Universidade Federal do Rio de Janeiro

Escola Politécnica

Departamento de Eletrônica e de Computação

Separação de Fontes Sonoras por Análise de Subespaços Independentes

Autor:

Renan Mariano Almeida

Orientador:

Prof. Luiz Wagner Pereira Biscainho, DSc

Examinador:

Prof. Marcello Luiz Rodrigues de Campos, PhD

Examinador:

Eng. Leonardo de Oliveira Nunes, MSc

DEL

Março 2011

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
Escola Politécnica - Departamento de Eletrônica e de Computação
Centro de Tecnologia, bloco H, sala H-217, Cidade Universitária
Rio de Janeiro - RJ CEP 21949-900

DEDICATÓRIA

Dedico este trabalho a Deus, minha família e amigos.

AGRADECIMENTOS

Primeiramente, a Deus.

Aos meus pais e ao meu irmão pela base familiar sólida, que eu acredito ser muito importante para atingir qualquer objetivo na vida, seja ele pessoal ou profissional.

Ao meu orientador, Luiz Wagner Pereira Biscainho, por acreditar em mim, sempre mantendo-me motivado.

Ao aluno de doutorado Alexandre Leizor, pelo excelente suporte prestado ao longo do desenvolvimento deste trabalho.

Ao amigo Carlos Vinícius Caldas Campos, que sempre esteve ao meu lado como um bom amigo e uma boa dupla em inúmeros trabalhos ao longo da faculdade, sempre procurando me prestar o mais rigoroso suporte matemático possível quando solicitado (e quando não solicitado também).

Ao amigo Felipe Sander Pereira Clark, pela sua prontidão em ajudar seja qual fosse o problema, além das tentativas que ele fazia para me "colocar para a frente" sempre que eu ficava desanimado ou preocupado com algo. E ele sempre conseguia.

À amiga Dayana Sant'Anna Lole, que, como mulher no meio de seus amigos todos homens, dava-nos um toque especial de sensibilidade e carinho.

Ao amigo Bernardo Cardoso de Aquino Cruz, por me mostrar que a competência nos estudos e o *carpe diem* podem coexistir, além de ter tido comigo conversas extracurriculares muito humanas, que me faziam sair da rotina fria dos estudos, mesmo estando em pleno ambiente acadêmico.

Ao amigo Diego dos Santos, simplesmente uma das melhores pessoas que alguém pode ter como amigo e que, através de sua personalidade dócil, responsável e madura, sempre me ensina muito.

Por fim, agradeço ao povo brasileiro que contribuiu de forma significativa à minha formação e estada nesta Universidade. Este projeto é uma pequena forma de retribuir o investimento e confiança em mim depositados.

RESUMO

A mistura de estímulos sonoros é recorrente na natureza, e a separação destes faz-se necessária em diversas aplicações na área de Processamento de Áudio. Este trabalho aborda a separação de fontes sonoras a partir de uma única mistura, utilizando especificamente um método estatístico da literatura, conhecido como Análise de Subespaços Independentes (do inglês *Independent Subspace Analysis*, ISA). Sua meta é implementar a ISA e avaliar o seu desempenho com relação a variações de parâmetros e nos tipos de sinais envolvidos, com o fim de formar uma base prática para futuras investigações.

Palavras-Chave: processamento de sinais de áudio, separação de fontes sonoras, análise de subespaços independentes.

ABSTRACT

The mixture of sound stimuli is recurrent in nature, and their separation is necessary in many aplications in the audio signal processing area. This work addresses the separation of sound sources, using specifically a statistical method from the literature, named ISA (Independent Subspace Analysis). Its goal is to evaluate the performance of ISA with respect to different parameter values and for various types of input signals in order to build a pratical basis for future investigations.

Key-words: audio signal processing, sound source separation, independent subspace analysis

SIGLAS

UFRJ - Universidade Federal do Rio de Janeiro

STFT - *Short-Time Fourier Transform*

STFTM - *Short-Time Fourier Transform Magnitude*

ISA - *Independent Subspace Analysis*

ICA - *Independent Component Analysis*

PCA - *Principal Component Analysis*

PDF - *Probability Density Function*

BSS - *Blind Source Separation*

SVD - *Singular Value Decomposition*

SDR - *Signal-to-Distortion Ratio*

SIR - *Signal-to-Interference Ratio*

SAR - *Signal-to-Artifact Ratio*

SER - *Signal-to-Error Ratio*

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

1.1 Tema

O tema deste trabalho está inserido na área de processamento digital de sinais. Realizou-se um estudo sobre o processamento computacional de misturas de estímulos sonoros para separar as contribuições das diversas fontes geradoras desses estímulos; pretende-se, portanto, obter e identificar cada sinal que compõe uma determinada mistura. Essa é uma tarefa de fácil solução para a percepção humana, porém de difícil tratamento no âmbito computacional. A Análise de Subespaços Independentes (do inglês *Independent Subspace Analysis* - ISA) foi a abordagem escolhida neste projeto. Trata-se de uma técnica estatística baseada na Análise de Componentes Independentes (do inglês *Independent Component Analysis* - ICA), por sua vez capaz de separar sinais estatisticamente independentes não-gaussianos que tenham sido misturados por combinação linear.

1.2 Delimitação

Os objetos de estudo são as misturas de sinais contendo áudio ou fala. Considera-se sinal de áudio todo aquele que é audível pelo ser humano, ou seja, situado em uma faixa de frequência entre 20 e 20 kHz. Sinal de fala é todo aquele que é emitido pelo aparelho fonador humano, cuja inteligibilidade depende principalmente da faixa que vai até 4 kHz (telefonia). Naturalmente, um sinal de fala é também um sinal de áudio. Entretanto, por suas características e aplicações próprias, costuma ser

tratado e classificado separadamente. Aqui, serão considerados sinais de áudio de diversas naturezas, incluindo músicas, voz falada e sons do dia-a-dia.

1.3 Justificativa

A mistura de estímulos sonoros é uma ocorrência comum no cotidiano. O canto de um pássaro e o barulho de um automóvel são exemplos de sinais de áudio que podem se misturar em um ambiente. Agora, imagine-se uma sala com várias pessoas conversando, de forma que seja impossível até para um ser humano distinguir o que cada pessoa está falando. Suponha-se agora que nesta sala tenha ocorrido um crime, e que tenha sido feita uma gravação sonora deste ambiente no momento em que o crime ocorreu. A separação de fontes sonoras encontra aí uma aplicação forense, permitindo tentar identificar (separar e amplificar) o que estava sendo dito por cada pessoa, auxiliando assim, talvez, de forma decisiva na perícia. Existem diversas outras aplicações para a separação de fontes sonoras, como na transcrição automática de músicas, na remixagem de gravações, etc.

1.4 Objetivo

Este trabalho tem por objetivo estudar uma técnica específica da literatura, voltada para a separação de fontes sonoras a partir de uma única mistura, a ISA. Pretende-se caracterizar seu desempenho sob diversas condições, em especial variando-se seus parâmetros e os tipos de sinais de áudio envolvidos, a fim de formar uma base prática para futuras investigações.

1.5 Metodologia

O problema da identificação de fontes desconhecidas a partir de uma mistura é chamado de Separação Cega de Fontes (do inglês *Blind Source Separation*, BSS), e pode ser solucionado através da ICA.

A ICA é uma abordagem muito conhecida na literatura, e possui aplicação em diversas áreas do conhecimento, tais como telecomunicações e processamento digital

de imagens. Contudo, a ICA apresenta algumas limitações que tornam algumas soluções pouco práticas, como por exemplo, a exigência de que o número de sensores seja maior ou igual ao número de fontes que se deseja separar.

Na tentativa de contornar essas limitações, desenvolveu-se um método que é uma forma mais genérica da ICA, a ISA [1]. Na ISA, realiza-se o processamento sobre uma única mistura no domínio tempo-frequencial, sendo representada através de um espectrograma, buscando separar as funções-base de tempo ou frequência. Essas funções de base precisam, então, ser agrupadas em conjuntos, de forma que cada um gere um espectrograma que seja capaz de descrever uma fonte. É necessário associar uma fase adequada a esses espectrogramas, para se reconstruir o sinal de áudio no domínio do tempo.

A Figura 1.1 esquematiza o sistema completo de separação estudado neste trabalho.

A pesquisa foi realizada a partir de artigos publicados sobre ICA e ISA, incluindo principalmente os artigos dos desenvolvedores do segundo método. É indicado pela literatura que a ISA é um método ideal para separação de misturas de sinais de bateria. Entretanto, neste trabalho, aplicou-se a ISA sobre misturas envolvendo diversos instrumentos musicais e sinais de voz, a fim de se verificar o desempenho do método de forma abrangente.

Para se atingir o objetivo deste trabalho, realizaram-se testes de forma sistemática, variando parâmetros inerentes à ISA, bem como os tipos dos sinais envolvidos nas misturas, de forma que estes pudessem explorar diversas características do algoritmo a ser investigado.

1.6 Descrição

Segue nos próximos parágrafos a descrição da estrutura deste documento, que foi dividido em 6 capítulos. O presente capítulo apresentou a introdução ao trabalho. No Capítulo 2 é mostrado o processamento realizado sobre um sinal de mistura

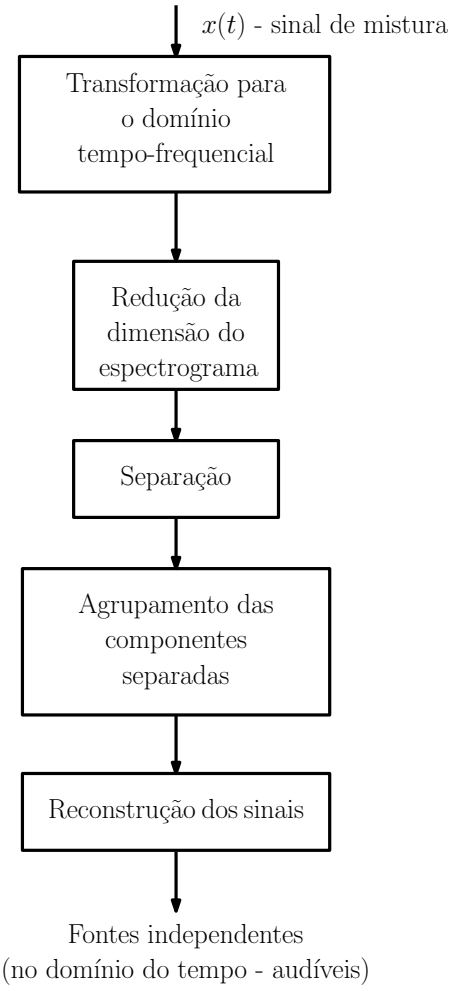

Figura 1.1: Esquema simplificado de um sistema de ISA.

de fontes antes de ser submetido à ISA. Mais especificamente, é discutido o pré-processamento de cada segmento do sinal.

O Capítulo 3 é o capítulo mais importante deste trabalho, pois apresenta o estudo sobre separação de fontes sonoras, abordando a ICA através de uma linha teórica didática que culmina na abordagem da ISA. No fim deste capítulo, aborda-se a *clusterização*, técnica pela qual, após a separação estatística das funções de base, busca-se identificar tais funções e agrupá-las de forma a produzir os espectrogramas de cada fonte sonora que compõe o sinal de mistura.

A reconstrução de fase para os espectrogramas de cada fonte separada é o tema apresentado no Capítulo 4. Nele, são mostradas técnicas que estimam e associam uma fase adequada para esses espectrogramas, resultando em sinais de áudio válidos e permitindo a avaliação da separação.

No Capítulo 5 são apresentadas as simulações computacionais, cujos resultados são avaliados criticamente. As conclusões do projeto são apresentadas no Capítulo 6.

Capítulo 2

Análise de sinais em blocos

2.1 Transformada de Fourier de Curta Duração

No processamento digital de sinais, é muito comum realizar-se a segmentação de um dado sinal no domínio do tempo em blocos com durações iguais. De maneira geral, esse procedimento é feito para facilitar a modelagem, já que o modelo de um bloco de sinal, por conter somente as informações referentes a esse trecho, tende a ser mais simples do que resultaria para o sinal inteiro.

A segmentação também permite uma análise mais "detalhada" do sinal em estudo, e pode ser aplicada, portanto, quando se deseja lidar com a sua não-estacionariedade[1]. A divisão é feita de forma que os blocos obtidos possam ser considerados estacionários, já que, dentro de um intervalo curto, o sinal costuma ser mais bem comportado do que indicaria sua análise completa. Dessa forma, pode-se descrever um sinal não-estacionário através de seus blocos modelados como sendo estacionários.

É também muito comum analisar-se um sinal originalmente descrito no domínio do tempo através da sua representação no domínio da frequência. A Transformada de Fourier [2] é uma ferramenta muito utilizada para obter-se essa representação,

[1]No sentido amplo, um sinal é considerado estacionário quando a média e a variância do processo aleatório que o modela não se alteram ao longo do tempo.

sendo a sua versão em tempo discreto dada matematicamente por

$$X(\omega) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n}, \tag{2.1}$$

onde $x(n)$ é o sinal no domínio do tempo (uma sequência temporal, sendo n um número inteiro representando o tempo discreto) e $X(\omega)$ é o sinal representado no domínio da frequência (em que ω representa a frequência angular normalizada).

Na ISA, um sinal (de entrada, que contém dois ou mais estímulos sonoros misturados) é transformado do domínio do tempo para o domínio de tempo-frequência, que é uma forma de representar as características frequenciais (espectrais) do sinal ao longo do tempo. Faz-se necessário, então, o cálculo de tais características para intervalos temporais determinados. Para isso, utiliza-se a Transformada de Fourier de Curta Duração (do inglês *Short-Time Fourier Transform, STFT*), que nada mais é do que a Transformada de Fourier convencional aplicada a um trecho do sinal finito (segmento) no tempo [2]. A equação da STFT para tempo discreto é dada por

$$X(mS, \omega) = \sum_{n=-\infty}^{\infty} x(n)w(n - mS)e^{-j\omega n}, \tag{2.2}$$

onde $x(n)$ é o sinal no domínio do tempo, w é uma função que será descrita mais adiante, responsável por fazer a segmentação desse sinal, mS é o deslocamento de w de forma a abranger trechos diferentes de $x(n)$ (em que S é o passo de análise e m é um número inteiro) e $X(mS, \omega)$ é a STFT do sinal $x(n)$.

Observa-se que as duas equações apresentadas anteriormente são parecidas, com a diferença de que na equação (2.2) existe a função $w(n)$ deslocada de mS multiplicando o sinal temporal $x(n)$. Essa função de segmentação é conhecida como *janela*. Ao ser feita a multiplicação desse sinal com a janela, todo o restante dele, que está fora da abrangência de $w(n)$, é ignorado. Dessa forma, $x(n)w(n - mS)$ representa os segmentos do sinal $x(n)$, e, para cada um deles (ou seja, para cada valor de m), é calculada a Transformada de Fourier, produzindo a STFT $X(mS, \omega)$.

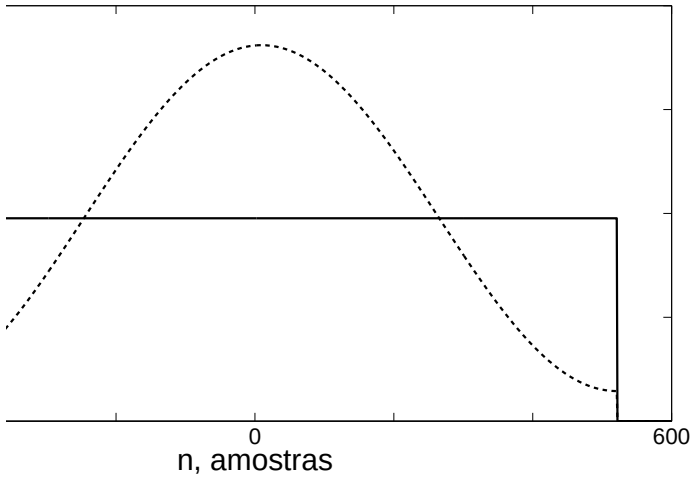

Figura 2.1: Representação no tempo da janela retangular (linha contínua) e da janela de Hamming (linha tracejada) de 1024 amostras, ambas normalizadas para $\sum_n w(n) = 1$.

A variável m indexa os segmentos de $X(mS, \omega)$ no tempo, enquanto ω representa a frequência. A magnitude de $X(mS, \omega)$ resulta no chamado espectrograma de $x(n)$:

$$\text{esp}(x(n)) = |X(mS, \omega)|. \tag{2.3}$$

2.2 Tipos de janelas

Para segmentar o sinal, é razoável pensar, inicialmente, em apenas dividi-lo em intervalos, sem a preocupação de usar uma janela para isso. No entanto, essa ideia também corresponde a multiplicar o sinal por uma janela, retangular, como a mostrada em linha contínua na Figura 2.1.

Como foi visto na equação (2.2), a segmentação do sinal é feita multiplicando-o pela janela deslocada no tempo. É conhecido da teoria de sistemas lineares que a multiplicação de duas funções no domínio do tempo é equivalente à operação de convolução entre esses dois sinais representados no domínio da frequência. Devido a essa característica, deve-se ter cuidado na escolha da janela $w(n)$ de forma que, ao multiplicá-la com a função $x(n)$ que se deseja analisar, esta não tenha suas características espectrais comprometidas além do aceitável.

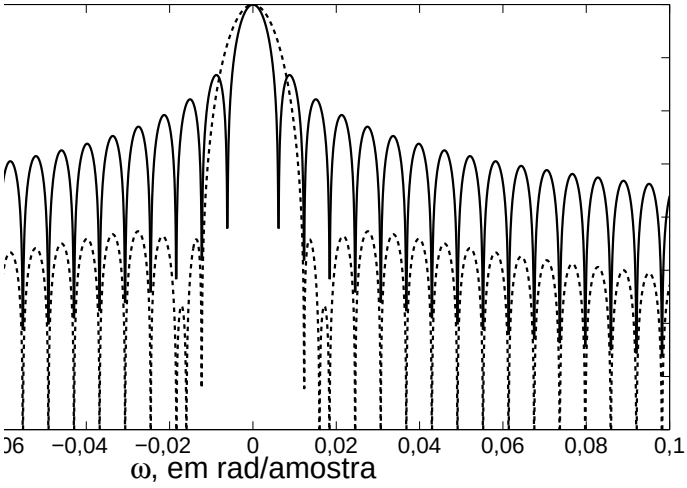

Figura 2.2: Espectros das janelas retangular (em linha contínua) e de Hamming (em linha tracejada). Observa-se que os lobos secundários da janela de Hamming são mais atenuados que os da janela retangular.

A representação frequencial de uma janela retangular é uma função *sampling*[2], que apesar de possuir o lobo principal mais estreito dentre todos os tipos de janela, possui lobos laterais elevados. Essa grande dispersão espectral da janela retangular está relacionada ao corte abrupto que ela realiza no sinal. Pode-se desejar uma janela cujo espectro de frequências tenha lobos laterais menores, o que corresponde a uma representação temporal em que a janela, nas suas extremidades, tenha uma transição mais suave até zero.

Dentre as diversas janelas da literatura com tal característica, a de *Hamming*—mostrada em linha tracejada na Figura 2.1—é bastante aplicada em processamento de áudio. Neste trabalho, foi utilizada uma versão modificada dela, que será explicada no último capítulo. A Figura 2.2 mostra a vantagem espectral da janela de Hamming sobre a janela retangular: a primeira possui lobos secundários mais atenuados que a segunda, ao custo de ter o lobo principal mais largo.

Cabe ressaltar que o sinal $x(n)$ sempre sofrerá alterações quando multiplicado por uma janela. Somente uma janela cuja representação no domínio da frequência fosse um impulso não distorceria o espectro do sinal; mas esta corresponderia a uma janela plana de tamanho infinito no domínio do tempo, o que, em outras palavras,

[2]A função *sampling* é definida como $Sa(u) = \frac{\text{sen}(u)}{u}$.

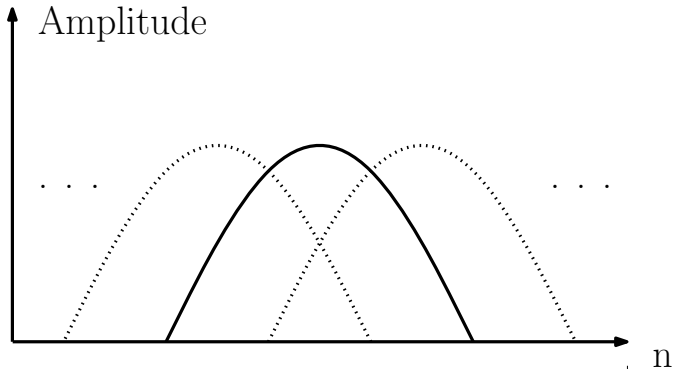

Figura 2.3: As janelas em pontilhado devem superpor-se à janela central de forma a compensar as atenuações causadas por ela na amplitude do sinal.

seria o mesmo que não realizar nenhum tipo de segmentação.

2.3 Sobreposição de janelas

Conforme visto na seção anterior, a janela retangular corta o sinal abruptamente. Porém, ela garante que todas as amostras do trecho sejam igualmente consideradas. Todavia, uma janela suavizadora como a de Hamming acaba por atenuar gradualmente mais as amostras do segmento em direção às extremidades das janelas, o que significa perda de informação. Para mitigar esse problema, uma solução adotada é a sobreposição das janelas, mostrada na Figura 2.3, em que o trecho onde o sinal foi atenuado por uma janela é compensado por outra. Em outras palavras, janelas adjacentes podem abranger um mesmo trecho do sinal, mas é desejável que suas atenuações sejam complementares, ou seja, que a soma das suas contribuições totalize 1.

Na equação (2.2), o parâmetro S é a distância entre duas janelas sobrepostas, considerando pontos correspondentes dos segmentos (centro a centro, por exemplo). Observa-se, então, que S está diretamente relacionado à sobreposição das janelas, e que deve ser menor do que o comprimento L da janela para que haja sobreposição. Geralmente o valor de L é muito menor que o número N de pontos do sinal a ser analisado, fazendo com que seja necessário um número razoável de janelas para abranger todo o sinal. Já o valor de S costuma ser tipicamente metade ou 1/4 do

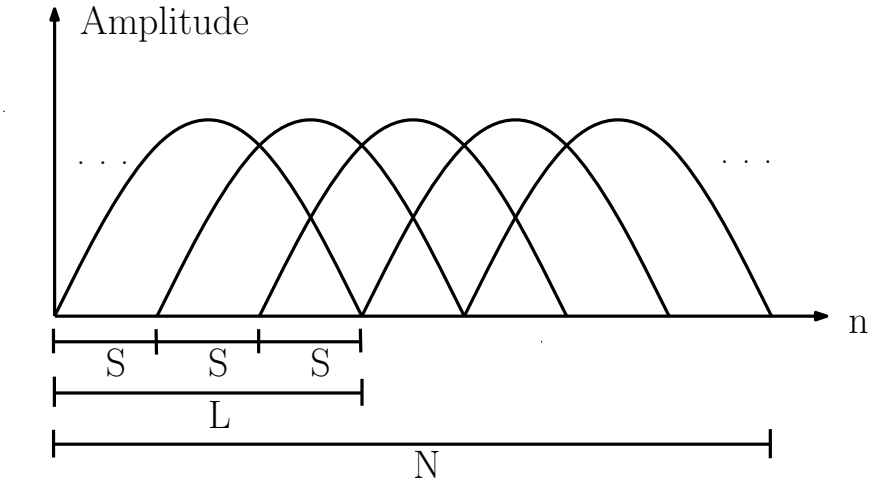

Figura 2.4: Janelamento com sobreposição mostrando a relação entre N, L e S.

valor de L. A Figura 2.4 ilustra a relação comum entre esses três parâmetros em um janelamento com sobreposição de janelas. Matematicamente,

$$S < L << N. \tag{2.4}$$

É importante destacar um problema, relacionado ao janelamento com sobreposição, que ocorre com as amostras iniciais e finais do sinal $x(n)$. No caso das amostras iniciais, estas são contempladas somente pelas primeiras janelas, e são atenuadas por elas sem haver compensação por outras anteriores. Como solução, pode-se preencher o início do sinal com um número adequado de zeros, conforme mostra a Figura 2.5, antes de se fazer o janelamento. A ideia é garantir que desde a primeira amostra válida do sinal ocorra a complementação das janelas sobrepostas. Esse aumento de amostras do sinal original é muito pequeno se comparado a N e, portanto, não resulta em uma diferença computacionalmente significativa. O mesmo se aplica às últimas amostras do sinal.

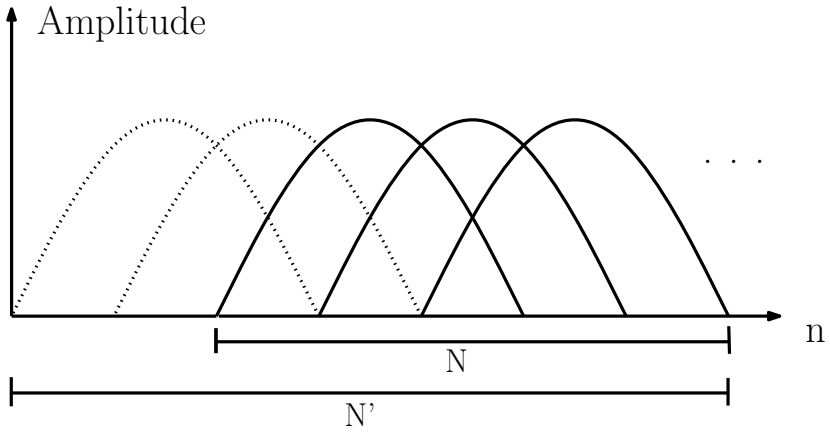

Figura 2.5: Extensão do sinal com zeros em seu início, aumentando o número de amostras de N para N'.

2.4 Transformada de Fourier Discreta

Nas equações (2.1) e (2.2), a variável ω é contínua na frequência. Tal característica não é adequada quando se deseja realizar o processamento computacional do sinal. Portanto, é necessário o mapeamento da variável discreta no tempo n para uma variável discreta na frequência k [3]. Esse mapeamento é denominado Transformada de Fourier Discreta (do inglês, *Discrete Fourier Transform, DFT*), sendo o responsável por tornar a Transformada de Fourier convencional e a STFT aplicáveis computacionalmente. As equações (2.1) e (2.2) representam, respectivamente, a Transformada de Fourier e a STFT discretas no tempo, somente.

Sabe-se que todo sinal periódico—e, consequentemente, de duração infinita—pode ser representado por uma Série de Fourier [4], que é uma soma ponderada de exponenciais complexas. A DFT se baseia no conceito de Série de Fourier: adotando como restrição que o sinal a se representar seja de duração finita, associa-se esta duração ao período de um sinal periódico subjacente; com isso, usa-se diretamente a Série de Fourier a este associada como a DFT do sinal sob análise.

Esta formulação se enquadra perfeitamente no conceito de STFT. Pode-se provar que a STFT discreta na frequência é a versão amostrada da STFT contínua na frequência [4], tal que a variável de frequência discretizada k indica a frequência

$k\omega_0$, em que ω_0 é a frequência fundamental do sinal periódico $x_p(n)$[3] associado, dada por

$$\omega_0 = \frac{2\pi}{N_0}, \tag{2.5}$$

onde N_0 corresponde ao número de pontos da DFT.

Na DFT, por padrão, o número de amostras N do sinal no tempo será igual ao número de pontos N_0 na representação frequencial. Entranto, pode-se escolher N_0 maior do que N. Quanto maior for N_0, menor será o ω_0, o que resulta em uma amostragem mais fina do espectro, ou seja, maior resolução na sua leitura. Um sinal com número de amostras menor do que aquele que se deseja para o espectro—a fim de se garantir boa resolução—deve ser preenchido com zeros no final (técnica chamada de *zero-padding*), de forma a atingir o número de amostras desejado.

Fazendo-se a devida substituição, a STFT discreta na frequência é dada por

$$X(mS, k) = \sum_{n=-\infty}^{\infty} x(n)\omega(n - mS)e^{-j\frac{2\pi kn}{N_0}}. \tag{2.6}$$

Há algoritmos rápidos para realizar a DFT, chamados genericamente de Transformada de Fourier Rápida (do inglês, *Fast Fourier Transform*, FFT) [5]. O mais popular deles, chamado de raiz 2, requer que o número de amostras da DFT seja uma potência de 2. Para isso, usualmente é feito um *zero-padding* de forma que o comprimento da DFT seja a potência de 2 mais próxima do sinal.

2.5 Janelamento de fase zero

A informação espectral extraída de um segmento janelado do sinal é comumente associada ao instante que corresponde ao centro da janela. Entretanto, a forma como o janelamento foi abordado até então refere-se ao uso de janelas causais (pois os segmentos são causais), implicando um deslocamento de $L/2$. Lembrando que os espectros do sinal e da janela são convoluídos, fica claro que o janelamento resulta em alterações indesejáveis (não-lineares) na informação de fase dos segmentos do

[3]para o qual $x_p(n) = x_p(n + N_0) \ \ \forall n$.

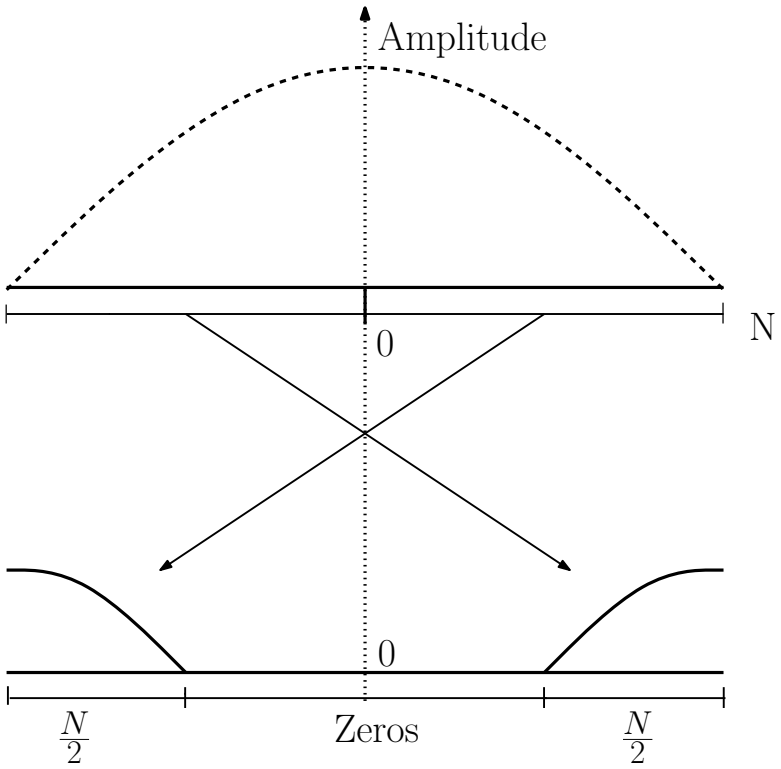

Figura 2.6: Procedimento para realizar o janelamento de fase zero.

sinal [6], já que a janela possui fase não-nula.

Tendo em vista que os segmentos do sinal mapeados pela DFT são tomados como períodicos, pode-se retirar a segunda metade das amostras de cada segmento e inseri-las na frente da primeira metade, sem alterar a representação frequencial. Isso fará com que a janela seja simétrica em relação à origem. Dessa forma, ficará com fase nula e seus efeitos na fase do sinal serão mitigados. Esse procedimento é chamado de *janelamento de fase zero*, e está mostrado na Figura 2.6. Se for necessário fazer o *zero-padding* do segmento para que se tenha maior resolução na frequência, este deve ser feito entre os trechos que foram invertidos, na região central.

Capítulo 3

Separação de fontes sonoras

3.1 Discussão inicial

A mistura de estímulos sonoros é uma ocorrência comum no dia-a-dia. Em um mesmo ambiente, o barulho do motor de um veículo que passa pela rua e o som de um pássaro cantando podem chegar misturados aos ouvidos de alguém que ali se encontra. Mesmo assim, é imediato para qualquer pessoa com audição normal perceber qual é o som do pássaro e qual é o do motor, pois o cérebro humano é treinado, ou seja, ele já "conhece" tais estímulos e, portanto, consegue individualizar suas possíveis fontes geradoras.

Imagine-se agora um exemplo em que dezenas de pessoas estão conversando ao mesmo tempo em uma sala, e há um observador ouvindo a mistura de vozes que ali ocorre. Pode-se facilmente imaginar aquele ruído típico causado por dezenas de conversas simultâneas em um único ambiente, em que nada que está sendo dito é compreensível. Porém, eventualmente, o observador pode identificar algumas frases provenientes de algumas pessoas devido a uma das seguintes características:

(1) Há, no meio dos falantes, alguém que o observador conheça, o que torna mais fácil a identificação dessa voz e, consequentemente, do que é dito por essa pessoa;

(2) Há, no meio dos falantes, alguém que fale mais alto do que os outros, e que portanto tem suas frases destacadas;

(3) Há, no meio dos falantes, pessoas que estão mais próximas do observador,

facilitando também a segregação de suas vozes e conversas.

Faça-se agora a seguinte analogia: um microfone acoplado a um computador seria como os ouvidos do observador, e a Unidade Central de Processamento (CPU) desse computador seria o cérebro do observador. Cabe ao computador, a partir de um sinal de áudio misturado, identificar as fontes geradoras (quem são as pessoas) dos estímulos que compõem essa mistura. Não conseguirá, por um simples motivo: diferentemente do cérebro, uma CPU não é capaz de realizar nenhum tipo de processamento por conta própria. Para o computador, um sinal que contém o barulho do motor de um veículo e o canto de um pássaro é análogo ao que um ser humano percebe do ruído de vozes não-identificáveis mencionado no parágrafo anterior.

Dessa forma, é necessário fazer uso de algum tipo de método capaz de tornar esse "ruído de vozes não-identificáveis" em "conversas mais inteligíveis". Assim como um observador humano procura aspectos proeminentes que identifiquem uma das vozes, o processamento computacional deve extrair determinadas características da mistura que permitam identificar as fontes geradoras dos estímulos que a compõem. Uma das formas de realizar essa extração é o tema principal deste trabalho: a Análise de Subespaços Independentes.

3.2 Descorrelação e independência

Antes de serem apresentadas as técnicas para a resolução do problema das misturas de fontes sonoras, faz-se necessário relembrar dois conceitos fundamentais relativos à matemática estatística.

Duas variáveis aleatórias vetoriais $\mathbf{X}$ e $\mathbf{Y}$ são descorrelacionadas se a covariância $\mathbf{C_{XY}}$ entre elas for nula [7], ou seja,

$$\mathbf{C_{XY}} = E[(\mathbf{X} - \bar{\mathbf{X}})(\mathbf{Y} - \bar{\mathbf{Y}})^T] = \mathbf{0}, \tag{3.1}$$

onde $\mathbf{0}$ é uma matriz nula, E [.] é a função Valor Esperado, $\bar{\mathbf{X}}$ e $\bar{\mathbf{Y}}$ são as médias de $\mathbf{X}$ e $\mathbf{Y}$ respectivamente, e T é o operador de transposição. A condição $\mathbf{C_{XY}} = \mathbf{0}$

leva a:

$$\mathbf{R_{XY}} = E[\mathbf{X}\mathbf{Y}^T] = E[\mathbf{X}]E[\mathbf{Y}] = \bar{\mathbf{X}}\bar{\mathbf{Y}}^T, \tag{3.2}$$

onde $\mathbf{R_{XY}}$ é a função de correlação cruzada entre $\mathbf{X}$ e $\mathbf{Y}$ (a prova dessa passagem pode ser vista no Apêndice A). Assim, pode-se dizer também que as variáveis aleatórias $\mathbf{X}$ e $\mathbf{Y}$ são descorrelacionadas se a correlação entre elas for fatorável como o produto das suas médias.

Entretanto, um conceito mais forte que a descorrelação é a independência estatística entre duas variáveis aleatórias. $\mathbf{X}$ e $\mathbf{Y}$ são independentes se a Função de Densidade de Probabilidade (do inglês *Probability Density Function*, PDF) conjunta $f_{\mathbf{X},\mathbf{Y}}(\mathbf{x}, \mathbf{y})$ for fatorável como o produto das densidades de probabilidade individuais [8], ou seja,

$$f_{\mathbf{X},\mathbf{Y}}(\mathbf{x}, \mathbf{y}) = f_{\mathbf{X}}(\mathbf{x}) f_{\mathbf{Y}}(\mathbf{y}). \tag{3.3}$$

De maneira mais descritiva, pode-se dizer que duas variáveis aleatórias são independentes se os valores de $\mathbf{X}$ não fornecem nenhuma informação sobre os valores de $\mathbf{Y}$ e vice-versa. $\mathbf{X}$ e $\mathbf{Y}$ são variáveis produzidas por dois experimentos aleatórios desconexos, assim como—retornando ao exemplo já visto—o canto do pássaro e o barulho do motor de um veículo. É importante mencionar que se duas variáveis aleatórias são independentes, então elas também serão descorrelacionadas. Entretanto, o inverso não é necessariamente verdadeiro.

3.3 Análise de Componentes Principais

O problema das inúmeras conversas que acontecem em uma sala é conhecido como *cocktail party problem*, cuja tradução é "problema do coquetel"—como se as pessoas estivessem conversando descontraidamente em um momento informal, como acontece em um coquetel. Muitas abordagens têm sido propostas para resolver o problema de individualizar uma ou mais dentre as diversas fontes sonoras presentes em uma ou mais misturas, o que se costuma chamar de separação de fontes.

Uma técnica tradicional da literatura é a denominada Análise de Componentes Principais (do inglês *Principal Component Analysis*, PCA), utilizada em análise

estatística de dados e extração das características principais de uma mistura de fontes [7]. A PCA é capaz de descorrelacionar um conjunto de variáveis antes correlacionadas, devolvendo as componentes em ordem decrescente de importância, isto é, da mais intensa para a menos intensa.

Para melhor entendimento, pode-se fazer uma analogia com o *cocktail party problem*. Supõe-se que entre os falantes há alguém que fale com uma voz muito aguda, bem destacada das vozes dos demais. Essa, portanto, é a característica de maior destaque dessa mistura e que será apontada pela PCA como a característica de maior relevância[1].

Considerando que se escolham apenas as primeiras componentes mais relevantes, que mais contribuem para o aumento da variância da mistura, para aproximá-la, pode-se dizer que a PCA serve como técnica de redução de redundância. As componentes retidas são as chamadas componentes principais e, devido ao fato de a PCA descartar o restante das componentes, diz-se que ela reduz a dimensão do espaço que gerou a mistura [9] [10] [11].

3.4 Análise de Componentes Independentes

3.4.1 Introdução

A PCA é uma técnica que realiza a descorrelação entre as componentes subjacentes que, somadas, formam a(s) mistura(s). Contudo, exceto para distribuições gaussianas [12], ela não garante a independência entre elas. Caso se deseje obter componentes independentes, faz-se necessária uma técnica mais poderosa do que a PCA. Essa técnica é chamada de Análise de Componentes Independentes (do inglês *Independent Component Analysis*, ICA) [7] [12] [13].

A ICA baseia-se no conceito de independência estatística entre duas variáveis aleatórias. Tomando-se como exemplo sinais de áudio, que são o foco deste trabalho, se há uma combinação linear de estímulos sonoros gerados por fontes completamente

[1]Neste modelo, cada sinal no tempo é associado a uma variável aleatória, recorrendo ao conceito de ergodicidade.

independentes, a ICA é o procedimento capaz de identificar e fornecer cada um desses estímulos. Para misturas mais complexas, por se tratar de um método estatístico, a ICA fornece como resultados estimativas—nem sempre aceitáveis—dos estímulos originais (não-misturados).

Pode-se interpretar também a ICA como capaz de efetivar descorrelação dos momentos de alta ordem entre as variáveis, enquanto que a PCA produz descorrelação de segunda ordem. Portanto, é de se esperar que a ICA seja uma técnica com desempenho superior ao da PCA na separação de fontes. Mas esse fato não descarta o uso da PCA, muito utilizada como uma etapa de pré-processamento para a ICA.

3.4.2 Não-gaussianidade

Conforme mencionado anteriormente, a descorrelação entre duas variáveis aleatórias não implica necessariamente a independência entre elas. Todavia, essa correspondência é verdadeira quando as variáveis em questão são gaussianas, também chamadas de normais (a prova para essa afirmação pode ser vista no Apêndice B). Sendo assim, a ICA precisa assumir que as fontes que se deseja separar são não-gaussianas. Caso contrário, a ICA não fará nada além do que foi feito pela PCA.

A não-gaussianidade é um conceito tão importante para a ICA que pode ser usada inclusive como princípio (ou parâmetro) de separação. O Teorema do Limite Central expressa o fato de que o resultado da soma de variáveis aleatórias independentes é uma distribuição mais próxima de uma gaussiana do que as distribuições de cada uma dessas variáveis sozinhas. A prova desse teorema pode ser vista em [8]. Assim, busca-se na ICA encontrar um ponto ótimo em que a gaussianidade das variáveis aleatórias (medida de quão próxima de uma gaussiana é sua distribuição) a serem estimadas seja mínima; isso quer dizer que as variáveis aleatórias estimadas estão "minimamente misturadas" e, portanto, são independentes uma das outras.

3.4.3 Modelo da ICA

Imagine-se uma situação parecida com o *cocktail party problem*, com a diferença de que existem apenas duas pessoas (1 e 2) conversando na sala. Há também dois

microfones (1 e 2) em pontos diferentes dessa sala gravando a conversa. Não é difícil imaginar que a conversa entre as duas pessoas não chegará da mesma forma nos dois microfones, pois isso depende da localização destes na sala. Talvez o microfone 1 esteja mais perto da pessoa 2 do que da pessoa 1, e por isso consiga captar melhor a voz da pessoa 2. Pode ser também que o microfone 2 esteja localizado igualmente próximo das duas pessoas, enquanto o microfone 1 está mais distante delas, captando a conversa com som baixo. O importante é perceber que existem diferentes formas de como essas duas pessoas e esses dois microfones podem estar dispostos na sala.

A modelagem dessa situação deveria descrever diferentes tempos de chegada e de intensidade com que as duas vozes chegam aos dois microfones, além dos múltiplos percursos que modelam as condições acústicas da sala. O modelo descrito a seguir contempla, entretanto, apenas as possíveis diferenças de intensidade com que a informação proveniente de cada fonte chega a cada um dos sensores. Isso é feito através da ponderação por coeficientes reais. Sendo $s_1(t)$ e $s_2(t)$ o som emitido pelas pessoas 1 e 2 respectivamente (t é o índice temporal), e $x_1(t)$ e $x_2(t)$ os sinais gravados pelos microfones 1 e 2 respectivamente, pode-se modelar esse problema da seguinte maneira [12]:

$$x_1(t) = a_{11}s_1(t) + a_{12}s_2(t) \tag{3.4}$$

$$x_2(t) = a_{21}s_1(t) + a_{22}s_2(t), \tag{3.5}$$

onde a_{11}, a_{12}, a_{21} e a_{22} são os ponderadores.

A ICA precisa assumir que as fontes $s_1(t)$ e $s_2(t)$ são estatisticamente independentes. A partir daí, a ideia é estimar essas fontes a partir das misturas $x_1(t)$ e $x_2(t)$. Como mais nada é sabido sobre as fontes, esse problema é conhecido como Separação Cega de Fontes (do inglês *Blind Source Separation*, BSS) em suas diversas formulações, e possui várias aplicações em processamento de sinais [1]. Uma das formas de se realizar a BSS é através da ICA.

O modelo de mistura apresentado nas equações (3.4) e (3.5) pode ser reescrito da forma matricial

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} . \begin{bmatrix} s_1(t) \\ s_2(t) \end{bmatrix}, \tag{3.6}$$

onde o vetor das misturas pode ser chamado de $\mathbf{x}$, o vetor das fontes de $\mathbf{s}$ e a matriz dos coeficientes de $\mathbf{A}$, levando assim à representação simplificada

$$\mathbf{x} = \mathbf{A}\mathbf{s}. \tag{3.7}$$

A matriz $\mathbf{A}$ é suposta de posto completo[2] e pode ser entendida como uma transformação linear pela qual o vetor de fontes $\mathbf{s}$ passa, resultando no vetor de misturas $\mathbf{x}$. Devido ao fato de realizar a mistura das fontes, a matriz $\mathbf{A}$ é conhecida como **matriz de mistura**.

Como o objetivo do método é estimar as fontes $s_1(t)$ e $s_2(t)$, deseja-se fazer a operação

$$\hat{\mathbf{s}} = \mathbf{A}^{-1}\mathbf{x}, \tag{3.8}$$

onde $\hat{\mathbf{s}}$ é a estimativa do vetor $\mathbf{s}$, e $\mathbf{A}^{-1}$ é a inversa da matriz $\mathbf{A}$. Essa equação expressa a forma de se obter uma estimativa das fontes a partir das misturas. Como o vetor $\mathbf{x}$ é dado do problema (pois contém os sinais de mistura), fica claro que o processamento por ICA objetiva calcular a matriz $\mathbf{A}^{-1}$. Essa matriz realiza uma transformação linear sobre o vetor $\mathbf{x}$ de forma a retornar as fontes independentes, e por isso é conhecida como **matriz de *desmistura***, sendo representada por $\mathbf{W}$ para simplificar a notação:

$$\mathbf{W} \approx \mathbf{A}^{-1}, \tag{3.9}$$

ou seja,

$$\hat{\mathbf{s}} \approx \mathbf{W}\mathbf{x}. \tag{3.10}$$

O sinal $\approx$ na equação (3.9) indica a natureza estatística do método, em que busca-se uma estimativa das fontes independentes. Observa-se que a matriz $\mathbf{A}$ também deve ser invertível para viabilizar o método (o que é verdade se ela possuir posto

[2]Uma matriz $i \times j$ é dita de posto completo se tem $\min\{i, j\}$ linhas e colunas linearmente independentes. Em outras palavras, uma mistura não pode ser uma combinação linear das outras.

completo). A condição de invertibilidade de uma matriz é a de que deve existir uma outra matriz cujo produto entre as duas resulte na matriz-identidade[3] I_2. Pode-se substituir $\mathbf{x}$ na equação (3.10) pela equação (3.7), obtendo-se

$$\hat{\mathbf{s}} \approx \mathbf{WAs}. \tag{3.11}$$

Se $\mathbf{W}$ é a inversa de $\mathbf{A}$, tem-se que $\mathbf{WA} = \mathbf{I}$, onde $\mathbf{I}$ é a matriz identidade. A equação $\hat{\mathbf{s}} \approx \mathbf{Is}$ indica que $\hat{\mathbf{s}} \approx \mathbf{s}$. Descrevendo de uma outra forma, o objetivo da ICA na BSS é fazer com que o produto $\mathbf{WA}$ seja o mais próximo possível da matriz identidade, encontrando o valor de $\mathbf{W}$ mais adequado para essa finalidade.

3.4.4 Estendendo o modelo

O modelo da ICA mostrado na equação (3.6) se refere ao caso em que se têm duas fontes e dois sensores. Entretanto, pode-se estender este modelo para um caso geral, em que se tenha diferentes números desses elementos. Sendo n a quantidade de fontes e k a quantidade de sensores, o modelo para o caso geral pode ser escrito como:

$$\begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ \vdots \\ x_k(t) \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & ... & a_{1n} \\ a_{21} & a_{22} & a_{23} & ... & a_{2n} \\ a_{31} & a_{32} & a_{33} & ... & a_{3n} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{k1} & a_{k2} & a_{k3} & ... & a_{kn} \end{bmatrix} \cdot \begin{bmatrix} s_1(t) \\ s_2(t) \\ s_3(t) \\ \vdots \\ s_n(t) \end{bmatrix}, \tag{3.12}$$

podendo também ser representado na forma matricial mostrada na equação (3.7).

Uma restrição importantíssima do modelo da ICA é que o número k de sensores tem que ser maior ou igual ao número de n de fontes [7],

$$k \geq n. \tag{3.13}$$

Essa restrição é necessária para garantir uma solução para o problema[4]—retornando, assim, as componentes independentes corretamente—e é um dos principais motivos

[3] $I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

[4] A solução é exata para $k = n$ e aproximada para $k > n$.

que levam à adoção da ISA (tema central deste trabalho) como um método de separação de fontes alternativo à ICA, conforme será visto mais adiante.

3.4.5 Ambiguidades da ICA

A ICA possui duas características que se referem às diferentes formas de como o método pode devolver as componentes independentes estimadas. Essas características não podem ser controladas e são chamadas de ambiguidades da ICA. São elas [12]:

(1) Não é possível determinar a ordem das componentes independentes;

(2) Não é possível determinar a energia de cada componente independente.

A primeira ambiguidade pode ser explicada reescrevendo-se o modelo[5] da equação (3.6) da forma

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \begin{bmatrix} a_{11} \\ a_{21} \end{bmatrix} s_1(t) + \begin{bmatrix} a_{12} \\ a_{22} \end{bmatrix} s_2(t). \tag{3.14}$$

Devido ao fato de $s_1(t)$, $s_2(t)$ e a matriz $\mathbf{A}$ serem dados desconhecidos do problema (lembrando que o modelo da ICA baseia-se no BSS, que é um modelo "cego"), pode-se trocar livremente a ordem dos termos da equação (3.14) e considerar qualquer uma das componentes independentes como sendo a primeira ($x_1(t)$) ou a segunda ($x_2(t)$). De forma mais analítica, pode-se tomar uma matriz de permutação[6] $\mathbf{P}$ e sua inversa $\mathbf{P}^{-1}$ e inseri-las no modelo da ICA, obtendo

$$\mathbf{x} = \mathbf{A}\mathbf{P}^{-1}\mathbf{P}\mathbf{s}, \tag{3.15}$$

em que o vetor $\mathbf{Ps}$ representa as fontes sonoras, porém em ordem diferente, e a matriz $\mathbf{AP}^{-1}$ é uma nova matriz de mistura a ser encontrada pela ICA.

A segunda ambiguidade também está relacionada ao fato de não se ter nenhuma informação sobre as fontes sonoras e a matriz $\mathbf{A}$. Por causa disso, qualquer constante

[5]Utilizando o modelo de duas fontes e dois sensores.

[6]Matriz de pemutação é uma matriz quadrada binária que possui exatamente um único valor 1 em cada linha e em cada coluna, e 0 no restante. É utilizada para realizar a permutação dos elementos de uma matriz ao multiplicá-la.

α_i que multiplique umas das fontes $s_i(t)$ pode ser cancelada pela mesma constante que esteja realizando a divisão da coluna correspondente,

$$\begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} = \frac{1}{\alpha_1} \begin{bmatrix} a_{11} \\ a_{21} \end{bmatrix} \alpha_1 s_1(t) + \frac{1}{\alpha_2} \begin{bmatrix} a_{12} \\ a_{22} \end{bmatrix} \alpha_2 s_2(t), \tag{3.16}$$

o que resulta em uma impossibilidade de se determinar as energias das componentes independentes. Todavia, uma maneira para contornar esse problema é assumir que a variância das fontes sonoras seja unitária ($E[s_i^2(t)] = 1$). O vetor $\mathbf{x}$ será modificado por um pré-processamento—conforme visto mais adiante—a fim de levar em consideração essa restrição, entre outras.

3.4.6 Centralização e branqueamento

Antes de se aplicar a ICA sobre o vetor de misturas $\mathbf{x}$, é interessante realizar dois tipos de pré-processamento sobre ele a fim de tornar a ICA mais simples e melhor condicionada [12].

O primeiro, chamado de centralização, tem o objetivo de fazer com que as misturas tenham média zero. Isso é feito subtraindo de $\mathbf{x}$ o valor de sua média $\bar{\mathbf{X}}$. Ao final da ICA, o vetor de estimativa $\hat{\mathbf{s}}$ também estará centralizado, mas poderá ter sua média adicionada a ele pela operação $\mathbf{A}^{-1}\bar{\mathbf{X}}$.

O segundo, chamado de branqueamento, tem por objetivo tornar as misturas descorrelacionadas e com variância unitária, e é feito após a centralização. Sendo assim, deve-se realizar uma transformação linear sobre $\mathbf{x}$, já centralizado, de tal forma que o vetor branqueado possua uma matriz de covariância igual à identidade. Maiores detalhes sobre o branqueamento podem ser encontrados em [7].

3.4.7 Negentropia, uma medida de não-gaussianidade

Conforme visto anteriormente, a não-gaussianidade pode ser utilizada como parâmetro para estimar as componentes independentes de uma mistura de fontes sonoras, já que, pelo Teorema do Limite Cental, a soma de variáveis aleatórias tende a ter

a sua distribuição mais próxima de uma gaussiana do que as distribuições dessas variáveis sozinhas.

Existem algumas formas de se fazer a medição da não-gaussianidade das misturas. Entretanto, neste trabalho, foi abordada apenas uma delas, que é utilizada pelo algoritmo descrito na próxima subseção: a **negentropia**.

Todavia, antes de ser definido o conceito de negentropia, é preciso entender o que significa entropia. Entropia é um conceito básico na Teoria da Informação que mede a quantidade de informação que uma variável aleatória pode conter. Quanto mais aleatória (ou imprevisível, não-estruturada) é uma variável, maior é a sua entropia [12].

A entropia é definida para uma variável aleatória discreta, mas também pode ser generalizada para variáveis e vetores aleatórios contínuos, sendo chamada neste caso de entropia diferencial. A entropia diferencial H de um vetor aleatório $\mathbf{y}$ com densidade de probabilidade $f(\mathbf{y})$ é definida como

$$H(\mathbf{y}) = -\int f(\mathbf{y}) \log f(\mathbf{y}) d\mathbf{y}. \tag{3.17}$$

Uma característica fundamental da entropia na Teoria da Informação é que variáveis aleatórias gaussianas possuem a maior entropia dentre todas com mesma covariância. Devido a esse fato, a entropia diferencial pode ser utilizada como medida de não-gaussianidade.

A entropia diferencial pode ser ainda modificada para funcionar de forma que variáveis gaussianas tenham entropia diferencial igual a zero, ao passo que todos os outros tipos de distribuição possuam entropia maior que zero. Essa adaptação é conhecida como negentropia, em que o prefixo "neg" refere-se ao fato de a negentropia fornecer somente valores não-negativos. A negentropia J de $\mathbf{y}$ é definida como

$$J(\mathbf{y}) = H(\mathbf{y}_{\text{gauss}}) - H(\mathbf{y}), \tag{3.18}$$

onde $H(\mathbf{y}_{\text{gauss}})$ é a entropia diferencial de uma variável com distribuição gaussiana, $\mathbf{y}_{\text{gauss}}$, com a mesma matriz de covariância de $\mathbf{y}$. Dessa forma, a negentropia de $\mathbf{y}$ será sempre nula caso este possua distribuição gaussiana, ou maior que zero caso este possua um outro tipo de distribuição.

Apesar de ser considerada uma medida ideal de não-gaussianidade, a negentropia é difícil de ser computada pela sua definição. Sendo assim, pode ser usado um tipo de aproximação para o seu cálculo, definido como

$$J(\mathbf{y}) \propto [E[G(\mathbf{y})] - E[G(\nu)]]^2, \tag{3.19}$$

onde ν é um vetor de variáveis gaussianas de média zero e variância unitária, G é uma função não-quadrática, e assume-se que $\mathbf{y}$ já esteja centralizado e branqueado. Essa aproximação mantém a propriedade de anular a negentropia caso a distribuição de $\mathbf{y}$ seja gaussiana, e resultar em valores maiores que zero em caso contrário [12].

3.4.8 Outros aspectos da ICA

A ICA é um método de separação de fontes com inúmeras aplicações que se estendem para muito além da área de áudio: um exemplo é a redução de ruído em imagens.

Existem múltiplas implementações próprias para a ICA. Por exemplo, a maximização da não-gaussianidade não é a única forma de se realizá-la. Métodos como a "minimização da informação mútua" e a "estimativa da máxima verossimilhança" também são utilizados [7], porém não foram abordados neste trabalho.

Dentro do próprio uso de maximização da não-gaussianidade para realização da ICA, a negentropia, por sua vez, também não é a única medida utilizada. A curtose, que é uma estatística de quarta ordem, também pode ser usada como medida de não-gaussianidade, porém não tão robusta quanto a negentropia [7].

3.4.9 FastICA, um algoritmo de ICA

As seções anteriores limitaram-se a discutir qual é o objetivo da ICA na BSS (encontrar a matriz $\mathbf{W}$ que desfaz a mistura das fontes), qual princípio será utilizado para atingir este objetivo (maximização da não-gaussianidade) e qual a medida que sustentará este princípio (negentropia). Contudo, falta a discussão de como a matriz $\mathbf{W}$ é encontrada. Assim, será apresentado agora o funcionamento do FastICA [12], um algoritmo de ICA muito conhecido na literatura e que foi utilizado neste trabalho. O nome é devido ao fato de este algoritmo convergir rapidamente, mas essa discussão será omitida.

O algoritmo da FastICA realiza previamente a centralização e o branqueamento do vetor de misturas, ou seja, assume-se que o vetor $\mathbf{x}$ já esteja com média zero, variância unitária e seus elementos estejam descorrelacionados. A sequência do algortimo a ser mostrada é referente à estimativa de uma única componente por vez, sendo assim chamada de aproximação por deflação.

Pode-se reescrever o modelo apresentado na equação (3.10) da seguinte maneira:

$$\hat{\mathbf{s}}_l = \sum_{i=1}^{n} \mathbf{w}_i^T \mathbf{x}, \tag{3.20}$$

onde $\mathbf{w}_i^T$ é cada vetor-linha da matriz $\mathbf{W}$. Para estimar a l-ésima componente $\hat{\mathbf{s}}_l$, deve-se maximizar a não-gaussianidade do termo $\mathbf{w}_i^T\mathbf{x}$ correspondente. Portanto, a negentropia é medida sobre $\mathbf{w}_i^T\mathbf{x}$, ou seja, faz-se $J(\mathbf{w}_i^T\mathbf{x})$.

A ideia do algoritmo é encontrar a direção, expressa por um vetor unitário[7] $\mathbf{w}_i$, que maximize a não-gaussianidade da projeção $\mathbf{w}_i^T\mathbf{x}$, medida pela aproximação da negentropia mostrada na equação (3.19). Um esquema iterativo para esse fim é descrito em [7].

A variância de $\mathbf{w}_i^T\mathbf{x}$ deve ser igual à unidade. Como se assume que o vetor $\mathbf{x}$ já passou pelo processo de branqueamento, basta restringir a norma de $\mathbf{w}_i$ à unidade. O valor ótimo de $E[G(\mathbf{w}_i^T\mathbf{x})]$ (que anula a negentropia) sob a restrição $E[(\mathbf{w}_i^T\mathbf{x})^2] = 1$

[7]Vetor unitário é aquele que possui norma quadrática igual a 1.

é obtido através de uma sequência de métodos algébricos que não serão elucidados aqui. Basta que se entenda que a ideia é encontrar valores de $\mathbf{w}_i^T\mathbf{x}$ (um de cada vez) que serão iguais, cada um, a uma componente independente diferente.

A convergência de $\mathbf{w}_i$ significa que o novo e o antigo valor de $\mathbf{w}_i$ estão na mesma direção. Entretanto, antes de se atingir a convergência, faz-se necessário a cada iteração descorrelacionar as saídas encontradas até então, a fim de prevenir que diferentes vetores convirjam para uma mesma componente. Um modo de fazer essa descorrelação é, após a atualização de $\mathbf{w}_i$, subtrair dele os termos $\mathbf{w}_j\mathbf{w}_j^T\mathbf{w}_i$ referentes aos passos anteriores, onde $j = 1, ..., i-1$. Logo após, deve-se normalizar $\mathbf{w}_i$, para preservar sua norma unitária.

Considerando g a derivada da função não-linear G da equação (3.19), a sequência de operações do FastICA é apresentada abaixo de forma sintética:

Passo (1): Centralizar e branquear os dados.
Passo (2): Escolher do número n de componentes a serem estimadas. Inicializar o contador (i = 1).
Passo (3): Inicializar $\mathbf{w}_i$ com um valor aleatório.
Passo (4): Atualizar $\mathbf{w}_i$: $\mathbf{w}_i = E[\mathbf{x}g(\mathbf{w}_i^T\mathbf{x})] - E[\dot{g}(\mathbf{w}_i^T\mathbf{x})]\mathbf{w}_i$, sendo $\dot{g}$ a derivada de g e, consequentemente, derivada segunda de G.
Passo (5): Descorrelacionar $\mathbf{w}_i$: $\mathbf{w}_i = \mathbf{w}_i - \sum_{j=1}^{i-1} \mathbf{w}_j\mathbf{w}_j^T\mathbf{w}_i$.
Passo (6): Normalizar $\mathbf{w}_i$: $\mathbf{w}_i = \frac{\mathbf{w}_i}{\|\mathbf{w}_i\|}$.
Passo (7): Verificar convergência de $\mathbf{w}_i$. Se não convergiu, voltar para o passo 4.
Passo (8): Finalizar a estimativa da atual componente e passar para a próxima: $i = i + 1$.
Passo (9): Se $i \leq n$, retornar para o passo 3.

O passo 4 desse algoritmo mostra a fórmula de atualização de $\mathbf{w}_i$ que é obtida através da otimização de $E[G(\mathbf{w}_i^T\mathbf{x})]$ sob a restrição $E[(\mathbf{w}_i^T\mathbf{x})^2] = 1$.

3.4.10 Limitação da ICA

Apesar de a ICA ser um método atraente por retornar a estimativa das componentes independentes a partir de uma mistura de fontes sonoras (além de outros tipos de misturas), sua aplicação se torna pouco prática em algumas situações, como por exemplo, no próprio problema do coquetel. A exigência de haver um número de sensores no mínimo igual ao número de fontes é praticamente impossível de ser atendida quando existem várias pessoas falando ao mesmo tempo.

Devido a esse fato, faz-se necessário o desenvolvimento de uma nova abordagem, em que poucas misturas ou até mesmo somente uma mistura seja(m) suficiente(s) para a obtenção das fontes originais. Na seção a seguir, será abordada em detalhes uma tal técnica, a ISA, que é a alma deste trabalho.

3.5 Análise de Subespaços Independentes

3.5.1 Introdução

A Análise de Supespaços Independentes (ISA) surge como uma forma de contornar a limitação da ICA quanto a o número de sensores dever ser obrigatoriamente maior que ou igual ao número de fontes. A proposta da ISA é utilizar, em vez de muitas misturas onde a ponderação das fontes é feita de forma diferente em cada uma delas, uma única mistura, sendo capaz de extrair dela toda a informação necessária para a separação das fontes.

Uma outra questão relevante é o fato de nem sempre ser possível obter-se componentes independentes utilizando a ICA, o que sugere uma reinterpretação dos resultados [7]: ao invés de se dizer que "a ICA retorna componentes independentes uma das outras", troca-se a afirmação para "a ICA retorna componentes maximamente independentes uma das outras"; em outras palavras, em vez de se falar em independência, fala-se em redução de dependência.

Uma forma diferente de contornar esse problema é descartar o próprio pressuposto de independência entre as componentes. E é isso exatamente o que a ISA faz:

as fontes que compõem determinada mistura não precisam ser independentes. A independência agora é uma característica atribuída a *subespaços* de representação, que serão explicados na próxima subseção.

Apesar de a ISA ser um método alternativo ao anterior para a separação de fontes, não prescinde do uso da ICA. Na realidade, a técnica da separação continua sendo a ICA, apenas aplicada aos subespaços já mencionados, conforme será visto em maiores detalhes mais adiante. E, devido ao fato de relaxar o pressuposto de independência entre as fontes e de acabar com a restrição quanto ao número mínimo de sensores, pode-se entender a ISA como uma generalização da ICA.

3.5.2 O que é um subespaço na ISA?

Inicialmente, é importante redefinir um conceito. Na discussão sobre a ICA, da seção anterior, ficou implícito que "componente" era o mesmo que "fonte de sinal", ou seja, retornar as componentes era o mesmo que retornar as fontes que compunham a mistura em questão. Entretanto, na ISA, o termo "componente" significa apenas aquilo que é gerado ao fim da etapa de ICA, não sendo a fonte propriamente dita.

Enquanto o objetivo da ICA é retornar componentes independentes[8], a ISA deve entregar subespaços independentes como resultado. Um subespaço aqui é um conjunto gerado por determinado número de componentes retornadas pela ICA, o qual se espera que descreva uma fonte. As componentes que geram um mesmo subespaço podem até ter certa dependência entre si, porém as componentes que geram subespaços diferentes devem ser independentes. Por isso diz-se que os subespaços são independentes.

Em outras palavras, componentes resultantes da ICA podem ser agrupadas de forma que cada grupo gere subespaços independentes. Adiante, cada subespaço é processado de forma a recompor o sinal de uma fonte que compõe determinada mistura. Logo, ao final da ISA, tem-se as fontes independentes separadas. E tudo isso a partir de uma única mistura dada como entrada. A Figura 3.1 demonstra essa explicação com um exemplo.

[8]Ou maximamente independentes, conforme visto na subseção anterior.

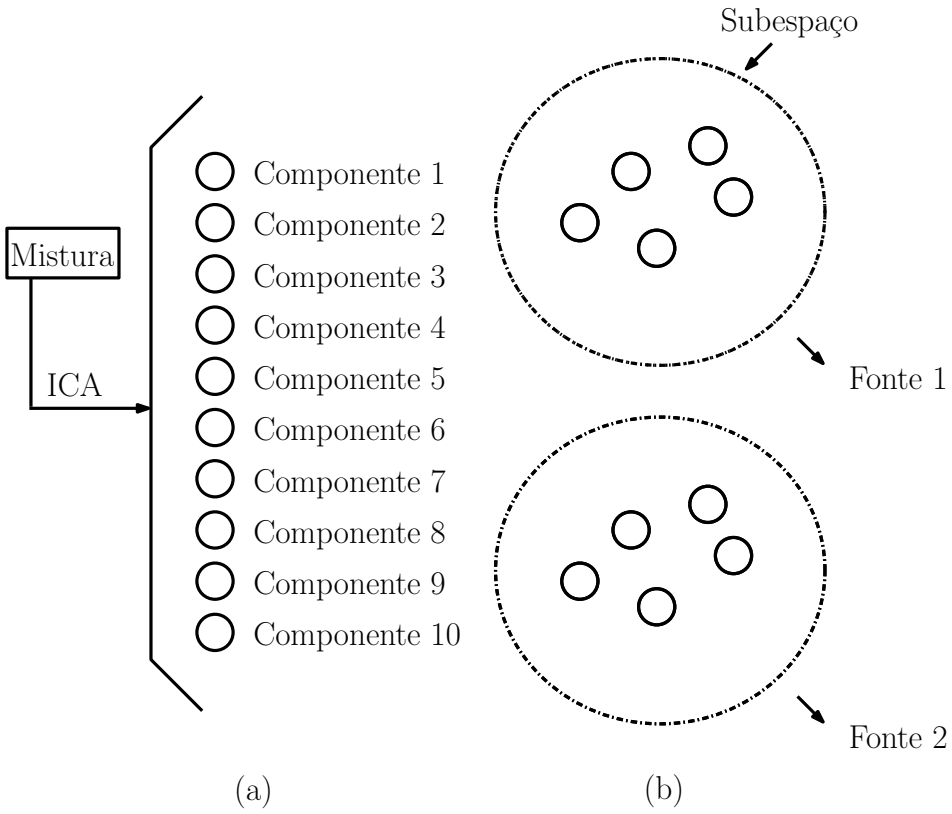

Figura 3.1: Conceito de subespaço na ISA. (a) Supondo que a ICA retorne 10 componentes que caracterizam a mistura dada, (b) essas componentes podem ser agrupadas em dois grupos com cinco componentes cada, por exemplo, gerando, cada um desses grupos, os chamados subespaços. Esses dois subespaços descrevem, cada um, as fontes de sinal que estão contidas na mistura. Como são dois subespaços neste exemplo, então são duas fontes que a serem obtidas.

Cabe aqui uma pergunta a ser respondida: se agora, na discussão sobre ISA, diz-se que as componentes retornadas não são mais equivalentes às fontes de sinal contidas na mistura, o que são, então, essas componentes? É o que será visto a seguir.

3.5.3 O espectrograma na ISA

No Capítulo 2 foi discutida a utilidade de se dividir um sinal em blocos, fazendo-se uso de um janelamento adequado, e transformá-lo para o domínio da frequência, resultando em uma representação tempo-frequencial que é chamada de espectrograma[9].

Como o espectrograma descreve as variações de frequência do sinal ao longo do tempo, pode-se entendê-lo como sendo formado por combinações de um conjunto representativo de funções na frequência e funções no tempo, ambas chamadas de *funções de base* [11], ou vetores de base. Uma função de base frequencial é um padrão frequencial que descreve o sinal em determinado instante (na verdade, determinado intervalo) de tempo, enquanto uma função de base temporal é um padrão temporal que descreve o sinal para uma determinada (na verdade, determinada faixa de) frequência.

A ISA faz uso dessa representação; contudo, em vez de a ICA ser processada sobre as amostras do sinal de mistura no domínio do tempo, ela é feita sobre as componentes do espectrograma que descreve este sinal. Ao final da ICA, têm-se as funções de base independentes, que, combinadas, geram o espectrograma da mistura. A ideia é que essas funções de base independentes descrevam, cada uma, uma característica relativa a somente uma das fontes que compõem a mistura.

Na ISA, as funções de base de frequência podem ser as componentes independentes após a separação por ICA e que, ao serem agrupadas, formam um subespaço. Com as funções de base de frequência separadas e devidamente agrupadas em subespaços, pode-se obter os ponderadores temporais correspondentes para cada grupo, resul-

[9]O processo de janelamento do sinal no domínio do tempo e transformação deste para o domínio da frequência resulta na STFT do sinal. A definição de espectrograma do sinal aqui adotada foi o módulo de sua STFT.

tando assim nos espectrogramas de cada subespaço independente, que descrevem—por sua vez—as características tempo-frequenciais de cada fonte.

Pode-se também aplicar a separação sobre as componentes de tempo, resultando em funções de base de tempo independentes e, após isso, encontrar os ponderadores frequenciais correspondentes.

A forma como esses vetores separados são agrupados para formarem os subespaços e o processamento aplicado sobre cada espectrograma independente com o objetivo de se obter as fontes de sinal são assuntos discutidos na Seção 3.6 e no Capítulo 4, respectivamente.

3.5.4 Modelo da ISA

A ISA é um método de separação que foi desenvolvido especificamente para aplicações em sinais de áudio, mas suas ideias podem ser aplicadas em outras áreas do conhecimento, como no processamento de imagens [14]. A ideia é caracterizar qualquer tipo de som cotidiano através da representação tempo-frequencial.

Pode-se dizer que a ISA baseia-se no conceito de redução de redundância da representação tempo-frequencial do sinal de mistura [9] [10] [11] [14], sendo então as fontes de sinal representadas por subespaços de baixa dimensão. Essa redução de redundância (e de dimensão) é feita através da PCA, cujo conceito foi discutido na Seção 3.3, e que será vista de forma sistemática mais adiante, para o caso específico da ISA. Por enquanto, foca-se no modelo da ISA, que está baseado nos conceitos de função de base, componente e subespaço discutidos anteriormente.

Como qualquer outro modelo, a ISA precisa assumir algumas características do sinal a ser separado. A primeira delas é o sinal de mistura $x(t)$, composto pela soma de sinais gerados por um número n fontes independentes $s_i(t)$, ou seja,

$$x(t) = \sum_{i=1}^{n} s_i(t). \tag{3.21}$$

Nota-se que, comparado ao modelo de mistura da ICA, este modelo não realiza ponderação por coeficientes, modelando a disposição dos sensores no ambiente, pois o sinal de mistura é um só (em inglês, *single-channel*), ou seja, a captação da mistura é feita por apenas um sensor.

Aplicando-se a Transformada de Fourier de Curta-Duração (STFT), discutida no Capítulo 2, sobre o sinal de mistura $x(t)$ e tomando-se o seu módulo, obtém-se o seu espectrograma (módulo da STFT) $\mathbf{X}$. A razão pela qual se usa o módulo da STFT em vez da STFT (valores complexos) da mistura é que as informações perceptivamente relevantes do sinal que a ISA procura capturar não são observáveis quando se utilizam os valores complexos da STFT.

A dimensão de $\mathbf{X}$ é $k \times m$, onde k é o número de pontos (ou canais) de frequência e m é o número de pontos (ou intervalos) de tempo. Cada coluna de $\mathbf{X}$ é um vetor que representa o espectro de frequências para um determinado intervalo de tempo. De forma similiar, cada linha é um vetor que representa as variações no tempo de um determinado canal de frequência.

Cada trecho de tempo m de $\mathbf{X}$ pode ser expresso como uma soma ponderada de l funções de base independentes $\mathbf{z}_j \in \mathbf{R}^k$ que representam as características espectrais da mistura [1]. Esses vetores são definidos como estáticos, mas os ponderadores que os multiplicam variam no tempo:

$$\mathbf{x}^{(m)} = \sum_{j=1}^{l} y_j^{(m)} \mathbf{z}_j, \tag{3.22}$$

onde $\mathbf{x}^{(m)}$ é o vetor-coluna de $\mathbf{X}$ que representa o espectro do sinal de mistura para o instante de tempo m, e $y_j^{(m)}$ são os coeficientes para o instante de tempo m. Dessa forma, a ISA está calcada no princípio de invariância dos vetores $\mathbf{z}_j$, pois estes descrevem todos os trechos $\mathbf{x}^{(m)}$ de $\mathbf{X}$, com diferença nos pesos, estes sim variantes.

A equação (3.22) pode ser escrita de outra maneira para representar o espectrograma total da mistura $\mathbf{X}$ ao invés de somente um trecho (coluna) deste. Conforme foi discutido na subseção anterior, o espectrograma é formado por funções de

base frequenciais e temporais associadas, e a forma como essa associação é feita é assumindo-se que $\mathbf{X}$ seja descrito pela soma de produtos vetoriais entre essas funções,

$$\mathbf{X} = \sum_{j=1}^{l} \mathbf{z}_j \mathbf{y}_j^T, \tag{3.23}$$

onde $\mathbf{z}_j$ são as funções de base frequenciais e $\mathbf{y}_j$ (cujos elementos são $y_j^{(m)}$) são as funções de base temporais. Os vetores tidos como invariantes e independentes sobre os quais realizar-se-á a separação podem ser as funções de base de frequência ou as tempo. Contudo, aqui, consideram-se invariância e independência entre as funções de base de frequência $\mathbf{z}_j$, como é feito nos artigos estudados.

Cada fonte que se deseja estimar pode ser descrita por um subespaço independente $\mathbf{Z}_i$ (com $i = 1, 2, ..., n$), que é representado por uma matriz contendo determinado número l_i de vetores de base,

$$\mathbf{Z}_i = \begin{bmatrix} \mathbf{z}_1^{(i)} & \mathbf{z}_2^{(i)} & \mathbf{z}_3^{(i)} & \cdots & \mathbf{z}_{l_i}^{(i)} \end{bmatrix}, \tag{3.24}$$

onde $l_i < l$ e $\mathbf{Z}_i \subseteq \mathbf{Z}$, em que $\mathbf{Z}$ é o espaço espectral total da mistura, contendo todos os vetores de base. $\mathbf{Z}_i$ possui dimensão $k \times l_i$.

Com os vetores de base agrupados em n subspaços, pode-se representar o espectrograma total da mistura $\mathbf{X}$ como sendo formado pela soma de espectrogramas desconhecidos que são estatisticamente independentes entre si, o que é uma aproximação válida em muitos casos[10] [14]. Sendo assim,

$$\mathbf{X} = \sum_{i=1}^{n} \mathbf{X}_i, \tag{3.25}$$

onde $\mathbf{X}_i$ representa os n espectrogramas independentes. Lembrando que n é o número de fontes que compõem o sinal de mistura, diz-se que cada espectrograma independente $\mathbf{X}_i$, correpondente a um subespaço $\mathbf{Z}_i$, descreve uma fonte independente $s_i(t)$.

[10]Isso é verdadeiro quando não há superposição no tempo e na frequência entre espectrogramas subjacentes.

Os espectrogramas $\mathbf{X}_i$, por sua vez, são representados pelo produto entre o seu subespaço (composto por vetores de base frequenciais) e a matriz $\mathbf{Y}_i$ que contém os vetores temporais correspondentes, ou seja,

$$\mathbf{X}_i = \mathbf{Z}_i \mathbf{Y}_i^T. \tag{3.26}$$

Logo, tem-se:

$$\mathbf{X} = \sum_{i=1}^{n} \mathbf{Z}_i \mathbf{Y}_i^T. \tag{3.27}$$

Nota-se que as equações (3.23) e (3.27) são parecidas. Ambas denotam uma soma de produtos entre os vetores de base frequenciais e temporais, com a diferença de que: na equação (3.23), as funções frequenciais ainda não estão agrupadas, fazendo-se a soma dos l produtos entre os vetores livres; e na equação (3.27) temos o agrupamento das funções de base frequenciais em n subespaços, fazendo-se portanto a soma dos n produtos entre cada matriz de vetores frequenciais e a matriz de ponderadores temporais correspondentes.

3.5.5 Sistema completo

Após a apresentação do modelo da ISA na subseção anterior, é interessante mostrar a visão macro de sua implementação. A ISA pode ser definida como uma técnica que engloba uma PCA para a redução de dimensão do espectrograma que descreve a mistura (discutida na Seção 3.3), seguida da separação dos vetores de base por ICA [9] (discutida na Seção 3.4).

A ideia ao se utilizar a PCA sobre o espectrograma está em reter os l vetores de base independentes que descrevem os aspectos originais desse espectrograma, porém descartando componentes que contribuem com variância mínima. Essa etapa é necessária devido ao grande número de vetores de base que formam o espectrograma da mistura.

Como exemplo simples, toma-se um sinal no domínio do tempo com 20.000.000 de amostras, dividido em janelas de 1.000 amostras cada. Isso resulta em 20.000 instantes de tempo no espectrograma, o que corresponderia a 20.000 componentes

frequenciais. Se for considerada a superposição entre as janelas, esse valor será ainda maior, tornando o processamento pouco prático. Por isso, a PCA é fundamental não só para descorrelacionar os vetores de base antes da ICA, como também para reduzir a dimensão dos dados, retendo as componentes principais.

Com os l vetores de base principais retidos, faz-se a separação entre eles através da ICA. Conforme descrito no modelo da ISA, os vetores de base agora maximanente independentes precisam ser agrupados em subespaços independentes, de forma que cada subespaço contenha os vetores de base que descrevem características espectrais de uma mesma fonte. O agrupamento desses vetores é feito através de um processo chamado de "clusterização"[11], que será descrito na próxima seção.

Com os subespaços e, consequentemente, com os espectrogramas obtidos, faz-se necessário um processamento que transforme cada espectrograma independente em um sinal de áudio no domínio do tempo, o qual corresponde a uma fonte que compõe a mistura. Isso pode ser feito através da reconstrução de fase, discutida no próximo capítulo.

Conforme observado, a ISA apesar de ser definida como PCA seguida de ICA, necessita ainda das etapas de clusterização e reconstrução de fase para que se possa ter as fontes de sinal originais no domínio do tempo, audíveis. A Figura 3.2 mostra o diagrama em blocos do sistema completo que envolve o processamento por ISA.

3.5.6 PCA e ICA

Conforme já mencionado, a importância da PCA na ISA é reter as componentes principais, que são aquelas que contribuem de forma mais significativa para o aumento da variância do sinal de mistura, em detrimento das demais componentes.

Também é chamado de PCA o conjunto do pré-processamento composto pela centralização e branqueamento dos dados, discutidos na Subseção 3.4.6. Entretanto,

[11] O nome vem do inglês *cluster*, que, no contexto da ISA, equivale a cada subespaço que engloba determinado número de vetores similares.

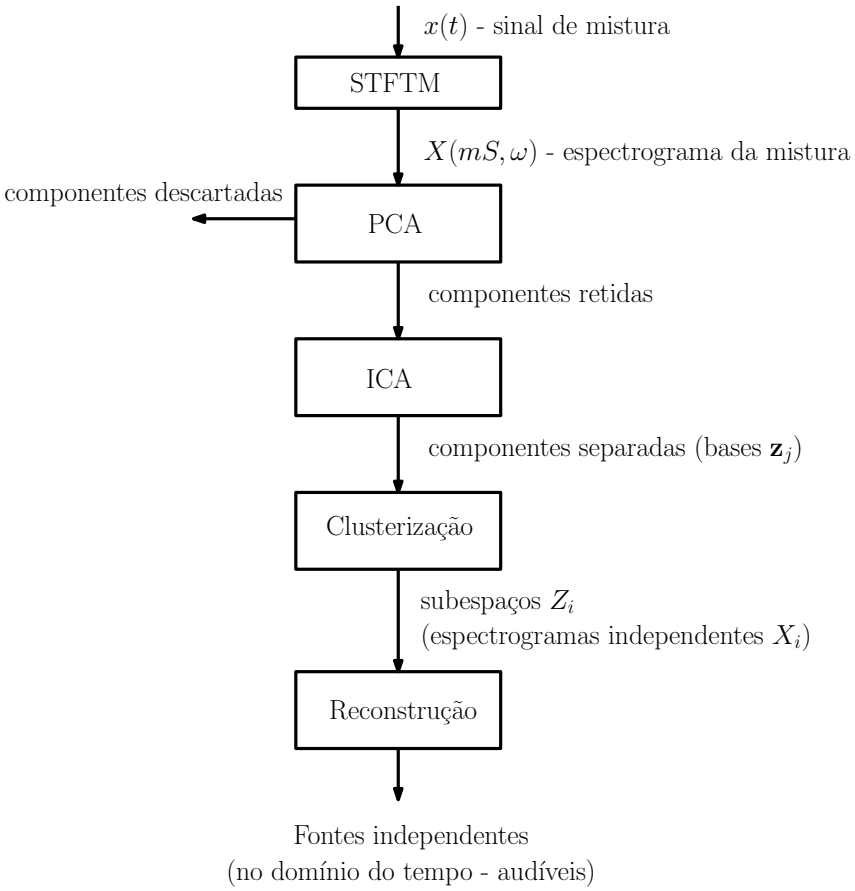

Figura 3.2: Sistema completo relativo à ISA.

a PCA que será descrita a seguir ocorre antes ainda dessas duas etapas, com a função de somente reduzir a dimensão dos dados.

Na descrição do modelo da ISA foi visto que o espectrograma total da mistura pode ser representado como a soma dos produtos entre l funções (vetores) de base frequenciais (independentes, obtidas através da ICA) e temporais. Mas de onde vêm essas l funções de base frequenciais? O valor l é justamente o número de componentes do espectrograma total da mistura $\mathbf{X}$ retidas pela PCA que darão origem a essas funções após a ICA.

A PCA pode ser realizada através da Decomposição em Valores Singulares (do inglês, *Singular Value Decomposition*, SVD) [10] [1] [14], em que $\mathbf{X}$, que é uma matriz $k \times m$, é decomposto da forma

$$\mathbf{X} = \mathbf{USV}^T, \tag{3.28}$$

onde $\mathbf{U}$ é uma matriz ortogonal $k \times k$, $\mathbf{V}$ é uma matriz ortogonal $k \times m$ e $\mathbf{S}$ é uma matriz $k \times m$ composta pelos valores singulares em sua diagonal principal. As colunas de $\mathbf{U}$ correspondem às componentes principais de $\mathbf{X}$ baseadas na frequência, enquanto as colunas de $\mathbf{V}$ correspondem às componentes principais de $\mathbf{X}$ baseadas no tempo. Aqui, considera-se que foram retidas as componentes principais de $\mathbf{U}$.

Pode-se estimar o número l de componentes a serem retidas na PCA através da soma cumulativa normalizada dos valores singulares, fazendo-se

$$\frac{\sum_{i=1}^{l} \sigma_i}{\sum_{i=1}^{n} \sigma_i} \geq \phi, \tag{3.29}$$

onde σ_i são os valores singulares da matriz $\mathbf{S}$ e ϕ é um limiar, tal que

$$0 \leq \phi \leq 1. \tag{3.30}$$

Dessa forma, o valor de ϕ controla a quantidade de informação retida pela PCA. Fazendo-se $\phi = 1$, resulta $l = k$, ou seja, todas as componentes são retidas, não havendo redução de dimensão da mistura. Isso faz com que as componentes de

frequência (colunas de **U**) obtidas—que estão em grande quantidade—correspondam a regiões limitadas do espectro de frequências.

Em contrapartida, fazendo-se $\phi << 1$, resultam poucas componentes de frequência, fazendo com que cada uma destas carregue características essenciais de alguma fonte. Deve haver um equilíbrio entre a quantidade de informação a ser retida e o reconhecimento das características das fontes, significando que é necessária a escolha cautelosa do número de componentes retidas pela PCA, a fim de se otimizar a separação.

Supondo que se deseja obter como funções de base independentes as componentes de frequência, deve-se realizar a ICA sobre as colunas de **U** de forma a encontrar uma matriz **F**, cujas colunas são as funções de base frequenciais independentes, ou seja,

$$\mathbf{F} = \mathbf{W}\mathbf{U}^T, \tag{3.31}$$

onde **W** é a matriz de desmistura, já discutida na Seção 3.4.

Com as funções de base frequenciais obtidas, pode-se então calcular os ponderadores temporais correspondentes através da multiplicação do espectrograma da mistura **X** pela pseudo-inversa[12], $\mathbf{F}_p$, da matriz de funções de base frequenciais **F**:

$$\mathbf{T} = \mathbf{F}_p\mathbf{X}. \tag{3.32}$$

As matrizes de vetores de base frequenciais e ponderadores temporais obtidas podem, agora, ser multiplicadas para formar o espectrograma da mistura **X**. O agrupamento dos vetores em subespaços é discutido na próxima seção.

3.5.7 Aspectos complementares

Conforme visto na explicação da equação (3.22), as funções de base independentes (no caso, frequenciais) que descrevem o espectrograma da mistura **X** são modeladas

[12]A pseudo-inversa $\mathbf{A}^+$ de uma matriz **A** é tal que $\mathbf{A}^+\mathbf{A} = I$, podendo ser vista como uma generalização da inversa de uma matriz, válida para matrizes não-quadradas.

como sendo invariantes. Essa invariância é devido ao fato de que as mesmas funções de base são formadoras de todos os trechos de **X**.

Na prática, isso significa que, dado um padrão espectral, não são permitidas modulações no seu *pitch*[13] ao longo do sinal de mistura [14], ou seja, a utilidade da ISA é maior para a separação de misturas em que as sonoridades dos instrumentos que as compõem não variam de *pitch*, sendo válida para a maioria dos sons de bateria. Entranto, conforme já foi dito, neste trabalho a ISA foi explorada utilizando-se misturas entre outros instrumentos também.

Apesar do apelo da ISA em se utilizar somente uma mistura para a realizar a separação das fontes, o método possui uma limitação relativa ao número de componentes, para a identificação das fontes, que devem ser retidas pela PCA. A dificuldade é a de que esse número varia dependendo da amplitude relativa entre as fontes, já que a etapa da PCA baseia-se na energia das componentes do espectrograma, retendo as de maior energia.

3.6 Clusterização para ISA

3.6.1 Introdução

Na seção anterior, foi visto em detalhes o método de Análise de Subespaços Independentes, ISA, para a separação de fontes sonoras. Discutiu-se a ideia de agrupar os vetores de base independentes em subespaços independentes que descrevem, cada um, uma fonte independente contida na mistura. Entretanto, não foi vista a forma como esse agrupamento pode ser feito. Cabe agora discutir essa etapa, chamada de clusterização.

Na clusterização, especificamente para a ISA, busca-se subdividir o espaço total **Z** (gerado por todas as funções ou vetores de base) em subespaços, cada qual abrigando os vetores mais similares entre si. Além de na ISA, a clusterização encontra várias aplicações, tais como no reconhecimento de padrões e na compressão de dados [15].

[13] *Pitch* é a altura percebida de um som, medida em Hz.

3.6.2 Definindo o problema

Cada subespaço pode ser representado por um vetor $\mathbf{c}_i$—em que $1 \leq i \leq n$ e n é o número total de subespaços—chamado de *codevector*. O conjunto formado por todos os *codevectors* é denominado *codebook*, sendo representado por $\mathbf{C}$. O agrupamento das funções $\mathbf{z}_j$—em que $1 \leq j \leq l$ e l é o número de componentes retidas pela PCA—em n subespaços (com $l > n$) é feito através de um processo iterativo cujo objetivo é encontrar os *codevectors* que minimizem uma função de custo chamada de *distorção esperada*. Esta função quantifica a diferença entre um vetor $\mathbf{z}_j$ e um *codevector* $\mathbf{c}_i$, sendo definida por [16]

$$D(\mathbf{Z}, \mathbf{C}) = \sum_{j=1}^{l} \sum_{i=1}^{n} P(\mathbf{z}_j, \mathbf{c}_i) d(\mathbf{x}_j, \mathbf{c}_i), \tag{3.33}$$

onde $\mathbf{Z}$—que pode ser entendida como uma variável vetorial aleatória—indica o espaço total que contém todos os vetores de base $\mathbf{z}_j$, $P(\mathbf{z}_j, \mathbf{c}_i)$ indica a função de probabilidade conjunta entre $\mathbf{Z}$ e o *codebook* $\mathbf{C}$—que também é uma variável vetorial aleatória—e $d(\mathbf{z}_j, \mathbf{c}_i)$ é uma medida de distorção, que pode ser simplesmente a distância euclidiana[14] entre os vetores de base e os *codevectors*.

Minimizar a distorção esperada significa tornar um conjunto de vetores $\mathbf{z}_j$ maximamente similares a um dos *codevectors* $\mathbf{c}_i$, formando assim, com ele, um subespaço do espaço total. Essa similaridade entre os vetores de base é avaliada através da probabilidade que eles possuem de pertencer a um mesmo subespaço.

Havendo um conjunto com l vetores de base e desejando-se clusterizá-los em n subespaços (para originar as n fontes independentes), o objetivo é encontrar n *codevectors* que minimizem a distorção esperada. A Figura (3.3) ilustra este problema com um exemplo para $l = 10$ e $n = 2$. É importante esclarecer que cada vetor de base relaciona-se com apenas um dos *codevectors*.

[14]Distância euclidiana é a distância mínima—reta—entre dos pontos no espaço, sendo matematicamente escrita como $d(P, Q) = \sqrt{\sum_{t=1}^{r} (p_t - q_t)^2}$, onde $P = (p_1, p_2, ..., p_r)$ e $Q = (q_1, q_2, ..., q_r)$.

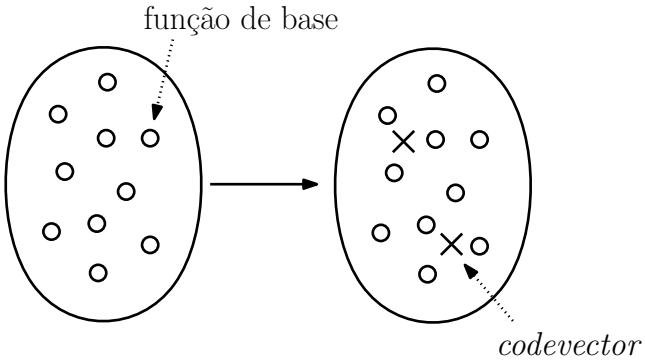

Figura 3.3: Para o caso em que se tenha 10 funções de base e se deseje clusterizá-las em 2 subespaços, deve-se encontrar 2 *codevectors*, cada um deles relacionado com algumas dessas funções de base, agrupando-as em 2 subespaços.

A equação (3.33) mostra que é feita uma varredura do espaço total $\mathbf{Z}$ a fim de se verificar a medida de distorção $d(\mathbf{x}_j, \mathbf{c}_i)$ entre cada vetor de base e cada *codevector*, combinada com a probabilidade conjunta entre todos os $\mathbf{z}_j$ e $\mathbf{c}_i$. Através da definição de Probabilidade Condicional[15] [8], pode-se reescrever essa equação da forma

$$D = \sum_{j=1}^{l} P(\mathbf{z}_j) \sum_{i=1}^{n} P(\mathbf{c}_i|\mathbf{z}_j) d(\mathbf{x}_j, \mathbf{c}_i), \tag{3.34}$$

pois

$$P(\mathbf{z}_j, \mathbf{c}_i) = P(\mathbf{c}_i|\mathbf{z}_j) P(\mathbf{z}_j), \tag{3.35}$$

onde $P(\mathbf{z}_j)$ é a probabilidade da variável $\mathbf{Z}$ e $P(\mathbf{c}_i|\mathbf{z}_j)$ é a probabilidade condicional de, dado o vetor $\mathbf{z}_j$, o *codevector* $\mathbf{c}_i$ estar associado a ele.

A distorção esperada $D(\mathbf{Z}, \mathbf{C})$ geralmente é minimizada com relação aos *codevectors* $\mathbf{c}_i$ e a probabilidade $P(\mathbf{c}_i|\mathbf{z}_j)$. Dessa forma, a melhor solução encontrada é aquela que associa o vetor $\mathbf{z}_j$ ao *codevector* $\mathbf{c}_i$ mais próximo [15] [16]. Por outro lado, o método de clusterização proposto em [1] objetiva minimizar $D(.)$—agrupando assim os vetores de base—considerando inicialmente certa aleatoriedade, medida

[15]Sendo A e B dois eventos probabilísticos, a probabilidade condicional de ocorrência de A, dado que B ocorreu, é definida por $P(A|B) = \frac{P(A \cap B)}{P(B)}$.

através da entropia conjunta $H(\mathbf{Z}, \mathbf{C})$, que é definida como [17]

$$H(\mathbf{Z}, \mathbf{C}) = -\sum_{j=1}^{l}\sum_{i=1}^{n} P(\mathbf{z}_j, \mathbf{c}_i) \log P(\mathbf{z}_j, \mathbf{C} = \mathbf{c}_i). \tag{3.36}$$

Sendo assim, faz-se a minimização da distorção esperada $D(\mathbf{Z}, \mathbf{C})$ restrita ao valor da entropia conjunta $H(\mathbf{Z}, \mathbf{C})$ por meio do método langrangeano[16]:

$$F(D, T, H) = D(\mathbf{Z}, \mathbf{C}) - TH(\mathbf{Z}, \mathbf{C}), \tag{3.37}$$

onde $F(D, T, H)$ é o lagrangeano e T é o multiplicador de Langrange. Sendo assim, a minimização de $D(\mathbf{Z}, \mathbf{C})$ é feita agora através da minimização de $F(D, T, H)$.

Quando T é um valor alto, $F(D, T, H)$ deve ser minimizado através da maximização da entropia conjunta $H(\mathbf{Z}, \mathbf{C})$. Quando T é um valor baixo, $F(D, T, H)$ deve ser minimizado através da minimização da distorção esperada $D(\mathbf{Z}, \mathbf{C})$. Quando T é um valor intermediário, a minimização de $F(D, T, H)$ baseia-se em um equilíbrio entre maximizar $H(\mathbf{Z}, \mathbf{C})$ e minimizar $D(\mathbf{Z}, \mathbf{C})$.

Esse método para minimizar a distorção esperada utilizando multiplicador de Lagrange é chamado de *Deterministic Annealing* [16]—cuja tradução livre é *Recozimento Determinístico*.

3.6.3 Por que usar o *Deterministic Annealing*?

Os métodos de clusterização tradicionais da literatura (como o k-médias[17]), podem atingir apenas um mínimo local de uma função de custo com múltiplos mínimos, que não coincide com o mínimo global desejado. O algoritmo cai nessa espécie de "armadilha", dependendo da forma como foi inicializado.

Uma analogia para o exposto no parágrafo anterior pode ser feita imaginando-se que $D(\mathbf{Z}, \mathbf{C})$ seja uma superfície acidentada, chamada de superfície de custo,

[16]Em problemas de otimização, o método lagrangeano é utilizado para encontrar os extremos de determinada função cujas variáveis estão suscetíveis a uma ou mais restrições.

[17]Mais sobre o k-médias pode ser visto em [15].

cujos buracos são os seus mínimos. Coloca-se, então, uma bola em algum ponto dessa superfície e deseja-se que ela caia no buraco mais fundo, ou seja, aquele que corresponde ao mínimo global. Entretanto, se ela cair em outro buraco (um mínimo local), dali ela não poderá mais sair.

O método *deterministic annealing* visa a contornar o problema da inicialização e, consequentemente, escapar da armadilha do mínimo local de $D(\mathbf{Z}, \mathbf{C})$. O valor de T na equação 3.37 é inicialmente muito elevado, fazendo com que o valor da entropia conjunta $H(\mathbf{Z}, \mathbf{C})$ seja mais significativo do que $D(\mathbf{Z}, \mathbf{C})$. Dessa forma, admite-se, de início, um nível de "aleatoriedade" maior no modelo. Na analogia com a superfície de custo e a bola, isso corresponde a esticar a superfície, tornando-a plana, e localizar a bola no centro dela, ou seja, a bola inicialmente não tende a rolar para nenhum local específico.

Ao diminuir-se um pouco o valor de T, diminui-se o nível de aleatoriedade e a característica acidentada da superfície de custo começa a aparecer (picos e vales começam a se formar), já que a distorção $D(\mathbf{Z}, \mathbf{C})$ sobressai um pouco mais com a redução da influência da entropia conjunta $H(\mathbf{Z}, \mathbf{C})$ na equação (3.37). Assim, com a otimização de $F(D, T, H)$, a bola se acomodará no mínimo da superfície de custo modificada. Gradualmente, diminui-se o valor de T e a superfície vai, gradativamente, tomando a forma da superfície correspondente a $D(\mathbf{Z}, \mathbf{C})$, com a diminuição da "aleatoriedade" proporcionada por $H(\mathbf{Z}, \mathbf{C})$. Logo, através da otimização de $F(D, T, H)$ para cada valor T, a bola se acomodará sempre no mínimo da superfície. Por isso o procedimento é dito como sendo determinístico: apesar do seu início possuir certa "aleatoriedade", ela é diminuída ao passo que o valor de T é reduzido.

No processo final, quando $T = 0$, retorna-se ao problema original de simplesmente minimizar $D(\mathbf{Z}, \mathbf{C})$. Entretanto, a bola já foi devidamente "encaminhada" (restringida a uma localidade) e, ao se realizar a último procedimento de otimização, ela cairá no mínimo global da superfície de custo, que, agora, com o fim da "aleatoriedade", é a superfície correspondente a $D(\mathbf{Z}, \mathbf{C})$. Dessa forma, encontra-se o mínimo global de $D(\mathbf{Z}, \mathbf{C})$, ou seja, um *codebook* $\mathbf{C}$ cujos *codevectors* otimizam o

agrupamento dos vetores de base $\mathbf{z}_j$. Não há prova de convergência do *deterministic annealing* para o mínimo global.

No *deterministic annealing*, costuma-se chamar o multiplicador de Lagrange de "temperatura" e representá-lo por T (como foi feito neste trabalho), pois este método derivou-se de outro, chamado *simulated anneling*, por sua vez desenvolvido para a física estatística, em que diminui-se gradativamente a temperatura de um sistema para se otimizar determinada função de custo. O termo *annealing* (recozimento, em português) deve-se à analogia com o procedimento de redução da dureza do aço através do seu resfriamento lento.

3.6.4 Desenvolvimento

Da Teoria da Informação, sabe-se que a entropia conjunta $H(\mathbf{Z}, \mathbf{C})$ pode ser decomposta da seguinte forma [17]:

$$H(\mathbf{Z}, \mathbf{C}) = H(\mathbf{Z}) + H(\mathbf{C}|\mathbf{Z}), \tag{3.38}$$

onde $H(\mathbf{Z})$ é a entropia relativa aos vetores de base e $H(\mathbf{C}|\mathbf{Z})$ é a entropia condicional dos *codevectors* $\mathbf{c}_i$ dado $\mathbf{z}_j$. A entropia $H(\mathbf{Z})$ não depende do procedimento de clusterização, sendo um valor constante. Portanto, deve-se focar o desenvolvimento do método na entropia condicional $H(\mathbf{C}|\mathbf{Z})$, ignorando $H(\mathbf{Z})$. Inserindo-se as definições da entropia conjunta decomposta $H(\mathbf{Z}, \mathbf{C})$ e da distorção esperada $D(\mathbf{Z}, \mathbf{C})$ na equação (3.37), obtém-se

$$F = \sum_{j=1}^{l} P(\mathbf{z}_j) \sum_{i=1}^{n} P(\mathbf{c}_i|\mathbf{z}_j) d(\mathbf{x}_j, \mathbf{c}_i) - T \left[H(\mathbf{Z}) - \sum_{j=1}^{l} P(\mathbf{z}_j) \sum_{i=1}^{n} P(\mathbf{c}_i|\mathbf{z}_j) \log P(\mathbf{c}_i|\mathbf{z}_j) \right]. \tag{3.39}$$

Minimizando-se o lagrangeano $F(D, T, H)$ em relação à probabilidade condicional $P(\mathbf{c}_i|\mathbf{z}_j)$, obtém-se que para este valor ser ótimo, deve ser descrito por uma distribuição de Gibbs [16], que possui a forma

$$P(\mathbf{c}_i|\mathbf{z}_j) = \frac{1}{K_N} e^{(-\frac{d(\mathbf{z}_j, \mathbf{c}_i)}{T})}, \tag{3.40}$$

onde K_N é uma constante de normalização dada por:

$$K_N = \sum_{i=1}^{n} e^{(-\frac{d(\mathbf{z}_j, \mathbf{c}_i)}{T})}. \tag{3.41}$$

Observa-se que quando T tende ao infinito, $P(\mathbf{c}_i|\mathbf{z}_j)$ tende a ter distribuição uniforme. Isso significa que todos os vetores $\mathbf{z}_j$ estão igualmente associados (com probabilidade $\frac{1}{l}$) a todos os *codevectors*, justicando a explicação anterior de "aleatoriedade" presente no agrupamento para este caso. Por outro lado, quando T se aproxima de zero, $P(\mathbf{c}_i|\mathbf{z}_j)$ tende a seguir uma distribuição impulsiva, em que cada vetor $\mathbf{z}_j$ é associado com probalidade 1 a somente um dos *codevectors*, o mais próximo espacialmente.

No algoritmo, T deve ser inicializado com um valor alto e ser reduzido aos poucos até o valor zero. A cada T, estima-se iterativamente os *codevectors* $\mathbf{c}_i$ de forma que eles possuam distribuição de Gibbs, minimizando-se assim o lagrangeano $F(D, T, H)$ e, consequentemente, a distorção esperada $D(\mathbf{Z}, \mathbf{C})$.

Para atualizar os *codevectors* em cada iteração, toma-se a derivada de $D(\mathbf{Z}, \mathbf{C})$ da equação (3.34) em relação a cada *codevector* c_i e se a iguala a zero:

$$\sum_{j=1}^{l} P(\mathbf{z}_j) P(\mathbf{c}_i|\mathbf{z}_j) \frac{d}{d\mathbf{c}_i} d(\mathbf{z}_j, \mathbf{c}_i) = 0. \tag{3.42}$$

Nota-se que o somatório $\sum_{i=1}^{n}$ desaparece, pois, ao derivar a equação (3.34), o único termo deste somatório que não é zero é o termo que envolve o *codevector* de índice i.

Conforme já dito, a probabilidade $P(\mathbf{z}_j)$ não depende da clusterização. O seu valor é constante e igual a $\frac{1}{l}$ (lembrando que l é o número de funções ou vetores de base que foram retidas pela PCA). Então:

$$\frac{1}{l} \sum_{j=1}^{l} P(\mathbf{c}_i|\mathbf{z}_j) \frac{d}{d\mathbf{c}_i} d(\mathbf{z}_j, \mathbf{c}_i) = 0. \tag{3.43}$$

Considerando a medida de distorção $d(\mathbf{z}_j, \mathbf{c}_i)$ como a distância euclidiana, encontra-se a fórmula de atualização dos *codevectors*, dada por:

$$\mathbf{c}_i = \sum_{j=1}^{l} P(\mathbf{z}_j|\mathbf{c}_i)\mathbf{z}_j. \tag{3.44}$$

Entretanto, é importante lembrar que outras medidas de distorção podem ser utilizadas.

Logo, através das equações (3.40) e (3.44), tem-se a maneira de otimizar os *codevectors* $\mathbf{c}_i$ para cada valor de T. Para o último valor de T, $T = 0$, realiza-se mais uma sequência de iterações para encontrar-se os *codevectors* definitivos. Assim, cada um deles representará um *cluster*, e a eles serão associadas aos vetores $\mathbf{z}_j$ mais próximos, resultando nos subespaços que descrevem as fontes independentes.

Capítulo 4

Reconstrução de fase

4.1 Por que recuperar a fase dos sinais?

A separação de fontes sonoras por ISA é realizada através do espectrograma do sinal que contém a mistura de fontes. Para isso, busca-se separar as funções de base que compõem esse espectrograma e, então, agrupá-las em subespaços independentes formadores das fontes independentes. Como o espectrograma do sinal é calculado como o módulo de sua STFT ao longo do tempo, conforme visto na equação (2.3), pode-se notar que a fase da mistura é completamente descartada pelo algoritmo de separação.

Como a ISA opera somente sobre o módulo da STFT e negligencia a informação de fase da mistura, obtêm-se como resultado apenas os módulos das representações frequenciais dos componentes independentes. Em decorrência disso, é comum associar-se a fase da mistura aos módulos obtidos. Entretanto, apesar de tal prática ser simples, ela é rudimentar, e está longe de ser eficaz, já que a fase associada não pode ser a correta para as magnitudes em questão [14]. Para se obter os sinais separados em uma forma que permita a avaliação auditiva da separação, é necessário que a fase correspondente a cada um seja recuperada.

4.2 Estimação do sinal através do módulo de sua STFT

Conforme visto na seção anterior, a ISA devolve apenas o módulo da STFT (referido como STFTM, sendo M a inicial de *magnitude*) dos sinais separados, sendo necessário associar uma fase correspondente a cada uma dessas magnitudes. Mas como obter a fase de um sinal? A solução para esse problema está em realizar uma estimativa de fase através da própria STFTM do sinal. Chama-se aqui de $|Y(mS,\omega)|$ a STFTM do sinal cuja fase se deseja obter, sendo m, S e ω os mesmos parâmetros (índice do segmento, passo de deslocamento da janela e variável de frequência, respectivamente) discutidos no Capítulo 2.

O procedimento adotado em [18] visa a recriar uma fase adequada para ser associada à $|Y(mS,\omega)|$, reconstruindo o sinal por completo. Para tal, reconstrói-se um sinal $x(n)$ cuja STFTM $|X(mS,\omega)|$ seja a mais próxima possível de $|Y(mS,\omega)|$, trazendo consigo a fase obtida ao longo do processo. O sinal $x(n)$ é então uma estimativa da representação temporal de $Y(mS,\omega)$. Essa estimativa pode ser feita minimizando-se, a cada iteração, o erro quadrático médio (MSE, do inglês *Mean Squared Error*) entre $|Y(mS,\omega)|$ e $|X(mS,\omega)|$. A expressão do MSE, referido como $\varepsilon[.]$, é dada por

$$\varepsilon[x(n), Y(mS,\omega)] = \sum_{m=-\infty}^{\infty} \frac{1}{2\pi} \int_{\omega=-\pi}^{\pi} |X(mS,\omega) - Y(mS,\omega)|^2 d\omega, \tag{4.1}$$

com

$$X(mS,\omega) = \sum_{n=-\infty}^{\infty} x(n) w(n - mS) e^{-j\omega n}, \tag{4.2}$$

onde $w(.)$ é a janela de análise.

Analisa-se essa equação da seguinte maneira: o termo dentro do somatório é a Tranformada de Fourier inversa do quadrado da diferença entre as duas STFTMs. Dessa forma, tem-se dentro do somatório a diferença representada no domínio do tempo em termos da variável m. Essa diferença é somada para todos os valores de m, resultando em um só número que a caracteriza. Observa-se que a notação

$\varepsilon[x(n), Y(mS, \omega)]$ só explicita em função de que parâmetros o cálculo é realizado: $x(n)$ é a saída e $Y(mS, \omega)$ é a entrada.

Através do Teorema de Parseval[1], a equação (4.1) pode ser escrita da forma

$$\varepsilon[x(n), Y(mS, \omega)] = \sum_{m=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} [x(mS, l) - y(mS, l)]^2, \tag{4.3}$$

onde $x(mS, l)$ e $y(mS, l)$ são as transformadas inversas de $X(mS, \omega)$ e $Y(mS, \omega)$ respectivamente, e no somatório interno, l é a variável temporal.

Para minimizar a equação (4.3), pode-se tomar a derivada parcial de $\varepsilon[.]$ em relação a $x(n)$ e igualar o seu valor a zero (ver Apêndice C), gerando o resultado

$$x(n) = \frac{\sum_{m=-\infty}^{\infty} y(mS, n) w(n - mS)}{\sum_{m=-\infty}^{\infty} w^2(n - mS)}, \tag{4.4}$$

onde $w(.)$ é a janela de análise, e n é a nova variável temporal.

4.3 Algoritmo de Griffin & Lim (G&L)

4.3.1 Introdução

O método de G&L para a reconstrução do sinal baseia-se na equação (4.4). Funciona de forma iterativa, utilizando a fórmula de atualização dada por

$$x^{i+1}(n) = \frac{\sum_{m=-\infty}^{\infty} w(n - mS) \frac{1}{2\pi} \int_{\omega=-\pi}^{\pi} \hat{X}^i(mS, \omega) e^{-j\omega n} d\omega}{\sum_{m=-\infty}^{\infty} w^2(n - mS)}, \tag{4.5}$$

com

$$\hat{X}^i(mS, \omega) = X^i(mS, \omega) \frac{|Y(mS, \omega)|}{|X^i(mS, \omega)|}, \tag{4.6}$$

[1]O Teorema de Parseval diz que a soma (ou a integral) do quadrado de uma função é igual à soma (ou a integral) do quadrado de sua transformada.

onde $x^{i+1}(n)$ é a estimativa $x(n)$ na iteração $i+1$ do laço de repetição, que deve ser obtida a partir da STFT da estimativa imediatamente anterior, denominada $X^i(mS,\omega)$.

A equação (4.6) mostra que a fase de $x^i(n)$ (calculada na iteração anterior) é associada ao módulo de $Y(mS,\omega)$ para gerar o sinal intermediário $\hat{X}^i(mS,\omega)$. Pode-se analisar essa associação escrevendo $X^i(mS,\omega)$ na equação (4.6) na forma polar. Assim, seu módulo se cancelará com o denominador, restando apenas sua fase, que, multiplicada com $|Y(mS,\omega)|$, resulta na STFT $\hat{X}^i(mS,\omega)$. Então, a equação (4.6) também pode ser escrita da forma

$$\hat{X}^i(mS,\omega) = |Y(mS,\omega)|e^{j\angle(X^i(mS,\omega))}, \tag{4.7}$$

onde $\angle(X^i(mS,\omega))$ se refere à fase de $X^i(mS,\omega)$ atribuída para $\hat{X}^i(mS,\omega)$. Por fim, inserindo a equação (4.6) na equação (4.5), obtém-se o valor de $x^i(n)$ em termos de $|Y(mS,\omega)|$ e da estimativa $x(n)$ da iteração imediatamente anterior. Matematicamente:

$$x^{i+1}(n) = \frac{\sum_{m=-\infty}^{\infty} w(n-mS)\frac{1}{2\pi}\int_{\omega=-\pi}^{\pi} X^i(mS,\omega)\frac{|Y(mS,\omega)|}{|X^i(mS,\omega)|}e^{-j\omega n}d\omega}{\sum_{m=-\infty}^{\infty} w^2(n-mS)}. \tag{4.8}$$

Vale lembrar que, neste trabalho, $|Y(mS,\omega)|$ é a STFTM de uma fonte independente, estimada por ISA, para a qual se deseja estimar a fase, e $x(n)$, o sinal estimado.

4.3.2 Descrição

O algoritmo de G&L pode ser melhor compreendido através do diagrama em blocos da Figura 4.1. O sinal $x(n)$ pode ser inicializado com fase nula, aleatória ou até mesmo com a própria fase da mistura. Na primeira iteração do laço, essa fase inicial é, então, associada à STFTM $|Y(mS,\omega)|$ (conhecida), da forma mostrada na equação (4.6), para que se obtenha $\hat{X}^i(mS,\omega)$. Sobre $\hat{X}^i(mS,\omega)$, realiza-se a Transformada de Fourier inversa. Esses dois últimos procedimentos são conhecidos juntos pelo termo *magnitude spectrum constrained transform* (ou somente *M-constrained*

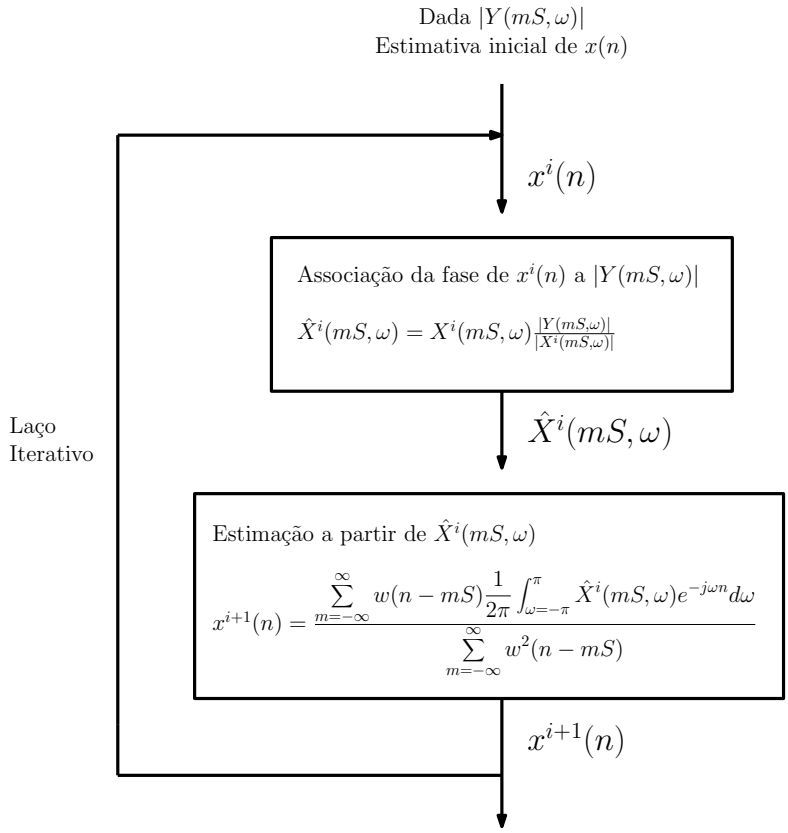

Figura 4.1: Diagrama em blocos do algoritmo de Griffin-Lim. Figura baseada em [18].

transform) [19], o que, em tradução livre, corresponde a *transformação restrita ao espectro de magnitude*. Esse procedimento gera a representação de $\hat{X}^i(mS,\omega)$ no domínio do tempo, que pode ser chamada de $\hat{x}^i(n)$, e que é equivalente ao sinal $y(mS,n)$ da equação (4.4). Com isso, pode-se obter a próxima estimativa $x(n)$, através da equação (4.5). A fase calculada para $x(n)$ é, então, utilizada na *M-constrained transform* da próxima iteração. Essa sequência é repetida até que se atenda a um determinado critério de parada (que será discutido mais adiante), produzindo, assim, o sinal $x(n)$ completamente estimado.

4.3.3 Aspectos complementares

O algoritmo de G&L faz com que, a cada iteração, a distância $\varepsilon[.]$ (equação (4.1)) seja menor. Trata-se de uma convergência decrescente e monotônica [18]. Devido ao fato de minimizar a distância (ou diferença) $\varepsilon[.]$, o algoritmo de G&L é também referido como *Least Square Error Estimation* (LSEE), ou Estimação por Erro Quadrático Mínimo.

Em geral, adota-se como critério para parada do algoritmo um número máximo de iterações.

4.4 Real-Time Iterative Spectrogram Inversion (RTISI)

4.4.1 Introdução

O algoritmo de G&L, da forma como foi apresentado na seção anterior, poderia ser aplicado, por exemplo, em uma comunicação na presença de ruído (onde o ruído é separado da informação relevante), desde que esta não precise ser feita em tempo real. Isso se deve ao fato de que a sequência mostrada na Figura (4.1) realiza a estimativa de todas as amostras do sinal a cada iteração, ou seja, faz uso de todos os *frames*[2] da STFTM em cada iteração. Logo, é necessário que se tenha a STFTM completa para que, após, esta possa ter sua fase estimada, e somente então o sinal $x(n)$ ser enviado.

Mediante o exposto, conclui-se que o algoritmo de G&L não é adequado quando se deseja separar o ruído da informação principal em aplicações em tempo real: é preciso modificar o algoritmo para abranger tais aplicações. Um novo método também baseado no LSEE e denominado *Real-Time Iterative Spectrogram Inversion* (RTISI—que em tradução livre significa *Inversão Iterativa do Espectrograma em Tempo Real*) se propõe a resolver isso [20].

No RTISI, um só *frame* é processado de cada vez, tendo sua fase estimada ao fim do processo, através do LSEE. Dessa forma, as equações mostradas na Seção 4.3 são as mesmas, com a diferença de que agora operam em tempo real. Os *frames* do sinal que já foram reconstruídos são sobrepostos e somados (*overlap-added*) ao *frame* atual para que se obtenha uma estimativa inicial de fase para este (o primeiro *frame* do sinal pode ser inicializado através de uma das formas já foram discutidas). Devido às características de estimativa *frame* a *frame* (não utilizando o sinal inteiro em cada

[2] *Frame* (ou quadro) neste contexto é um trecho com duração limitada de um sinal.

iteração) e ao uso de informação prévia, o RTISI é mais veloz computacionalmente se comparado ao G&L.

4.4.2 Descrição do método

Supõe-se que já foram reconstruídos os $m-1$ primeiros *frames*. Denota-se esse sinal parcial por $x_{m-1}(n)$. Analisa-se, agora, a reconstrução do *frame* m. Para a sobreposição das janelas, discutida no Capítulo 2, adota-se $L = 4S$, ou seja, sobreposição de 75%. Com isso, o m-ésimo *frame* é formado através do *overlap-add* dos *frames* $m-1$, $m-2$ e $m-3$, conforme mostra a Figura 4.2. Para distinguir somente o *frame* m parcialmente reconstruído do sinal x_{m-1}, denota-se o primeiro por $x_{m-1}(n)w(n-mS)$, ou seja, com a janela $w(.)$ selecionando somente o local de reconstrução.

O procedimento do RTISI está mostrado na Figura 4.3 para uma iteração relativa à reconstrução de um *frame*. Computa-se a fase inicial para o *frame* m posicionando-se a janela de análise na posição em que ele deverá se encontrar, ou seja, $x_{m-1}(n)w(n-mS)$. Isso permite que se consiga uma boa continuidade de fase ao longo da reconstrução. Após o *janelamento*, realiza-se a Transformada de Fourier e obtém-se uma estimativa de fase. Essa estimativa é associada à STFTM dada, $|Y(mS,\omega)|$, e, sobre o sinal gerado por essa associação, realiza-se a operação *M-constrained*. Assim, através da fórmula de atualização da equação (4.5), é obtida a estimativa atualizada do *frame* m.

Se o critério de parada ainda não é atingido, a estimativa atual do *frame* m é somada ao *frame* parcial $x_{m-1}(n)w(n-mS)$, produzindo x_m, que, ao ser novamente *janelado* e passar pela Transformada de Fourier, resulta em uma nova estimativa de fase para o *frame* m, dando continuidade ao processo iterativo. Quando o critério de parada é atingido, a reconstrução do *frame* m é finalizada, faz-se o *overlap-add* dos segmentos e a reconstrução do próximo *frame*, $m+1$, é inicializada. O critério de parada pode ser um dos que foram discutidos na Seção 4.3.

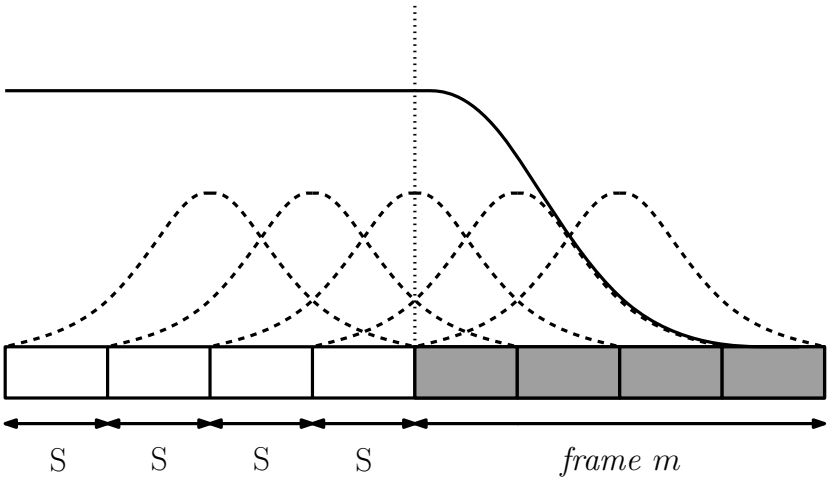

Figura 4.2: Reconstrução do *frame* m para $L = 4S$ para $L = 4S$ (75% de sobreposição). A linha sólida na parte superior da figura mostra o contorno do sinal previamente estimado no domínio do tempo. Figura baseada em [19].

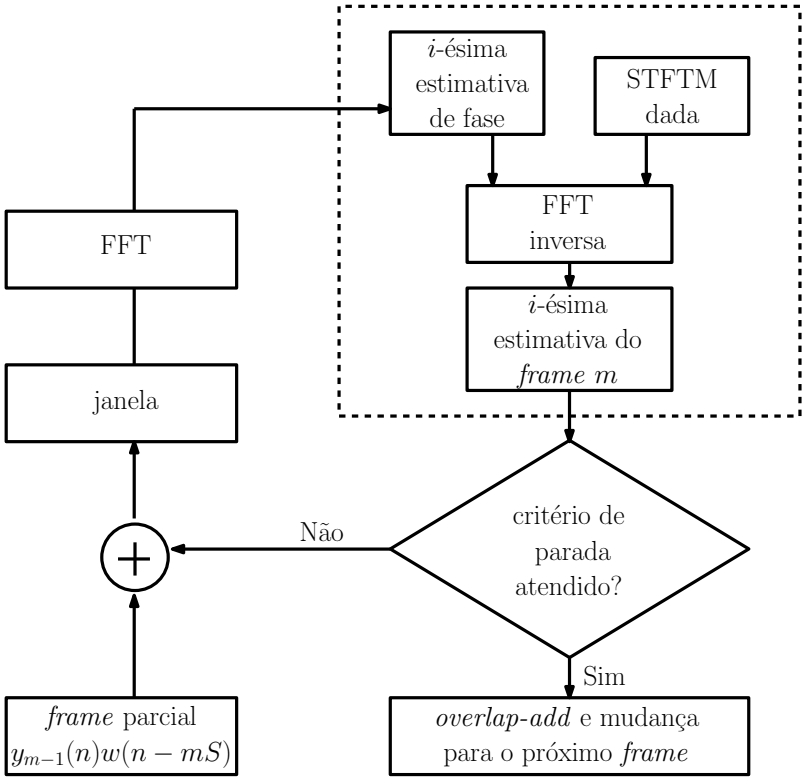

Figura 4.3: Diagrama em blocos do RTISI mostrando a reconstrução *frame* a *frame*. A linha tracejada é a operação que foi chamada de *M-constrained Transform*. Figura baseada em [20].

4.4.3 Aspectos Complementares

É importante destacar que, no caso $L = 4S$, o método RTISI utiliza somente informações dos 3 *frames* anteriores (que se sobrepõem com o atual), enquanto o G&L utiliza informações dos 3 anteriores e dos 3 posteriores, já que, nesse método, o sinal inteiro tem a sua estimativa atualizada a cada iteração. A Tabela 4.1 resume as principais diferenças entre os algoritmos de G&L e RTISI.

Tabela 4.1: *Diferença entre os algoritmos de G&L e RTISI*

G&L	**RTISI**
Não pode ser usado em aplicações em tempo real	Ideal para aplicações em tempo real
Estimativa de todos os *frames* a cada iteração	Estimativa *frame* a *frame*
Utilizada toda a informação do sinal a cada iteração	Utiliza informação de *frames* já reconstruídos
Processamento mais lento	Processamento mais rápido
Maior qualidade de reconstrução	Menor qualidade de reconstrução

O RTISI atende requisitos estruturais e computacionais da reconstrução em tempo real. O primeiro requisito diz respeito ao uso de somente informações provenientes dos *frames* anteriores e do corrente para uma reconstrução *frame* a *frame*. Já o segundo diz respeito à pouca quantidade de computação requerida para que o algoritmo seja rápido o suficiente para ser aplicável em tempo real.

4.5 Real-Time Iterative Spectrogram Inversion with Look-Ahead (RTISI-LA)

4.5.1 Introdução

Embora o algoritmo RTISI atenda aos requisitos estruturais e computacionais mencionados anteriormente, não é atendido o requisito de flexibilidade, que é: um bom algoritmo de reconstrução em tempo real deve reconstruir sinais com melhor qualidade se forem usados mais recursos computacionais. Dessa forma, o algoritmo

deve ser adaptável (flexível) para funcionar em aplicações com diferentes demandas de qualidade *versus* velocidade de processamento. No RTISI, o desempenho, em termos do erro entre as STFTMs (do sinal dado e do reconstruído), converge para uma assíntota e não é melhorado, mesmo que mais iterações sejam realizadas. Isso acontece devido ao fato de o RTISI somente utilizar informações proveninentes do *frame* corrente e dos anteriores. Visando a atender o requisito de flexibilidade, desenvolveu-se o algoritmo denominado *Real-Time Iterative Spectrogram Inversion with Look-Ahead (RTISI-LA)* [19].

O RTISI-LA é uma extensão do RTISI, em que o termo adicional *look-ahead* (olhar adiante) indica uma estratégia em que um determinado número de *frames* futuros influenciam na reconstrução do *frame* corrente. Essa estratégia não compromete a adequação do algoritmo aos requisitos estruturais e computacionais necessários para aplicações em tempo real, pois em geral somente um pequeno número de *frames* futuros são utilizados. Além do mais, a reconstrução continua a ser feita *frame* a *frame*.

Cabe aqui, agora, adotar uma denominação: no RTISI, o *frame* m torna-se "comprometido" [21] logo após ser reconstruído através do processo iterativo. Estar comprometido significa que, após o *overlap-add* com os *frames* adjacentes, o *frame* m torna-se parte estimada do sinal, estando associado às estimativas dos *frames* anteriores e formando com estes um sinal de áudio válido. No RTISI-LA, o *frame* m permanece "não-comprometido" até que o *frame* $m + K$ seja gerado, sendo K o número do *frames* do *look-ahead*.

4.5.2 Descrição do método

A Figura 4.4 mostra o conceito de comprometimento sobre o *frame* m (sombreado na Figura 4.4(a)), com $L = 4S$ (sobreposição de janelas de 75%) e $K = 3$. Utiliza-se um *buffer* (Figura 4.4(b)) para armanezar o *frame* m (corrente), os 3 anteriores (que são os que se sobrepõem a m para $L = 4S$) e os 3 futuros. Quando $K = 0$, ou seja, quando não são utilizados *frames* futuros, o RTISI-LA é idêntico ao RTISI comum.

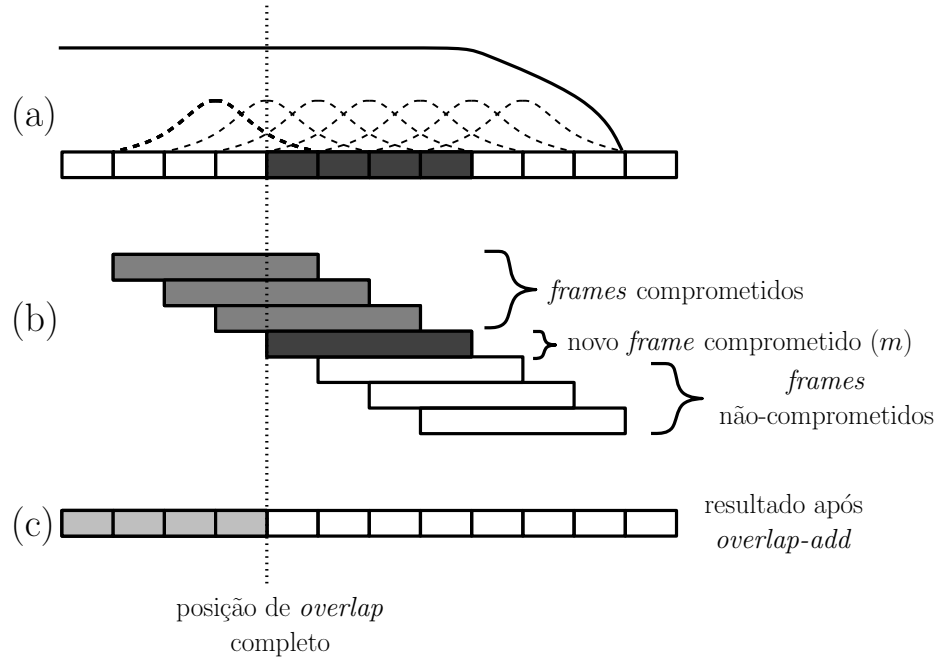

Figura 4.4: (a) Sinal reconstruído, mostrando o *frame* corrente (m, que está sombreado) e o contorno (linha sólida na parte superior). (b) Os *frames* do $m-3$ ao $m+K$ sendo processados no *buffer*. (c) Resultados após o *overlap-add*. O trecho sombreado é a saída do algoritmo já pronta para ser utilizada na aplicação. Figura baseada em [21].

Conforme foi visto, quando o *frame* m é inicialmente gerado, ele permanece não-comprometido dentro do *buffer*, e a reconstrução segue até que se atinja o *frame* $m+K$. Daí, estima-se uma fase inicial para o *frame* $m+K$ e aplica-se a *M-constrained Transform* paralelamente sobre todos os *frames* não-comprometidos do *buffer* (do *frame* m ao $m+K$), utilizando o espectro de magnitude (STFTM) de referência. Dessa forma, os *frames* contidos no *buffer* terão fases associadas a eles, constituindo-se sinais temporais de áudio. Então, é feito o *overlap-add* sobre esses sinais, resultando na Figura 4.4(c).

Contudo, o *frame* m ainda permancerá não-comprometido. Portanto, deve-se fazer o janelamento dos *frames* m ao $m+K$ sobre o resultado do *overlap-add* para que se utilizem, na próxima iteração, as informações obtidas. Logo após, aplica-se novamente a *M-constrained Transform* sobre esses *frames*, obtendo as novas estimativas desses segmentos, sobre os quais deverá ser feito novamente o *overlap-add*. Esse processo é repetido até que o critério de parada seja atendido. Após isso, o *frame* m torna-se comprometido e a posição de *overlap* completo, indicada na Figura 4.4(c),

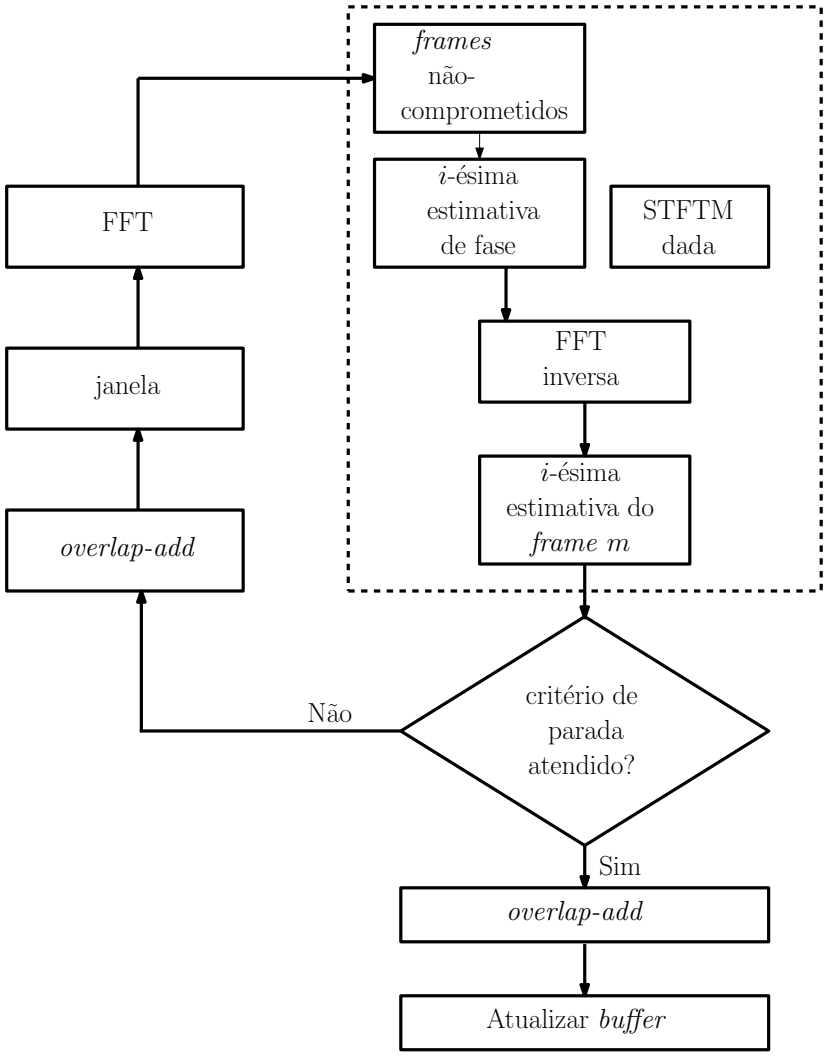

Figura 4.5: Diagrama em blocos do algoritmo RTISI-LA. O quadrado tracejado indica a operação *M-constrained Transform*. Figura baseada em [21].

é movida um passo S ($= L/4$) à frente.

A posição de *overlap* completo indica que o sinal antes dela (sombreado) é uma saída (*output*) de áudio que já pode ser utilizada na aplicação. Ao avançar um passo, deve-se retirar o *frame* $m-3$ do *buffer*, deslocar os demais para a esquerda, a adicionar o *frame* $m+K+1$, de forma a viabilizar a reconstrução do *frame* $m+1$.

Para melhor entendimento, a Figura 4.5 mostra o diagrama em blocos do algoritmo RTISI-LA. Sobre os *frames* não-comprometidos (m ao $m+K$) é aplicada a *M-constrained Transform*, resultando em segmentos no domínio do tempo cujas fases foram atualizadas. Se o critério de parada for atendido, faz-se o *overlap-add* final e S amostras do sinal são adicionadas à saída. Se não for atendido, *janela*-se o resultado para obter novamente os *frames* a aplicar sobre eles a *M-constrained Transform*.

4.5.3 Aspectos Complementares

É importante notar que no processo de reconstrução do primeiro *frame* não há *frames* no *buffer*. Dessa forma, pode-se adotar qualquer fase como fase inicial para esse segmento. O processo, então, é feito normalmente, adicionando e mantendo não-comprometidos os K primeiros *frames* no *buffer*. Ao final do processo iterativo para a reconstrução do primeiro *frame*, este torna-se comprometido, e as primeiras S amostras após o *overlap-add* tornam-se a primeira saída para a aplicação.

Tabela 4.2: *Diferença entre os algoritmos de G&L, RTISI e RTISI-LA*

G&L	**RTISI**	**RTISI-LA**
Não pode ser usado em aplicações em tempo real	Ideal para aplicações em tempo real	Ideal para aplicações em tempo real
Estimativa de todos os *frames* a cada iteração	Estimativa *frame* a *frame*	Estimativa *frame* a *frame*
Não utiliza informação prévia	Utiliza informação de *frames* já reconstruídos	Utiliza informação de *frames* já reconstruídos e de *frames* futuros
Processamento mais lento	Processamento mais rápido	Flexível. Depende do número de *frames* futuros utilizados
Maior qualidade de reconstrução	Menor qualidade de reconstrução	Flexível. Depende do número de *frames* futuros utilizados

A carga computacional do RTISI-LA é um pouco maior do que a do RTISI comum. Entretanto, como já foi dito, dependendo do número de *frames* do *look-ahead*, o requisito computacional necessário para o uso do RTISI-LA em aplicações em tempo real não é comprometido. O que deve ser levado em consideração é a flexibilidade que o RTISI-LA oferece. Em uma aplicação cuja preferência seja a qualidade da reconstrução (podendo haver certo retardo), basta serem utilizados mais *frames* para o *look-ahead*. Por outro lado, em uma aplicação cuja meta principal seja a instantaneidade da reconstrução (em detrimento de certa qualidade), basta que poucos *frames* (ou até mesmo nenhum) sejam utilizados para o *look-ahead*.

A Tabela 4.2 mostra as principais diferenças entre os algoritmos expostos neste capítulo.

4.6 Consideração final

É interessante notar que, mesmo não havendo informação de fase (nas fontes estimadas resultantes da ISA), esta informação está contida na STFTM, e só precisa ser explicitada através de algum método de reconstrução. Existe prova de biunivocidade entre magnitude e fase de um sinal real de duração finita representado na frequência [22].

Capítulo 5

Experimentos

5.1 Introdução

Neste capítulo são apresentadas as descrições dos experimentos realizados e os resultados obtidos através de cada um deles. Porém, antes disso, apresentam-se os avaliadores de qualidade para a separação e para a reconstrução de fase, e também informações sobre janelamento e tabelas de parâmetros da ISA e sinais utilizados para os testes.

5.2 Avaliação de Qualidade

5.2.1 Avaliação da separação

Para que se possa avaliar a qualidade de uma separação sob vários aspectos, é possível modelar um sinal de áudio separado $\hat{s}$ da forma [23]

$$\hat{s} = s + e_i + e_a + e_r, \tag{5.1}$$

onde s corresponde ao sinal original (que se desejaria obter), e_i é a interferência causada por outras fontes (ou seja, o resíduo de outra fonte presente na que se deseja avaliar), e_a corresponde aos artefatos gerados pelo processo de separação e e_r é o ruído (caso haja ruído na mistura).

Uma das medidas mais importantes utilizadas para avaliar quantitativamente a separação é denominada de Razão Fonte-Distorção (do inglês *Source-to-Distortion Ratio*, SDR), calculada como:

$$\mathrm{SDR} = 10 \log \frac{\|s\|^2}{\|e_i + e_a + e_r\|^2}. \tag{5.2}$$

Uma outra medida é denominada Razão Fonte-Interferência (do inglês *Source-to-Intererence Ratio*, SIR), e não considera o ruído e os artefatos gerados pelo método (caso haja). É calculada como:

$$\mathrm{SIR} = 10 \log \frac{\|s\|^2}{\|e_i\|^2}. \tag{5.3}$$

Há ainda uma terceira medida, que considera somente os artefatos gerados. É a Razão Fonte-Artefato (do inglês *Source-to-Artifact Ratio*, SAR):

$$\mathrm{SAR} = 10 \log \frac{\|s\|^2}{\|e_a\|^2}. \tag{5.4}$$

Existem também formas de avaliação da separação baseadas em Psicoacústica, em que o sinal de áudio é transformado do domínio do tempo para o domínio psicoacústico. Isso significa que se busca, através de um complexo processamento não-linear, simular a percepção sonora humana. O PEAQ (*Perceptual Evaluation of Audio Quality*) é um exemplo desse tipo de avaliação [23]. A saída desse medidor é a ODG *Objective Difference Grade*, que é uma nota dada pelo sistema ao sinal de áudio separado com base no sinal original, podendo variar de -4 (degradação muito incômoda) a 0 (degradação imperceptível).

O PEAQ trabalha com sinais amostrados em 48 kHz, ou seja, para se obter a ODG foi necessário reamostrar os sinais através da função **resample** do MATLAB©. A implementação do PEAQ utilizada está disponível em:
`http://www-mmsp.ece.mcgill.ca/Documents/Downloads/PQevalAudio/`

5.2.2 Avaliação da reconstrução de fase

Uma forma quantitativa para avaliar a reconstrução do sinal é utilizar a Razão Sinal-Ruído (do inglês *Signal-to-Error Ratio*, SER) [19] [20], dada por:

$$\mathrm{SER} = 10\log \frac{\sum_{m=-\infty}^{\infty} \frac{1}{2\pi} \int_{\omega=-\pi}^{\pi} |Y(mS,\omega)|^2 d\omega}{\sum_{m=-\infty}^{\infty} \frac{1}{2\pi} \int_{\omega=\pi}^{\pi} [|Y(mS,\omega| - |X(mS,\omega|]^2 d\omega}, \tag{5.5}$$

onde $|Y(mS,\omega)|$ é a STFTM dada e $|X(mS,\omega)|$ é a STFTM do sinal reconstruído. Observa-se que o termo presente no denominador trata-se da distância $D[.]$, descrita matematicamente na equação (4.1) do Capítulo 4. Minimizar a distância $D[.]$ é o mesmo que maximizar a SER. Dessa forma, quanto maior for a SER, melhor é a qualidade da reconstrução.

5.3 Informações prévias

5.3.1 Janelamento

Conforme visto no Capítulo 2, a janela de Hamming é muito utilizada em processamento de áudio. Ela é um caso particular de uma família de janelas na forma

$$w_h(n) = \frac{2w_r(n)}{\sqrt{4a^2 + 2b^2}} \left[a + b\cos\left(\frac{2\pi n}{L} + \frac{\pi}{L}\right)\right], \tag{5.6}$$

onde a e b são constantes reais escolhidas adequadamente, n é a variável temporal discreta, L é o tamanho da janela e $w_r(n)$ é a janela retangular, geralmente com amplitude unitária. Fazendo-se $a = 0{,}54$, $b = -0{,}46$ e $w_r(n) = \frac{\sqrt{S}}{\sqrt{L}}$, sendo S o passo de sobreposição, obtém-se valor 1 no somatório contido no denominador da equação (4.5) do Capítulo 4 [18], eliminando assim a necessidade da normalização e, consequentemente, reduzindo o volume de processamento. Esta é a janela de Hamming.

Também pode ser visto em [18] que, para a janela retangular, a propriedade do somatório unitário é atendida com $w_r(n) = \frac{\sqrt{S}}{\sqrt{L}}$.

Foi feito o janelamento de fase zero em todos os testes, pois é imprescindível que a janela não prejudique a fase do sinal a ser segmentado. Testes relativos ao parâmetros L e S foram feitos na Subseção 5.4.3.

5.3.2 Clusterização

O método de clusterização utilizado foi o *deterministic annealing*, visto no final do Capítulo 3. O parâmetro de temperatura T, inerente ao método, é definido com base nos dados de entrada pelo próprio algoritmo utilizado neste trabalho, e não foi explorado nos testes.

Em [1] e [14], menciona-se um algoritmo de clusterização por *deterministic annealing* baseado na distância de *Kullback-Leibler* como medida de distorção $d(\mathbf{z}_j, \mathbf{c}_i)$—e que pode ser encontrado em [24]. Neste projeto, por simplicidade, optou-se por utilizar neste trabalho a distância euclidiana como medida de distorção. A implementação do *deterministic annealing* com a distância euclidiana como medida de distorção é de autoria de Justin Muncaster, autor de [15].

5.3.3 Parâmetros da ISA

A Tabela 5.1 apresenta a descrição dos parâmetros mais importantes relativos à ISA.

Tabela 5.1: *Parâmetros relativos à ISA*

Representação	**Descrição**
L	Tamanho da janela.
S	Passo de sobreposição.
w	Janela utilizada.
ϕ	Parâmetro da PCA. Controla o número de componentes retidas.
l	Número de componentes retidas. É função de ϕ.
C	Parâmetro de clusterização. Número de *clusters* nas quais as funções de base devem ser agrupadas, = número de fontes.
I	Parâmetro de reconstrução. Número máximo de iterações.
K	Parâmetro de reconstrução. Máximo de frames do *look-ahead* (no caso em que utiliza-se o RTISI-LA).

5.3.4 Banco de sinais

Todos os sinais utilizados são *monoaurais*[1], possuem taxa de amostragem de 44,1 kHz[2], estão no formato *wav* e foram extraídos da base RWC [25]. As misturas entre eles foram produzidas sinteticamente. As Tabelas 5.2 e 5.3 apresentam a lista das misturas que foram utilizadas, com uma descrição para cada uma, contendo o nome que servirá para referenciar os sinais ao longo deste capítulo, sua classificação quanto ao *pitch* (*pitched*[3] - P ou *unpitched*[4] - U) e sua duração.

[1]Provenientes de um único canal.

[2]Para se utilizar o PEAQ, os resultados e as fontes originais foram reamostrados para 48 kHz, mas foram processados pela ISA com 44,1 kHz

[3]com *pitch* definido, ou tonal.

[4]sem *pitch* definido, ou não-tonal.

Tabela 5.2: *Sinais utilizados: descrição*

Nome	**Descrição**
bumbo_piano	6 batidas de bumbo de bateria misturadas com uma nota aguda de piano.
apito_prato_1	Apito misturado com 3 batidas de prato de bateria, sendo a primeira ocorrência a mais forte e a terceira a mais fraca.
apito_prato_2	Versão modificada de *apito_prato_1*, em que as batidas de prato só iniciam após o término do apito.
contrabaixo_violino	Uma nota grave de contrabaixo misturada com uma nota média de violino.
piano_gr_ag_1	Uma nota grave e uma nota aguda de piano, em que a aguda ocorre somente após o término da grave.
piano_gr_ag_2	Uma nota grave e uma nota aguda de piano, em que a aguda ocorre pouco após o início da grave, havendo sobreposição entre elas.
piano_gr_ag_3	Uma nota grave e uma nota aguda de piano, em que ambas ocorrem ao mesmo tempo.
piano_ag_1	Duas notas idênticas de piano, em que uma ocorre pouco após o início da outra, havendo sobreposição entre elas.
piano_ag_2	Duas notas idênticas de piano, em que uma ocorre somente após o término da outra.
violino_prato	Uma nota aguda de violino misturada com 3 batidas de prato de bateria (o mesmo prato da mistura com apito).
corneta_vibrafone	Uma frase melódica tocada por uma corneta sintetizada misturada com uma sequência de 3 notas iguais de vibrafone.
corneta_vozC	O mesmo sinal de corneta da mistura anterior misturado com voz masculina cantada.
corneta_ruido	O mesmo sinal de corneta das misturas anteriores misturado com um ruído branco gaussiano de baixa potência.
vozC_vozF	O mesmo sinal de voz cantada misturado com um sinal de voz falada masculina.
vibrafone_prato	2 ocorrências da mesma nota de vibrafone intercaladas com 2 batidas iguais de prato. Vibrafone inicia primeiro.

Tabela 5.3: *Sinais utilizados: natureza e duração*

Nome	**P/U**	**Duração (s)**
bateria_piano	P/U	2,27
apito_prato_1	U	3,81
contrabaixo_violino	P	3,00
piano_gr_ag_1	P	2,50
piano_gr_ag_2	P	5,00
piano_gr_ag_3	P	3,00
piano_ag_1	P	2,50
piano_ag_2	P	3,75
violino_prato	P/U	3,81
corneta_vibrafone	P	4,00
corneta_ruido	P/U	4,00
corneta_vozC	P	4,00
vozC_vozF	P	3,00
vibrafone_prato	P/U	3,00

5.4 Testes de ajuste de parâmetros

Este conjunto de testes teve por objetivo definir alguns parâmetros a serem utilizados nos testes sistemáticos sobre a ISA nas seções seguintes.

5.4.1 Parâmetros da reconstrução de fase

5.4.1.1 Objetivo

Definir o melhor método de reconstrução de fase—G&L, RTISI ou RTISI-LA—a ser adotado nos demais testes. Vários testes foram realizados para verificar o desempenho dos métodos de reconstrução, todos com resultados similares. A seguir, é mostrado um deles.

5.4.1.2 Teste de exemplo

Mostra-se, como exemplo, o sinal *bateria_piano*, por ser uma mistura que contém uma fonte tonal e uma não-tonal, buscando maior generalidade. Não foi feita a separação nesses testes. Através do MATLAB©, obteve-se a STFTM da mistura, que foi entregue como dado de entrada para os três algoritmos de reconstrução para

que eles pudessem reconstruir uma fase associada à STFTM em questão. A SER foi calculada como descrito na Subseção 5.2.2.

O método de aproximação e a não-linearidade utilizadas para o FastICA foram as padrões do algoritmo (por deflação e $g(u) = u^3$, em que $g(.)$ é a derivada de $G(.)$— ver Subseção 3.4.7 do Capítulo 3). Fixaram-se $L = 1024$ e $S = 256$, ou seja, 75% de sobreposição, conforme [21]. Para o caso do RTISI-LA, utilizaram-se número de iterações $I = 100$ e número de *frames* do *look-ahead* $K = 3$. Esses valores estão justificados em [19], onde é mostrado que eles garantem uma reconstrução aceitável para sinais musicais e de voz.

Conforme pode ser observado na Tabela 5.4, o RTISI-LA possui melhor desempenho do que os métodos G&L e RTISI para $I = 100$ e $K = 3$, estando esse resultado de acordo com o exposto em [19].

Tabela 5.4: *Comparando a SER dos três métodos de reconstrução*

Método	**G&L**	**RTISI**	**RTISI-LA**
SER	34,33	27,08	36,50

Assim, por resultar em maior SER, o RTISI-LA foi o método de reconstrução de fase utilizado em todos os testes seguintes, com $I = 100$ e $K = 3$.

5.4.2 Parâmetros do FastICA

5.4.2.1 Objetivo

Definir o método de aproximação (por deflação ou simétrico[5] e a não-linearidade (que pode ser *gauss* ($g(u) = u\exp^{(\frac{-u^2}{2})}$), *pow3* ($g(u) = u^3$), *skew* ($g(u) = u^2$) ou *tanh* ($g(u) = \text{tgh}(u)$)) para o FastICA. Vários testes foram realizados a fim de se verificar a separação das fontes, todos com resultados similares. Dois deles, encadeados, são mostrados a seguir.

[5]Outra alternativa de cálculo fornecida pelo pacote de FastICA para MATLAB© disponível em `http://research.ics.tkk.fi/ica/fastica/`.

5.4.2.2 Teste de exemplo 1

Utilizou-se o sinal *apito_prato_1*, um mistura entre dois sinais: o toque de um apito (com *pitch* definido e variável) e batidas de um prato (sem *pitch* definido). Na PCA, escolheu-se reter as componentes frequenciais, que são as colunas de **U** da Decomposição em Valores Singulares (SVD) do espectrograma da mistura (ver Subseção 3.5.6 do Capítulo 3). O valor ϕ que controla a quantidade de informação retida pela PCA foi ajustado para 0,7, pois resultou em uma separação perceptivamente razoável para este sinal. Esse valor acarretou a retenção de 21 componentes frequenciais, ou seja, $l = 21$ (Tabela 5.1). Novamente fixaram-se $L = 1024$ e $S = 256$ (75% de sobreposição). Fez-se a clusterização das 21 funções de base frequenciais para 2 fontes, ou seja, $C = 2$. Para cada método de aproximação foram testados subjetivamente[6] os quatro tipos de não-linearidades.

Pela audição informal, percebeu-se que o método por deflação com as não-linearidades *pow3* e *tanh* forneceu os melhores resultados, porém não foi possível perceber diferença entre os desempenhos de uma e outra. Para definir a escolha, foram feitos novos testes, todos com resultados similares, um deles mostrado a seguir.

5.4.2.3 Teste de exemplo 2

Foi utilizado o sinal *contrabaixo_violino* (agora uma mistura entre dois instrumentos com *pitch* definido), mantendo-se todos os outros parâmetros idênticos aos do teste anterior, com exceção de l, que depende do sinal, ainda que para o mesmo ϕ. Fez-se avaliação perceptiva sobre esse sinal para as não-linearidades *pow3* e *tanh* sob o método de aproximação por deflação (que foram os parâmetros que produziram os melhores resultados no teste de exemplo 1.

A não-linearidade *pow3* teve o melhor resultado. Dessa forma, o método por deflação e a não-linearidade *pow3* foram os parâmetros da ICA utilizados em todos os testes seguintes.

[6] Avaliação subjetiva é aquela em que uma pessoa julga a qualidade dos sinais através da audição.

5.4.3 Definindo parâmetros de janelamento

5.4.3.1 Objetivo

Verificar os efeitos do tamanho da janela L e do passo de sobreposição S na separação. Vários testes foram realizados a fim de verificar a melhor relação entre L e S, todos com resultados parecidos. Um deles é mostrado a seguir.

5.4.3.2 Teste de exemplo

Utilizou-se também o sinal *apito_prato_1*. Para cada valor de L (512, 1024 e 2048), fez-se $S = \frac{L}{2}$ e $\frac{L}{4}$ e mantiveram-se os outros parâmetros fixos, com ϕ de 0,7 e $C = 2$. Para a relação $S = L$ (sem sobreposição), naturalmente utilizou-se janela retangular.

O processo de análise da separação também foi subjetivo. Obtiveram-se, em geral, melhores resultados com a relação $S = \frac{L}{4}$ (sobreposição de 75%) e $L = 1024$. Verificou-se que para alguns sinais poderia ser vantajoso utilizar outros valores de L; contudo, fixaram-se esses valores para os testes seguintes.

5.5 Testes de esparsidade

5.5.1 Objetivo

Mostrar o funcionamento da ISA para sinais com diferentes graus de esparsidade no tempo e na frequência.

5.5.2 Testes com notas de piano

Foram realizados cinco testes diferentes: $A1$, $A2$, $A3$, $A4$ e $A5$. Para todos eles, o valor de ϕ foi 0,7, $L = 1024$ e $C = 2$.

Em $A1$, utilizou-se o sinal *piano_gr_ag_1*, cujas notas (uma grave e outra aguda, nessa ordem) estão completamente isoladas no tempo. Embora suas frequências fundamentais sejam distantes uma da outra, elas possuem harmônicos coincidentes. Porém, constatou-se que estes são harmônicos bem distantes da frequência fundamental.

Em $A2$, utilizou-se o sinal *piano_gr_ag_2*, com as mesmas notas de $A1$, porém com sobreposição parcial no tempo: uma inicia após o início da outra, porém esta ter terminado.

Em $A3$, utilizou-se o sinal *piano_gr_ag_3*, com as mesmas notas de $A1$ e $A2$, porém tocadas ao mesmo tempo.

Em $A4$, utilizou-se o sinal *piano_2ag_1*, com duas notas agudas repetidas, isoladas no tempo.

Em $A5$, utilizou-se o sinal *piano_2ag_2*, com as duas notas agudas iguais parcialmente sobrepostas no tempo.

As Figuras 5.1 e 5.2 mostram as misturas de $A1$ e de $A4$ como exemplo, respectivamente.

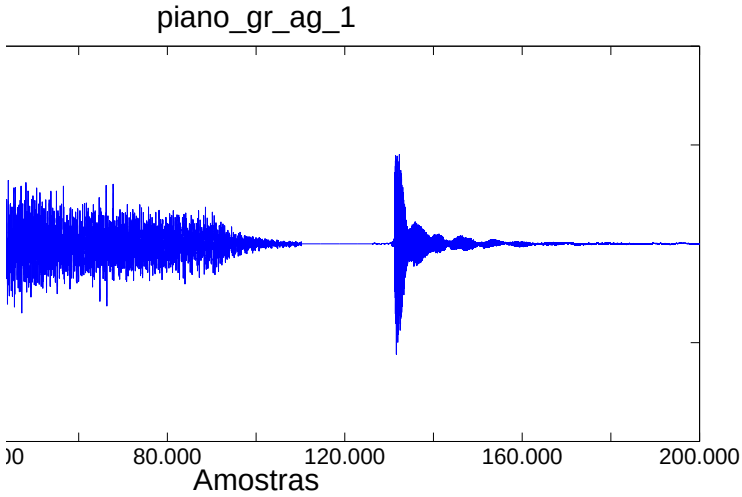

Figura 5.1: Sinal *piano_gr_ag_1*.

Para $A1$ e $A2$, verificou-se que boa parte dos harmônicos da nota grave estão presentes em uma das fontes estimadas, praticamente sem informação da nota aguda; o *pitch* daquela nota surge levemente alterado em relação ao que se ouve na mistura original. Na outra fonte, encontram-se os harmônicos restantes da nota grave (que assume um *pitch* mal definido) juntamento com a nota aguda praticamente intacta. O resultado para $A1$ pode ser visto na Figura 5.3.

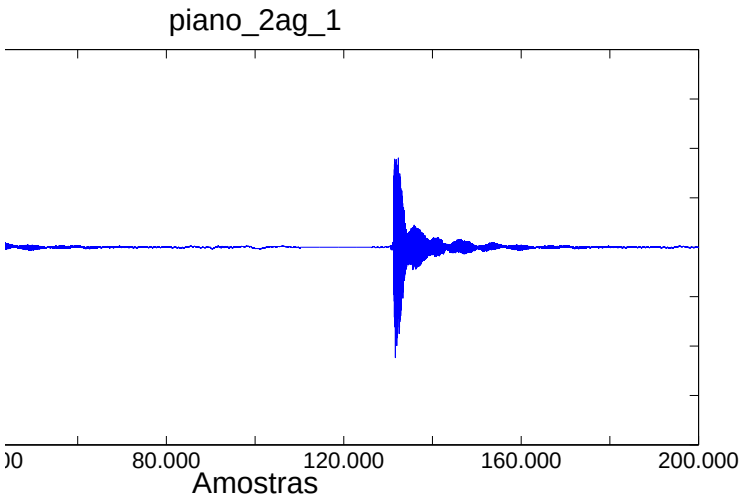

Figura 5.2: Sinal *piano_2ag_1*.

Para $A3$, verificou-se que uma das fontes contém grande parte do conteúdo da nota grave associada com a nota aguda praticamente inalterada. A outra fonte estimada contém os harmônicos residuais da nota grave.

Para $A4$ e $A5$, não houve separação entre as duas notas agudas. Em uma das fontes estimadas extraiu-se a parcela tonal das notas; na outra, restaram os sons causados pelas "batidas" na tecla. O resultado para $A4$ pode ser visto na Figura 5.4.

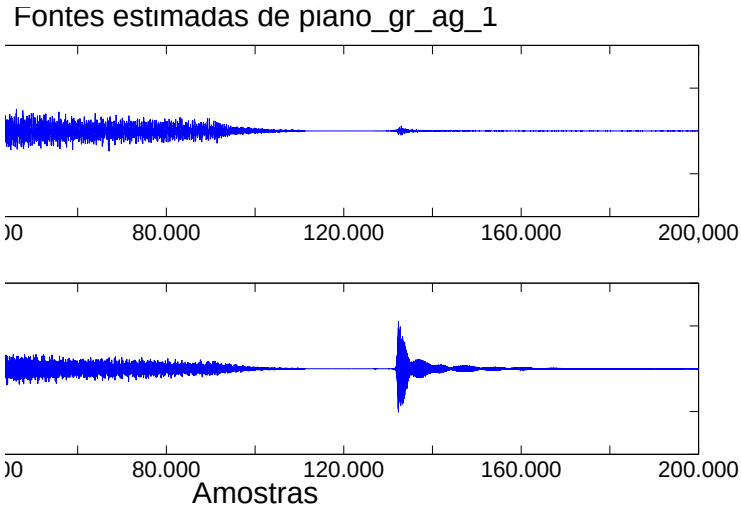

Figura 5.3: Fontes estimadas para $A1$.

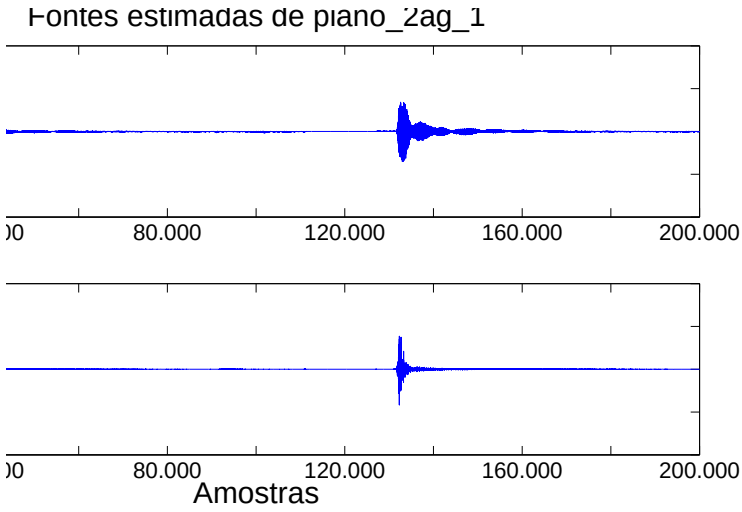

Figura 5.4: Fontes estimadas para $A4$.

5.5.3 Testes com apito e prato

A fim de verificar a esparsidade também para sinais mistos (percussivo + não-percussivo), foram feitos dois testes: $B1$ e $B2$. Em $B1$, utilizou-se o sinal *apito_prato_1* (em que apito e prato são ouvidos ao mesmo tempo) e em $B2$, o sinal *apito_prato_2* (em que o prato só é ouvido após o término do apito). Os parâmetros foram os mesmos dos testes com as notas de piano. A Figura 5.5 mostra o sinal de $B1$ no tempo.

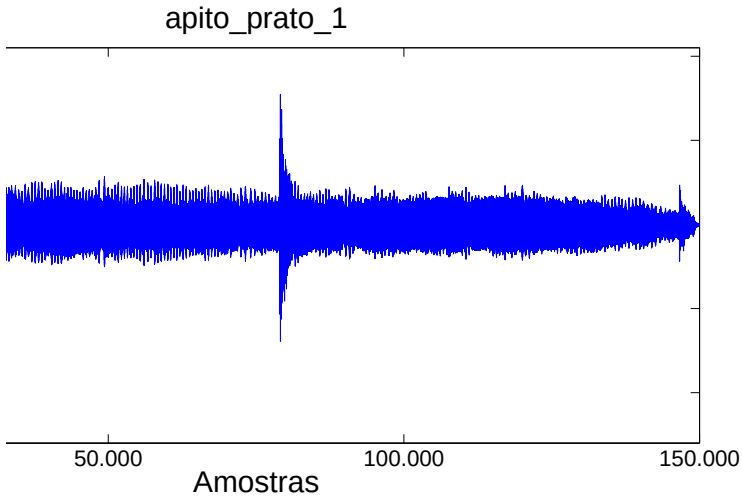

Figura 5.5: Sinal *apito_prato_1*.

A Figura 5.6 mostra o resultado de $B1$. Observa-se que uma das fontes estimadas contém somente as batidas de prato, enquanto na outra fonte estimada verifica-se ainda mistura entre o apito e prato. Em $B2$, verificou-se razoável separação.

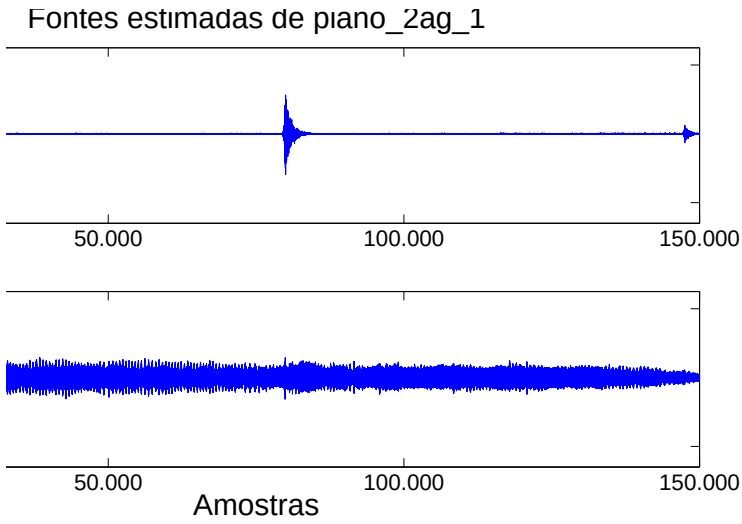

Figura 5.6: Fontes estimadas em $B1$.

5.5.4 Conclusão

Os resultados para esses tipos de misturas (entre notas de piano e entre apito e prato) mostram que a ISA consegue separar sinais com componentes de frequências distintas; porém, a separação é melhor verificada quando as fontes soam isoladas no tempo. Os exemplos com notas agudas repetidas de piano ($A4$ e $A5$) mostraram que a ISA "interpretou" o ataque percussivo como sendo provenientes de uma fonte e a parcela tonal subsequente como de outra; isso sugere que nos exemplos com duas notas diferentes talvez a separação possa ser favorecida especificando-se um número maior de fontes para o algoritmo.

5.6 Testes de funções de base

5.6.1 Objetivo

Conforme visto na Subseção 3.5.6 do Capítulo 3, o espectrograma da mistura pode ser decomposto, pela Decomposição em Valores Singulares, na multiplicação das matrizes $\mathbf{U}$, $\mathbf{S}$ e $\mathbf{V}$, sendo que as colunas de $\mathbf{U}$ são as componentes frequenciais desse

espectrograma e as colunas de **V** são as componentes temporais. Pela teoria da ISA, pode-se extrair tanto uma quanto a outra, dependendo da escolha, as componentes independentes serão as funções de base frequenciais (no caso de se extrair as colunas de **U**) ou temporais (no caso de se extrair as colunas de **V**) após a ICA.

Nestes testes, buscou-se comparar os resultados entre separações obtidas através da retenção de componentes frequenciais e através da retenção de componentes temporais pela PCA.

Nas seções anteriores, os testes de avaliação de qualidade foram subjetivos, já que suas intenções eram somente fixar parâmetros de operação e constatar qualitativamente a extensão da necessidade da esparsidade para o bom desempenho da ISA. A seguir, serão adotados avaliadores objetivos (descritos na Seção 5.2.1).

5.6.2 Testes com violino e prato

Aqui, utilizou-se o sinal *violino_prato* para três valores diferentes de ϕ, 0,5 e 0,7 e 0,9. A ISA foi aplicada duas vezes para cada um desses valores: uma retendo componentes frequenciais e outra retendo componentes temporais da PCA. A escolha desta mistura deveu-se ao fato de esta conter um sinal percussivo e um sinal tonal, com propriedades bem distintas entre si.

As Tabelas 5.5, 5.6 e 5.7 mostram os resultados desses testes para os três valores de ϕ com os respectivos números de componentes retidas (l) comparados pelas métricas SIR, SDR e SAR de cada fonte estimada. $F1$ corresponde ao violino e a $F2$ corresponde ao prato. Os melhores resultados obtidos para cada valor de ϕ em $F1$ e em $F2$ estão em negrito. Nota-se que os valores da SDR são negativos; isso indica que a potência da fonte efetivamente separada é menor que a de todos os defeitos presentes em sua estimativa. Como os valores da SIR são positivos, não é a interferência da outra fonte a responsável pela SDR baixa. Os valores da SAR (que é a razão entre as potências do sinal e do artefato, somente) corroboram este fato; esta medida não será mais utilizada daqui em diante, por não ser especialmente informativa.

Através dos valores da SIR, nota-se melhora crescente na separação em $F1$, com auge para $\phi = 0{,}90$ e pior para $\phi = 0{,}95$, retendo-se componentes frequenciais (colunas de **U**) da PCA. $F2$ leva a resultados parecidos. Ainda, nota-se certa coerência entre os valores em negrito: por exemplo, a SIR, a SDR e a SAR para $F1$, retendo-se as componentes frequenciais, indicam $\phi = 0{,}90$ como o fornecedor do melhor resultado. Adicionalmente, o valor da SIR de 65,46 dB para $\phi = 0{,}95$, retidas componentes temporais, é alto devido a os valores da SDR e da SAR correspondentes terem sido os mesmos nos arredondamentos feitos, o que indica muito baixa interferência.

Entretanto, na avaliação subjetiva, percebe-se melhor separação para $\phi = 0{,}5$, que piora conforme o aumento desse parâmetro—pois percebem-se batidas de prato mais fortes junto com o violino. Essa diferença entre as avaliações objetiva e subjetiva deve-se ao fato de a grande potência de artefatos gerados pelo método não ser considerada na SIR, mas percebida subjetivamente.

Nota-se que a qualidade da separação não somente depende da escolha de ϕ como também depende do tipo de componentes que são retidas pela PCA, se as frequenciais ou as temporais. Porém, cabe mencionar que não se pode dizer que reter componentes frequenciais ou temporais tenha sido especialmente melhor.

Tabela 5.5: *Separação da mistura violino_prato por funções frequenciais (colunas de **U**) e temporais (colunas de **V**). Medição através da SIR.*

PCA	ϕ	l	F1 (SIR)	F2 (SIR)
U	0,50	4	1,81 dB	24,43 dB
	0,70	12	15,80 dB	**38,86 dB**
	0,90	39	**30,02 dB**	27,69 dB
	0,95	72	2,26 dB	-0,56 dB
V	0,50	4	37,93 dB	**38,88 dB**
	0,70	12	33,33 dB	30,35 dB
	0,90	39	26,77 dB	0,18 dB
	0,95	72	**65,46 dB**	11,66 dB

Tabela 5.6: *Separação da mistura violino_prato por funções frequenciais (colunas de* ***U****) e temporais (colunas de* ***V****). Medição através da SDR.*

PCA	ϕ	l	F1 (SDR)	F2 (SDR)
U	0,50	4	-35,08 dB	-30,33 dB
	0,70	12	-7,38 dB	**-25,19 dB**
	0,90	39	**-6,19 dB**	-30,39 dB
	0,95	72	-28,62 dB	-28,00 dB
V	0,50	4	**-18,86 dB**	-18,92 dB
	0,70	12	-30,23 dB	**-18,25 dB**
	0,90	39	-31,38 dB	-41,02 dB
	0,95	72	-28,88 dB	-32,61 dB

Tabela 5.7: *Separação da mistura violino_prato por funções frequenciais (colunas de* ***U****) e temporais (colunas de* ***V****). Medição através da SAR.*

PCA	ϕ	l	F1 (SAR)	F2 (SAR)
U	0,50	4	-32,88 dB	-30,32 dB
	0,70	12	-7,24 dB	-25,15 dB
	0,90	39	**-6,19 dB**	-30,39 dB
	0,95	72	-26,59 dB	**-24,69 dB**
V	0,50	4	**-18,86 dB**	-18,92 dB
	0,70	12	-30,23 dB	**-18,24 dB**
	0,90	39	-31,37 dB	-38,10 dB
	0,95	72	-28,88 dB	-32,32 dB

5.7 Testes de número de fontes

5.7.1 Objetivo

Variar o número de fontes—parâmetro C da Tabela 5.1—a serem encontradas pela ISA a fim de verificar a separação.

5.7.2 Testes com violino e prato

Para este teste utilizou-se novamente o sinal *violino_prato*. Contudo, aqui, utilizou-se 0,98 como valor fixo de ϕ, retendo 172 componentes (foram escolhidas as de frequência), e variou-se C. Para viabilizar a avaliação objetiva do resultado, nos testes com números de fontes maiores que 2, fez-se clusterização manual entre as com-

ponentes perceptivamente mais parecidas, somando-as, retornando então ao número original de fontes. Foram retidas componentes frequenciais pela PCA.

Para todos os valores de C, perceberam-se as componentes do violino misturadas com algumas componentes de prato em apenas um dos subespaços reconstruídos, enquanto os outros continham, em sua grande maioria, o restante das componentes de prato, somente. Isso se deve ao fato de que as batidas de prato, além de possuirem alta energia, ocupam também grande faixa de frequências; portanto, com $\phi = 0,98$ espera-se que dos 172 padrões frequenciais retidos, a maioria responda pela modelagem do prato. Resultados similares foram observados quando reteve-se componentes temporais pela PCA.

As Tabelas 5.8 e 5.9 apresentam os valores da SIR e da SDR, respectivamente, para cada número de fontes, sendo que $F1$ corresponde ao sinal de violino e $F2$ corresponde ao sinal do prato. Observa-se o melhor valor da SIR, tanto em $F1$ como em $F2$, para $C = 4$. Entretanto, comprovou-se que, perceptivamente, o aumento de C não produziu melhora na separação neste exemplo.

Tabela 5.8: *Valores SIR para diferentes números de fontes requeridas.*

C	F1	F2
2	11,95 dB	27,84 dB
4	42,23 dB	39,17 dB
5	19,38 dB	43,55 dB
8	40,59 dB	36,52 dB
10	13,41 dB	36,54 dB

Tabela 5.9: *Valores SDR para diferentes números de fontes requeridas.*

C	F1	F2
2	-13,83 dB	-34,98 dB
4	3,11 dB	-24,39 dB
5	-14,81 dB	-26,01 dB
8	0,0504 dB	-29,30 dB
10	-31,60 dB	-26,55 dB

5.8 Testes complementares

5.8.1 Objetivo

Mostrar o funcionamento da ISA para misturas entre outros tipos de sinais. Além da SDR e da SIR, o PEAQ também foi utilizado para avaliar os sinais. Em todos os casos, adotaram-se $L = 1024$, clusterização para duas fontes ($C = 2$) e componentes frequenciais da PCA.

5.8.2 Teste com corneta e vibrafone

Foi utilizado o sinal *corneta_vibrafone*, em que os dois instrumentos (um de sopro, outro de percussão) possuem *pitch* definido. O ϕ ajustado foi 0,7.

A Tabela 5.10 apresenta a SDR, a SIR e o PEAQ para cada fonte estimada. A fonte estimada $F1$ corresponde à corneta e a fonte estimada $F2$ corresponde ao vibrafone. O valor do PEAQ parece estar coerente com os valores de SIR e SDR; entretanto, as medidas estão tão próximas do limite inferior da escala (-4) que podem ser consideradas iguais. Subjetivamente, percebeu-se boa separação entre as fontes.

É interessante comentar que, na avaliação subjetiva, a fonte estimada $F2$ possui mais resquícios da corneta do que $F1$ do vibrafone. Isso acontece porque as ocorrências de vibrafone são as mesmas (uma única nota) e a frase melódica tocada pela corneta possui mais notas. Assim, a ISA encontra maior dificuldade em separar a frase melódica em uma única fonte. Portanto, pode ser interessante aumentar o número de fontes desejadas a serem retornadas nesta separação ou aplicar um método de ISA para sinais não-estacionários.

Os valores do PEAQ obtidos em todos os testes complementares desta seção estão próximos do limite inferior (-4) devido ao fato de esta ser uma medida muito exigente, que "tolera" somente diferenças mínimas entre as fontes separadas e as originais.

Tabela 5.10: *Resultado da separação para corneta_vibrafone.*

Avaliação	F1	F2
SDR	-34,55 dB	-16,83 dB
SIR	21,36 dB	41,35 dB
PEAQ	-3,909	-3,883

5.8.3 Teste com corneta e voz cantada

Foi utilizado o sinal *corneta_vozC*. O ϕ utilizado foi $0,7$.

A Tabela 5.11 apresenta a SDR, a SIR e o PEAQ para cada fonte estimada. A fonte estimada $F1$ corresponde à corneta e a fonte estimada $F2$ corresponde à voz cantada. O valor do PEAQ está coerente com a SIR e a SDR. Apesar de a SIR para a componente da corneta estar negativa, percebeu-se, subjetivamente, pequena separação entre as fontes, que se explica pelo fato de que, aqui, tanto a frase melódica tocada pela corneta quanto a voz cantada são sinais que possuem *pitch* variante no tempo. Novamente, pode ser interessante aumentar o número de fontes desejadas a serem retornadas nesta separação (e reuni-las convenientemente com base em um critério perceptivo) ou aplicar um método não-estacionário para a separação.

Tabela 5.11: *Resultado da separação para corneta_vozC.*

Avaliação	F1	F2
SDR	-68,01 dB	-41,28 dB
SIR	-30,33 dB	6,79 dB
PEAQ	-3,913	-3,907

5.8.4 Teste com corneta e ruído

Foi utilizado o sinal *corneta_ruido*. O ϕ utilizado foi 0,7.

A Tabela 5.12 apresenta a SDR, a SIR e o PEAQ para cada fonte estimada. A fonte estimada $F1$ corresponde à corneta e a fonte estimada $F2$ corresponde ao ruído. Não se percebe presença de ruído na componente em $F1$; e também não se percebem componentes da corneta em $F2$, sendo este um caso muito bom de

separação, perante os demais, em que o PEAQ forneceu seus melhores valores (ainda ruins, mas já significativamente melhores que o fundo de escala).

Cabe mencionar que há diferença entre o ruído original e o que se percebe em $F2$, o que leva a crer que a componente de ruído não foi bem reconstruída. Essa observação não é uma surpresa, visto que o ruído branco não possui padrões espectrais.

Tabela 5.12: *Resultado da separação para corneta_ruido*

Avaliação	F1	F2
SDR	-27,88 dB	1,69 dB
SIR	22,15 dB	66,41 dB
PEAQ	-3,189	-3,539

5.8.5 Teste com voz cantada e voz falada

Foi utilizado o sinal *vozC_vozF*. O ϕ utilizado foi 0,7.

Não foi percebida separação entre as vozes para o processo inicial. Foram feitas, então, outras tentativas, variando-se o valor de ϕ e o número de fontes. Também se experiementou reter componentes temporais ao invés das frequenciais. Mesmo assim, não foi observada separação aproveitável, por motivo semelhante ao explicado no teste com corneta e vibrafone e no teste com corneta e voz cantada, em que os sinais misturados possuem múltiplos *pitches* variando no tempo. Em particular, a voz falada ainda apresenta maior dificuldade, por suas variações contínuas de *pitch* ao longo do tempo.

5.8.6 Teste com vibrafone e prato

Foi utilizado o sinal *vibrafone_prato*. O ϕ utilizado foi 0,5. Foram retidas as componentes frequenciais pela PCA.

A Tabela 5.13 apresenta a SDR, a SIR e o PEAQ para cada fonte estimada. A fonte estimada $F1$ corresponde às duas ocorrências de vibrafone e a fonte estimada $F2$ corresponde às duas ocorrências do prato. Subjetivamente, notou-se separação

entre as fontes, com resquícios de baixa amplitude da intercalação da outra fonte em cada fonte estimada.

Tabela 5.13: *Resultado da separação para corneta_prato*

Avaliação	F1	F2
SDR	-72,87 dB	-35,28 dB
SIR	2,19 dB	28,91 dB
PEAQ	-3,909	-3,913

Capítulo 6

Conclusões

6.1 Conclusões

Apesar de as métricas objetivas SDR e SAR e a métrica psicoacústica PEAQ terem sido desfavoráveis ao avaliarem as separações dos sinais aqui testados, constatou-se subjetivamente que houve razoáveis separações. Essas medidas geralmente são utilizadas para avaliar diferenças moderadas entre o sinal original e o separado. Entretanto, a ISA gera alta potência de artefatos e, mesmo que as separações sejam razoáveis perceptivamente, a SDR, a SAR e o PEAQ acabam por atribuir notas indistintamente ruins, não sendo adequados para a avaliação.

No que diz respeito à esparsidade dos sinais no tempo e na frequência, a ISA possui melhor desempenho para sinais que podem ser encontrados isolados (tanto no tempo como na frequência) na mistura. Essa observação está de acordo com o que foi mencionado na modelagem da ISA, descrita na Subseção 3.5.4, onde se lê que o somatório dos espectrogramas independentes corresponde aproximadamente ao espectrograma da mistura somente quando não há sobreposição entre os espectrogramas subjacentes.

Observou-se, ainda, que a ISA não se mostrou eficaz para misturas entre sinais não-estacionários. Contudo, em [1], menciona-se uma forma de aplicar a ISA para esses tipos de sinais.

Por fim, cabe relembrar que os autores dos artigos sobre a ISA mencionam que ela funciona melhor para sinais percussivos, especificamente sinais de bateria. De fato, nas misturas contendo sinal de prato, tal sinal teve parte de suas componentes melhor separadas da outra fonte.

6.2 Trabalhos Futuros

A ISA é uma técnica com muitas possibilidades de aprimoramento, sendo que algumas já foram abordadas na literatura.

Em [10] e [11] são mostradas técnicas que buscam contornar a dificuldade da ISA padrão quanto ao número de componentes que devem ser retidas pela PCA de forma a resultar em uma separação ótima para cada tipo de sinal de mistura, já que, por exemplo, fontes que possuem baixa amplitude em relação a outras requerem maior número de componentes para serem reconhecidas. Essas técnicas necessitam de informações prévias das fontes que se deseja separar. Em [9] é mostrada a ISA utilizando *locally linear embedding* (o que, em tradução livre, corresponde a "incorporação localmente linear"), que, segundo este artigo, é uma técnica alternativa às duas anteriores para a redução da dimensão do sinal de mistura, porém com a vantagem de não necessitar de informações prévias das fontes. Tais soluções podem ser incorporadas ao sistema já desenvolvido.

Conforme já foi dito, neste trabalho utilizou-se como método de clusterização o *deterministic annealing* com medida de distorção igual a distância euclidiana. Entretanto, em [14] menciona-se o uso do *ixegram*, uma matriz contendo a informação de diferença entre as bases através da divergência de Kullback-Leibler, que resulta em uma implementação mais complexa para o *deterministic annealing*. Essa é mais uma variação a ser testada.

Referências Bibliográficas

[1] CASEY, M. A., *Separation of Mixed Audio Sources by Independent Subspace Analysis*, Technical Report TR-2001-31, Mitsubishi Eletric Research Labs, Massachusetts, EUA, Setembro 2001.

[2] RABINER, L. R., SCHAFER, R. W., *Digital Processing of Speech Signals*. Nova Jersey, EUA, Prentice-Hall, 1978.

[3] DINIZ, P. S. R., SILVA, E. A. B., NETTO, S. L., *Processamento Digital de Sinais: Projeto e Análise de Sistemas*. Porto Alegre, Brasil, Bookman, 2004.

[4] HAYKIN, S., VEEN, B. V., *Sinais e Sistemas*. Porto Alegre, Brasil, Bookman, 2001.

[5] MITRA, S. K., *Digital Signal Processing, a computer-based approach*. 2 ed. Nova Iorque, EUA, McGraw-Hill, 2001.

[6] ESQUEF, P. A. A., BISCAINHO, L. W. P., "Spectral-Based Analysis and Synthesis of Audio Signals". In: Pérez-Meana, H. (ed.), *Signal Processing Methods for Music Transcription*, chapter 3, Hershey, EUA, Idea Group, pp. 56–92, 2007.

[7] HYVÄRINEN, A., KARHUNEN, J., OJA, E., *Independent Component Analysis*. Nova Iorque, EUA, John Wiley, 2001.

[8] PEEBLES, Jr, P. Z., *Probability, Random Variables and Random Signal Principles*. 4 ed. Nova Iorque, EUA, McGraw-Hill, 2001.

[9] FITZGERALD, D., LAWLOR, B., COYLE, E., "Independent Subspace Analysis using Locally Linear Embedding". In: *Proceedings of the 6^{th} Conference on Digital Audio Effects*, Londres, Inglaterra, Setembro 2003.

[10] FITZGERALD, D., LAWLOR, B., COYLE, E., "Sub-band Independent Subspace Analysis for Drum Transcription". In: *Proceedings of the 5th Conference on Digital Audio Effects*, Hamburgo, Alemanha, Setembro 2002.

[11] FITZGERALD, D., LAWLOR, B., COYLE, E., "Prior Subspace Analysis For Drum Transcription". In: *114th AES Convention, Preprint 5808*, Audio Engineering Society, Amsterdã, Holanda, Março 2003.

[12] HYVÄRINEN, A., OJA, E., "Independent Component Analysis: Algorithms and Applications", *Neural Networks*, v. 13, n. 4-5, pp. 411–430, Maio-Junho 2000.

[13] VIRTANEN, T., "Unsupervised Learning Methods for Source Separation in Monoaural Music Signals". In: Klapuri, A., Davy, M. (eds.), *Signal Processing Methods for Music Transcription*, Signal Processing Methods for Music Transcription, chapter 9, Nova Iorque, EUA, Springer, pp. 267–296, 2006.

[14] FITZGERALD, D., *Automatic Drum Transcription and Source Separation*. Ph.D. thesis, Conservatory of Music and Drama, Dublin Institute of Technology, Dublim, Irlanda, 2004.

[15] MUNCASTER, J., *A Brief Introduction to Deterministic Annealing*, Technical report, Department of Computer Science, Universidade da Califórnia, Santa Bárbara, EUA, Junho 2006.

[16] HAYKIN, S., *Neural Networks, a comprehensive foundation*. 2 ed. Upper Saddle River, EUA, Prentice Hall, 1999.

[17] PROAKIS, J. G., SALEHI, M., *Communication Systems Engineering*. Upper Saddle River, EUA, Prentice-Hall, 1994.

[18] GRIFFIN, D. W., LIM, J. S., "Signal Estimation from Modified Short-Time Fourier Transform", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, v. ASSP-32, n. 2, pp. 236–243, Abril 1984.

[19] ZHU, X., BEAUREGARD, G. T., WYSE, L. L., "Real-Time Signal Estimation From Modified Short-Time Fourier Transform Magnitude Spectra", *IEEE*

Transactions on Acoustics, Speech, and Signal Processing, v. 15, n. 5, pp. 1645–1653, Julho 2007.

[20] ZHU, X., BEAUREGARD, G. T., WYSE, L. L., "An Efficient Algorithm For Real-Time Spectrogram Inversion". In: *Proceedings of the 8th Conference on Digital Audio Effects (DAFX-05)*, pp. 116–121, Madri, Espanha, Setembro 2005.

[21] ZHU, X., BEAUREGARD, G. T., WYSE, L. L., "Real-Time Iterative Spectrum Inversion With Look-Ahead". In: *Proceedings of the IEEE International Conference on Multimedia & Expo (ICME 2006)*, pp. 229–232, Ontario, Canadá, Julho 2006.

[22] SANZ, J. L. C., "Mathematical Considerations for the Problem of Fourier Transform Phase Retrieval from Magnitude", *SIAM Journal of Applied Mathematics*, v. 45, n. 4, pp. 651–664, 1985.

[23] TYGEL, A. F., *Métodos de Fatoração de Matrizes Não-negativas Para Separação de Sinais Musicais*. M.Sc. dissertation, Universidade Federal do Rio de Janeiro, Rio de Janeiro, Brasil, Dezembro 2009.

[24] HOFMANN, T., BUHMANN, J. M., "Pairwise Data Clustering by Deterministic Annealing", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 19, n. 1, pp. 1–14, Janeiro 1997.

[25] GOTO, M., HASHIGUCHI, H., NISHIMURA, T., *et al.*, "Music Genre Database and Musical Instrument Sound Database". In: *Proceedings of the* 4^{th} *International Conference om Music Information Retrieval*, pp. 229–230, Baltimore, Outubro 2003.

Apêndice A

Prova da condição $\mathbf{R_{XY}} = \bar{\mathbf{X}}\bar{\mathbf{Y}}^T$ a partir da sua descorrelação

Supõe-se inicialmente que as variáveis aleatórias $\mathbf{X}$ e $\mathbf{Y}$ são descorrelacionadas, ou seja, que a covariância entre elas seja nula. Portanto:

$$\mathbf{C_{XY}} = E[(\mathbf{X} - \bar{\mathbf{X}})(\mathbf{Y} - \bar{\mathbf{Y}})^T] = 0. \tag{A.1}$$

Inserindo o operador de transposição nos parênteses:

$$E[(\mathbf{X} - \bar{\mathbf{X}})(\mathbf{Y}^T - \bar{\mathbf{Y}}^T)] = 0. \tag{A.2}$$

Aplicando a propriedade distributiva:

$$E[\mathbf{X}\mathbf{Y}^T - \mathbf{X}\bar{\mathbf{Y}}^T - \bar{\mathbf{X}}\mathbf{Y}^T + \bar{\mathbf{X}}\bar{\mathbf{Y}}^T] = 0. \tag{A.3}$$

O operador Valor Esperado é linear. Portanto, o valor esperado de uma soma de termos é igual à soma dos valores esperados desses termos. Logo:

$$E[\mathbf{X}\mathbf{Y}^T] - E[\mathbf{X}\bar{\mathbf{Y}}^T] - E[\bar{\mathbf{X}}\mathbf{Y}^T] + E[\bar{\mathbf{X}}\bar{\mathbf{Y}}^T] = 0. \tag{A.4}$$

Devido ainda à linearidade do operador Valor Esperado, os valores constantes podem sair de dentro dessa operação. As médias $\bar{\mathbf{X}}$ e $\bar{\mathbf{Y}}^T$ são constantes. Então:

$$E[\mathbf{X}\mathbf{Y}^T] - \bar{\mathbf{Y}}^T E[\mathbf{X}] - \bar{\mathbf{X}}E[\mathbf{Y}^T] + \bar{\mathbf{X}}\bar{\mathbf{Y}}^T = 0. \tag{A.5}$$

Os valores $E[\mathbf{X}]$ e $E[\mathbf{Y}^T]$ são as médias $\bar{\mathbf{X}}$ e $\bar{\mathbf{Y}}^T$, respectivamente. Com isso, pode-se cancelar um dos termos negativos com o último termo. Então:

$$E[\mathbf{X}\mathbf{Y}^T] - \bar{\mathbf{X}}\bar{\mathbf{Y}}^T = 0. \tag{A.6}$$

$$\mathbf{R_{XY}} = E[\mathbf{X}\mathbf{Y}^T] = \bar{\mathbf{X}}\bar{\mathbf{Y}}^T \tag{A.7}$$

Apêndice B

Prova de que variáveis gaussianas não-correlacionadas são também independentes

Supoẽ-se inicialmente que X e Y são variáveis aleatórias gaussianas, ou seja, que suas PDFs são da forma

$$f_X(x) = \frac{1}{\sqrt{2\pi\sigma_X^2}} e^{\frac{-(x-\bar{X})^2}{2\sigma_X^2}} \tag{B.1}$$

e

$$f_Y(y) = \frac{1}{\sqrt{2\pi\sigma_Y^2}} e^{\frac{-(y-\bar{Y})^2}{2\sigma_Y^2}}, \tag{B.2}$$

onde $\bar{X}$, $\bar{Y}$ são as médias estatísticas de X e Y, respectivamente, e σ_X^2 e σ_Y^2 são as variâncias de X e Y, respectivamente. A PDF conjunta de X e Y é da forma

$$f_{X,Y}(x,y) = \frac{1}{2\pi\sigma_X\sigma_Y\sqrt{1-\rho_{XY}^2}} e^{\frac{-1}{2(1-\rho_{XY}^2)}\left[\frac{(x-\bar{X})^2}{\sigma_X^2} - \frac{2\rho_{XY}(x-\bar{X})(y-\bar{Y})}{\sigma_X\sigma_Y} + \frac{(y-\bar{Y})^2}{\sigma_Y^2}\right]}, \tag{B.3}$$

onde

$$\rho_{XY} = \frac{C_{XY}}{\sigma_X\sigma_Y}. \tag{B.4}$$

Se X e Y são variáveis não-correlacionadas, isso significa que $C_{XY} = 0$ (sua covariância é nula). Sendo assim, o valor de ρ_{XY} também será nulo. Substituindo-se

$\rho_{XY} = 0$ na expressão da PDF conjunta, obtém-se:

$$f_{X,Y}(x,y) = \frac{1}{2\pi\sigma_X\sigma_Y} e^{-\frac{1}{2}\left[\frac{(x-\bar{X})^2}{\sigma_X^2} + \frac{(y-\bar{Y})^2}{\sigma_Y^2}\right]}. \tag{B.5}$$

Essa expressão pode ser reescrita da forma

$$f_{X,Y}(x,y) = \frac{1}{\sqrt{2\pi\sigma_X^2}} e^{\frac{-(x-\bar{X})^2}{2\sigma_X^2}} . \frac{1}{\sqrt{2\pi\sigma_Y^2}} e^{\frac{-(y-\bar{Y})^2}{2\sigma_Y^2}}, \tag{B.6}$$

que nada mais é do que o produto das PDFs de X e Y, ou seja,

$$f_{X,Y}(x,y) = f_X(x) f_Y(y). \tag{B.7}$$

Essa última expressão é a condição de independência entre duas variáveis aleatórias X e Y. Logo, duas variáveis gaussianas não-correlacionadas são também independentes [8].

Apêndice C

Expressão para minimização do MSE

A ideia é derivar a equação $D[.]$ e igualar o seu valor a zero. Devido à propriedade de linearidade, a operação derivada pode ser inserida nos somatórios. Com isso, tem-se

$$\frac{\partial D[x(n), Y(mS, \omega)]}{\partial x(n)} = \sum_{m=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \frac{\partial}{\partial x(n)} [x(mS, l) - y(mS, l)]^2 = 0. \tag{C.1}$$

Utilizando a Regra da Cadeia[1], obtém-se:

$$\sum_{m=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} 2[x(mS, l) - y(mS, l)] \frac{\partial}{\partial x(n)} x(mS, l) = 0. \tag{C.2}$$

Sendo $x(mS, l)$ o sinal $x(l)$ janelado, faz-se $x(mS, l) = x(l)w(l - mS)$ na expressão acima. Então:

$$\sum_{m=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} 2[x(l)w(l - mS) - y(mS, l)] \frac{\partial}{\partial x(n)} x(l)w(l - mS) = 0. \tag{C.3}$$

[1]Trata-se de uma fórmula para o cálculo da derivada de uma função composta de duas funções. Sendo $f(x)$ e $g(x)$ duas funções, a Regra da Cadeia é matematicamente escrita como $(f(g(x)))' = f'(g(x))g'(x)$

O valor de $\frac{\partial}{\partial x(n)}x(l)$ é igual a 1 para $l = n$ e 0 para $l \neq n$. Com isso, tem-se:

$$\sum_{m=-\infty}^{\infty} 2[x(n)w(n - mS) - y(mS, n)]w(n - mS) = 0 \tag{C.4}$$

$$\sum_{m=-\infty}^{\infty} 2x(n)w(n - mS)w(n - mS) - \sum_{m=-\infty}^{\infty} 2y(mS, n)w(n - mS) = 0 \tag{C.5}$$

$$\sum_{m=-\infty}^{\infty} 2x(n)w(n - mS)w(n - mS) = \sum_{m=-\infty}^{\infty} 2y(mS, n)w(n - mS). \tag{C.6}$$

$$\sum_{m=-\infty}^{\infty} x(n)w^2(n - mS) = \sum_{m=-\infty}^{\infty} y(mS, n)w(n - mS). \tag{C.7}$$

Por fim, deixando em termos de $x(n)$, chega-se a

$$x(n) = \frac{\sum_{m=-\infty}^{\infty} y(mS, n)w(n - mS)}{\sum_{m=-\infty}^{\infty} w^2(n - mS)}. \tag{C.8}$$