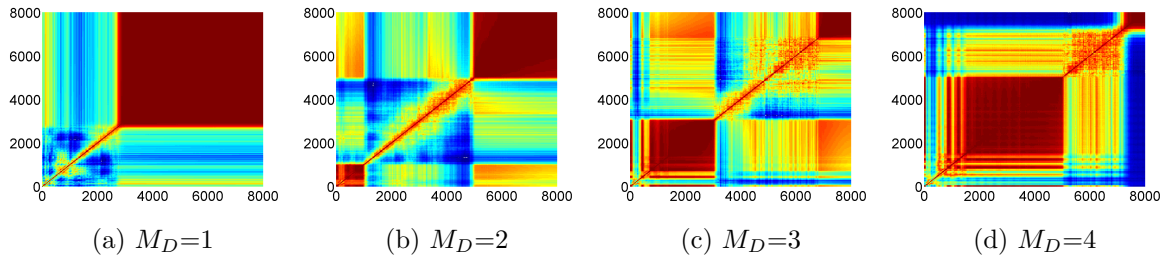


Figura 3.35: Matrizes de covariância - 4 subbandas - Voz masculina 2

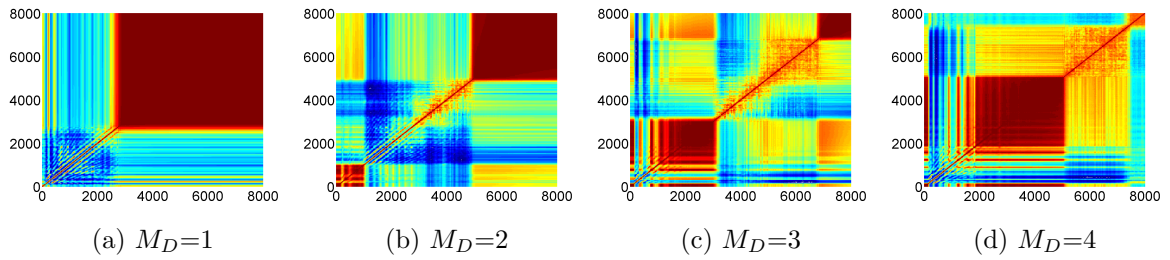


Figura 3.36: Matrizes de covariância - 4 subbandas - Voz feminina 1

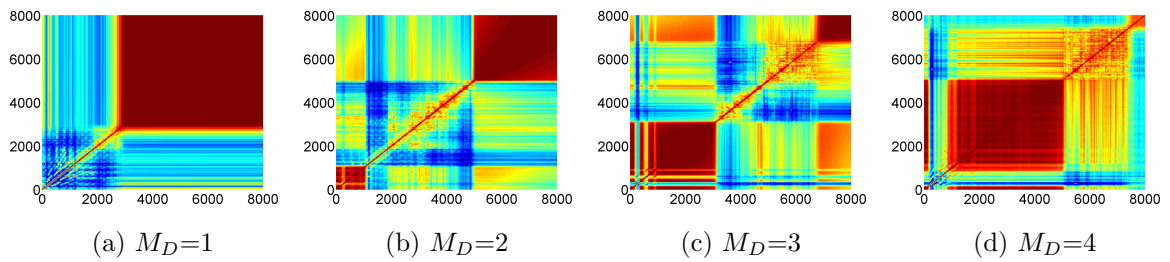


Figura 3.37: Matrizes de covariância - 4 subbandas - Voz feminina 2

As Figuras 3.38 a 3.41 apresentam as matrizes de covariâncias para os quatro sinais de voz distintos decompostos em duas subbandas. Para a decomposição em subbandas empregou-se o banco de filtros modulado por cosseno, descrito na seção 1.5, com filtro protótipo de comprimento 64.

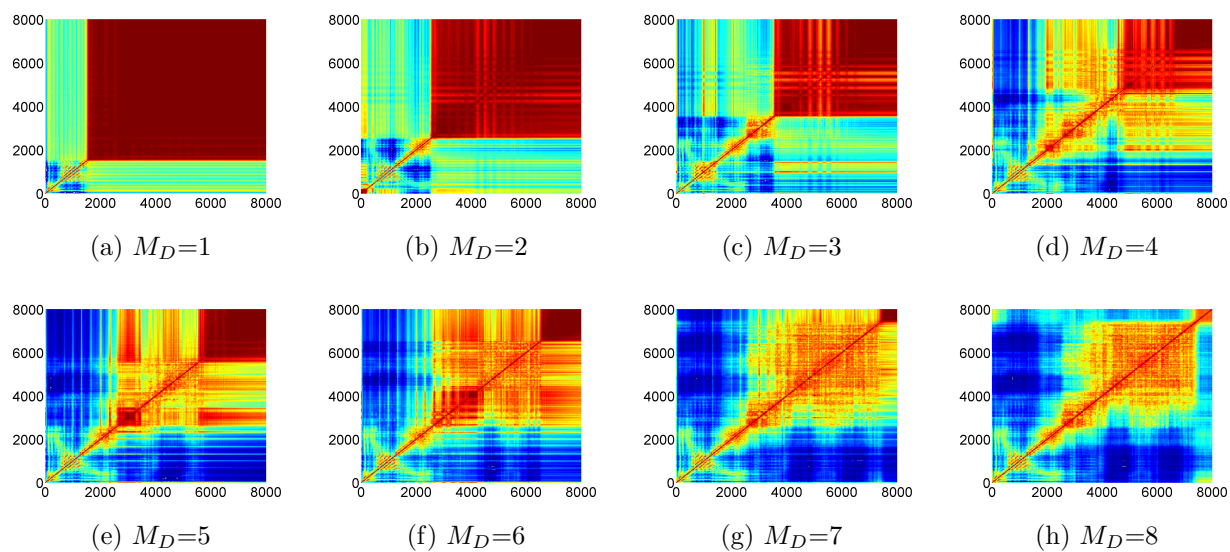


Figura 3.38: Matrizes de covariância - 8 subbandas - Voz masculina 1

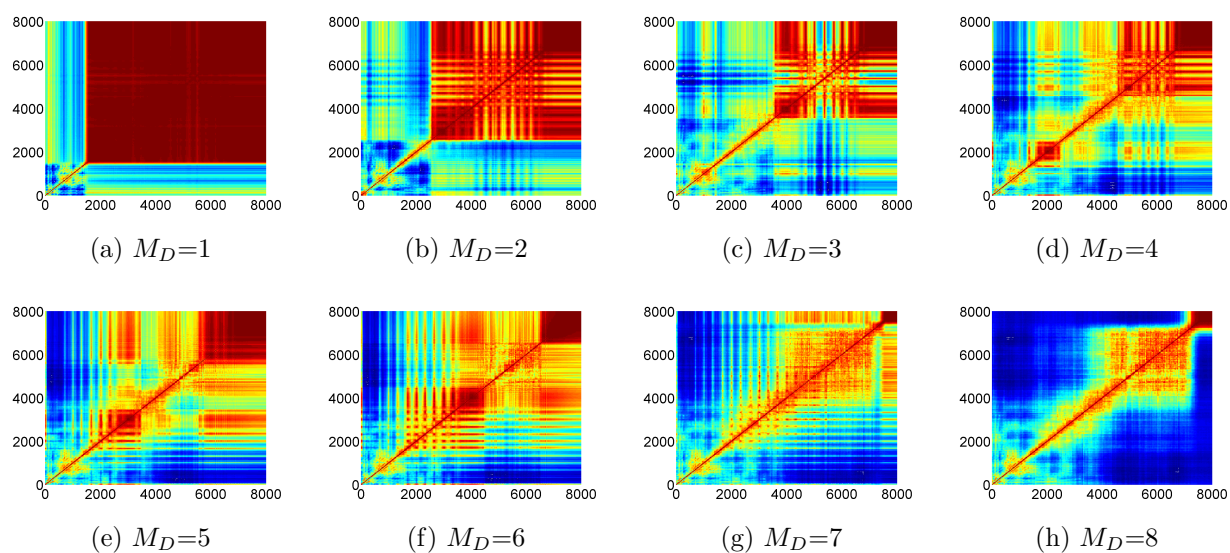


Figura 3.39: Matrizes de covariância - 8 subbandas - Voz masculina 2

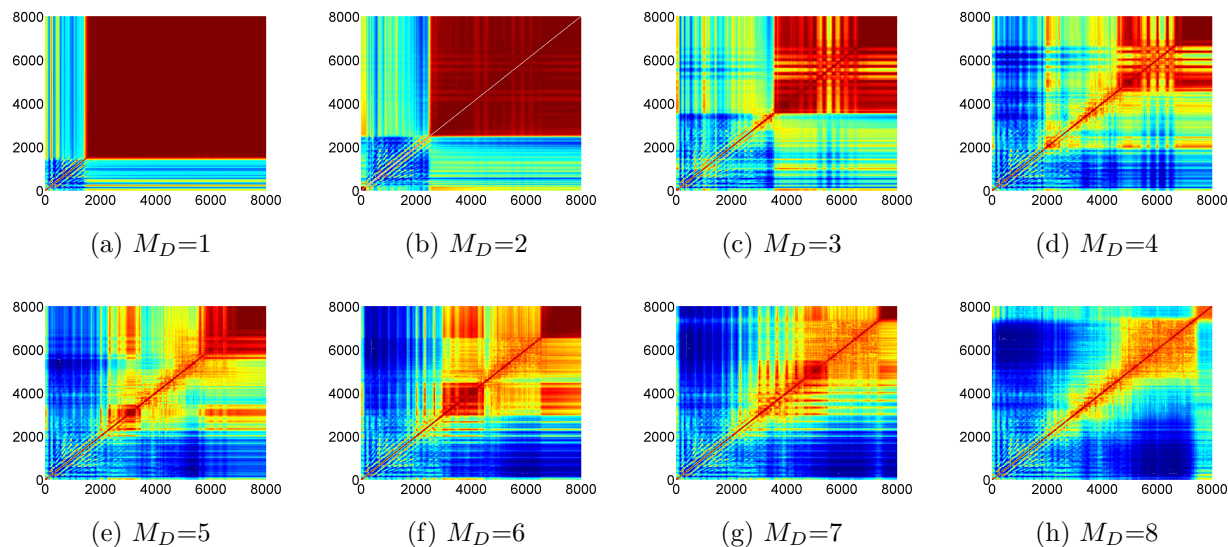


Figura 3.40: Matrizes de covariância - 8 subbandas - Voz feminina 1

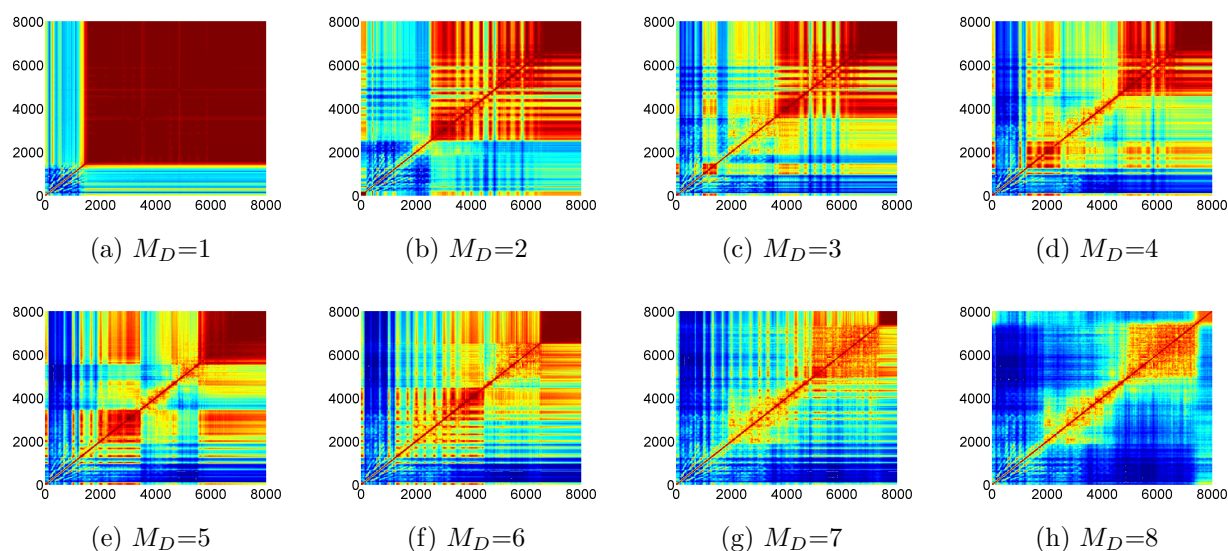


Figura 3.41: Matrizes de covariância - 8 subbandas - Voz feminina 2

Nas subbandas de mais altas frequências, observa-se um espalhamento maior das componentes de frequência, enquanto que nas subbandas de baixa frequência as raias são mais bem definidas.

### 3.4 Comentários gerais

Nesta seção, serão apresentadas as observações pertinentes a cada uma das características do sinal de voz que podem ser utilizadas para melhoria do algoritmo de separação.

Através da análise da não-gaussianidade, é possível perceber que para as subbandas em alta frequência, esta característica é mais forte. Para os sinais decompostos em duas, quatro ou oito subbandas, a subbanda mais baixa apresenta distribuição bem semelhante à distribuição do sinal em banda cheia. Já nas outras subbandas, a distribuição fica bem concentrada no centro, com exceção da decomposição em 8 subbandas, que na subbanda mais alta apresentou, em alguns casos, distribuição mais gaussiana até do que o sinal em banda cheia.

Analisando os histogramas gerados para o estudo da não-estacionariedade, é possível notar que para os sinais em baixa frequência, foram encontradas janelas de até 8000 amostras enquanto que para os sinais em alta frequência os valores de janela se concentram no lado esquerdo do gráfico, não passando de aproximadamente 2048 amostras. Isto significa que nas baixas frequências podemos utilizar blocos maiores.

Observando as matrizes de covariância, percebe-se que uma escolha adequada seria decompor o sistema em duas ou quatro subbandas e também o emprego de banco de filtros não-uniformes.

# Capítulo 4

## Separação de fontes com algoritmos adaptativos supervisionados

Nesse capítulo serão avaliadas as limitações do modelo de separação por filtros FIR (*Finite Impulse Response*). Para isso, utilizaremos algoritmos adaptativos supervisionados para encontrar os coeficientes ótimos do sistema de separação e calcularemos a SIR resultante em ambientes com diferentes características de reverberação. Os algoritmos utilizados foram o NLMS (*Normalized Least-Mean Squares*) que é bem conhecido na literatura de filtragem adaptativa e o NSAF [32] (*Normalized Subband Adaptive Filters*).

Primeiramente, será descrita a configuração do sistema adaptativo utilizada, especificando o sinal que queremos encontrar para o problema de separação. Depois, serão apresentados os algoritmos adaptativos adotados e como foram empregados no problema. Por fim, serão mostrados resultados de simulações nos diferentes ambientes e conclusões relevantes.

### 4.1 Sinal desejado no problema de separação cega

No problema de separação cega não é possível recuperar o sinal exatamente como o sinal original, sendo o sinal desejado o sinal original filtrado, como demonstrado na seguinte análise no domínio da frequência. Os sinais de mistura  $\mathbf{X}_i$  e de saída do sistema de separação  $\mathbf{Y}_i$  são dados por:

$$\mathbf{X}_1 = \mathbf{H}_{11}\mathbf{S}_1 + \mathbf{H}_{21}\mathbf{S}_2 \quad (4.1)$$

$$\mathbf{X}_2 = \mathbf{H}_{12}\mathbf{S}_1 + \mathbf{H}_{22}\mathbf{S}_2 \quad (4.2)$$

$$\mathbf{Y}_1 = \mathbf{W}_{11}\mathbf{X}_1 + \mathbf{W}_{21}\mathbf{X}_2 \quad (4.3)$$

$$\mathbf{Y}_2 = \mathbf{W}_{12}\mathbf{X}_1 + \mathbf{W}_{22}\mathbf{X}_2 \quad (4.4)$$

Substituindo as equações (4.1) e (4.2) nas equações (4.3) e (4.4), tem-se:

$$\begin{aligned} \mathbf{Y}_1 &= \mathbf{W}_{11}\mathbf{H}_{11}\mathbf{S}_1 + \mathbf{W}_{11}\mathbf{H}_{21}\mathbf{S}_2 + \mathbf{W}_{21}\mathbf{H}_{12}\mathbf{S}_1 + \mathbf{W}_{21}\mathbf{H}_{22}\mathbf{S}_2 \\ &= (\mathbf{W}_{11}\mathbf{H}_{11} + \mathbf{W}_{21}\mathbf{H}_{12})\mathbf{S}_1 + (\mathbf{W}_{11}\mathbf{H}_{21} + \mathbf{W}_{21}\mathbf{H}_{22})\mathbf{S}_2 \end{aligned} \quad (4.5)$$

e

$$\begin{aligned} \mathbf{Y}_2 &= \mathbf{W}_{12}\mathbf{H}_{11}\mathbf{S}_1 + \mathbf{W}_{12}\mathbf{H}_{21}\mathbf{S}_2 + \mathbf{W}_{22}\mathbf{H}_{12}\mathbf{S}_1 + \mathbf{W}_{22}\mathbf{H}_{22}\mathbf{S}_2 \\ &= (\mathbf{W}_{12}\mathbf{H}_{11} + \mathbf{W}_{22}\mathbf{H}_{12})\mathbf{S}_1 + (\mathbf{W}_{12}\mathbf{H}_{21} + \mathbf{W}_{22}\mathbf{H}_{22})\mathbf{S}_2 \end{aligned} \quad (4.6)$$

Para encontrar uma solução possível para o problema foi imposto ao sistema que o sinal desejado de saída  $\mathbf{D}_1$  deveria representar o sinal original  $\mathbf{S}_1$  e o sinal desejado de saída  $\mathbf{D}_2$ , o sinal  $\mathbf{S}_2$ . Para isso, devemos igualar a zero a parcela do sinal correspondente a  $\mathbf{S}_2$  na equação de  $\mathbf{Y}_1$  e também a parcela do sinal correspondente a  $\mathbf{S}_1$  na equação de  $\mathbf{Y}_2$ , ou seja,

$$\mathbf{W}_{11}\mathbf{H}_{21} + \mathbf{W}_{21}\mathbf{H}_{22} = 0 \quad (4.7)$$

$$\mathbf{W}_{11} = -\frac{\mathbf{W}_{21}\mathbf{H}_{22}}{\mathbf{H}_{21}} \quad (4.8)$$

$$\mathbf{W}_{12}\mathbf{H}_{11} + \mathbf{W}_{22}\mathbf{H}_{12} = 0 \quad (4.9)$$

$$\mathbf{W}_{22} = -\frac{\mathbf{W}_{12}\mathbf{H}_{11}}{\mathbf{H}_{12}} \quad (4.10)$$

As equações (4.8) e (4.10) possuem infinitas soluções e a solução adotada para as simulações foi a seguinte:

$$\mathbf{W}_{11} = \mathbf{H}_{22} \quad (4.11)$$

$$\mathbf{W}_{12} = \mathbf{H}_{12} \quad (4.12)$$

$$\mathbf{W}_{21} = -\mathbf{H}_{21} \quad (4.13)$$

$$\mathbf{W}_{22} = -\mathbf{H}_{11} \quad (4.14)$$

Assim, os sinais  $\mathbf{D}_1$  e  $\mathbf{D}_2$  passam a ser:



$$\begin{aligned}\mathbf{D}_1 &= (\mathbf{W}_{11}\mathbf{H}_{11} + \mathbf{W}_{21}\mathbf{H}_{12})\mathbf{S}_1 \\ &= (\mathbf{H}_{22}\mathbf{H}_{11} - \mathbf{H}_{21}\mathbf{H}_{12})\mathbf{S}_1\end{aligned}\tag{4.15}$$

e

$$\begin{aligned}\mathbf{D}_2 &= (\mathbf{W}_{12}\mathbf{H}_{21} + \mathbf{W}_{22}\mathbf{H}_{22})\mathbf{S}_2 \\ &= (\mathbf{H}_{12}\mathbf{H}_{21} - \mathbf{H}_{11}\mathbf{H}_{22})\mathbf{S}_2\end{aligned}\tag{4.16}$$

Esses valores de  $\mathbf{D}_1$  e  $\mathbf{D}_2$  é que serão os sinais desejados utilizados nos algoritmos adaptativos.

## 4.2 Algoritmos adaptativos

Nesta seção serão apresentados os algoritmos adaptativos utilizados, o NLMS e o NSAF. Também será descrito como cada algoritmo foi adaptado para o problema de separação de fontes.

### 4.2.1 Algoritmo NLMS

O algoritmo NLMS (*Normalized Least-Mean Squares*) é dado pelo algoritmo LMS (*Least-Mean Squares*) com um passo de adaptação normalizado para minimizar o problema que o algoritmo LMS sofre de amplificação de ruído do gradiente quando o tamanho do vetor de entrada é muito grande [33].

O algoritmo LMS é bastante utilizado em várias aplicações de filtragem adaptativa pela sua simplicidade computacional, convergência comprovada em ambientes estacionários, convergência na média para solução de Wiener e comportamento estável quando implementado com a aritmética de precisão finita [34].

Ambos os algoritmos LMS e NLMS atualizam os coeficientes do filtro utilizando o erro entre a saída desejada e a saída obtida pelo filtro, conforme ilustrado na Figura 4.1.

Os vetores de entrada e de coeficientes podem ser definidos por:

$$\mathbf{x}(n) = \begin{bmatrix} x(n) & x(n-1) & \dots & x(n-L+1) \end{bmatrix}^T\tag{4.17}$$

$$\mathbf{w}(n) = \begin{bmatrix} w_0(n) & w_1(n) & \dots & w_{L-1}(n) \end{bmatrix}^T\tag{4.18}$$

Portanto, temos a seguinte saída e erros:

$$y(n) = \mathbf{w}^T(n)\mathbf{x}(n)\tag{4.19}$$

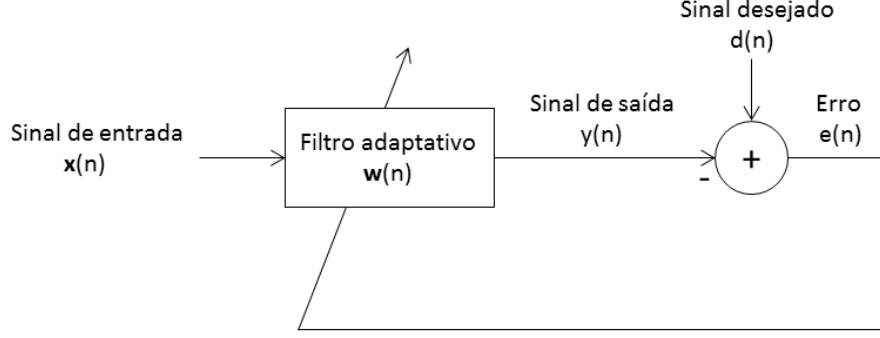


Figura 4.1: Algoritmo NLMS

$$e(n) = d(n) - y(n) \quad (4.20)$$

Para a solução do algoritmo LMS temos que o erro e a atualização dos coeficientes são dados, respectivamente, por [33] e [34]:

$$e(n) = d(n) - \mathbf{w}^T(n)\mathbf{x}(n) \quad (4.21)$$

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e(n)\mathbf{x}(n) \quad (4.22)$$

em que  $\mu$  é o fator de convergência, que no algoritmo NLMS torna-se variável conforme a seguinte equação:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu_n e(n)\mathbf{x}(n) = \mathbf{w}(n) + \Delta\tilde{\mathbf{w}}(n) \quad (4.23)$$

Partindo da estratégia de minimizar o erro quadrático instantâneo [34], temos que este erro é dado por:

$$e^2(n) = d^2(n) + \mathbf{w}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}(n) - 2d(n)\mathbf{w}^T(n)\mathbf{x}(n) \quad (4.24)$$

Aplicando uma mudança no vetor de coeficientes do filtro tal que  $\tilde{\mathbf{w}}(n) = \mathbf{w}(n) + \Delta\tilde{\mathbf{w}}(n)$ , o erro quadrático com os coeficientes  $\tilde{\mathbf{w}}(n)$  passa a ser dado por:

$$\begin{aligned} \tilde{e}^2(n) = e^2(n) + 2\Delta\tilde{\mathbf{w}}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}(n) + \Delta\tilde{\mathbf{w}}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)\Delta\tilde{\mathbf{w}}(n) \\ - 2d(n)\Delta\tilde{\mathbf{w}}^T(n)\mathbf{x}(n) \end{aligned} \quad (4.25)$$

Substituindo  $d(n)$  por  $d(n) = e(n) + \mathbf{w}^T(n)\mathbf{x}(n)$  e  $\Delta\tilde{\mathbf{w}}(n)$  por  $\Delta\tilde{\mathbf{w}}(n) = 2\mu_n e(n)\mathbf{x}(n)$ , temos que:

$$\begin{aligned}\Delta e^2(n) &\triangleq \tilde{e}^2(n) - e^2(n) \\ &= -4\mu_n e^2(n) \mathbf{x}^T(n) \mathbf{x}(n) + 4\mu_n^2 e^2(n) [\mathbf{x}^T(n) \mathbf{x}(n)]^2\end{aligned}\quad (4.26)$$

Derivando a equação acima em relação a  $\mu_n$  e igualando a zero, para minimizar a função  $\Delta e^2(n)$ , encontramos o valor de  $\mu_n$ :

$$\mu_n = \frac{1}{2\mathbf{x}^T(n) \mathbf{x}(n)} \quad (4.27)$$

Utilizando  $\mu_n$  na equação (4.23), a atualização dos coeficientes passa a ser dada por:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{e(n) \mathbf{x}(n)}{\mathbf{x}^T(n) \mathbf{x}(n)} \quad (4.28)$$

Normalmente, um fator de convergência fixo  $\mu$  também é introduzido para controlar o desalinhamento e um parâmetro  $\gamma$  para evitar passos muito grandes quando  $\mathbf{x}^T(n) \mathbf{x}(n)$  se torna muito pequeno. Assim, a equação final para atualização dos coeficientes é dada por:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \frac{\mu}{\gamma + \mathbf{x}^T(n) \mathbf{x}(n)} e(n) \mathbf{x}(n) \quad (4.29)$$

Os valores adequados para escolha de  $\mu$  são [34]:

$$0 \leq \mu \leq 2 \quad (4.30)$$

### Algoritmo NLMS para o problema de separação

Para adaptar o algoritmo NLMS ao problema de separação, basta entender que, para duas fontes e dois sensores, o algoritmo deverá gerar duas saídas,  $y_1(n)$  e  $y_2(n)$ , e dois erros,  $e_1(n)$  e  $e_2(n)$ :

$$y_1(n) = \mathbf{w}_1^T(n) \mathbf{x}(n) \quad (4.31)$$

$$y_2(n) = \mathbf{w}_2^T(n) \mathbf{x}(n) \quad (4.32)$$

$$e_1(n) = d_1(n) - y_1(n) \quad (4.33)$$

$$e_2(n) = d_2(n) - y_2(n) \quad (4.34)$$

sendo que os sinais desejados  $d_1(n)$  e  $d_2(n)$  são os definidos na Seção 4.1. O sinal de entrada  $\mathbf{x}(n)$  deverá ser um vetor composto pelos dois vetores do sinal de mistura:

$$\mathbf{x}_1(n) = \begin{bmatrix} x_1(n) & x_1(n-1) & \dots & x_1(n-L+1) \end{bmatrix}^T \quad (4.35)$$

$$\mathbf{x}_2(n) = \begin{bmatrix} x_2(n) & x_2(n-1) & \dots & x_2(n-L+1) \end{bmatrix}^T \quad (4.36)$$

$$\mathbf{x}(n) = \begin{bmatrix} \mathbf{x}_1^T(n) & \mathbf{x}_2^T(n) \end{bmatrix}^T \quad (4.37)$$

Os vetores de coeficientes  $\mathbf{w}_1(n)$  e  $\mathbf{w}_2(n)$  representam os coeficientes dos dois filtros que geram cada saída:

$$\mathbf{w}_{11}(n) = \begin{bmatrix} w_{11,0}(n) & w_{11,1}(n) & \dots & w_{11,N-1}(n) \end{bmatrix}^T \quad (4.38)$$

$$\mathbf{w}_{12}(n) = \begin{bmatrix} w_{12,0}(n) & w_{12,1}(n) & \dots & w_{12,N-1}(n) \end{bmatrix}^T \quad (4.39)$$

$$\mathbf{w}_{21}(n) = \begin{bmatrix} w_{21,0}(n) & w_{21,1}(n) & \dots & w_{21,N-1}(n) \end{bmatrix}^T \quad (4.40)$$

$$\mathbf{w}_{22}(n) = \begin{bmatrix} w_{22,0}(n) & w_{22,1}(n) & \dots & w_{22,N-1}(n) \end{bmatrix}^T \quad (4.41)$$

$$\mathbf{w}_1(n) = \begin{bmatrix} \mathbf{w}_{11}^T(n) & \mathbf{w}_{21}^T(n) \end{bmatrix}^T \quad (4.42)$$

$$\mathbf{w}_2(n) = \begin{bmatrix} \mathbf{w}_{12}^T(n) & \mathbf{w}_{22}^T(n) \end{bmatrix}^T \quad (4.43)$$

Portanto, para cada iteração, haverá uma atualização para cada vetor de coeficientes,  $\mathbf{w}_1(n)$  e  $\mathbf{w}_2(n)$ , de acordo com a equação (4.29).

## 4.2.2 Algoritmo NSAF

O algoritmo NSAF [32] decompõe o sinal de entrada e o sinal desejado em subbandas permitindo explorar as propriedades de cada sinal resultante. Este algoritmo utiliza o princípio da perturbação mínima, em que, de uma iteração para a seguinte, os coeficientes do filtro adaptativo devem ser alterados de forma mínima, sujeito a uma restrição imposta sobre o filtro de atualização da saída. Estas várias restrições em subbandas forçam cada uma delas a convergir quase que independentemente.

Como a atualização dos coeficientes é realizada em uma taxa mais baixa, decimando-se os sinais de erro, o custo computacional quase não se altera quando comparado ao NLMS, mas a convergência é dada de forma mais rápida.

Uma característica única do algoritmo NSAF é que o erro do sinal é calculado

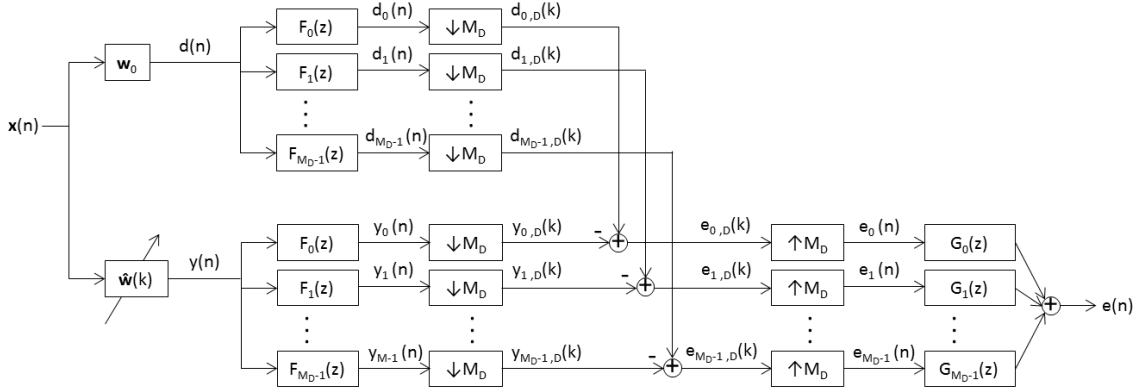


Figura 4.2: Algoritmo NSAF

em subbandas porém a atualização dos coeficientes do filtro adaptativo é feita em banda completa, conforme ilustrado na Figura 4.2. O índice  $n$  será utilizado para as sequências originais do sinal enquanto que o índice  $k$  será utilizado para as sequências decimadas.

No algoritmo NSAF, o sinal desejado  $d(n)$  e a saída do filtro  $y(n)$  são decompostos em  $M_D$  subbandas pelos filtros de análise  $\{F_0(z), \dots, F_{M_D-1}(z)\}$  e decimados. Assim, o sinal de saída decimado é dado por

$$y_{i,D}(k) = \sum_{l=0}^{L-1} \hat{w}_l(k) u_i(kM_D - l) = \hat{\mathbf{w}}^T(k) \mathbf{x}_i(k) \quad (4.44)$$

em que

$$\mathbf{x}_i(k) = [x_i(kM_D) \quad x_i(kM_D - 1) \quad \dots \quad x_i(kM_D - L + 1)]^T \quad (4.45)$$

$$\hat{\mathbf{w}}(k) = [\hat{w}_0(k) \quad \hat{w}_1(k) \quad \dots \quad \hat{w}_{L-1}(k)]^T \quad (4.46)$$

Assim, sendo  $d_{i,D}$  o sinal desejado decomposto em  $M_D$  subbandas e decimado, o erro será dado por

$$e_{i,D}(k) = d_{i,D}(k) - \hat{\mathbf{w}}^T(k) \mathbf{x}_i(k) \quad (4.47)$$

Após a derivação do algoritmo proposto baseado no princípio da perturbação mínima [32], os autores chegaram na seguinte equação de atualização dos coeficientes do filtro de separação:

$$\hat{\mathbf{w}}(k+1) = \hat{\mathbf{w}}(k) + \mu \sum_{i=0}^{M_D-1} \frac{\mathbf{x}_i(k)}{\delta + \|\mathbf{x}_i(k)\|^2} e_{i,D}(k) \quad (4.48)$$

O passo de atualização  $\mu$  foi introduzido para exercer controle sobre a mudança nos coeficientes do filtro e, conseqüentemente, sobre o erro médio quadrático em regime permanente.

### Algoritmo NSAF para o problema de separação

Para adaptar o algoritmo NSAF ao problema de separação, foi feito o mesmo procedimento que para o algoritmo NLMS. Assim, as equações (4.47) e (4.48) podem ser empregadas para gerar a estimativa do sinal de cada fonte. Para duas fontes e dois sensores, obtém-se:

$$e_{1,i,D}(k) = d_{1,i,D}(k) - \hat{\mathbf{w}}_1^T(k)\mathbf{x}_i(k) \quad (4.49)$$

$$e_{2,i,D}(k) = d_{2,i,D}(k) - \hat{\mathbf{w}}_2^T(k)\mathbf{x}_i(k) \quad (4.50)$$

$$\hat{\mathbf{w}}_1(k+1) = \hat{\mathbf{w}}_1(k) + \mu \sum_{i=0}^{M_D-1} \frac{\mathbf{x}_i(k)}{\delta + \|\mathbf{x}_i(k)\|^2} e_{1,i,D}^1(k) \quad (4.51)$$

$$\hat{\mathbf{w}}_2(k+1) = \hat{\mathbf{w}}_2(k) + \mu \sum_{i=0}^{M_D-1} \frac{\mathbf{x}_i(k)}{\delta + \|\mathbf{x}_i(k)\|^2} e_{2,i,D}^2(k) \quad (4.52)$$

em que

$$\mathbf{x}_i(k) = \left[ \mathbf{x}_{1,i}^T(k) \quad \mathbf{x}_{2,i}^T(k) \right]^T \quad (4.53)$$

$$\mathbf{x}_{1,i}(k) = \left[ x_{1,i}(kM_D) \quad x_{1,i}(kM_D - 1) \quad \dots \quad x_{1,i}(kM_D - L + 1) \right]^T \quad (4.54)$$

$$\mathbf{x}_{2,i}(k) = \left[ x_{2,i}(kM_D) \quad x_{2,i}(kM_D - 1) \quad \dots \quad x_{2,i}(kM_D - L + 1) \right]^T \quad (4.55)$$

$$\hat{\mathbf{w}}_1(k) = \left[ \hat{\mathbf{w}}_{11}^T(k) \quad \hat{\mathbf{w}}_{21}^T(k) \right]^T \quad (4.56)$$

$$\hat{\mathbf{w}}_2(k) = \left[ \hat{\mathbf{w}}_{12}^T(k) \quad \hat{\mathbf{w}}_{22}^T(k) \right]^T \quad (4.57)$$

$$\hat{\mathbf{w}}_{11}(k) = \left[ \hat{w}_{11,0}(k) \quad \hat{w}_{11,1}(k) \quad \dots \quad \hat{w}_{11,L-1}(k) \right]^T \quad (4.58)$$

$$\hat{\mathbf{w}}_{12}(k) = \left[ \hat{w}_{12,0}(k) \quad \hat{w}_{12,1}(k) \quad \dots \quad \hat{w}_{12,L-1}(k) \right]^T \quad (4.59)$$

$$\hat{\mathbf{w}}_{21}(k) = \left[ \hat{w}_{21,0}(k) \quad \hat{w}_{21,1}(k) \quad \dots \quad \hat{w}_{21,L-1}(k) \right]^T \quad (4.60)$$

$$\hat{\mathbf{w}}_{22}(k) = \left[ \hat{w}_{22,0}(k) \quad \hat{w}_{22,1}(k) \quad \dots \quad \hat{w}_{22,L-1}(k) \right]^T \quad (4.61)$$

### Modificação no algoritmo NSAF para o problema de separação

Partindo da observação de que existe correlação entre os sinais das misturas, para acelerar a convergência do algoritmo NSAF, foi introduzida a matriz de autocorrelação  $\hat{\mathbf{R}}$  na equação de atualização dos coeficientes, da seguinte forma

$$\hat{\mathbf{w}}_j(k+1) = \hat{\mathbf{w}}_j(k) + \mu \sum_{i=0}^{M_D-1} \hat{\mathbf{R}}_i^{-1}(k) \frac{\mathbf{x}_i(k)}{\delta + \|\mathbf{x}_i(k)\|^2} e_{i,D}^j(k) \quad (4.62)$$

em que

$$\hat{\mathbf{R}}_i(k) = \begin{bmatrix} \hat{\mathbf{R}}_{i,11}(k) & \hat{\mathbf{R}}_{i,12}(k) \\ \hat{\mathbf{R}}_{i,12}(k) & \hat{\mathbf{R}}_{i,22}(k) \end{bmatrix} \quad (4.63)$$

e as matrizes  $\hat{\mathbf{R}}_{i,11}(k)$ ,  $\hat{\mathbf{R}}_{i,12}(k)$  e  $\hat{\mathbf{R}}_{i,22}(k)$  são matrizes diagonais tais que

$$\hat{\mathbf{R}}_{i,11}(k) = \sigma_{i,11}^2(k) \mathbf{I} \quad (4.64)$$

$$\hat{\mathbf{R}}_{i,12}(k) = \sigma_{i,12}^2(k) \mathbf{I} \quad (4.65)$$

$$\hat{\mathbf{R}}_{i,22}(k) = \sigma_{i,22}^2(k) \mathbf{I} \quad (4.66)$$

$$\sigma_{i,11}^2(k) = \mathbf{x}_{1,i}^T(k) \mathbf{x}_{1,i}(k) \quad (4.67)$$

$$\sigma_{i,12}^2(k) = \mathbf{x}_{1,i}^T(k) \mathbf{x}_{2,i}(k) \quad (4.68)$$

$$\sigma_{i,22}^2(k) = \mathbf{x}_{2,i}^T(k) \mathbf{x}_{2,i}(k) \quad (4.69)$$

Como a matriz  $\hat{\mathbf{R}}_i(k)$  é composta por blocos de matrizes diagonais, a sua inversa pode ser encontrada de maneira simples. Definindo a matriz de correlação inversa por

$$\hat{\mathbf{R}}_i(k)^{-1} = \begin{bmatrix} \hat{\mathbf{S}}_{i,11}(k) & \hat{\mathbf{S}}_{i,12}(k) \\ \hat{\mathbf{S}}_{i,21}(k) & \hat{\mathbf{S}}_{i,22}(k) \end{bmatrix} \quad (4.70)$$

temos que

$$\begin{bmatrix} \hat{\mathbf{R}}_{i,11}(k) & \hat{\mathbf{R}}_{i,12}(k) \\ \hat{\mathbf{R}}_{i,12}(k) & \hat{\mathbf{R}}_{i,22}(k) \end{bmatrix} \cdot \begin{bmatrix} \hat{\mathbf{S}}_{i,11}(k) & \hat{\mathbf{S}}_{i,12}(k) \\ \hat{\mathbf{S}}_{i,21}(k) & \hat{\mathbf{S}}_{i,22}(k) \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \quad (4.71)$$

de onde

$$\hat{\mathbf{R}}_{i,11}(k)\hat{\mathbf{S}}_{i,11}(k) + \hat{\mathbf{R}}_{i,12}(k)\hat{\mathbf{S}}_{i,21}(k) = \mathbf{I} \quad (4.72)$$

$$\hat{\mathbf{R}}_{i,11}(k)\hat{\mathbf{S}}_{i,12}(k) + \hat{\mathbf{R}}_{i,12}(k)\hat{\mathbf{S}}_{i,22}(k) = \mathbf{0} \quad (4.73)$$

$$\hat{\mathbf{R}}_{i,12}(k)\hat{\mathbf{S}}_{i,12}(k) + \hat{\mathbf{R}}_{i,22}(k)\hat{\mathbf{S}}_{i,22}(k) = \mathbf{I} \quad (4.74)$$

$$\hat{\mathbf{R}}_{i,12}(k)\hat{\mathbf{S}}_{i,11}(k) + \hat{\mathbf{R}}_{i,22}(k)\hat{\mathbf{S}}_{i,21}(k) = \mathbf{0} \quad (4.75)$$

Resolvendo as equações (4.73) e (4.75) para encontrar  $\hat{\mathbf{S}}_{i,12}(k)$  e  $\hat{\mathbf{S}}_{i,21}(k)$ , obtém-se:

$$\hat{\mathbf{S}}_{i,12}(k) = -\hat{\mathbf{R}}_{i,11}^{-1}(k)\hat{\mathbf{R}}_{i,12}(k)\hat{\mathbf{S}}_{i,22}(k) \quad (4.76)$$

$$\hat{\mathbf{S}}_{i,21}(k) = \hat{\mathbf{R}}_{i,22}^{-1}(k)\hat{\mathbf{R}}_{i,12}(k)\hat{\mathbf{S}}_{i,11}(k) \quad (4.77)$$

Combinando as equações (4.72), (4.74), (4.76) e (4.77), temos que

$$\hat{\mathbf{S}}_{i,11}(k) = (\hat{\mathbf{R}}_{i,11} - \hat{\mathbf{R}}_{i,12}\hat{\mathbf{R}}_{i,22}^{-1}\hat{\mathbf{R}}_{i,12})^{-1} \quad (4.78)$$

$$\hat{\mathbf{S}}_{i,22}(k) = (\hat{\mathbf{R}}_{i,22} - \hat{\mathbf{R}}_{i,12}\hat{\mathbf{R}}_{i,11}^{-1}\hat{\mathbf{R}}_{i,12})^{-1} \quad (4.79)$$

Assim, de forma mais simples, podemos escrever

$$\hat{\mathbf{S}}_{i,11}(k) = (\sigma_{i,11}^2(k) - (\sigma_{i,12}^2(k))^2\sigma_{i,22}^{-2}(k))^{-1}\mathbf{I} \quad (4.80)$$

$$\hat{\mathbf{S}}_{i,22}(k) = (\sigma_{i,22}^2(k) - (\sigma_{i,12}^2(k))^2\sigma_{i,11}^{-2}(k))^{-1}\mathbf{I} \quad (4.81)$$

$$\hat{\mathbf{S}}_{i,12}(k) = -\sigma_{i,11}^{-2}(k)\sigma_{i,12}^2(k)(\sigma_{i,22}^{-2}(k) - (\sigma_{i,12}^2(k))^2\sigma_{i,11}^2(k))^{-1}\mathbf{I} \quad (4.82)$$

$$\hat{\mathbf{S}}_{i,21}(k) = \sigma_{i,22}^{-2}(k)\sigma_{i,12}^2(k)(\sigma_{i,11}^2(k) - (\sigma_{i,12}^2(k))^2\sigma_{i,22}^{-2}(k))^{-1}\mathbf{I} \quad (4.83)$$



### 4.3 Resultados das simulações

Foram feitas simulações para três diferentes tempos de reverberação: 0.1 s, 0.25 s e 0.5 s. A medida de avaliação utilizada foi a SIR (*Signal to Interference Ratio*). O tamanho do filtro de separação utilizado foi o mesmo do filtro de mistura. Portanto, para  $T_{60}=0.1$  s, 0.25 s e 0.5 s, o tamanho de  $L$  utilizado foi 623, 1599 e 3227, respectivamente.

Na Figura 4.3 são mostradas as evoluções da SIR ao longo das iterações para os algoritmos NLMS, NSAF e NSAF modificado utilizando  $T_{60} = 0.1$  s. Percebe-se que o número de iterações necessárias para convergência do NSAF modificado é bem menor.

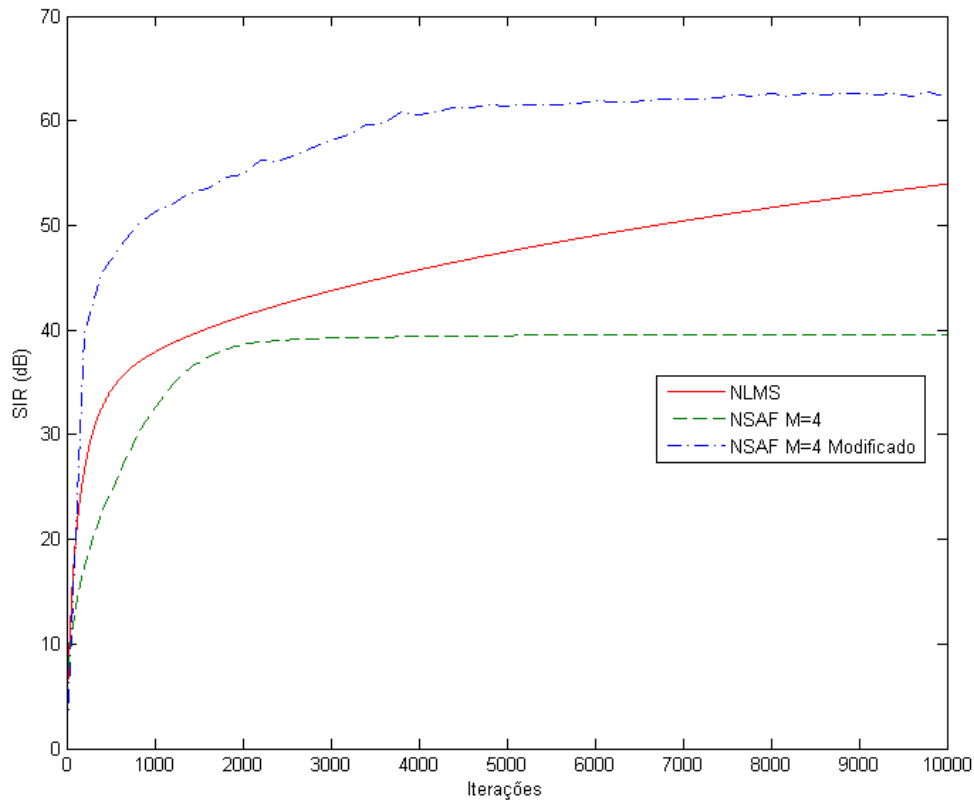


Figura 4.3: Comparação entre os algoritmos para tempo de reverberação igual a 0.1 s

Na Figura 4.4 são mostradas as evoluções da SIR ao longo das iterações para os algoritmos NLMS, NSAF e NSAF modificado utilizando  $T_{60} = 0.25$  s.

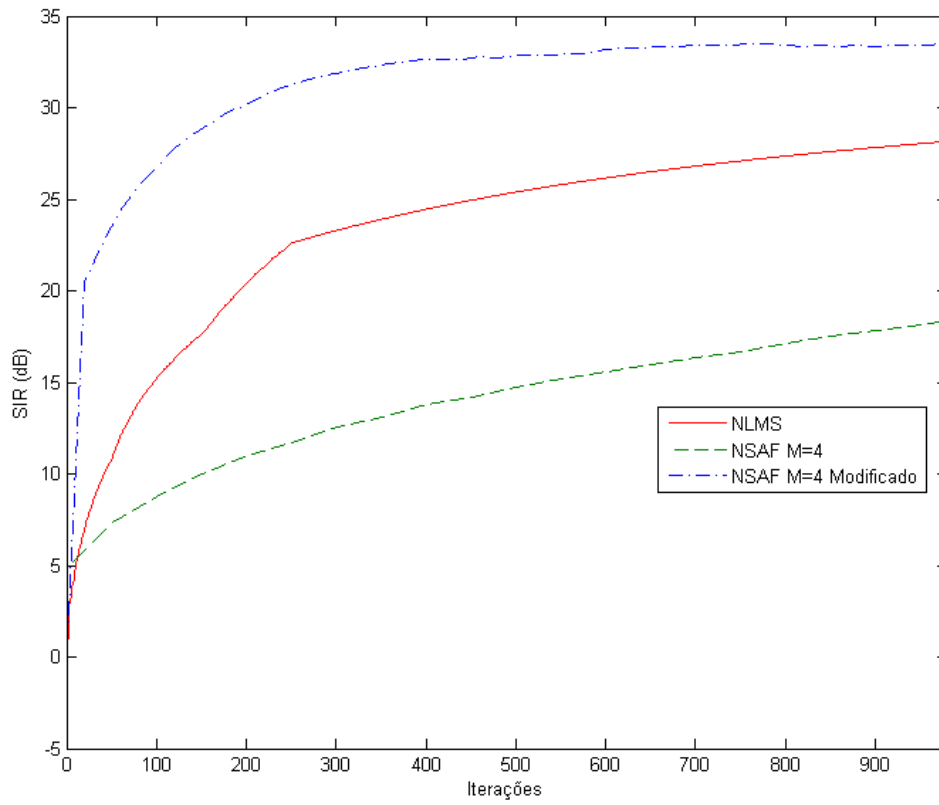


Figura 4.4: Comparação entre os algoritmos para tempo de reverberação igual a 0.25 s

Na Figura 4.5 são mostradas as evoluções da SIR ao longo das iterações para os algoritmos NLMS e NSAF utilizando  $T_{60} = 0.5$  s.

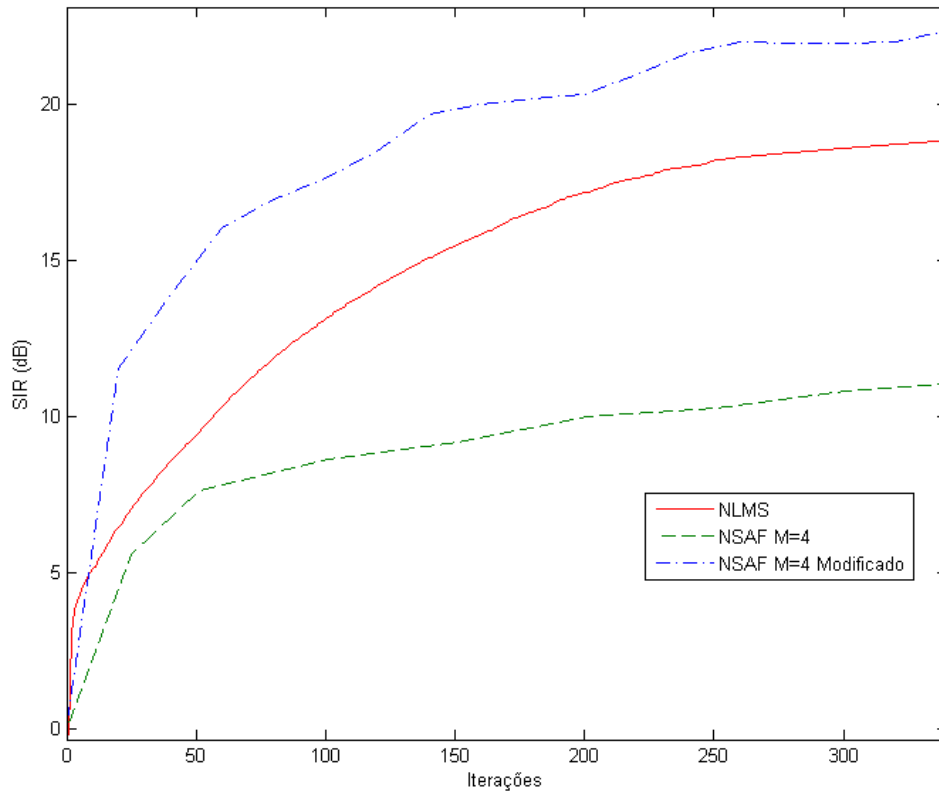


Figura 4.5: Comparação entre os algoritmos para tempo de reverberação igual a 0.5 s

## 4.4 Avaliação dos resultados

Para todos os tempos de reverberação simulados, observa-se uma convergência significativamente mais rápida com a modificação proposta para o algoritmo NSAF.

Nesse capítulo, conclui-se que existe uma limitação no modelo e que essa limitação aumenta conforme o tempo de reverberação também aumenta. Portanto, por melhor que seja o algoritmo de separação implementado, para os casos de alta reverberação, como 0.5 s, a SIR ótima a ser alcançada pode ultrapassar 20 dB. Enquanto que uma reverberação de 0.1 s, não limita tanto o modelo, sendo possível ultrapassar 60 dB.

## Capítulo 5

# Exploração das características dos sinais de voz e dos ambientes reverberantes para melhoria do algoritmo de separação TRINICON

Neste capítulo, algumas modificações foram feitas no algoritmo TRINICON na tentativa de melhorar a medida de avaliação SIR diante dos resultados obtidos nos capítulos anteriores.

Para as simulações em subbandas, foi utilizada a estrutura uniforme proposta em [35], ilustrada na Figura 5.1.

Cada sinal observado da mistura,  $x_i(k)$  é decomposto pelo filtro de análise  $F_n(z)$  sendo então decimados e filtrados pelo subfiltro de separação da  $n$ -ésima subbanda dado por  $W_{ij}^n(z)$ . Posteriormente, os sinais são interpolados e recombinaados através do filtro de síntese  $G_n(z)$  para gerar o sinal de saída para a  $n$ -ésima subbanda. A estimativa do sinal de cada fonte é encontrada por meio da soma dos sinais em cada subbanda, ou seja,  $\sum_{n=0}^{M_D-1} y_i^n(k)$ . Assim, as soluções são independentes nas diferentes subbandas.

As simulações foram feitas supondo  $Q = P$ , ou seja, número de fontes igual ao número de sensores e  $Q = 2$ , apenas duas fontes. Além disso, para gerar as misturas, foram utilizados os parâmetros da Tabela 5.1.

Os sinais de voz utilizados foram os mesmos do Capítulo 3, sendo a mistura composta de um sinal de voz masculino e outro feminino, disponibilizados em [30], com duração de 10 segundos e frequência de amostragem 16 kHz.

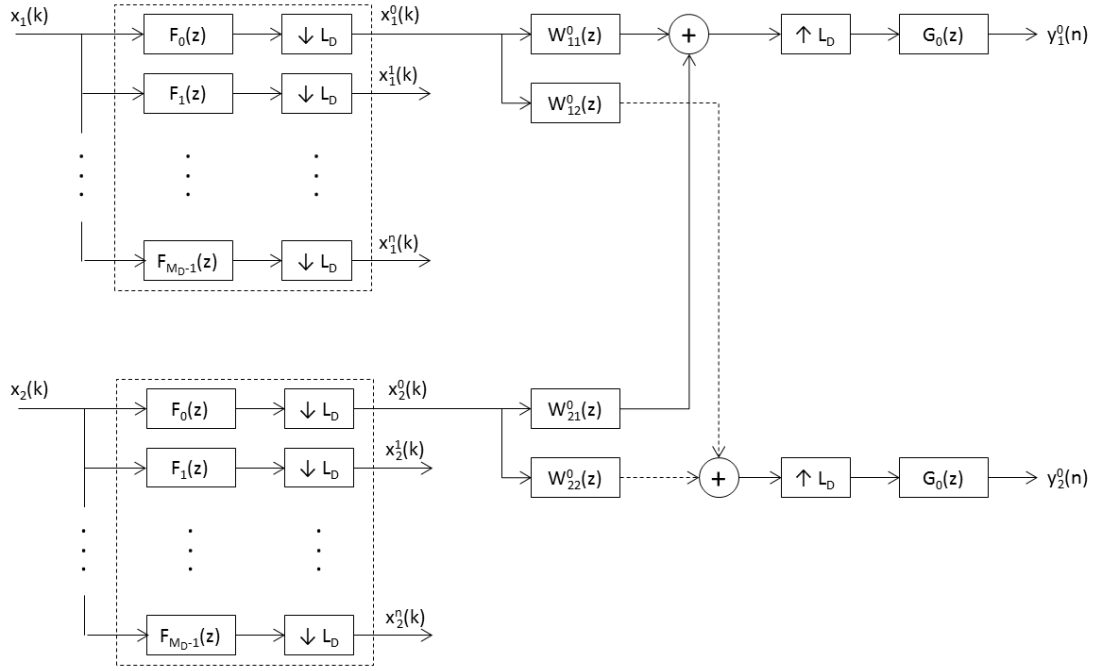


Figura 5.1: Estrutura empregada para separação em subbandas para o caso determinado e número de fontes igual a dois. Encontra-se explicitada apenas a geração da saída para a primeira subbanda.

Tabela 5.1: Parâmetros utilizados para geração do sinal de mistura

Parâmetro	Valor
Ângulo da fonte 1	$-45^\circ$
Ângulo da fonte 2	$45^\circ$
Coefficiente de absorção das paredes	0.8
Distância entre os microfones	5 cm
Distância das fontes até o centro do arranjo de microfones	1 m

## 5.1 Simulações para os sinais em banda completa

Primeiramente, foram feitas algumas simulações para o sinal em banda completa para servir de comparação posteriormente.

Na Tabela 5.2 pode-se ver os resultados obtidos após a convergência do algoritmo para  $T_{60} = 0.1$  s, para diferentes tamanhos do filtro de separação.

Tabela 5.2: Valores de SIR para simulação em banda completa com  $T_{60} = 0.1$  s

L	PDF gaussiana	PDF laplaciana
512	19.7868	20.6946
1024	21.6462	21.5733
2048	22.2089	21.9763
4096	20.9031	19.7562

Na Tabela 5.3 pode-se ver os resultados obtidos após a convergência do algoritmo

para  $T_{60} = 0.25s$ , para diferentes tamanhos do filtro de separação.

Tabela 5.3: Valores de SIR para simulação em banda completa com  $T_{60} = 0.25$  s

L	PDF gaussiana	PDF laplaciana
512	10.2997	10.2072
1024	11.8090	11.8443
2048	12.5039	12.5767
4096	11.8286	11.9309

Na Tabela 5.4 pode-se ver os resultados obtidos após a convergência do algoritmo para  $T_{60} = 0.5s$ , para diferentes tamanhos do filtro de separação.

Tabela 5.4: Valores de SIR para simulação em banda completa com  $T_{60} = 0.5$  s

L	PDF gaussiana	PDF laplaciana
1024	6.2815	6.0513
2048	5.9368	5.4572
4096	6.6756	6.7141
8192	6.1614	5.8406

## 5.2 Simulações para os sinais em subbandas

Nesta seção, o algoritmo foi modificado de maneira que as informações obtidas nos capítulos anteriores fossem incorporadas ao algoritmo TRINICON.

Para isso, nas simulações foram utilizadas quatro subbandas, sem decimação, com a configuração do algoritmo TRINICON, em cada subbanda, conforme mostrado na Tabela 5.5.

Tabela 5.5: Configuração do algoritmo TRINICON em subbandas

Subbanda	Função <i>score</i>	Tamanho do bloco (N)	Número de atrasos (D)
$M_D = 1$	PDF gaussiana	4L	L
$M_D = 2$	PDF laplaciana	2L	L
$M_D = 3$	PDF laplaciana	2L	L
$M_D = 4$	PDF laplaciana	2L	L

Para os tempos de reverberação 0.1 s, 0.25 s e 0.5 s, os tamanhos do filtro de separação utilizados, dados por  $L$ , foram 1024, 2048 e 4096 respectivamente.

Na Figura 5.2 são mostradas as evoluções da SIR ao longo das iterações para os algoritmos TRINICON em banda cheia e TRINICON em subbandas com os parâmetros da Tabela 5.5 para  $T_{60} = 0.1$  s.

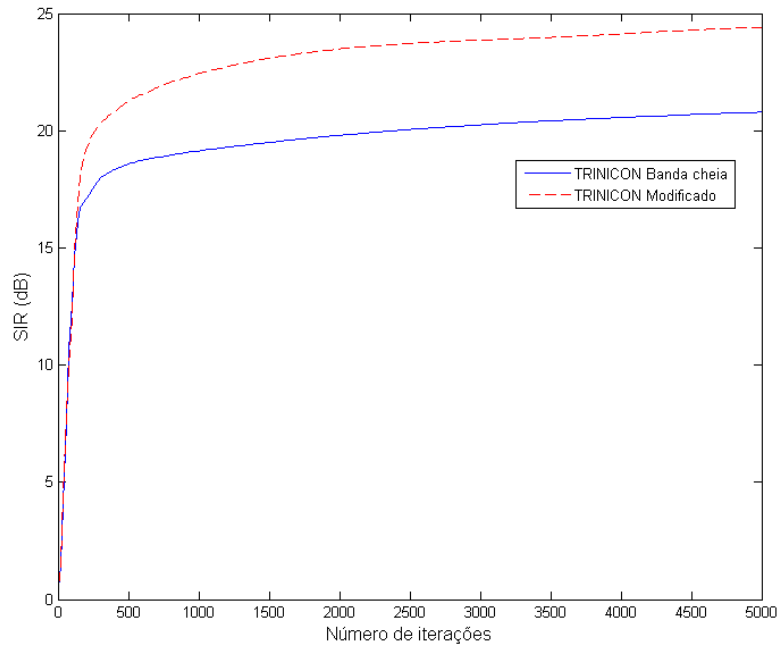


Figura 5.2: Comparação entre os algoritmos TRINICON para tempo de reverberação igual a 0.1 s

Na Figura 5.3 são mostradas as evoluções da SIR ao longo das iterações para os algoritmos TRINICON em banda cheia e TRINICON em subbandas com os parâmetros da Tabela 5.5 para  $T_{60} = 0.25$  s.

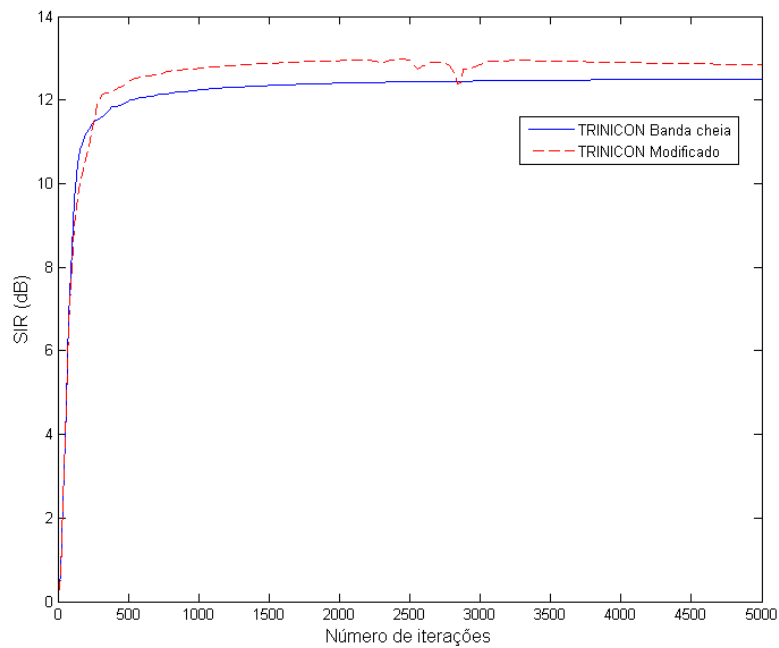


Figura 5.3: Comparação entre os algoritmos TRINICON para tempo de reverberação igual a 0.25 s

Na Figura 5.4 são mostradas as evoluções da SIR ao longo das iterações para os algoritmos TRINICON em banda cheia e TRINICON em subbandas com os parâmetros da Tabela 5.5 para  $T_{60} = 0.5$  s.

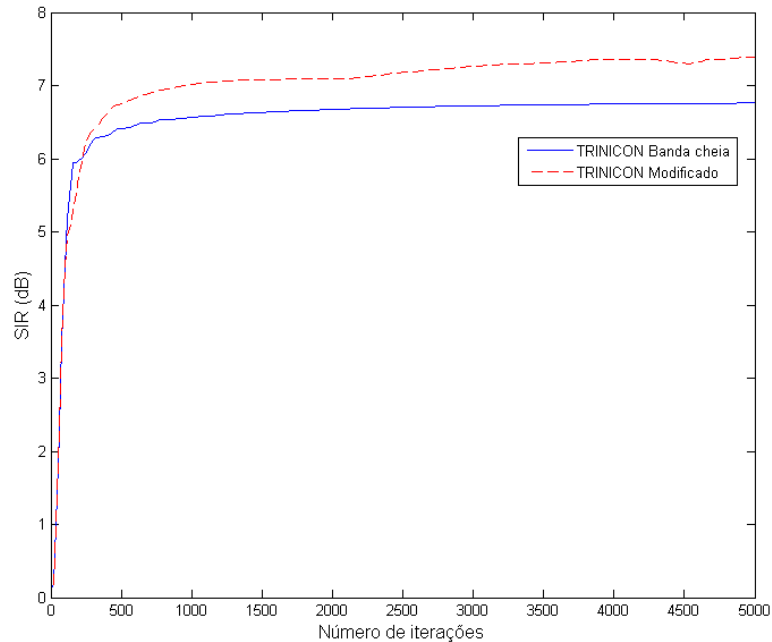


Figura 5.4: Comparação entre os algoritmos TRINICON para tempo de reverberação = 0.5 s

Observando os resultados, tem-se que para o tempo de reverberação igual a 0.1 s, a SIR passou de 20.79 dB, com o algoritmo TRINICON em banda cheia, para 24.4 dB, com o algoritmo TRINICON modificado; para o tempo de reverberação igual a 0.25 s, a SIR passou de 12.5 dB para 12.85 dB; e para o tempo de reverberação igual a 0.5 s, a SIR passou de 6.761 dB para 7.384 dB. Com isso, percebe-se que as modificações feitas melhoraram o desempenho do algoritmo sendo que o ganho com o processamento em subbandas foi maior para o menor tempo de reverberação.



# Capítulo 6

## Conclusões

Neste capítulo serão apresentadas as conclusões obtidas ao longo do trabalho.

No Capítulo 3, conclui-se que o sinal de voz possui características que podem ser exploradas para modificar os algoritmos de separação. Quando os sinais são decompostos em subbandas, percebe-se que, principalmente, a subbanda com as frequências mais baixas se destaca das demais. Por exemplo, nas bandas de baixa frequência a curtose é menor do que nas bandas de alta frequência indicando que sua PDF é mais larga do que as PDFs das demais subbandas. Na análise sobre não-estacionariedade, pode ser observado que o sinal da subbanda de baixa frequência gerou janelas maiores, o que sugeriu o uso de blocos de tamanhos maiores nesta faixa de frequência.

No Capítulo 4, foram identificadas as limitações dos algoritmos adaptativos supervisionados NLMS e NSAF para o problema de separação. Dessa forma, foi visto que para os coeficientes ótimos, em ambientes reverberantes,  $T_{60} = 0.5$  s, a SIR pode passar de 20 dB.

Por último, no Capítulo 5, foram apresentadas as simulações do algoritmo TRINICON com as modificações propostas. Houve uma melhoria na evolução da SIR, mas nenhum resultado chegou próximo do resultado possível utilizando os algoritmos adaptativos do Capítulo 4. Quanto menor o tempo de reverberação, maior foi a diferença entre as SIRs dos algoritmos TRINICON e TRINICON modificado. Para  $T_{60} = 0.1$  s, a melhoria foi de 4 dB, enquanto que para  $T_{60} = 0.25$  s, foi apenas 2 dB.

Este é um trabalho inicial que mostrou que é possível melhorar o desempenho do algoritmo TRINICON explorando as características do sinal de voz em cada subbanda. Portanto, existem ainda trabalhos futuros que podem ser feitos de forma a complementar esta dissertação. Espera-se que sejam obtidos ganhos maiores modificando a função custo, como as funções citadas em [25], e também utilizando banco de filtros não-uniforme. Também pode-se reduzir a complexidade do método empregando decimação, mas seria necessário compensar efeitos de *aliasing*.

# Referências Bibliográficas

- [1] ZHENG, Y., JOHNSTON, D., BERWICK, J., et al. “Signal Source Separation in the Analysis of Neural Activity in Brain”, *NeuroImage*, , n. 13, pp. 447–458, jan. 2001.
- [2] LATHAUWER, L. D., MOOR, B. D., VANDEWALLE, J. “Fetal Electrocardiogram Extraction by Blind Source Subspace Separation”, *IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING*, v. 47, n. 5, pp. 567–572, maio 2000.
- [3] WANG, L., ODANI, K., KAI, A., et al. “Speech recognition using blind source separation and dereverberation method for mixed sound of speech and music”, *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific*, pp. 1–4, nov. 2013.
- [4] MIAO, L., QI, H. “A Blind Source Separation Perspective on Image Restoration”, *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–7, jun. 2007.
- [5] LEHMANN, E. A., JOHANSSON, A. M. “Prediction of energy decay in room impulse responses simulated with an image-source model”, *Journal of the Acoustical Society of America*, v. 124, n. 1, pp. 269–277, July 2008.
- [6] LEHMANN, E. A., JOHANSSON, A. M., NORDHOLM, S. “Reverberation-Time Prediction Method for Room Impulse Responses Simulated with the Image-Source Model”. In: *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA'07)*, pp. 159–162, New Paltz, NY, USA, October 2007.
- [7] LEHMANN, E. A. “Código-fonte MATLAB para simulação de sala reverberante”. <http://www.eric-lehmann.com>, jun. 2015.
- [8] AICHNER, R. *Acoustic Blind Source Separation in Reverberant and Noisy Environments*. Tese de D.Sc., Universität Erlangen-Nürnberg, Erlangen, Alemanha, 2007.

- [9] KHANAGHA, V., KHANAGHA, A. “Solving global permutation ambiguity of time domain BSS using speaker specific features of speech signals”, *IEEE Symposium on Industrial Electronics and Applications*, pp. 1007–1011, 2009.
- [10] HIROSHI SAWADA, RYO MUKAI, S. A., MAKINO, S. “A Robust and Precise Method for Solving the Permutation Problem of Frequency-Domain Blind Source Separation”, *IEEE Transactions on Speech and Audio Processing*, v. 12, n. 5, pp. 530–538, 2004.
- [11] KIM, M., CHOI, S. “ICA-Based Clustering for Resolving Permutation Ambiguity in Frequency-Domain Convolutional Source Separation”, *18th International Conference on Pattern Recognition*, v. 1, pp. 950–954, 2006.
- [12] HYVÄRINEN, A., KARHUNEN, J., E. OJA. *Independent Component Analysis*. John-Wiley & Sons, 2001.
- [13] MITRA, S. K. *Digital Signal Processing: A Computer-Based Approach*. 4 ed. New York, McGraw-Hill, 2011.
- [14] VAIDYANATHAN, P. P. *Multirate Systems and Filter Banks*. New Jersey, Prentice-Hall, 1993.
- [15] E. VINCENT, R. G., FÉVOTTE, C. “Performance measurement in blind audio source separation”, *IEEE Transactions on Audio, Speech and Language Processing*, v. 14, pp. 1462–1469, 2006.
- [16] VINCENT, E. “Código-fonte MATLAB para medidas de avaliação”. [http://bass-db.gforge.inria.fr/bss\\_eval/](http://bass-db.gforge.inria.fr/bss_eval/), jun. 2015.
- [17] PARRA, L., SAJDA, P. “Blind Source Separation via Generalized Eigenvalue Decomposition”, *Journal of Machine Learning Research*, v. 4, pp. 1261–1269, 2003.
- [18] PARRA, L., SPENCE, C. “Convolutional Blind Separation of Non-Stationary Sources”, *IEEE Transactions on Speech and Audio Processing*, v. 8, pp. 320–327, 2000.
- [19] WEINSTEIN, E., FEDER, M., OPPENHEIM, A. “Multi-Channel Signal Separation by Decorrelation”, *IEEE Trans. Speech Audio Processing*, v. 1, n. 4, pp. 405–413, 1993.
- [20] MITSURU KAWAMOTOA, K. M., OHNISHIA, N. “A method of blind separation for convolved non-stationary signals”, *Neurocomputing*, v. 22, pp. 157–171, 1998.

- [21] MOLGEDEY, L., SCHUSTER, H. G. “Separation of a Mixture of Independent Signals Using Time Delayed Correlations”, *Physical Review Letters*, v. 72, n. 23, pp. 3634–3637, jun. 1994.
- [22] LIU, R., S. V., HUANG, Y.-F. “Indeterminacy and identifiability of blind identification”, *IEEE Trans. on Circuits and Systems*, v. 38, n. 5, pp. 499–509, 1991.
- [23] GERVEN, S. V., COMPERNOLLE, D. V. “Signal separation by symmetric adaptive decorrelation: stability, convergence, and uniqueness”, *IEEE Trans. Signal Processing*, v. 43, n. 7, pp. 1602–1612, 1995.
- [24] SMARAGDIS, P. “Blind Separation of convolved mixtures in the frequency domain”, *Neurocomputing*, v. 22, pp. 21–34, 1998.
- [25] VICTORIO, L. *Algoritmos de separação cega de sinais de áudio no domínio da frequência em ambientes reverberantes: Estudo e Comparações*. Dissertação de m.sc., Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil, 2010.
- [26] BUCHNER, H., AICHNER, R., KELLERMANN, W. “Blind Source Separation for Convolutional Mixtures Exploiting Nongaussianity, Nonwhiteness, and Nonstationarity”, *International Workshop on Acoustic Echo and Noise Control (IWAENC2003)*, pp. 275–278, 2003.
- [27] COVER, T., THOMAS, J. *Elements of Information Theory*. New York, Wiley & Sons, 1991.
- [28] S. AMARI, A. C., YANG, H. H. “A New Learning Algorithm for Blind Signal Separation”, *Advances in Neural Information Processing Systems*, v. 8, pp. 757–763, 1996.
- [29] BREHM, H., STAMMLER, W. “Description and generation of spherically invariant speech-model signals”, *Signal Processing*, v. 12, pp. 119–141, 1987.
- [30] VINCENT, E., SAWADA, H. “Código-fonte MATLAB para simulação de sala reverberante”. <http://www.irisa.fr/metiss/SASSECO7>, jun. 2015.
- [31] RUDOY, D., BASU, P., QUATIERI, T. F., et al. “Adaptive short-time analysis-synthesis for speech enhancement.” In: *ICASSP*, pp. 4905–4908. IEEE, 2008. ISBN: 1-4244-1484-9. Disponível em: <<http://dblp.uni-trier.de/db/conf/icassp/icassp2008.html#RudoyBQDW08>>.

- [32] LEE, K. A., GAN, W. S. “Improving Convergence of the NLMS Algorithm Using Constrained Subband Updates”, *IEEE Signal Processing Letters*, v. 11, n. 9, pp. 736–739, 2004.
- [33] HAYKIN, S. O. *Adaptive Filter Theory*. Prentice-Hall, 2013.
- [34] *Adaptive Filtering: Algorithms and Practical Implementation*. 3 ed. New York, Springer, 2008.
- [35] HADDAD, D. B. *Estruturas em Subbandas para Filtragem Adaptativa e Separação Cega e Semi-Cega de sinais de voz*. Tese de D.Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2013.