

# ANÁLISE DE COMPONENTES ESPARSOS: SEPARAÇÃO CEGA DE FONTES E APLICAÇÕES

**Diego Barreto Haddad**

PADS, COPPE/Poli/UFRJ, Rio de Janeiro, Brasil

Coord. Telecomunicações, CEFET-RJ UNED Nova Iguaçu, Rio de Janeiro, Brasil

diego@pads.ufrj.br

**Mariane Rembold Petraglia**

PADS, COPPE/Poli/UFRJ, Rio de Janeiro, Brasil

mariane@pads.ufrj.br

**Paulo Bulkool Batalheiro**

PROSAICO, PEL/DETEL/UERJ, Rio de Janeiro, Brasil

bulkool@pads.ufrj.br

**Jorge Costa Pires Filho**

PADS, COPPE/Poli/UFRJ, Rio de Janeiro, Brasil

Guerra Eletrônica, Instituto de Pesquisa da Marinha (IPqM), Rio de Janeiro, Brasil

jcpfilho@pads.ufrj.br

**Resumo** – Em diversas aplicações de separação cega de fontes, a hipótese de que o número de fontes não supera o número de misturas não é satisfeita. Neste contexto, as mais conhecidas técnicas de Análise de Componentes Independentes em geral não podem ser empregadas e cabe recorrer a outros métodos - entre os quais se destaca a Análise de Componentes Esparsos (SCA, do inglês *Análise de Componentes Esparsos*). Este tutorial almeja expor os fundamentos das principais abordagens de Análise de Componentes Esparsos, para o caso de misturas lineares e instantâneas, bem como apresentar algumas de suas aplicações. Ao final, o uso de SCA foi validado numa aplicação de reconhecimento automático de instrumentos musicais.

**Palavras-chave** – Análise de componentes esparsos, bancos de filtros, transformadas, separação cega de sinais.

**Abstract** – In several applications of blind source separation, the hypothesis that the number of sources does not exceed the number of mixtures is not satisfied. In this context, the best known techniques of Independent Component Analysis (ICA) in general cannot be employed and other methods are used - among which stands out the Sparse Component Analysis (SCA). This tutorial aims to present the fundamentals of the main approaches of Sparse Component Analysis, for the instantaneous linear case, as well as to describe some of its applications. At the end, the use of SCA is validated in an application of automatic recognition of musical instruments.

**Keywords** – Sparse component analysis, filterbanks, transforms, signals blind separation.

## 1. INTRODUÇÃO

Desde que sensores para gravação de sinais se disseminaram, surge o problema de recuperar sinais a partir de misturas. Este problema possui estreita relação com a habilidade humana denominada *cocktail party* (festa de coquetel) a qual permite-nos discernir, num ambiente onde coexistem diversas fontes acústicas, o sinal de interesse em detrimento das interferências. Embora nosso sistema cerebral se desincumba da tarefa de forma aparentemente simples, sabe-se que a reprodução desta propriedade em ambientes microprocessados é um desafio (o qual parecia “aparentemente impossível” [1]), que ensejou intensa pesquisa. A partir de meados da década de 90, surgem os primeiros algoritmos [2],[3],[4] relativamente bem-sucedidos em algumas configurações de misturas instantâneas (embora na década de 80 já haja alguns métodos para contemplar esta tarefa).

O problema da separação de fontes tem seu paradigma solicitado como etapa de pré-processamento para obtenção de dados não contaminados para análise posterior [5], também abordando interessantes problemas em estatísticas de mais alta ordem e sistemas não-lineares [6]. A mineração de dados e a compreensão de bancos de dados complexos também motivam técnicas de separação de fontes [7],[8]. Entre as inúmeras aplicações destas técnicas temos: sensoriamento remoto [9], análise de sinais biomédicos [10], sísmicos [11] e de comunicações digitais [12].

A abordagem mais empregada para contemplar o problema é a Análise de Componentes Independentes (ICA) [2], bem como extensões da ICA para configurações convolutivas [13],[14], porém a relação entre o número  $M$  de sensores e o número  $N$  de fontes é um parâmetro essencial para a definição do método a empregar. Quando  $M = N$  (configuração determinada), a abordagem padrão de ICA pode ser utilizada. Configurações nas quais o número de misturas supera o número de fontes (superdeterminadas) podem exigir uma etapa de pré-processamento, em geral baseado em PCA [15], antes de aplicar a ICA. Já configurações onde  $M < N$  (subdeterminadas ou degeneradas [16]), mais difíceis, devem ser divididas em dois casos: i)  $M = 1$  e ii)  $M > 1$ . O caso  $M = 1$  é extremamente difícil; a maioria das abordagens para este caso assumem um conhecimento estatístico mais detalhado das fontes [17],[18] (ou de seu espectro), sendo em geral necessário o acesso a trechos isolados das fontes para treinamento; outras possibilidades interessantes (com resultados ainda pouco satisfatórios para sinais de áudio), residem na fatoração não-negativa de matrizes [19],[7] ou em redes neurais de terceira geração [20].

Quando  $1 < M < N$ , empregar a ICA torna-se algo extremamente difícil [16], mas a diversidade espacial (oriunda do fato da contribuição de cada fonte diferir entre os sensores, devido às distintas localizações destes) ainda pode ser utilizada. Aqui reside a grande utilidade da SCA, a qual introduz a hipótese de esparsidade acerca das fontes, podendo ser útil mesmo quando a hipótese de independência entre as fontes não é satisfeita [16], sendo considerada mais poderosa do que a ICA [21].

Entre os modelos de mistura, enfatizamos os lineares, não ruidosos e instantâneos, os quais podem ser estendidos para o caso convolutivo [22]. Este tutorial tem por objetivo fornecer um panorama das técnicas de SCA, colocando o leitor a par dos conhecimentos primordiais para o início da pesquisa na área.

## 2. MISTURA E SEPARAÇÃO: MODELOS E PROBLEMAS

Seja  $s_i(n)$  a  $n$ -ésima amostra da  $i$ -ésima fonte (sinal a recuperar, na nomenclatura da separação cega de fontes - BSS, do inglês *Blind Source Separation*). Tendo  $N$  fontes, podemos formar o vetor  $\mathbf{s}(n)$  das fontes, definido como  $\mathbf{s}(n) = [s_1(n) \ \cdots \ s_N(n)]^T$ . Seja  $\mathbf{H}$  a matriz, denominada matriz de mistura ou de *crossstalk* [21], que contém os ganhos de cada percurso fonte-sensor, de forma que o elemento  $h_{ij}$  corresponda ao ganho do caminho entre a  $j$ -ésima fonte e o  $i$ -ésimo sensor. Sendo  $M$  o número de sensores (ou de misturas), podemos formar o vetor  $\mathbf{x}(n)$ , o qual contém as misturas, de forma que  $\mathbf{x}(n) = \mathbf{H}\mathbf{s}(n)$ , com a matriz  $\mathbf{H}$  possuindo dimensões  $M \times N$ . Este modelo é denominado de instantâneo e linear.

Tendo acesso somente às misturas, ignoramos tanto as fontes quanto a matriz de mistura. Isto significa que estamos *às cegas*, daí o adjetivo cega para esta configuração de separação de fontes. Recuperar as fontes com pouco conhecimento acerca das mesmas, assim como da matriz de misturas, torna a BSS, cuja estrutura é vista na Figura 1, extremamente poderosa. Porém, desta ignorância surgem alguns problemas, vistos a seguir.

Quando o número de fontes iguala o de misturas e  $\mathbf{H}$  é não-singular, basta-nos estimar a inversa  $\mathbf{W}$  de  $\mathbf{H}$ . Caso o posto de  $\mathbf{H}$

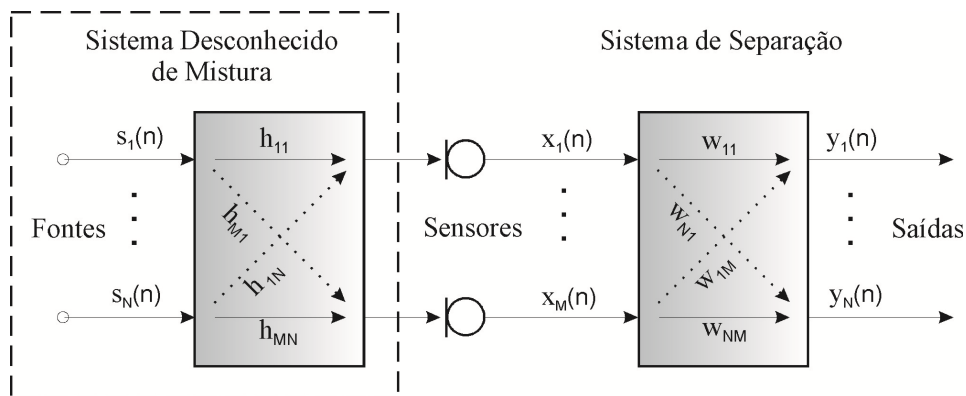


Figura 1: Estrutura de um algoritmo de separação de fontes para os casos em que o número de fontes não supera o de sensores.

seja inferior a  $N$ , combinações lineares das misturas podem fornecer algumas fontes, mas não todas [1]. Tanto num caso como no outro, as estimativas padecem da ambiguidade de escalamento e do problema da permutação.

Frequentemente no processo de separação é necessário empregar-se algum conhecimento estatístico das fontes (como supor suas amostras distribuídas segundo uma pdf<sup>1</sup> subgaussiana). Este conhecimento, ainda que inexato, é chamado de parâmetro de *nuisance* [23] dado que, estando primordialmente interessados em estimar as fontes, temos que conhecer algo de suas distribuições. Alterar a escala de uma fonte (ou seja, multiplicá-la por um número real) modifica sua pdf, mas o formato desta se mantém o mesmo, a menos de expansões ou compressões. Assim, conhecer (aproximadamente) o formato da distribuição das fontes não é suficiente para termos o nível dinâmico (ou a variância) das mesmas. Além disso, multiplicar a  $i$ -ésima coluna da matriz de mistura por um escalar ou a  $i$ -ésima fonte pelo mesmo escalar produz o mesmo efeito para as misturas. Como ignoramos tanto o nível dinâmico das fontes quanto a matriz de mistura, deparamo-nos com uma ambiguidade de escalamento

<sup>1</sup>Função densidade de probabilidade, do inglês *probability density function*.

impossível de ser contornada. Segue-se daí que podemos assumir que as fontes têm variância unitária<sup>2</sup> (o que é comum em algoritmos de ICA [8]) ou que a primeira linha da matriz de mistura seja formada por elementos 1s ou ainda que as colunas de  $\mathbf{H}$  possuem norma unitária (hipóteses comuns de diferentes etapas da SCA).

O problema da permutação reside no fato de não conhecermos *a priori* a rotulagem que se faz das fontes (ou seja, qual a primeira, qual a segunda, etc) na hora da separação. Ele implica que as nossas estimativas possam estar permutadas à ordem adotada para as fontes originais. A permutação é inerente aos métodos de BSS, podendo ser em alguns casos contornada com algum pós-processamento (por exemplo, quando é sabido que uma das fontes é um sinal de voz e as outras são ruídos com características espectrais distintas [24]).

A Figura 2 permite-nos visualizar a ambiguidade de escalamento, bem como o problema de permutação numa separação considerada bem-sucedida de separação cega de fontes. Observa-se, por exemplo, que a terceira estimativa (Fig. 2 (i)) contém a primeira fonte (Fig. 2 (a) - permutação e escalamento) e que a segunda fonte (Fig. 2 (b)) é estimada invertida (devido à ambiguidade de escalamento) e permutada (ver Fig. 2 (g)).

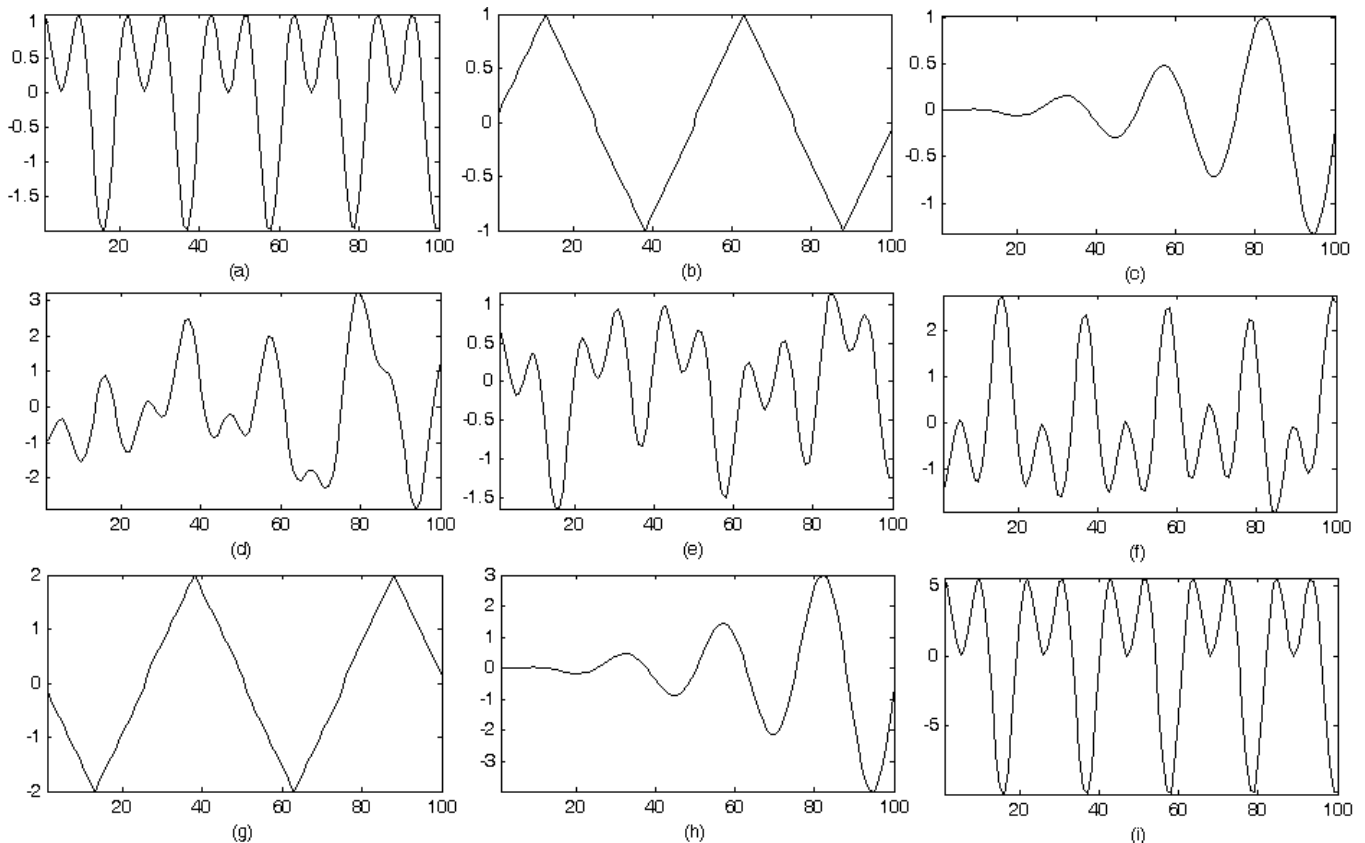


Figura 2: Sinais envolvidos em um sistema determinado de separação cega de fontes ( $M = N$ ). (a), (b) e (c): fontes originais; (d), (e) e (f): misturas instantâneas; (g), (h) e (i): estimativas.

### 3. ESPARSIDADE

A hipótese central da SCA consiste em supor fontes esparsas. A esparsidade significa que a maior parte das amostras de cada fonte estão próximas de zero ou que, alternativamente, relativamente poucas amostras concentrem a maior parte da energia de cada fonte. Observamos que o conceito de esparsidade aqui empregado é menos forte que o utilizado em análise numérica.

Desafortunadamente, muitos sinais de interesse, como sinais de áudio, voz e dados financeiros não apresentam esta propriedade. Este fato não impede o recurso à SCA desde que, através de uma transformada, encontremos uma representação esparsa das fontes [16], representação essa que consiste em importante tema de pesquisa [7].

Assim, é comum ter como primeiro passo da SCA o emprego de uma transformada esparsificadora, pois a projeção das misturas para um espaço apropriado de representação esparsa aumenta a qualidade da separação. Os elementos do espaço de representação não precisam ser linearmente independentes, podendo formar um dicionário redundante [21]. Entre as transformações mais utilizadas, temos a STFT (*Short-Time Fourier Transform*) [25], a *wavelet packet* [21],[26] e as correspondentes dos algoritmos derivados do *Matching Pursuit* [27]. Após a transformada, a SCA atua nos coeficientes da mesma tratando-os quais amostras das fontes. Após as estimativas de cada fonte serem obtidas, é necessário, quando utilizamos uma transformada esparsificadora,

<sup>2</sup>Mesmo que a variância das fontes seja conhecida (por exemplo em sinais de comunicações digitais), ainda poderia restar uma ambiguidade de sinal caso as fontes assumissem valores positivos e negativos. Caso as fontes possuam amostras complexas, pode advir uma ambiguidade de fase.

efetuar a transformada inversa da mesma, pois na realidade nestes casos a SCA estima os coeficientes das fontes no domínio da transformada. Logo, o último passo da SCA efetua a transformada inversa da empregada no primeiro passo, passando as estimativas no domínio da transformada para o domínio original de cada fonte (em geral, no domínio do tempo).

O segundo passo da SCA consiste na estimativa da matriz  $\mathbf{H}$  de mistura. Eis uma importante diferença entre a SCA e a ICA, já que, enquanto esta efetua sua busca no espaço das matrizes de separação, aquela almeja estimar diretamente a matriz de mistura. Uma das razões para este proceder reside no fato de  $\mathbf{H}$  não possuir inversa, por não ter posto completo nos casos em que  $M < N$  (configurações que são o principal foco da SCA).

Após estimada a matriz  $\mathbf{H}$  (a menos das ambiguidades de escalamento e permutação), a SCA empreende a reconstrução das fontes. Esta etapa não é trivial como nas configurações determinadas, onde basta-nos multiplicar a matriz das misturas pela estimativa de sua inversa. Em outras palavras, em contextos subdeterminados, um boa estimativa de  $\mathbf{H}$  não implica termos razoáveis estimativas das fontes.

Assim, podemos condensar a SCA em quatro passos: i) aplicação de uma transformada esparsificadora; ii) estimativa da matriz  $\mathbf{H}$ ; iii) estimativa dos coeficientes das fontes no domínio da transformada ou reconstrução das fontes e iv) aplicação da inversa da transformada empregada em (i). Em geral, a ICA não utiliza as etapas (i) e (iv) (embora recorrer a transformadas possa ser um procedimento útil [28],[29]), condensando as etapas (ii) e (iii) numa única (já que encontra diretamente uma estimativa da inversa da matriz de misturas, o que basta para reconstruir as fontes). Embora existam alguns métodos de SCA que estimam simultaneamente as fontes e a matriz de mistura [30], a maioria aborda estas tarefas de forma independente, resultando num custo computacional muito menor, sem aparentemente prejudicar a qualidade da separação. Por esta razão, estas técnicas serão o foco deste tutorial.

#### 4. ESTIMATIVA DA MATRIZ DE MISTURA

Seja  $\mathbf{h}^j$  a  $j$ -ésima coluna da matriz de mistura  $\mathbf{H}$ . Podemos então expressar  $\mathbf{x}(n)$  como:

$$\mathbf{x}(n) = \sum_{j=1}^N \mathbf{h}^j s_j(n), \quad (1)$$

donde se vê que cada vetor  $\mathbf{x}(n)$  é uma combinação linear das colunas da matriz de misturas. Por exemplo, se a matriz  $\mathbf{H}$  é:

$$\mathbf{H} = \begin{bmatrix} 0,3273 & -0,1867 \\ 0,1746 & 0,7258 \end{bmatrix},$$

e temos  $\mathbf{s}(n) = [1,5 \ 4]^T$ , então  $\mathbf{x}(n) = [-0,2559 \ 3,1651]^T$ , conforme ilustra a Figura 3.

Tendo pressuposta a esparsidade das fontes, espera-se que a maior parte dos coeficientes das fontes sejam próximos de

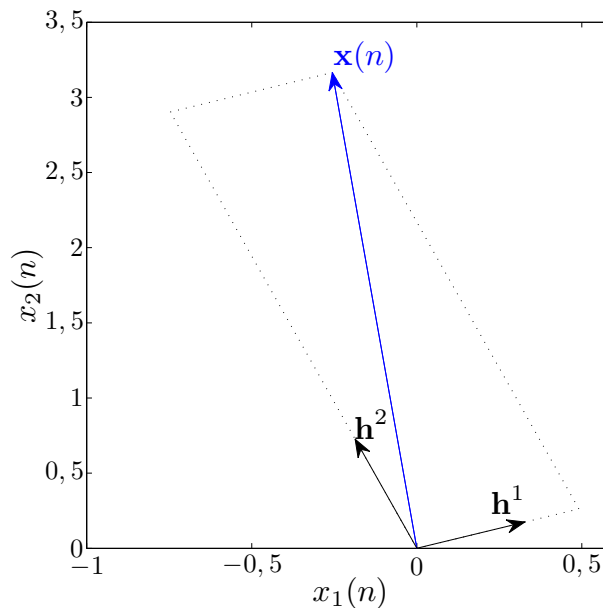


Figura 3: Exemplo dos vetores de mistura e da matriz de mistura numa configuração determinada; os vetores em preto são as colunas da matriz de mistura; o vetor  $\mathbf{x}(n)$  aparece em azul.

zero. Em outras palavras, temos que em cada instante  $n$ , em geral, apenas uma fonte (ou, equivalentemente, a coluna da matriz

de mistura correspondente) contribui fortemente na combinação linear que gera o vetor  $\mathbf{x}(n)$ . Esta fonte é chamada de fonte dominante. Isto significa que, caso plotemos todos os vetores  $\mathbf{x}(n)$ , verificaremos uma tendência de aglomeração destes vetores em direções específicas. Estas direções preferenciais são exatamente as direções das colunas da matriz  $\mathbf{H}$ .

A Figura 4 ilustra a formação de retas direcionais nos casos  $2 \times 3$  e  $3 \times 5$  (onde  $M \times N$  indica  $M$  misturas e  $N$  fontes), tanto no domínio do tempo quanto no domínio da transformada, para misturas de sinais de voz e música. Nos gráficos Fig. 4 (b) e Fig. 4 (d), a transformada utilizada foi a *wavelet packet* de 7 níveis com *wavelet* Daubechies 32. Verifica-se que as retas são melhor visualizadas no domínio da transformada, já que os coeficientes desta são mais esparsos do que as amostras das fontes no domínio do tempo.

Se as direções preferenciais dos vetores  $\mathbf{x}(n)$  coincidem com as direções dos vetores coluna da matriz de mistura, podemos

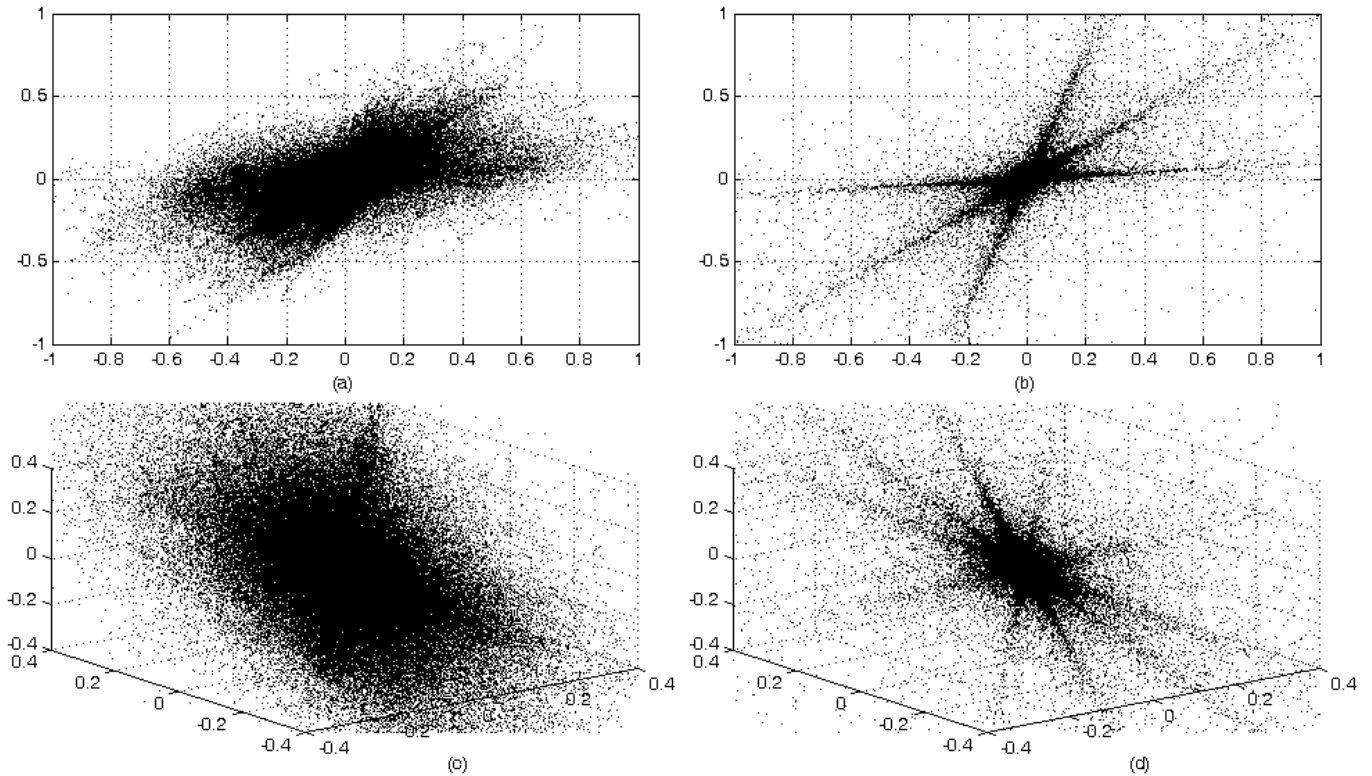


Figura 4: Pontos dos vetores  $\mathbf{x}(n)$ . (a) configuração  $2 \times 3$  (domínio do tempo); (b) configuração  $2 \times 3$  (domínio da transformada); (c) configuração  $3 \times 5$  (domínio do tempo) e (d) configuração  $3 \times 5$  (domínio da transformada).

estimar a matriz de mistura a partir do conhecimento destas direções preferenciais. Antes de apresentar alguns dos métodos que estimam estas direções preferenciais, vejamos exatamente qual é a relação matemática entre as direções das retas, as quais sempre passam pela origem do sistema de coordenadas, e os vetores coluna supracitados.

Assumindo 2 misturas ( $M = 2$ ) e admitindo que somente a  $i$ -ésima fonte não é nula no instante  $n$ , temos:

$$\begin{aligned} x_1(n) &= \sum_{j=1}^N h_{1j}s_j(n) = h_{1i}s_i(n), \\ x_2(n) &= \sum_{j=1}^N h_{2j}s_j(n) = h_{2i}s_i(n), \\ \frac{x_2(n)}{x_1(n)} &= \frac{h_{2i}}{h_{1i}} \Rightarrow x_2(n) = \frac{h_{2i}}{h_{1i}} x_1(n). \end{aligned} \quad (2)$$

Em outras palavras, quando somente a  $i$ -ésima fonte está ativa,  $x_2(n)$  torna-se proporcional a  $x_1(n)$ , com a constante de proporcionalidade dependendo da  $i$ -ésima coluna da matriz de mistura. Tendo acesso a  $\mathbf{x}(n)$ , podemos estimar esta razão. Porém sempre há um grau de liberdade na solução, pois a razão depende de 2 números. Um procedimento comum em SCA para contornar este problema consiste em supor que a primeira linha da matriz de mistura apresenta apenas elementos iguais a 1 (este não é um procedimento problemático, dada a ambiguidade de escalamento - vide Seção 1). Assim, chegamos a:

$$x_2(n) = h_{2i}x_1(n). \quad (3)$$

Tendo o vetor  $\mathbf{x}(n)$  por componentes  $x_1(n)$  e  $x_2(n)$ , a tangente do seu ângulo pode ser definida como:

$$\operatorname{tg}(\psi^{(i)}) = \frac{x_2(n)}{x_1(n)} = \frac{h_{2i}x_1(n)}{x_1(n)} = h_{2i}, \quad (4)$$

de onde vemos que, após estimados os ângulos preferenciais, a tangente destes consiste na estimativa da segunda linha da matriz de mistura<sup>3</sup>. Vejamos a seguir alguns dos métodos que almejam estimar estas direções preferenciais.

#### 4.1. VARREDURA DE ZIBULEVSKY

Proposta em [25], a abordagem da Varredura de Zibulevsky consiste num dos primeiros métodos da SCA. É um método simples e altamente competitivo em relação aos mais recentes na literatura. O método consiste na formação de um histograma angular, numa busca dos ângulos do gráfico  $x_2 \times x_1$  que possuem maior densidade de pontos. Para a formação do histograma, são escolhidos ângulos teste (tipicamente centenas deles) uniformemente distribuídos no intervalo  $[0, \pi]$ . Não é necessário contemplar o intervalo  $(\pi, 2\pi)$ , já que uma reta definida por um ângulo  $\theta$  equivale a uma definida pelo ângulo  $\theta + \pi$  (os ângulos são os únicos parâmetros necessários para determinar as retas, dado que todas elas passam pela origem), bastando que os vetores  $\mathbf{x}(n)$  cujos ângulos encontrem-se no intervalo  $(\pi, 2\pi)$  sejam refletidos simetricamente, de forma que seus ângulos passem a se encontrar no intervalo  $[0, \pi]$ . Para cada ângulo de teste  $\theta_t$ , a varredura calcula a contribuição de todos os ângulos  $\theta_n$  (associados aos vetores  $\mathbf{x}(n)$ ), somando estas contribuições ao final para gerar a “densidade” de pontos na vizinhança de  $\theta_t$ . A contribuição de  $\mathbf{x}(n)$  e de seu respectivo  $\theta_n$  a  $\theta_t$  é avaliada segundo a “função potencial” abaixo:

$$\Phi(\theta_t, \theta_n, \mathbf{x}(n)) = \begin{cases} \lambda |\mathbf{x}(n)| \left(1 - \frac{|\Delta|}{\pi/4}\right), & |\Delta| < \frac{\pi}{4} \\ 0, & |\Delta| \geq \frac{\pi}{4} \end{cases}, \quad (5)$$

onde  $\Delta = \theta_t - \theta_n$  e  $\lambda$  é um parâmetro arbitrário, geralmente da ordem de dezenas. A função potencial dá maior peso a ângulos  $\theta_n$  próximos de  $\theta_t$  (anulando a contribuição quando se encontram muito distantes) e atribui uma confiança maior aos  $\mathbf{x}(n)$  de maior módulo, dado que o erro angular destes é menos suscetível a ruído.

Na realidade, assim como na maior parte da literatura, a varredura de Zibulevsky, no cálculo da função potencial, despreza os vetores  $\mathbf{x}(n)$  de módulos baixos, pois estes são mais sensíveis a interferências. Este procedimento possui a vantagem de agilizar muito o processamento (dado que a maioria dos vetores  $\mathbf{x}(n)$  possuem módulo baixo). Tipicamente, desprezam-se os pontos cujo módulo seja inferior a um décimo do maior módulo de  $\mathbf{x}(n)$ , já que não contribuem tanto para uma distribuição supergaussiana [31]. Um histograma típico da varredura de Zibulevsky para uma configuração  $2 \times 3$  é mostrado na Figura 5, sendo que cada um dos três picos associa-se a uma das fontes.

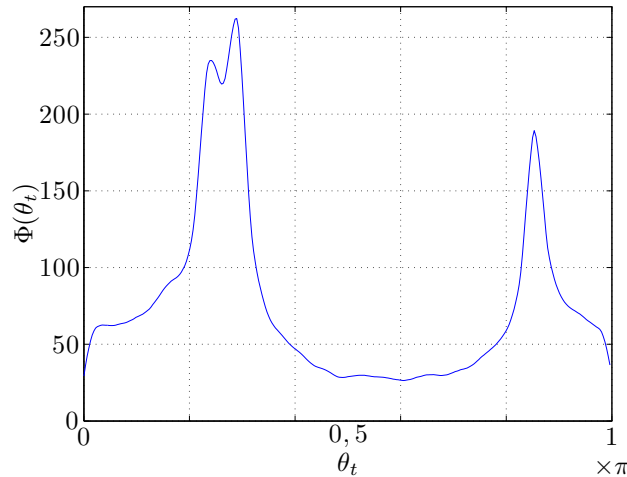


Figura 5: Histograma típico obtido pela varredura de Zibulevsky.

#### 4.2. MODELOS DE MISTURAS LAPLACIANAS

Uma função densidade de probabilidade é dita *laplaciana* se puder ser expressa do seguinte modo:

$$\mathcal{L}(\theta, c, m) = ce^{-2c|\theta-m|}, \quad (6)$$

<sup>3</sup>Note que a tangente do ângulo  $\frac{\pi}{2}$  é infinita. Esta situação implica que um dos sensores não capta nada de uma das fontes, o que significa que não temos uma mistura em cada sensor. Na prática, esta possibilidade é incomum e não será abordada neste tutorial.

onde  $m$  é a média do modelo e  $c$  (positivo) é um fator relacionado à dispersão da pdf. Já um modelo de  $N$  misturas de laplacianas pode ser escrito como:

$$p(\theta) = \sum_{i=1}^N \alpha_i \mathcal{L}(\theta, c_i, m_i) = \sum_{i=1}^N \alpha_i c_i e^{-2c_i|\theta-m_i|}, \quad (7)$$

onde o subíndice  $i$  denota os parâmetros correspondentes à  $i$ -ésima laplaciana da mistura,  $\alpha_i > 0$  é o fator de ponderação da  $i$ -ésima mistura e  $\sum_{i=1}^N \alpha_i = 1$ . O método proposto em [31],[32] modela a distribuição dos ângulos de  $\mathbf{x}(n)$  como uma mistura de laplacianas, estimando os parâmetros da mistura por meio de um algoritmo iterativo do tipo EM (*Expectation Maximization*). Supondo que temos  $K$  vetores  $\mathbf{x}(n)$ , as equações de atualização (vide demonstrações em [31]) dos parâmetros são:

$$p(i|\theta_n) = \frac{\alpha_i \mathcal{L}(\theta_n, m_i, c_i)}{\sum_{i=1}^N \alpha_i \mathcal{L}(\theta_n, m_i, c_i)}, \quad (8)$$

$$\alpha_i^+ \leftarrow \frac{1}{K} \sum_{n=1}^K p(i|\theta_n), \quad (9)$$

$$m_i^+ \leftarrow \frac{\sum_{n=1}^K \frac{\theta_n}{|\theta_n - m_i|} p(i|\theta_n)}{\sum_{n=1}^K \frac{1}{|\theta_n - m_i|} p(i|\theta_n)}, \quad (10)$$

$$c_i^+ \leftarrow \frac{\sum_{n=1}^K p(i|\theta_n)}{2 \sum_{n=1}^K |\theta_n - m_i| p(i|\theta_n)}, \quad (11)$$

onde  $p(i|\theta_n)$  consiste na probabilidade de a  $i$ -ésima Laplaciana possuir  $\theta_n$ . Após a convergência das iterações, os parâmetros  $m_i$  representam os ângulos preferenciais a estimar. A inicialização dos parâmetros  $m_i$  pode ser feita via  $K$ -médias [33], e o processo iterativo pode apresentar melhores resultados caso estabeleçamos limites superiores para  $\alpha_i$  e/ou  $c_i$ . Cumpre enfatizar que o algoritmo pode padecer de problemas de contorno [31]. A evolução das estimativas de  $m_i$  num caso particular de 2 misturas e 3 fontes pode ser vista na Figura 6. Para gerar esta figura, foram empregados três sinais de voz (duas femininas e uma masculina, com durações de 10 s em 16 kHz), decomposição via *wavelet packet* de 7 níveis com a *wavelet* bior6.8. Foi arbitrado um limite superior para  $c$  igual a 3.

O método EM empregado utiliza uma estimativa de máxima verossimilhança, o que significa que as estimativas  $m_i$  são

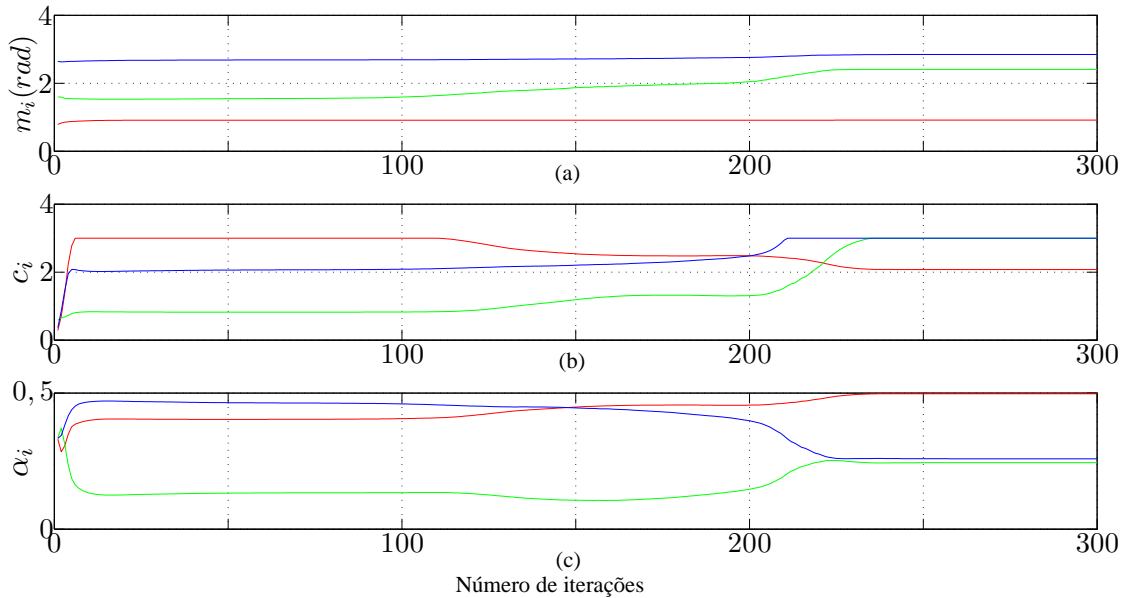


Figura 6: Evolução dos parâmetros estimados via modelos de misturas laplacianas. Para  $i = 1, 2$  e  $3$  temos, respectivamente, as cores vermelha, verde e azul.

as medianas de cada *cluster*. Embora [32] tente minimizar o efeito supostamente danoso desta escolha, em [34] argumenta-se que ela é superior à escolha da média. O artigo [35] oferece uma alternativa de misturas de laplacianas, utilizando modelos de misturas chamados de exponenciais generalizados.

### 4.3. EXTENSÃO PARA UM MAIOR NÚMERO DE MISTURAS - ALGORITMO SCAM

A maioria dos métodos que estimam a matriz de mistura não possuem uma trivial extensão para configurações nas quais  $M > N$ . Em [36], o algoritmo SCAM (SCA modificado) foi proposto, o qual pode empregar qualquer método para  $M = 2$  de forma a estimar matrizes que tenham mais do que 2 linhas. O algoritmo SCAM pode ser condensado nos seguintes passos:

- i. Estimar todos os coeficientes de cada linha da matriz de mistura (via varredura de Zibulevsky ou outro método equivalente);
- ii. Para cada linha obter todas as permutações possíveis para os coeficientes estimados;
- iii. Gerar todas as combinações entre as linhas (candidatas) distintas;
- iv. Gerar todas as matrizes candidatas, combinando os Passos (i) e (iii);
- v. Calcular as inversas de todas as matrizes de mistura candidatas, gerando as matrizes de separação candidatas;
- vi. Determinar as estimativas para cada matriz de separação candidata;
- vii. Determinar a matriz ótima maximizando alguma medida de independência das estimativas<sup>4</sup>.

### 4.4. OUTROS MÉTODOS

Outras técnicas de estimativa da matriz de mistura residem num agrupamento (*clustering*) dos ângulos, de forma que a média ou a mediana dos *clusters* encontrados estimem as direções preferenciais. Não é o objetivo deste tutorial detalhar algoritmos de clusterização. Portanto, citamos as principais abordagens da literatura: clusterização  $K$ -médias [37], *Fuzzy* [38] e hierárquica [22]. Alguns algoritmos de clusterização serão empregados em nossas simulações.

Existem outras técnicas de estimativa da matriz de mistura, como os métodos DUET [39],[40] (o qual apresenta a vantagem de trabalhar com misturas anecóicas, onde são permitidos atrasos entre sensores, embora não contemple a reverberação), TIFROM [41],[42],[43] e DUET/TIFROM estendido [44]. Estes métodos têm semelhanças importantes com a varredura de Zibulevsky, ainda que o método DUET exija a STFT como a transformada de esparsificação. Outros métodos recorrem a técnicas de treinamento não-supervisionado de mapas de Kohonen [45], seja de inspiração geométrica [46],[34], seja de motivação algébrica [47]. Estes métodos com frequência apresentam desempenhos inferiores aos de outros métodos, sendo também muito sensíveis à inicialização. Outra técnica importante, que utiliza a hipótese de que as distribuições no tempo e na frequência das fontes não se sobrepõem, é proposta em [48].

## 5. RECONSTRUÇÃO DAS FONTES: CAMINHO MAIS CURTO

A questão fundamental nas configurações de misturas subdeterminadas consiste na indeterminação das soluções já que, dadas  $\mathbf{H}$  não-singular e  $\mathbf{x}(n)$ , existem infinitas soluções  $\mathbf{s}(n)$  que satisfazem a restrição  $\mathbf{x}(n) = \mathbf{H}\mathbf{s}(n)$  (sob algumas restrições da pdf das fontes, este problema pode admitir mais de uma solução ou um número finito de soluções [49]).

Os métodos DUET e TIFROM (bem como seus derivados) utilizam um mascaramento binário das misturas no espaço tempo-frequência para obter estimativas das fontes, o que resulta em muitas discontinuidades na representação tempo-frequência dos sinais extraídos, produzindo ruído musical e distorção [50]. Este problema é maior nas situações onde regiões de alta energia que pertencem a diferentes fontes se sobrepõem. Podemos dizer que o gargalo nas técnicas de SCA reside na etapa de reconstrução das fontes, cujos resultados ainda revelam muitas distorções e artefatos.

O método mais empregado na reconstrução das fontes é o chamado de Caminho mais Curto (em inglês, *Shortest Path*), o qual será detalhado a seguir. Admitindo que as amostras de cada fonte são regidas por uma distribuição exponencial, a densidade de probabilidade conjunta da  $i$ -ésima fonte pode ser expressa como:

$$p(s_i(1), \dots, s_i(K)) = \prod_{n=1}^K c e^{-2c|s_i(n)|} = c^K e^{-2\sum_{n=1}^K |s_i(n)|}, \quad (12)$$

de onde decorre, em conjunto com a hipótese de fontes estatisticamente independentes, que a solução de máxima probabilidade é a estimativa das fontes cuja soma dos módulos é a menor. O método do caminho mais curto portanto estima, a cada instante  $n$ , o vetor  $\mathbf{s}(n)$  que possua norma  $l_1$  mínima, sujeito à restrição  $\mathbf{x}(n) = \mathbf{H}\mathbf{s}(n)$ . Pode-se provar que a solução deste problema apresenta  $N - M$  zeros [22],[51],[52].

Quando supomos que  $N - M$  fontes são nulas no instante  $n$ , o número de incógnitas torna-se igual a  $M$  e o sistema passa a ser determinado. Porém, ignoramos *a priori* quais são as  $N-M$  fontes que devem ser consideradas para que tenhamos a solução de máxima verossimilhança. O critério para determiná-la consiste em resolver os sistemas oriundos de todas as combinações possíveis de  $N - M$  fontes nulas. Temos então diversas estimativas de  $M$  fontes supostas não-nulas. A estimativa que apresenta a menor soma dos módulos será então a preferida.

Vamos a um exemplo de um sistema  $2 \times 3$ . Seja uma matriz de mistura (estimada) igual a:

$$\hat{\mathbf{H}} = \begin{bmatrix} 1 & 1 & 1 \\ 0,2 & 1,1 & -2 \end{bmatrix}. \quad (13)$$

<sup>4</sup>Em nossas simulações, a medida de correlação (mais fraca estatisticamente que a de independência) já foi suficiente.



Primeiramente, cumpre impor, de modo a não penalizar algumas soluções em detrimento de outras, uma norma unitária a todos os vetores correspondentes à cada uma das direções. Para tal, basta normalizar cada coluna da matriz de mistura. Devido à ambiguidade de escalamento, esta operação não é problemática. Daí obtemos:

$$\widehat{\mathbf{H}}_n = \begin{bmatrix} 0,98058 & 0,6726 & 0,44721 \\ 0,19612 & 0,73994 & -0,89443 \end{bmatrix}. \quad (14)$$

Suponhamos que num instante  $n$  tenha-se:

$$\mathbf{x}(n) = \begin{bmatrix} 0,91 \\ -0,356 \end{bmatrix}. \quad (15)$$

Supondo que apenas 2 fontes (considerando aqui  $M = 2$ ) contribuam para o vetor de misturas acima, efetuamos as seguintes estimativas:

$$\text{I: } \begin{bmatrix} \hat{s}_1(n) \\ \hat{s}_2(n) \end{bmatrix} = \begin{bmatrix} 0,9806 & 0,6726 \\ 0,1961 & 0,7399 \end{bmatrix}^{-1} \begin{bmatrix} 0,91 \\ -0,356 \end{bmatrix} = \begin{bmatrix} 1,5376 \\ -0,8887 \end{bmatrix},$$

$$\hat{s}_3(n) = 0 \text{ e } \sum_{i=1}^3 |\hat{s}_i(n)| = 2,4263$$

$$\text{II: } \begin{bmatrix} \hat{s}_1(n) \\ \hat{s}_3(n) \end{bmatrix} = \begin{bmatrix} 0,9806 & 0,4472 \\ 0,1961 & -0,8944 \end{bmatrix}^{-1} \begin{bmatrix} 0,91 \\ -0,356 \end{bmatrix} = \begin{bmatrix} 0,6789 \\ 0,5468 \end{bmatrix},$$

$$\hat{s}_2(n) = 0 \text{ e } \sum_{i=1}^3 |\hat{s}_i(n)| = 1,2255$$

$$\text{III: } \begin{bmatrix} \hat{s}_2(n) \\ \hat{s}_3(n) \end{bmatrix} = \begin{bmatrix} 0,6726 & 0,4472 \\ 0,7399 & -0,8944 \end{bmatrix}^{-1} \begin{bmatrix} 0,91 \\ -0,356 \end{bmatrix} = \begin{bmatrix} 0,702 \\ 0,9788 \end{bmatrix},$$

$$\hat{s}_1(n) = 0 \text{ e } \sum_{i=1}^3 |\hat{s}_i(n)| = 1,6809$$

Das 3 possibilidades, vemos que a segunda apresenta a menor soma dos módulos, devendo portanto ser a preferida. Cabe observar que as matrizes cujas inversas importa calcular só dependem da matriz de mistura estimada (e normalizada); logo, não é necessário recalculá-las para cada valor de  $n$ . Quando temos  $M = N$ , não precisamos supor nenhuma fonte como nula e então recaímos no procedimento padrão da ICA (pelo menos na etapa de reconstrução).

Uma possibilidade para a reconstrução consiste em utilizar, da solução escolhida, apenas a estimativa de maior módulo como não-nula (método chamado de *hard*, em contraposição ao método padrão intitulado *soft*). Assim, no exemplo acima, caso empregássemos o método *hard*, teríamos apenas a primeira fonte estimada como 0,6789 e todas as outras seriam estimadas como nulas.

## 6. EXPERIMENTOS

### 6.1. ESTIMATIVA DA MATRIZ DE MISTURA

A partir de três sinais de voz (duas femininas e uma masculina), amostrados em 16 kHz e com duração de 10 segundos, efetuamos misturas com 3 diferentes matrizes, mostradas a seguir:

$$\mathbf{H}_1 = \begin{bmatrix} 1 & 1 & 1 \\ -0,9 & -0,5 & 1,3 \end{bmatrix}$$

$$\mathbf{H}_2 = \begin{bmatrix} 1 & 1 & 1 \\ -0,9 & 0,5 & 1,3 \end{bmatrix}$$

$$\mathbf{H}_3 = \begin{bmatrix} 1 & 1 & 1 \\ -0,9 & 0,8 & 1,3 \end{bmatrix}$$

As dimensões das matrizes implicam a existência de apenas 2 misturas. O teste a ser efetuado foi inspirado na avaliação efetuada em [30]. Podemos expressar cada matriz de misturas como:

$$\mathbf{H}_i = \begin{bmatrix} 1 & 1 & 1 \\ \tan(\psi_i^{(1)}) & \tan(\psi_i^{(2)}) & \tan(\psi_i^{(3)}) \end{bmatrix},$$

com  $\tan(\psi_i^{(j)})$  sendo calculado pela Equação (4). A Figura 7 permite-nos visualizar graficamente as direções de cada coluna das matrizes  $\mathbf{H}_1$  (caso 1),  $\mathbf{H}_2$  (caso 2) e  $\mathbf{H}_3$  (caso 3).

Após efetuarmos as misturas, decompomo-las utilizando uma transformada do tipo *wavelet packet* com 7 níveis. A *wavelet*

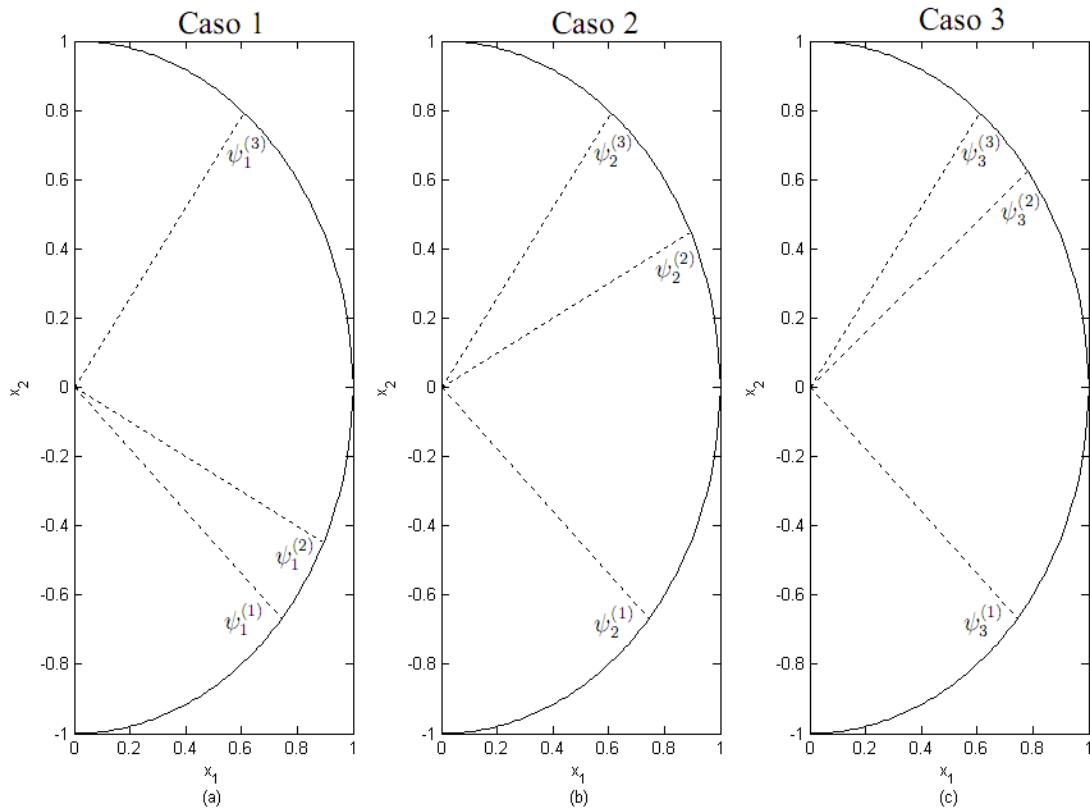


Figura 7: Visualização dos ângulos das matrizes de mistura. (a) matriz  $\mathbf{H}_1$ ; (b) matriz  $\mathbf{H}_2$  e (c) matriz  $\mathbf{H}_3$ .

utilizada foi a *Daubechies* 44. Os métodos que utilizamos para a estimativa da matriz de mistura foram a varredura de Zibulevsky, o modelo de misturas de laplacianas, as clusterizações hierárquica, de  $K$ -médias e *Fuzzy* (tanto com a média quanto com a mediana dos *clusters* para estas duas últimas) e a extensão dos métodos DUET e TIFROM.

Alguns dos métodos necessitam que arbitremos seus parâmetros. Escolhemo-los de forma a maximizar o desempenho de cada algoritmo. A Tabela 1 detalha as estimativas da segunda linha da matriz de mistura obtidas por cada método. Nesta tabela vemos que as clusterizações de  $K$ -médias e *Fuzzy* por vezes apresentam estimativas muito erráticas (assinaladas em negrito). Os outros métodos tendem a ser mais robustos a este problema.

A Tabela 2 apresenta os erros quadráticos médios dos coeficientes estimados em relação aos ótimos, não sendo apresentados os resultados dos casos em que algum método falhou de forma desastrosa (os métodos que nunca falharam de forma patente foram a varredura de Zibulevsky, a clusterização hierárquica e o DUET-TIFROM (estendido)).

## 6.2. TÉCNICAS DE RECONSTRUÇÃO

Na prática, as técnicas de reconstrução ocorrem após a etapa de estimativa da matriz de mistura. Nesta subseção, já que estamos primordialmente interessados nas técnicas de reconstrução, suporemos que a matriz de mistura é conhecida (ou, equivalentemente, que a estimativa da matriz de mistura teve excelente qualidade).

Uma das dificuldades das técnicas de reconstrução reside na relação entre o número  $N$  de fontes e o número  $M$  de misturas. Basicamente, o aumento de  $M$  facilita a reconstrução, já que aumenta a quantidade de informação disponível para a reconstrução. Já o acréscimo de  $N$  é um complicador da robustez da reconstrução.

Por isso, idealizamos um experimento com fontes de áudio e de voz. O número de fontes varia de 3 a 10. O número de misturas varia de 2 a  $N - 1$  (de modo a garantir a existência de diversidade espacial e que a configuração é subdeterminada). Para avaliar a reconstrução, utilizamos 3 métricas: SAR (razão sinal-artefato), SIR (razão sinal-interferência) e SDR (razão sinal-distorção), as quais, quando elevadas, expressam uma boa separação. Estas medidas são apresentadas na Fig. 8. Todas as medidas revelam a mesma tendência de apresentar melhor desempenho com o acréscimo de  $M$  (mantendo  $N$  fixo) ou com a redução de  $N$  (mantendo  $M$  fixo), conforme seria de se esperar.

Observamos que a SIR sempre encontra-se mais alta que as medidas de SDR e SAR, porque em misturas subdeterminadas a etapa de reconstrução insere artefatos e distorções significativas nas estimativas. Nos casos determinados (não contemplados na Figura 7) o fenômeno que ocorre em geral é inverso, com a SAR e a SDR em geral maiores do que a SIR.

Caso 1			
Valores reais	-0,9	-0,5	1,3
Varredura de Zibulevsky	-0,8935	-0,5063	1,3028
Mistura de Exponenciais	-0,7932	<b>-14,0721</b>	1,2843
Clusterização <i>K</i> -Médias (média dos <i>clusters</i> )	-0,7285	<b>-15,3975</b>	1,1843
Clusterização <i>K</i> -Médias (mediana dos <i>clusters</i> )	-0,8226	<b>-23,2922</b>	1,2966
Clusterização <i>Fuzzy</i> (média dos <i>clusters</i> )	-0,7396	<b>7,7369</b>	1,1951
Clusterização <i>Fuzzy</i> (mediana dos <i>clusters</i> )	-0,8306	<b>8,5181</b>	1,2942
Clusterização Hierárquica	-0,8995	-0,5062	1,2894
DUET-TIFROM (estendido)	-0,9250	-0,6092	1,2890
Caso 2			
Valores reais	-0,9	0,5	1,3
Varredura de Zibulevsky	-0,8966	0,5062	1,3027
Mistura de Exponenciais	-0,8961	0,4963	1,2883
Clusterização <i>K</i> -Médias (média dos <i>clusters</i> )	-0,8174	<b>-70,2300</b>	1,0931
Clusterização <i>K</i> -Médias (mediana dos <i>clusters</i> )	-0,8782	<b>72,2792</b>	1,2818
Clusterização <i>Fuzzy</i> (média dos <i>clusters</i> )	-0,8695	0,5290	1,3470
Clusterização <i>Fuzzy</i> (mediana dos <i>clusters</i> )	-0,8959	0,5213	1,3052
Clusterização Hierárquica	-0,8950	0,5096	1,2853
DUET-TIFROM (estendido)	-0,8215	0,9576	1,3000
Caso 3			
Valores reais	-0,9	0,8	1,3
Varredura de Zibulevsky	-0,8993	0,8103	1,3037
Mistura de Exponenciais	-0,9033	0,8008	1,2724
Clusterização <i>K</i> -Médias (média dos <i>clusters</i> )	-0,8116	<b>-17,4470</b>	1,1500
Clusterização <i>K</i> -Médias (mediana dos <i>clusters</i> )	-0,8795	<b>-24,8988</b>	1,2827
Clusterização <i>Fuzzy</i> (média dos <i>clusters</i> )	-0,8661	0,7730	1,3610
Clusterização <i>Fuzzy</i> (mediana dos <i>clusters</i> )	-0,8840	<b>11,4023</b>	1,2814
Clusterização Hierárquica	-0,9321	0,8142	1,2975
DUET-TIFROM (estendido)	-0,7939	0,9330	1,3306

Tabela 1: Estimativas por diferentes métodos da segunda linha das matrizes de mistura, considerando os 3 casos mostrados na Fig. 7.

Caso 1	
Varredura de Zibulevsky	$2,9927 \cdot 10^{-5}$
Clusterização Hierárquica	$5,0675 \cdot 10^{-5}$
DUET-TIFROM (est)	$4,2 \cdot 10^{-3}$
Caso 2	
Varredura de Zibulevsky	$1,9097 \cdot 10^{-5}$
Mistura de Laplacianas	$5,5263 \cdot 10^{-5}$
Clusterização <i>Fuzzy</i> (média)	$1,3 \cdot 10^{-3}$
Clusterização <i>Fuzzy</i> (mediana)	$1,6585 \cdot 10^{-4}$
Clusterização Hierárquica	$1,1111 \cdot 10^{-4}$
DUET-TIFROM (estendido)	$7,19 \cdot 10^{-2}$
Caso 3	
Varredura de Zibulevsky	$4,009 \cdot 10^{-5}$
Mistura de Laplacianas	$2,3970 \cdot 10^{-4}$
Clusterização <i>Fuzzy</i> (média)	$1,9 \cdot 10^{-3}$
Clusterização Hierárquica	$4,1221 \cdot 10^{-4}$
DUET-TIFROM (estendido)	$1 \cdot 10^{-2}$

Tabela 2: Erros quadráticos médios obtidos pelos diferentes métodos apresentados na Tabela 1.

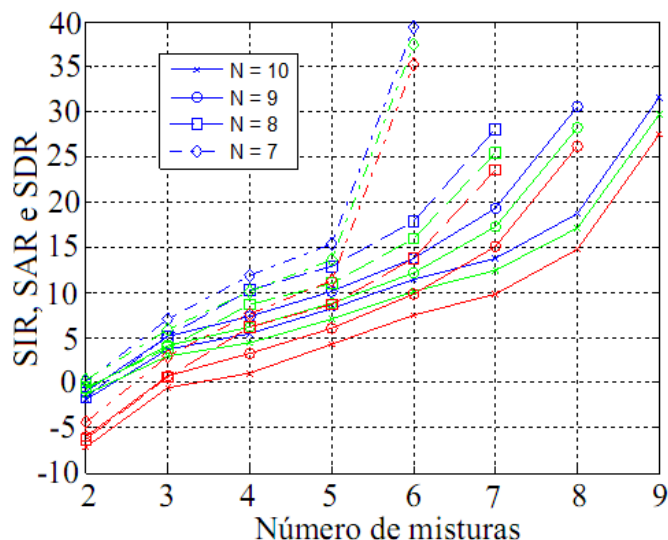


Figura 8: SIR (azul), SAR (verde) e SDR (vermelho) em função do número de misturas e de fontes.

### 6.3. APLICAÇÃO DE SCA EM UM SISTEMA DE RECONHECIMENTO AUTOMÁTICO DE INSTRUMENTOS MUSICAIS

Neste experimento utilizamos a separação cega de fontes como uma etapa de pré-processamento em um sistema de reconhecimento de instrumentos musicais. O procedimento elaborado para alcançar este objetivo segue as etapas:

- i. separação das  $M$  misturas polifônicas em  $N$  sequências monofônicas;
- ii. extração das notas de cada sequência monofônica;
- iii. classificação das notas de cada sequência monofônica através de um classificador previamente elaborado.

Na primeira etapa comparamos três algoritmos de BSS (FastICA [53], Infomax [2] e SCAM), que são aplicadas a casos de misturas polifônicas determinadas. Na etapa 2 um extrator de notas usa a média e o desvio padrão de uma janela de modo a determinar o início e o fim de cada nota [54]. Em seguida as notas são classificadas de acordo com os instrumentos de origem. Todas as notas são provenientes da base de dados RWC (Real World Computing [55]).

As sequências monofônicas (fontes originais) foram construídas por meio de notas do conjunto de teste (obtido a partir de um percentual sorteado da região central da tessitura do instrumento musical). Essas notas são sorteadas (formando um subconjunto menor) e separadas por intervalos (*gaps*) aleatórios, que variam entre 0,045 ms e 0,3 ms. Após a formação das sequências monofônicas, estas são misturadas por uma matriz de mistura de dimensões  $3 \times 3$  aleatória, gerando os sinais polifônicos (misturas instantâneas).

Foram utilizadas 10 matrizes de mistura distintas para gerar as misturas utilizadas neste experimento. Cada grupo de três sinais misturados foi formado por 3 conjuntos de 3 sequências de notas monofônicas. O Grupo A foi formado por sequências monofônicas contendo notas de instrumentos de palheta (oboé, saxofone barítono e saxofone contralto), o grupo B por instrumentos de percussão (xilofone, glockenspiel e vibrafone) e o grupo C por instrumentos de metal (trompa, trombone e trompete).

#### 6.3.1 CONSIDERAÇÕES PRELIMINARES

Devido ao problema da permutação, inerente às técnicas de separação cega (ver Seção 2), as estimativas obtidas podem não estar na ordem original. Sendo assim, logo após obtermos as estimativas através do separador cumpre ainda identificar qual fonte está associada a uma dada estimativa. Isso nem sempre representa um processo de fácil identificação porque em vários casos as estimativas obtidas possuem contaminação das demais fontes. Para resolver esse problema usamos uma métrica de comparação da estimativa com a fonte original: a razão sinal-distorção (SDR), a qual fornece uma medida de similaridade de cada estimativa em relação a cada uma das fontes.

Existe a possibilidade, quando o processo de separação não obtém um bom desempenho, de termos mais de uma estimativa (ou mesmo todas) associadas a uma mesma fonte predominantemente (caso 1), ou ainda apresentar-se como a melhor representação para todas as fontes originais (caso 2). Portanto, o uso de métricas para medir o grau de similaridade da estimativa com a fonte original pode não resultar numa boa avaliação do método de separação. Quando a separação ocorre de forma satisfatória haverá uma associação distinta entre as estimativas e as fontes. Para contornarmos essa possível distorção adotou-se

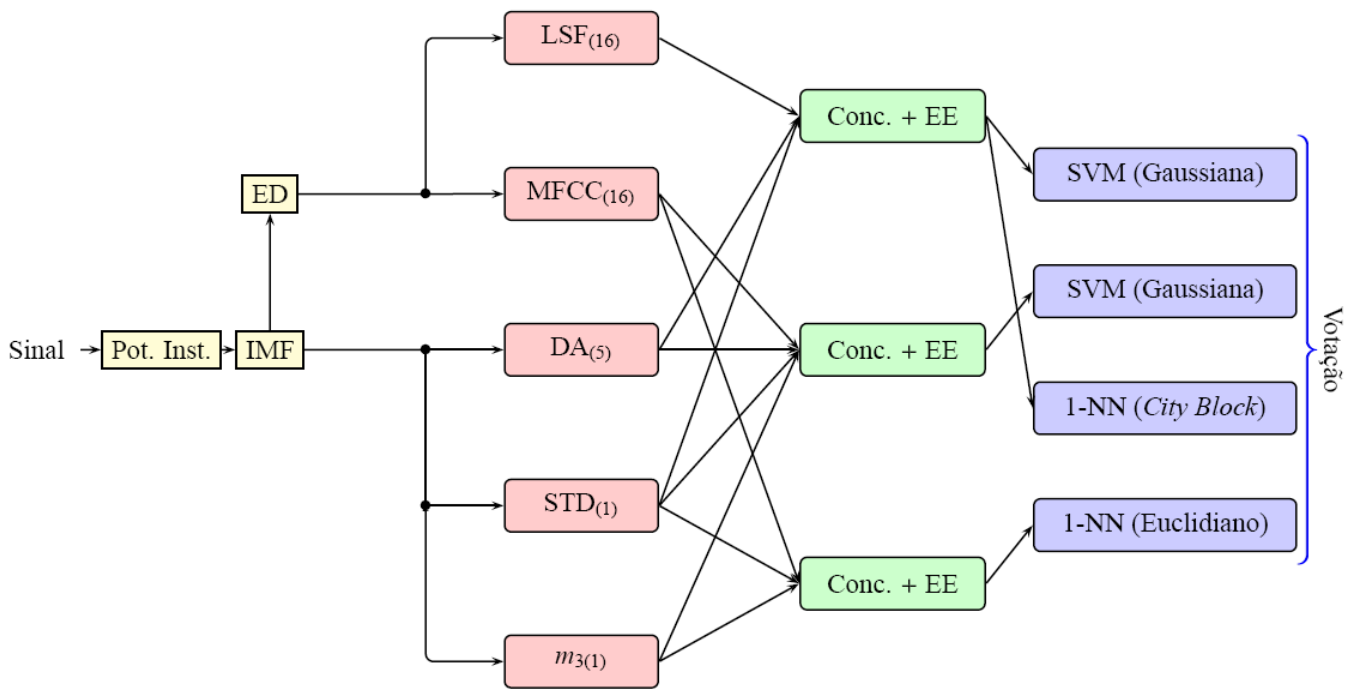


Figura 9: SRAIM empregado.

o seguinte critério para avaliarmos as estimativas da medida de separação (MS) obtida pelos algoritmos:

$$\bar{X} = \frac{1}{N} \sum_{i=1}^N \max \{ \text{SDR}(s_j, \hat{s}_i) \}, \quad (16)$$

$$\bar{Y} = \frac{1}{N} \sum_{j=1}^N \max \{ \text{SDR}(s_j, \hat{s}_i) \}, \quad (17)$$

$$\text{MS} = \min(\bar{X}, \bar{Y}), \quad (18)$$

onde  $s_j$  é a  $j$ -ésima fonte e  $\hat{s}_i$  representa a  $i$ -ésima estimativa. A estimação ótima ( $\bar{X} = \bar{Y}$ ) ocorre quando a separação é considerada satisfatória, ou seja, quando os casos 1 e 2 não ocorrem.

### 6.3.2 SISTEMA DE RECONHECIMENTO AUTOMÁTICO DE INSTRUMENTOS MUSICAIS (SRAIM)

O SRAIM empregado recorre a um banco de classificadores [54], conforme mostra a Figura 9. O desempenho do banco de classificadores supera o de qualquer um dos classificadores, caso empregados isoladamente. Este SRAIM foi projetado para o reconhecimento de notas isoladas, a partir de um segmento das mesmas.

Conforme pode ser visto na Figura 9, os três primeiros blocos (Pot. Inst., IMF e ED) condensam parte do pré-processamento aplicado ao sinal (nota de um instrumento musical). O bloco Pot. Inst. obtém a potência instantânea do sinal, através de um limiar definido como 90% da média da potência instantânea da nota musical. A segmentação da nota é efetuada pelo bloco IMF (Início, Meio e Fim), proposto em [54], o qual estabelece limiares dependentes da potência da nota de modo a estimar seus instantes de início e de fim. Após a detecção destes instantes, as medidas de desvio padrão (STD) e momento central de terceira ordem ( $m_3$ ) são extraídas, bem como cinco descritores de áudio (DA): taxa de cruzamento por zero, fluxo espectral, valor RMS do *frame*, centroide espectral e largura do centroide espectral. Por fim, o segmento extraído passa por um processo de escalamento dinâmico (bloco ED), visando sua normalização. A partir desse momento são extraídos os vetores de características da nota que são formados basicamente por dois tipos de codificadores: coeficientes LSF (do inglês *Line Spectral Frequencies*) [57] e MFCC (do inglês *Mel Frequencies Cepstral Coefficients* [58]-[60]). Ambos os codificadores utilizam 16 coeficientes representativos. Essas medidas são concatenadas<sup>5</sup> de 3 diferentes formas, resultando em 3 vetores de características distintos os quais, após passar por um processo de escalamento estatístico (EE), servem como entrada para 4 classificadores. Foram utilizados como classificadores 2 SVMs (do inglês *Support Vector Machines*) [61]-[63] e 2 K-NN (do inglês *K-nearest neighbor* [64]). Ao final a classe predita representa a classe mais votada entre os quatro classificadores (caso haja empate, há um sorteio entre as

<sup>5</sup>Dá o termo “Conc.” na Figura 9.

Algoritmo	Percussão	Palhetas	Metais	Total
<b>FastICA</b>	27,5	25,3	28,4	27,0
<b>Infomax</b>	38,6	35,0	38,4	37,3
<b>SCAm</b>	40,8	39,7	41,6	40,7

Tabela 3: SDR média das estimativas dos instrumentos musicais segundo o algoritmo de separação empregado.

Algoritmo	Percussão	Palhetas	Metais	Total
<b>FastICA</b>	87,3	90,2	83,0	86,6
<b>Infomax</b>	89,3	80,4	81,4	83,7
<b>SCAm</b>	91,7	92,2	84,8	89,5
<b>Fontes</b>	100,0	96,1	85,0	93,7

Tabela 4: Taxa de acerto (em porcentagem) do SRAIM no tocante às notas individuais de cada estimativa. A linha **Fontes** refere-se ao uso das próprias fontes (sem mistura) para o reconhecimento, resultando no limitante superior de desempenho do SRAIM.

predições). O SRAIM foi treinado com um subconjunto de notas<sup>6</sup> oriundo de três bases de dados distintas: RWC [55], MIS<sup>7</sup> [65] e MUMS<sup>8</sup> [66]. O sistema adotado é capaz de classificar 20 instrumentos distintos. O conjunto usado na fase de treinamento não possui notas em comum com o conjunto de notas usado para formar as sequências monofônicas de teste (10% das notas oriundas da base de dados RWC).

### 6.3.3 Separação de Fontes e SRAIM

Neste experimento foram testados os algoritmos Infomax, FastICA e SCAm em configurações determinadas (nas quais o número de fontes e de misturas é igual a três). Para a estimativa dos ângulos necessária à técnica SCAm foi efetuado um simples histograma dos ângulos dos vetores  $(x_1(n), x_i(n))$ , para  $i = 2, 3$ . Este histograma dividiu uniformemente o intervalo  $[0, \pi]$  em 36000 subintervalos. Os picos deste histograma são utilizados para estimar os ângulos referentes aos coeficientes da segunda e terceira linhas da matriz de mistura.

A Tabela 3 apresenta a SDR média das estimativas dos instrumentos musicais. A SDR é uma medida interessante por não ser afetada apenas por interferências, mas também por artefatos.

Já a Tabela 4 apresenta a taxa de acerto (em porcentagem) dos algoritmos cuja comparação é o nosso propósito, atentando-se para as notas individuais extraídas. O algoritmo SCAm obteve os melhores em todos os casos testados.

A Tabela 5 permite-nos investigar a taxa de acerto global (ou seja, o instrumento mais votado ao longo de cada estimativa), que é sempre superior à taxa de acerto das notas individuais. Apenas a técnica SCAm conseguiu obter a taxa de acerto de 100%. Estes resultados confirmam a competitividade das técnicas SCA, inclusive em contextos determinados, mostrando que o algoritmo SCAm obteve rendimento superior aos algoritmos derivados da ICA.

## 7. CONCLUSÕES

Este tutorial abordou de forma ampla as técnicas de SCA, num contexto de misturas instantâneas. Ressaltamos que os métodos empregados em misturas convolutivas não diferem muito dos aqui apresentados. Embora menos famosa do que a ICA, a SCA permite-nos estimar as fontes mesmo quando o número de fontes supera o de misturas, contemplando, portanto, um maior leque de contextos.

Dentre as técnicas de SCA, enfocamos as que estimam de forma independente a matriz de mistura e a reconstrução das fontes, pois estas são as técnicas que possuem menor custo computacional sem, todavia, comprometer a qualidade das estimativas. O

<sup>6</sup>90% das notas oriundas das 3 bases de dados empregadas.

<sup>7</sup>Do inglês *Musical Instruments Samples*.

<sup>8</sup>*McGill University Master Samples*.

Algoritmo	Percussão	Palhetas	Metais	Total
<b>FastICA</b>	96,7	100,0	100,0	98,9
<b>Infomax</b>	100,0	93,3	96,7	96,7
<b>SCAm</b>	100,0	100,0	100,0	100,0
<b>Fontes</b>	100,0	100,0	100,0	100,0

Tabela 5: Taxa de acerto (em porcentagem) dos instrumentos de cada estimativa.

conceito de esparsidade foi realçado, bem como evidenciada sua importância nas etapas principais da SCA. Destas etapas, a da reconstrução das fontes (em configurações subdeterminadas) consiste no gargalo da SCA, já que suas estimativas padecem de distorções e artefatos mesmo quando a matriz de mistura exata é utilizada. Importa enfatizar a incapacidade das técnicas-padrão de ICA de trabalhar em contextos subdeterminados.

Por fim, conclui-se que o uso de técnicas de separação cega na identificação de instrumentos musicais em misturas instantâneas revelou-se viável. O desempenho das técnicas de SCA em contextos determinados revelou-se bastante satisfatório, superando o obtido pelos algoritmos baseados na ICA.

## AGRADECIMENTOS

Os autores gostariam de agradecer o parcial apoio financeiro da FAPERJ e do CNPq.

## REFERÊNCIAS

- [1] X.-R. Cao e R.-W. Liu. “General Approach to Blind Source Separation,” *IEEE Transactions on Signal Processing*, vol. 44, no. 3, pp. 562-571, 1996.
- [2] P. Comon, “Independent Component Analysis, a new concept?,” *Signal Processing*, vol. 36, no. 3, pp. 287-314, 1994.
- [3] A. J. Bell e T. J. Sejnowski, “An information-maximization approach to blind separation and blind deconvolution,” *Neural Computation*, vol. 7, no. 6, pp. 1004-1034, 1995.
- [4] J.-F. Cardoso e B. Laheld, “Equivariant adaptive source separation,” *IEEE Transactions on Signal Processing*, vol. 44, no. 12, pp. 3017-3030, 1996.
- [5] N. Gadhok e W. Kinsner, “An Implementation of  $\beta$ -Divergence for Blind Source Separation,” *IEEE CCECE/CCGEL*, pp. 1446-1449, 2006.
- [6] C. C. Cheung e L. Xu, “Some Global and Local Convergence Analysis on the Information-Theoretic Independent Component Analysis Approach,” *Neurocomputing*, vol. 30, pp. 79-102, 2000.
- [7] A. Cichocki, R. Zdunek, A. H. Phan, S.-I. Amari, “Nonnegative Matrix and Tensor Factorizations,” Wiley, 1st Edition, 2009.
- [8] A. Hyvarinen, J. Karhunen, e E. Oja, *Independent Component Analysis*. John Wiley and Sons, 2001.
- [9] M. S. Naceur, M. A. Loghmani e M. R. Boussema, “The Contribution of the Sources Separation Method in the Decomposition of Mixed Pixels,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 42, no. 11, pp. 2642-2653, 2004.
- [10] J. Anemuller, T. Sejnowski e S. Makeig, “Complex Independent Component Analysis of Frequency-Domain Electroencephalographic Data,” *Neural Networks*, vol. 16, pp. 1311-1323, 2003.
- [11] F. Acernese, A. Ciaramella, S. De Martino, R. De Rosa, M. Falanga and R. Tagliaferri, “Neural Networks for Blind Source Separation of Stromboli Explosion Quakes,” *IEEE Transactions on Neural Networks*, vol. 14, no. 1, pp. 167-175, 2003.
- [12] A. T. Erdogan, “Globally Convergent Deflationary Instantaneous Blind Source Separation Algorithm for Digital Communication Signals,” *IEEE Transactions on Signal Processing*, vol. 55, no. 5, pp. 2182-2192, 2007.
- [13] H. Buchner, R. Aichner e W. Kellerman, “A Generalization of Blind Source Separation Algorithms for Convolutional Mixtures Based on Second-Order Statistics,” *IEEE Transactions on Speech and Audio Processing*, vol. 13, no. 1, pp. 120-134, 2005.
- [14] T. Kim, H. T. Attias, S. Lee e T. Lee, “Blind Source Separation Exploiting Higher-Order Frequency Dependencies,” *IEEE Transactions on Audio, Speech and Language Processing*, vol. 15, no. 1, pp. 70-78, 2007.
- [15] I. T. Jolliffe, *Principal Component Analysis*. Springer, 2a ed., 2002.
- [16] R. Gribonval e S. Lesage, “A Survey of Sparse Component Analysis for Blind Source Separation: Principles, Perspectives, and New Challenges,” *ESANN Proceedings*, pp. 323-330, 2006.
- [17] L. Benaroya, F. Bimbot e R. Gribonval, “Audio Source Separation with a Single Sensor,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 1, pp. 191-199, 2006.
- [18] H. Sameti, H. Sheikhzadeh, L. Deng e R. L. Brennan, “HMM Strategies for Enhancement of Speech Signals Embedded in Nonstationary Noise,” *IEEE Transactions on Speech and Audio Processing*, vol. 6, no. 5, pp. 445-455, 1998.
- [19] T. Virtanen, “Monaural Sound Source Separation by Nonnegative Matrix Factorization with Temporal Continuity and Sparseness Criteria,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 1066-1074, 2007.
- [20] J. Rouat, “Source Separation with One Ear: Proposition for an Anthropomorphic Approach,” *EURASIP Journal on Applied Signal Processing*, pp. 1365-1373, 2005.
- [21] P. Kisilev, M. Zibulevsky e Y. Y. Zeevi, “Multiscale Framework for Blind Separation of Linearly Mixed Signals,” *Journal of Machine Learning Research*, pp. 1-26, 2003.
- [22] S. Winter, W. Kellerman, H. Sawada e S. Makino, “MAP-Based Underdetermined Blind Source Separation of Convolutional Mixtures by Hierarchical Clustering and  $l_1$ -Norm Minimization,” *EURASIP Journal on Advances in Signal Processing*, vol. 2007, p. 1-12, 2007.
- [23] J.-F. Cardoso, “Blind Signal Separation: Statistical Principles,” *Proceedings of the IEEE*, vol. 86, no. 10, pp. 2009-2025, 1998.
- [24] S. Y. Low, S. Nordholm e R. Togneri, “Convolutional Blind Signal Separation with Post-Processing,” *IEEE Transactions on Speech and Audio Processing*, vol. 12, no. 5, pp. 539-548, 2004.
- [25] P. Bofill e M. Zibulevsky, “Underdetermined Blind Source Separation Using Sparse Representations,” *Signal Processing*, vol. 81, no. 11, pp. 2353-2362, 2001.

- [26] Y. Li, S. Amari, A. Cichocki, D. W. C. Ho e S. Xie, “Underdetermined Blind Source Separation Based on Sparse Representation,” *IEEE Transactions on Signal Processing*, vol. 54, no. 2, pp. 423-437, 2006.
- [27] S. Lesage, S. Krstulovic e R. Gribonval, “Underdetermined Source Separation: Comparison of Two Approaches Based on Sparse Decompositions,” *ICA 2006*, pp. 633-640, 2006.
- [28] F. Wang, H. Li e R. Li, “Applications of Wavelet to Independent Component Analysis,” *Proceedings of the 6th World Congress on Intelligent Control and Automation*, pp. 2930-2934, 2006.
- [29] B. Azzerboni, G. Finocchio, M. Ipsale, F. La Foresta e F.C. Morabito, “A New Approach to Detection of Muscle Activation by Independent Component Analysis and Wavelet Transform,” *Springer-Verlag, Lecture Notes in Computer Science*, vol. 2486, pp. 109-116, 2002.
- [30] C. Févotte e S. J. Godsill, “A Bayesian Approach for Blind Source Separation of Sparse Sources,” *IEEE Transactions on Audio and Speech Processing*, vol. 14, no. 6, pp. 2174-2188, 2006.
- [31] N. Mitianoudis e T. Stathaki, “Batch and Online Underdetermined Source Separation Using Laplacian Mixture Models,” *IEEE Transactions on Audio, Speech & Language Processing*, vol. 15, no. 6, pp. 1818-1832, 2007.
- [32] N. Mitianoudis e T. Stathaki, “Overcomplete Source Separation Using Laplacian Mixture Models,” *IEEE Signal Processing Letters*, vol. 12, no. 4, pp. 277-280, 2005.
- [33] C. M. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press, 1995.
- [34] F. J. Theis, C. G. Puntonet e E. W. Lang, “Median-based Clustering for Underdetermined Blind Signal Processing”, vol. 13, no. 2, pp. 96-99, 2006.
- [35] Z. Shi, H. Tang, W. Liu e Y. Tang, “Blind Source Separation of More Sources than Mixtures Using Generalized Exponential Mixture Models”, *Neurocomputing*, vol. 61, pp. 461-469, 2004.
- [36] J. C. P. Filho, M. R. Petraglia e D. B. Haddad, “Avaliação de Técnicas de Separação Cega de Fontes na Identificação de Instrumentos Musicais”, *CBA*, pp 1-6, 2010.
- [37] Y. Q. Li, A. C. e S.-I. Amari, “Analysis of sparse representation and blind source separation,” *Neural Computing*, vol. 16, pp.1193-1234, 2004.
- [38] P. Bofill e M. Zibulevsky, “Blind Source Separation by Sparse Decomposition,” *Neural Computing*, vol. 13, no. 4, pp. 863-882, 2001.
- [39] A. Jourjine, S. Rickard e O. Yilmaz, “Blind Separation of Disjoint Orthogonal Signals: Demixing N Sources from 2 Mixtures,” *Proc. 2000 IEEE Int. Conf. Acoustics, Speech, Signal Processing (ICASSP)*, vol. 5, pp. 2985-2988, 2000.
- [40] Ö. Yilmaz e S. Rickard, “Blind Separation of Speech Mixtures Via Time-Frequency Masking”, *IEEE Transactions on Signal Processing*, vol. 52, no. 7, pp. 1830-1847, 2004.
- [41] F. Abrard, Y. Deville e P. White, “From Blind Source Separation to Blind Source Cancellation in the Underdetermined Case: A New Approach Based on Time-Frequency Analysis,” *Proc. 3rd. Int. Conf. Independent Component Analysis*, pp. 734-739, 2001.
- [42] F. Abrard e Y. Deville, “Blind Separation of Dependent Sources Using the Time-Frequency Ratio of Mixtures Approach,” *7th. Int. Symp. Signal Processing Applications (ISSPA)*, vol. 2, pp. 81 - 84, 2003.
- [43] F. Abrard e Y. Deville, “A Time-frequency Blind Signal Separation Method Applicable to Underdetermined Mixtures of Dependent Sources,” *Signal Processing*, vol. 85, no. 7, pp. 1389-1403, 2005.
- [44] Y. Li, S. Amari, A. Cichocki, D. W. C. Ho e S. Xie, “Underdetermined Blind Source Separation Based on Sparse Representation,” *IEEE Transactions on Signal Processing*, vol. 54, no. 2, pp. 423-437, 2006.
- [45] S. Haykin, “*Neural Networks - A Comprehensive Foundation*,” Prentice Hall, 2nd Edition, 1999.
- [46] F. J. Theis, E. W. Lang e C. G. Puntonet, “A geometric Algorithm for Overcomplete Linear ICA,” *Neurocomputing*, vol. 56, pp. 381-398, 2004.
- [47] K. Waheed e F. M. Salem, “Algebraic Overcomplete Independent Component Analysis,” *4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA 2003)*, pp. 1077-1083, 2003.
- [48] Y. Luo, W. Wang, J. A. Chambers, S. Lambotharan e I. Proudler, “Exploitation of Source Nonstationarity in Underdetermined Blind Source Separation with Advanced Clustering Techniques,” *IEEE Transactions on Signal Processing*, vol. 54, no. 6, 2006.
- [49] J. Eriksson e V. Koivunen, “Identifiability and Separability of Linear ICA Models Revisited,” in *Proc. Int. Workshop on Independent Component Analysis and Blind Signal Separation (ICA2003)*, pp. 23-27, 2003.
- [50] S. Araki, S. Makino, A. Blin, Mukai R. e H. Sawada, “Underdetermined Blind Separation for Speech in Real Environments with Sparseness and ICA,” in *Proc. ICASSP 2004*, vol. 3, pp. 881-884, 2004.
- [51] F. J. Theis, “*Mathematics in Independent Component Analysis*,” PhD Thesis, Regensburg University, 2002.
- [52] I. Takigawa, M. Kudo e J. Toyama, “Performance Analysis of Minimum 11-norm Solutions for Underdetermined Source Separation,” *IEEE Transactions on Signal Processing*, vol. 52, no. 3, pp. 582-591, 2004.
- [53] A. Hyvärinen, “Fast and robust fixed-point algorithms for independent component analysis,” *IEEE Transactions on Neural Networks*, vol. 10, pp. 626-634, 1999.
- [54] J. C. Pires Filho, *Classificação de Instrumentos Musicais em Configurações Monfônicas e Polifônicas*, Dissertação de Mestrado, COPPE/UFRJ, 2009.
- [55] M. Goto e T. Nishimura, “RWC music database: Music genre database and musical instrument sound database,” via <http://sta.aist.go.jp/m.goto/RWC-MDB/m>, ISMIR, pp. 229-230, 2003.
- [56] E. Vincent e H. Sawada, “Stereo audio source separation evaluation campaign,” via <http://www.irisa.fr/metiss/SASSEC07/dev.zip>,



2010.

- [57] P. Kabal e R. P. Ramachandran, “The computation of line spectral frequencies using chebyshev polynomials,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 6, pp. 1419-1426, 1986.
- [58] P. Mermelstein, “Distance measures for speech recognition, psychological and instrumental,” In *Joint Workshop on Pattern Recognition and Artificial Intelligence*, pp. 374-388, 1976.
- [59] F. Opolko e J. Wapnick, “Mcgill university master samples,” via <http://www.music.mcgill.ca/resources/mums/html/mums.html>, 1987.
- [60] S.B. Davis e P. Mermelstein, “Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, pp. 357 - 366, 1980.
- [61] J. S. Bridle e M. D. Brown, “An experimental automatic word-recognition system,” *JSRU Report*, Vol. 1003, 1974.
- [62] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, vol. 5, pp. 273-297, 1995.
- [63] B. E. Boser, I. M. Guyon e V. N. Vapnik, “A training algorithm for optimal margin classifiers,” *Proceedings of the fifth annual workshop on Computational learning theory*, pp. 144-152, 1992.
- [64] A. Aizerman, E. Braverman e L. Rozonoer, “Theoretical foundations of the potential function method in pattern recognition learning,” *Automation and Remote Control*, vol. 25, pp. 821-837, 1964.
- [65] T. M. Cover e P. Hart, “Nearest neighbor pattern classification,” *IEEE Transactions on Information Theory*, vol. 13, pp. 21-27, 1967.
- [66] L. Fritts, “Musical instruments samples of iowa university, mis,” via <http://theremin.music.uiowa.edu/MIS.html>, 1997.
- [67] I. Daubechies, “Ten lectures on wavelets,” ISBN 0-89871-274-2, 1992.
- [68] T. Kitahara, M. Goto e H. G. Okuno, “Musical Instrument Identification Based on F0-Dependent Multivariate Normal Distribution,” *ICASSP*, vol. 5, pp. 421-424, 2003.