SÍNTESE ADITIVA APLICADA À TRANSFERÊNCIA DE TIMBRE EM VOCALISE

Lucas Lago Monteiro

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

Orientador: Luiz Wagner Pereira Biscainho

Rio de Janeiro
Junho de 2012

SÍNTESE ADITIVA APLICADA À TRANSFERÊNCIA DE TIMBRE EM VOCALISE

Lucas Lago Monteiro

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

Examinada por:

Prof. Luiz Wagner Pereira Biscainho, D.Sc.

Prof. Sergio Lima Netto, Ph.D.

Prof. Abraham Alcaim, Ph.D.

RIO DE JANEIRO, RJ – BRASIL
JUNHO DE 2012

Monteiro, Lucas Lago

Síntese Aditiva Aplicada à Transferência de Timbre em Vocalise/Lucas Lago Monteiro. – Rio de Janeiro: UFRJ/COPPE, 2012.

XIV, 102 p.: il.; 29,7cm.

Orientador: Luiz Wagner Pereira Biscainho

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2012.

Referências Bibliográficas: p. 91 – 97.

1. Voz Cantada. 2. Processamento de Sinais de Áudio. 3. Síntese. I. Biscainho, Luiz Wagner Pereira. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

Agradecimentos

Primeiramente, gostaria de agradecer a Deus, pois sem sua mão não teria sido possível iniciar e completar este mestrado. Reconheço que Ele é o maior responsável por tudo que conquistei e realizei até hoje.

Agradeço ao Luiz Wagner Pereira Biscainho, meu orientador, por ter me guiado ao longo desta etapa tão importante da minha vida, e por ter acreditado no sucesso deste trabalho, mesmo em momentos difíceis.

Aos meus colegas do LPS, que sempre estiveram prontos a me ajudar, dos quais gostaria de citar Anderson Vinícius, Rafael Amado, Axel Hollanda, Camila Gussen, Ana Quaresma, Tadeu Ferreira, Leonardo Nunes, Alexandre Leizor, Flávio Ávila, Wallace Martins, Markus Lima e Gabriel Matos.

Aos meus amigos do colégio, igreja, faculdade e de conjuntos musicais, por compreenderem minha ausência em diversos momentos ao longo destes anos.

Aos meus irmãos, Flávia e Daniel Monteiro, por sempre me incentivarem com grande companheirismo e inteligência. Agradeço, também, aos meus sogros e demais familiares.

Aos meus pais, Flávio e Dorimar Monteiro, por serem pessoas e pais maravilhosos, íntegros e exemplares. Agradeço por terem me dado todo o apoio e inspiração de que necessitei por todas as fases da minha vida.

À minha linda esposa Sarah Monteiro, ou como seu próprio nome diz, minha princesa, por desde o início do namoro ter compreendido os meus estudos que, por muitas vezes, nos privaram de momentos de qualidade juntos. Agradeço por sempre me incentivar em tudo que faço, por desejar o melhor para a minha vida, por ceder sua linda voz para o banco de dados de síntese e me ajudar, mesmo sem entender do assunto, a concluir este mestrado. Te amo pra sempre.

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

SÍNTESE ADITIVA APLICADA À TRANSFERÊNCIA DE TIMBRE EM VOCALISE

Lucas Lago Monteiro

Junho/2012

Orientador: Luiz Wagner Pereira Biscainho

Programa: Engenharia Elétrica

Esta dissertação tem por finalidade descrever técnicas de análise e síntese de sinais de voz cantada. Para implementar tais técnicas, foi desenvolvido um algoritmo que, a partir de um banco de dados contendo parâmetros de um determinado cantor alvo, recebe como entrada um sinal de referência que contenha a gravação de um cantor qualquer executando uma composição musical qualquer; como saída, o algoritmo deverá retornar um sinal sintetizado que contenha a mesma composição musical do sinal de referência, porém com o timbre do cantor alvo.

Para o desenvolvimento deste algoritmo, inicialmente esta dissertação descreve, compara e escolhe uma dentre as técnicas YIN, Cepstral e Logaritmo do Produto do Espectro Harmônico para detecção de *pitch*, a fim de obter a linha melódica de um sinal de voz cantada.

Outra ferramenta de análise de voz cantada necessária nesta dissertação é a detecção de vogais. Para isso, este texto descreve o conceito de envoltória espectral e seus formantes, os quais serão responsáveis por identificar as vogais contidas no sinal de referência.

Técnicas de *morphing* de envoltórias espectrais são descritas e utilizadas com o intuito de tornar o resultado da síntese mais natural aos ouvidos humanos.

Para a realização da síntese, são utilizadas envoltórias espectrais do cantor alvo juntamente com a técnica de síntese aditiva através da inversa da DFT, a qual tem por prerrogativa a construção do sinal sintético no domínio da frequência.

Com o intuito de avaliar os métodos implementados neste trabalho de mestrado, são apresentados alguns experimentos utilizando diferentes sinais de referência e diferentes sinais alvo.

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

ADDITIVE SYNTHESIS APPLIED TO TIMBRE TRANSFER IN VOCALISE

Lucas Lago Monteiro

June/2012

Advisor: Luiz Wagner Pereira Biscainho

Department: Electrical Engineering

This dissertation has the purpose of describing analysis and synthesis techniques of singing voice signals. To implement these techniques, it was developed an algorithm that, from a database containing parameters of a given target singer, takes as input a reference signal that contains the recording of a singer performing any musical composition; as output, the algorithm shall return a synthesized signal containing the same musical composition as the reference signal, but with the timbre of the target singer.

To develop this algorithm, this dissertation first describes, compares and chooses one among the techniques YIN, Cepstral and Logarithm of the Harmonic Product Spectrum for pitch detection, in order to acquire the melody of a singing voice signal.

Another tool for singing voice analysis necessary to this dissertation is vowel detection. In that matter, this dissertation describes the concept of spectral envelope and its formants, which are responsible for identifying the vowels contained in the reference signal.

Morphing techniques for spectral envelopes are described and used in order to make the result of the synthesis more natural to human ears.

To perform the synthesis, the proposed system uses spectral envelopes of the target singer along with additive synthesis via inverse DFT, which creates the synthetic signal in the frequency domain.

In order to evaluate the methods implemented in this dissertation, some experiments using different reference and target signals are presented.

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Na natureza, é possível encontrar inúmeros sons derivados das mais variadas fontes. Dentre elas, podemos destacar seres vivos capazes de emitir sons e os utilizar como ferramenta de comunicação, seja entre seres de uma mesma espécie, seja de espécies distintas.

Da mesma forma, o ser humano possui a capacidade de reproduzir diversos fonemas com o seu complexo sistema vocal. Isto possibilitou a criação de inúmeros dialetos, permitindo uma comunicação sofisticada entre os seus semelhantes.

Com a aceleração do desenvolvimento tecnológico no início do século passado, tornou-se possível, dentre uma vasta gama de benefícios, o armazenamento e reprodução destes sons antes somente produzidos pela natureza. Com isto, iniciaram-se os estudos de processamento de sinais de áudio, buscando analisar e manipular estes sinais já antes existentes.

Juntamente com a evolução da eletrônica, evoluíram as técnicas de processamento de sinais de áudio, ramificando-se em diferentes áreas de atuação para atender às mais variadas necessidades, como a restauração [3] e codificação [4] deste tipo de sinal.

Dentre estas áreas de atuação existe uma que busca reproduzir artificialmente, ou sinteticamente, os sons produzidos acusticamente por seres vivos e objetos encontrados na natureza. Esta área é conhecida como síntese de sinais de áudio, e apresenta, também, diversas vertentes de estudos e diferentes técnicas para atingir este objetivo.

As técnicas de síntese podem apresentar diversas finalidades. A primeira que nos vem à mente é a de simular sons de natureza real, buscando ao máximo que não haja diferença perceptível entre o som original e o sintetizado. Outra possível finalidade é a de mesclar (chamada na literatura de *morphing*) dois sinais distintos, com o fim de preservar determinadas características presentes em cada um destes sinais. Podemos, também, ter sintetizadores com o intuito de criar sons totalmente novos, o que é muito utilizado em música eletrônica e filmes de ficção.

1.1 Síntese de Voz

Uma das sínteses mais estudadas e almejadas é a síntese da voz de um ser humano, pois, conforme mencionado anteriormente, esta se origina de um complexo sistema vocal com a capacidade de emitir diversos fonemas, sem falar da variação de timbre entre diferentes locutores.

O estudo de síntese de voz pode ser dividido em duas áreas: voz falada e voz cantada (em inglês *speech* e *singing voice*, respectivamente), de forma que o primeiro trata de sinais que representam a fala de um locutor e o segundo, de sinais que representam um indivíduo executando uma determinada linha melódica através do canto.

Em muitos métodos de síntese de voz é importante diferenciar o tratamento que será feito para vogais e consoantes. Isso acontece devido ao fato de as vogais representarem sinais quase periódicos e harmônicos, possibilitando a distinção da altura da nota nela contida (em inglês, *pitch*). Já as consoantes tipicamente representam sinais de curta duração no domínio do tempo, resultando em um espectro não bem comportado e que pode ou não conter informação de *pitch*.

O primeiro sintetizador eletrônico de voz falada foi o *Voder*, produzido pelos laboratórios da Bell e demonstrado em 1939 na feira mundial em Nova York. Já em 1961, também a Bell apresentou a primeira síntese de voz cantada na música *Bicycle Built For Two*, onde o piano também foi sintetizado. Estas duas amostras podem ser encontradas e reproduzidas em [5]. Já em termos de síntese de instrumentos, o primeiro sintetizador comercial como conhecemos foi desenvolvido por Robert Moog e Herbert Deutsch em 1964, e chamado *Moog*.

Ao tentarmos simular qualquer som de natureza real através do processo de síntese pode ser útil certo conhecimento do seu modelo físico. Com isso, a síntese de um sinal de voz se apresenta como uma tarefa desafiadora, muitas vezes mais do que a síntese de um instrumento, pois frequentemente envolve um entendimento, mesmo que não muito aprofundado, da anatomia humana e da fonoaudiologia [6]. Outra difícil tarefa em síntese de voz é a simulação dos fonemas presentes em qualquer linguagem ou dialeto, o que é totalmente desnecessário na síntese de um instrumento.

1.2 Revisão Bibliográfica

A síntese de voz é um tema muito estudado e que pode ser ricamente encontrado na literatura. Vejamos, abaixo, alguns trabalhos de grande relevância em processamento de voz, síntese de áudio, *morphing* e voz cantada, sendo assim relacionados com esta dissertação.

Flanagan e Golden desenvolveram, em 1966, a técnica do *phase vocoder* [7] para

representar sinais de fala pelos seus espectros de fase e amplitude através da transformada de Fourier do sinal (STFT - *Short-Time Fourier Transform*). O intuito era fazer uma economia na largura de banda de transmissão e realizar compressão e expansão do sinal de fala no tempo. Após sua criação, o *phase vocoder* continuou a ser desenvolvido, como em 1973 por Schafer e Rabiner [8], em 1976 por Portnoff [9] e em 1977 por Allen [10], tornando-se uma das técnicas de análise/síntese mais estudadas e utilizadas até hoje.

Um som gerado por um instrumento, ou voz, se dá através da ressonância da onda sonora em seu interior; isto origina a quase periodicidade de uma nota, assim como os seus harmônicos. Ao fazer uma simples análise na frequência utilizando a STFT é fácil perceber que uma nota pura de um instrumento, ou voz, é representada essencialmente por picos no espectro de amplitude. Nesta análise vemos que, para um sinal bem comportado e harmônico, o primeiro pico representa a frequência fundamental (ou o *pitch*) e os demais são as outras parciais harmônicas; suas amplitudes relativas farão a diferenciação de timbres [11].

Com o intuito de analisar o comportamento de um sinal e seus harmônicos ao longo do tempo, Smith e Serra desenvolveram, em 1987, um analisador de espectro chamado *Parshl* [12]. Fazendo uso da STFT, o *Parshl* realiza o rastreamento das amplitudes, frequências e fases de um sinal através dos picos no espectro de amplitude ao longo do tempo. Isso permite a extração de parâmetros necessários para que se possa realizar a síntese aditiva de sinais harmônicos e inarmônicos, como o piano. McAulay e Quatieri apresentaram uma técnica semelhante ao *Parshl*, também com o intuito de rastrear as amplitudes, frequências e fases de cada componente senoidal [13]. Estes parâmetros são rastreados através de um algoritmo simples de detecção de picos. Mudanças bruscas de frequência de determinadas componentes podem ser rastreadas usando-se o conceito de nascimento e morte daquela determinada senoide. Estas técnicas podem ser aplicadas em sinais de fala e de áudio. Em 1992, Rodet e Depalle apresentaram um método de síntese aditiva baseado em envoltórias espectrais e na inversa da transformada de Fourier discreta (iDFT - *Inverse Discrete Fourier Transform*) [14]. Este método permitiu uma redução do custo computacional, facilitando o uso da síntese aditiva em tempo real.

A síntese aditiva permite uma fácil manipulação dos timbres, pois possibilita que alterações sejam feitas diretamente em cada componente harmônica de uma determinada nota. Porém, devido ao seu grande número de parâmetros a serem ajustados e ao seu elevado custo computacional, o método mais popular utilizado em sintetizadores de tempo real segue outro paradigma: o de síntese por amostra (*sampling synthesis*) [15]. Para este método são realizadas gravações de um determinado instrumento (em inglês, *samples*) que serão armazenadas em uma memória. Ao tocar uma nota, o sintetizador seleciona e reproduz uma gravação apropriada,

com o *pitch* e a dinâmica mais próximos dos desejados.

Mesmo com todas estas facilidades da síntese por amostra, Haken *et al.* [16] defendem que este tipo de síntese sofre de algumas limitações. A síntese por amostra possui um número limitado de gravações de cada instrumento acústico; com isso, quando a nota desejada não se encontra na memória é necessário que se faça uma modificação na amplitude e/ou na taxa de amostragem de uma gravação mais próxima do desejado. Este tipo de modificação, porém, dificilmente produzirá todas as variações espectrais associadas às mudanças de dinâmica e *pitch* de um instrumento acústico. Isso provoca uma não naturalidade no sinal sintetizado, tornando-o distinguível do sinal original. A síntese aditiva, por outro lado, permite um fácil controle de cada parcial (como visto anteriormente) e, com o avanço dos computadores e dos DSPs, ela se torna uma alternativa ainda mais atraente e confiável para o uso em tempo real.

Em 1995, Tellman, Haken e Holloway descrevem um algoritmo de interpolação de timbres (*morphing*) [17]. O processo de *morphing* permite uma combinação entre diferentes timbres, criando um novo som com uma duração e um timbre intermediários. O *morphing* também pode ser feito entre notas com diferentes alturas e dinâmicas. Haken *et al.*, em 1997, apresentam um sintetizador chamado *Continuum*, que utiliza um modelo de *morphing* tridimensional, onde o eixo x corresponde ao *pitch*, o eixo y ao timbre e o z à dinâmica [18]. Nos últimos anos, muitos estudos foram realizados com o fim de desenvolver técnicas de *morphing*, como em 2011 na tese de doutorado de Caetano [19], que gerou diversas publicações sobre o assunto [20] [21] [22] [23].

Duas abordagens diferentes de modelagem de voz cantada são apresentadas por Bonada *et al.* em 2001 [24], uma com o intuito de imitar a voz do cantor original em um sistema de caraoquê e outra com o intuito de sintetizar uma voz a partir da partitura e da transcrição fonética de uma música. Em 2003, Bonada *et al.* apresentam uma técnica de síntese de voz cantada utilizando métodos de síntese por amostra [25]. Está técnica foi utilizada no *Vocaloid*, um sintetizador comercial de voz cantada desenvolvido pela Yamaha [26], que participou do *Synthesis of Singing Challenge* (Desafio de síntese de voz cantada) realizado em 2007 [27]. Em 2009, Mayor, Bonada e Janer apresentaram o *Kaleivoicecope* [28], uma tecnologia que permite a transformação da voz humana, como por exemplo, de um homem para uma mulher ou um adolescente para uma senhora. Esta tecnologia tem o intuito de ser usada desde em instalações interativas até em videogames.

1.3 Visão Geral da Dissertação

A dissertação de mestrado de Paiva [29] apresentou técnicas de *morphing* e síntese de sinais de voz cantada com o intuito de realizar a modificação de *pitch* e a transformação de locutor. As técnicas por ele apresentadas são válidas para qualquer sinal de voz cantada, tratando potencialmente todos os fonemas presentes na letra de uma música.

O trabalho desenvolvido na presente dissertação tem por finalidade apresentar melhorias com relação à naturalidade da síntese de voz cantada, bem como na execução de uma melodia e reprodução de um timbre. A principal meta almejada é a eliminação de artefatos que dão a impressão de descontinuidade, inerentes às técnicas da literatura.

Com o intuito de analisar as dificuldades presentes na síntese de uma melodia e reprodução de um timbre, optou-se por restringir este estudo ao tratamento de vogais, pois elas são responsáveis pela maior parte das informações do timbre de um cantor e do *pitch* por ele executado, i.e., da linha melódica presente no sinal. Portanto, a síntese de consoantes não faz parte do escopo deste trabalho.

O algoritmo desenvolvido faz a aquisição de um sinal de referência, que consiste de uma gravação de voz de um cantor qualquer executando a vocalise[1] de uma melodia arbitrária. Este sinal será analisado para que se obtenham a linha melódica e as vogais executadas por ele. De posse destes dados, faz-se a ressíntese desta melodia, porém com um timbre contido em envoltórias espectrais armazenadas em um banco de dados de síntese. Assim, tem-se como saída deste algoritmo um sinal sintetizado contendo a mesma linha melódica, porém como se fosse cantada por outra pessoa.

Dentre as possíveis aplicações deste algoritmo, podemos destacar a substituição de timbre em gravações em estúdio, em shows de música ao vivo e em caraoquês.

Para a realização deste método, o algoritmo deve ser dividido em duas etapas:

1. análise do sinal de referência

2. síntese do sinal de saída

Para todos os passos destas duas etapas, o sinal será dividido em janelas (em inglês, *frames*) de 1024 amostras, que para uma taxa de amostragem de 44,1 kHz correspondem a 23,2 ms. As janelas utilizadas são janelas de Hann com 50 % de sobreposição.

[1] Vocalise é uma técnica vocal que consiste na execução de uma linha melódica somente através de vogais, sem a emissão de palavras. Esta técnica é muito utilizada em exercícios de canto.

1.3.1 Análise

A etapa de análise consiste na aquisição da melodia executada pelo sinal de referência através da detecção do *pitch* ao longo do tempo. Como está sendo realizada a síntese de um sinal de voz, além da melodia é necessário detectar o fonema emitido pelo sinal de referência (no caso, a vogal). Técnicas de detecção de *pitch* e vogal serão discutidas ao longo desta dissertação.

1.3.2 Síntese

Em setembro de 2010, Fonseca e Ferreira apresentaram um trabalho semelhante a este, com um algoritmo que também recebe um sinal de referência, analisa-o e faz a ressíntese da mesma música com o timbre de outra pessoa [30]. Porém, o método de síntese por eles utilizado vem a ser a síntese por amostra (*sampling synthesis*). Conforme visto na Seção 1.2, é comum vermos sintetizadores de voz cantada que utilizam a técnica de síntese por amostra.

A técnica de síntese escolhida para ser utilizada neste trabalho é a síntese aditiva, que, ao contrário da síntese por amostra, permite uma grande flexibilização de cada parcial do espectro. Para que haja uma redução do custo computacional, o sinal será criado diretamente no domínio da frequência e depois transformado ao domínio do tempo através da DFT inversa [14].

Para que isto seja realizado, envoltórias espectrais contendo o timbre do cantor desejado necessitam ser estimadas e armazenadas em um banco de dados de síntese. Estas envoltórias determinarão a amplitude de cada parcial no espectro de módulo, o que será responsável por reproduzir o timbre desejado.

Técnicas de *morphing* de envoltórias espectrais serão utilizadas para tornar mais suaves as transições entre diferentes vogais e entre os *frames* do sinal.

1.3.3 Organização do Trabalho

Após este capítulo introdutório, o Capítulo 2 deste texto descreverá técnicas de detecção de *pitch*, propondo melhorias e elegendo a técnica mais apropriada a este trabalho.

O Capítulo 3 descreverá os conceitos de envoltória espectral e de formantes, conceitos estes importantes para o desenvolvimento desta dissertação. Para este fim, este capítulo descreverá o modelo LPC e como ele pode ser utilizado para estimar a envoltória espectral de um sinal de áudio.

Algumas técnicas de detecção de vogal serão apresentadas no Capítulo 4, onde também será demonstrada a criação de um banco de dados contendo a probabilidade da posição no espectro de cada formante para cinco vogais da língua portuguesa.

Com o intuito de suavizar possíveis mudanças bruscas entre as janelas sintetizadas, o Capítulo 5 descreverá técnicas de *morphing* de timbres, para que haja uma interpolação dos timbres a cada grupo de janelas sintéticas. As técnicas de *morphing* descritas neste capítulo também serão utilizadas para a obtenção de valores intermediários entre as vogais *a*, *e*, *i*, *o* e *u*, obtidas pelo método de detecção de vogal do Capítulo 4.

Já o Capítulo 6 descreverá a técnica de síntese utilizada, bem como os resultados finais com ela obtidos. Este capítulo apresentará a criação de um banco de dados de síntese, que conterá envoltórias espectrais para cada vogal em diversos *pitches* para cada cantor do qual se deseja simular a voz.

Finalmente, no Capítulo 7 encontraremos a conclusão desta dissertação, apontando todos os pontos positivos e negativos encontrados nas técnicas e métodos aqui utilizados.

Capítulo 2

Algoritmos de Detecção de *Pitch*

2.1 Introdução

O *pitch* é a altura (associada à frequência) de um som percebida pelo sistema auditivo humano. Sendo assim, devidamente quantizado, associa-se a uma nota musical, que tanto pode ser reproduzida por um instrumento como pela voz humana.

Para sinais harmônicos, o *pitch* vem a ser igual à sua frequência fundamental. Já em sinais inarmônicos, apenas sob condições específicas se pode falar em um *pitch* único, e nesses casos (mesmo que sua determinação perceptiva seja simples), sua estimação automática vem a ser uma tarefa muito complexa.

Conforme mencionado no primeiro capítulo desta dissertação, a primeira etapa deste trabalho consiste na análise do sinal de referência; deseja-se extrair a melodia e os fonemas emitidos, para que com estas informações seja possível sintetizar um sinal com as mesmas características.

Para que isso seja possível, o sinal foi dividido em janelas (ou *frames*) de curta duração; em cada janela, foi aplicado um método de detecção de *pitch* para sua extração. Desta forma, a melodia do sinal de referência é adquirida.

Este capítulo relata três algoritmos diferentes que promovem esta detecção. Após serem descritos, é apresentada uma análise comparativa entre eles, em que se verifica a diferença quantitativa média entre os seus resultados, faz-se uma análise subjetiva e, por fim, analisa-se o custo computacional de cada método.

O objetivo final é selecionar o método mais eficaz, dentro das necessidades deste trabalho como um todo.

2.2 Algoritmo da Autocorrelação

Um dos métodos mais tradicionais de detecção de *pitch* é o da autocorrelação (ACF - *Autocorrelation Function*) [31] [32]. A autocorrelação é dada pelo somatório dos

produtos ponto a ponto de um sinal por ele mesmo defasado de τ amostras, conforme a equação abaixo.

$$r_t(\tau) = \sum_{j=t+1}^{t+W-\tau} x_j x_{j+\tau}, \tag{2.1}$$

onde x é o sinal em questão, W o tamanho da janela de observação e t o índice de tempo.

Se pensarmos que estamos lidando com sinais periódicos ou quase periódicos, pois não faria sentido falarmos em *pitch* para sinais sem periodicidade, é fácil deduzir que a autocorrelação será mais alta quando calculada para um atraso τ igual ao valor de um período do sinal. A maior autocorrelação possível se dá com τ igual a zero. Se o sinal tem período T, esse máximo também ocorrerá em valores de τ iguais a múltiplos de T. Na situação real, o período fundamental não coincide necessariamente com algum τ inteiro. Com isto, o *pitch* pode ser determinado aproximadamente pelo valor de τ correspondente ao maior pico (máximo global) presente nesta função, com $\tau > 0$.

Um erro comum neste método se dá quando a energia da frequência fundamental é muito baixa mediante as outras parciais harmônicas do sinal. Isto pode gerar erros em que a resposta do algoritmo será um múltiplo do *pitch* do sinal (frequentemente uma oitava).

2.2.1 Algoritmo YIN

Cheveigné e Kawahara desenvolveram um algoritmo para refinar os resultados do método da autocorrelação para detecção de *pitch*, chamado YIN [33]. Este algoritmo envolve uma sequência de seis passos.

O primeiro é o cálculo da autocorrelação, conforme descrito acima. A Figura 2.1 ilustra um exemplo de autocorrelação para um sinal de voz cantada.

No segundo passo, calcula-se a função diferencial $d_t(\tau)$ dada por

$$d_t(\tau) = r_t(0) + r_{t+\tau}(0) - 2r_t(\tau). \tag{2.2}$$

Se o algoritmo terminasse neste passo, o *pitch* seria tomado como o menor vale desta função.

O uso da função diferencial reduz o erro causado por um possível aumento da amplitude do sinal ao longo do tempo, o que pode fazer com que um atraso τ tenha amplitude maior do que a amplitude do τ referente ao próprio *pitch*. A função diferencial pode ser vista na Figura 2.2.

No terceiro passo, calcula-se a média normalizada acumulada da função diferencial $d'_t(\tau)$, que é igual a 1 quando $\tau = 0$, e para os demais valores de τ é dada

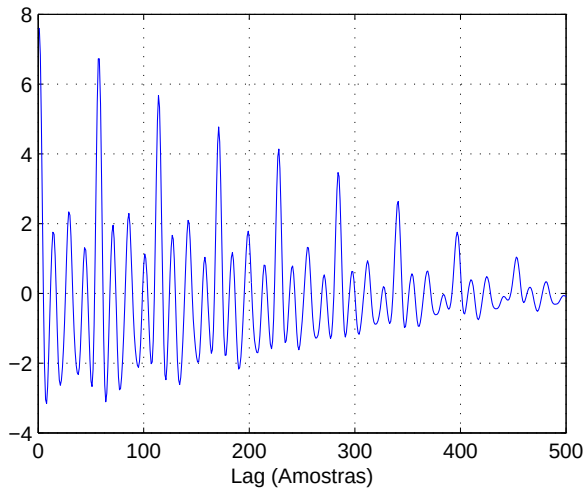


Figura 2.1: Primeiro passo do algoritmo YIN - Função Autocorrelação.

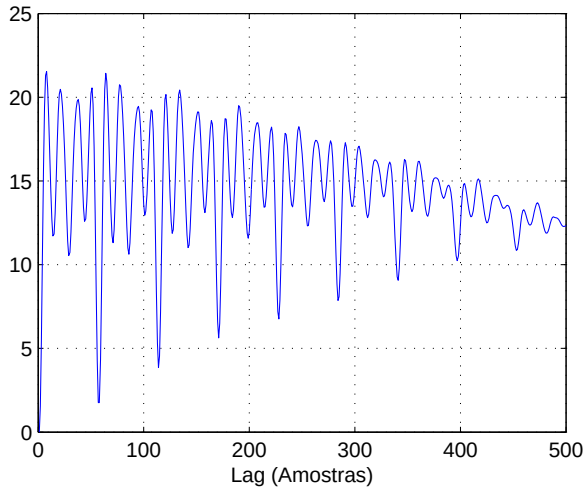


Figura 2.2: Segundo passo do algoritmo YIN - Função Diferencial.

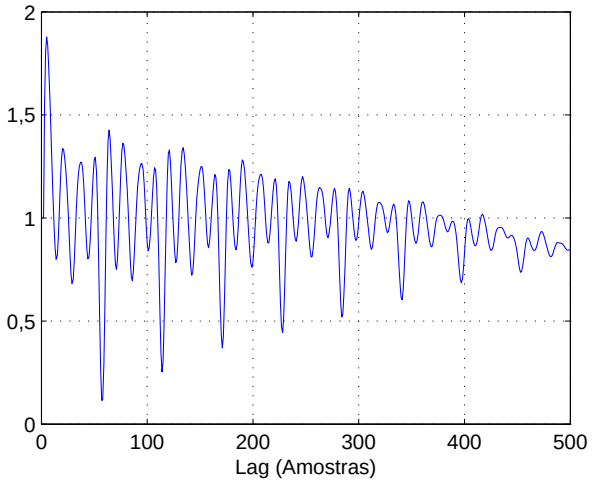


Figura 2.3: Terceiro passo do algoritmo YIN - Média Normalizada Acumulada da Função Diferencial.

por

$$d_t^{'}(\tau) = \frac{d_t(\tau)}{\left[(1/\tau)\sum_{j=1}^{\tau} d_t(j)\right]}. \tag{2.3}$$

Esta função sempre começará em 1 e tenderá a se manter alta para baixos valores de τ; com isso, evita-se encontrar um valor de *pitch* erroneamente muito alto. Outro benefício é o fato de normalizar a função para o quarto passo. Um exemplo pode ser visto na Figura 2.3.

No quarto passo, admite-se um valor de limiar para $d_t^{'}(\tau)$ tal que a localização do *pitch* é dada pelo menor valor de τ onde um mínimo da função $d^{'}$ encontra-se abaixo deste limiar. Não havendo valor de $d^{'}$ abaixo deste limiar, admite-se o mínimo global da função. Isso reduz o erro de oitavas, onde sub-harmônicos se encontrariam em vales mais profundos do que o do verdadeiro *pitch*.

No quinto passo, faz-se uma interpolação parabólica visando a contornar o erro causado quando o *pitch* não é um múltiplo da taxa de amostragem, o que pode ocasionar um erro de até metade do período de amostragem.

Em determinados intervalos não estacionários do sinal de voz, a estimativa pode falhar em algum ponto em que o período coincida com um valor relativamente alto de $d^{'}(T_t)$, onde T_t é o período estimado no tempo t. Já em um tempo $t^{'}$, a estimativa pode estar correta e o valor de $d^{'}(T_{t'})$ menor.

O sexto passo procura, para cada índice de tempo t, um mínimo de $d^{'}(T_\theta)$ para θ dentro de um pequeno intervalo $[t - T_{max}/2, t + T_{max}/2]$, sendo T_θ a estimativa no tempo θ e T_{max} o maior período esperado. Baseado na estimativa inicial, o algoritmo é aplicado novamente dentro de um pequeno intervalo de busca para que se obtenha a estimativa final.

Em [33], T_{max} é escolhido como o número de amostras que corresponde a 25 ms. Como foram utilizadas janelas de duração inferior a T_{max}, este passo não foi utilizado nesta dissertação; isso não acarretará grandes diferenças no resultado, já que segundo Cheveigné e Kawahara, este passo reduziria a taxa de erro de 0,77 % para 0,5 % [33].

2.3 Algoritmo Cepstral

A transformada cepstral, publicada em 1963 por Bogert, Healy e Tukey [34], possui diversas aplicações em processamento de sinais, como em reconhecimento de voz [35] e em análise de eco em sinais sísmicos [36]. Esta transformada também apresenta bons resultados em algoritmos de detecção de *pitch* [37].

O nome cepstral originou-se da inversão das letras da palavra *spectral* (espectral em inglês), assim como cepstrum da palavra *spectrum* (espectro em inglês). A variável independente do cepstrum é a *quefrency*, que também se originou de um jogo com a palavra *frequency* (frequência). Daqui em diante, será utilizada a palavra quefrência em vez de *quefrency*.

A razão para este jogo de palavras pode ser melhor compreendida ao analisarmos a construção do cepstrum de um sinal. A transformada cepstral de um sinal $x[n]$ é dada pela transformada de Fourier inversa do módulo de seu espectro $|X(e^{j\omega})|$, isto é

$$c[n] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log \left| X(e^{j\omega}) \right| e^{j\omega n} d\omega, \tag{2.4}$$

sendo o espectro do sinal dado por

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n}. \tag{2.5}$$

Observando o cepstrum de um trecho sonoro (em inglês, *voiced*) de um sinal, podemos identificar, além do pico central, a presença de um pico na quefrência referente ao *pitch*, o qual também aparece rebatido nas quefrências negativas. Por outro lado, ao analisarmos o cepstrum de um trecho surdo (em inglês, *unvoiced*) de um sinal, nenhum pico significativo, além do central, é apresentado. As Figuras 2.4 e 2.5 são exemplos de cepstrum para um sinal sonoro e um sinal surdo, respectivamente.

Desta forma, para determinação do *pitch* deste sinal, basta procurar pelo maior pico do cepstrum, desconsiderando picos referentes a quefrências muito baixas, pois referem-se a *pitches* muito altos e inalcançáveis por voz humana.

Este algoritmo também permite a identificação de trechos sonoros e surdos. Para que isto seja feito, basta verificar a existência ou não de picos significativos ao longo do sinal.

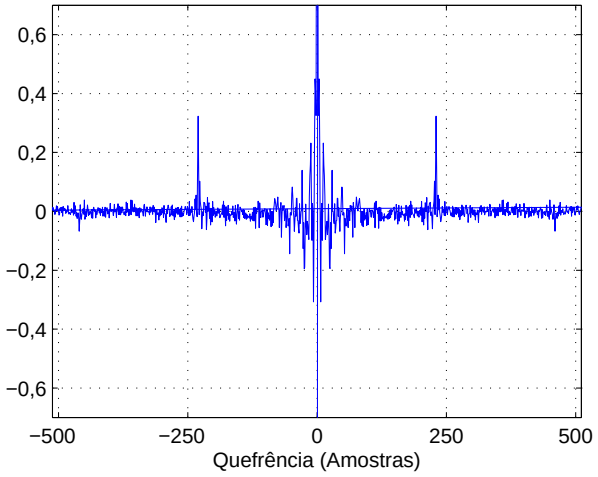


Figura 2.4: Cepstrum de um sinal sonoro (*voiced*).

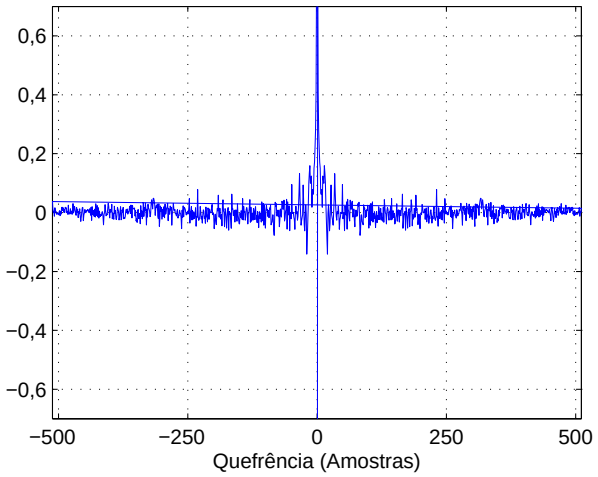


Figura 2.5: Cepstrum de um sinal surdo (*unvoiced*).

2.4 Algoritmo Logaritmo do Produto do Espectro Harmônico

O algoritmo Produto do Espectro Harmônico (HPS - *Harmonic Product Spectrum*) [38] [37], é dado por

$$P(e^{j\omega}) = \prod_{r=1}^{K} \left|X(e^{j\omega r})\right|^2, \tag{2.6}$$

onde $|X(e^{j\omega})|$ é o módulo do espectro do sinal $x[n]$ e r um número inteiro. Ao efetuarmos o logaritmo da equação acima, temos

$$\hat{P}(e^{j\omega}) = 2\sum_{r=1}^{K} \log\left|X(e^{j\omega r})\right|, \tag{2.7}$$

chamado de logHPS.

Conforme descrito na equação (2.7), a função $\hat{P}(e^{j\omega})$ é obtida através da soma do espectro de módulo de um sinal com $K-1$ compressões suas.

Para esclarecer este fato, lembremos da propriedade da expansão temporal da DTFT (*Discrete-Time Fourier Transform*) pela qual, ao multiplicarmos $j\omega$ por um número positivo r, teremos a expansão de $x_{(r)}[n]$ e a compressão de $X(e^{j\omega r})$ [39],

$$x_{(r)}[n] \xleftrightarrow{\mathcal{F}} X(e^{j\omega r}). \tag{2.8}$$

Note que foi utilizado o sinal $x_{(r)}[n]$. Isso porque $x[n]$ é um sinal discreto, e com isso não é possível realizar a sua expansão, já que não existem valores entre amostras do sinal. Por este motivo, definiu-se o sinal $x_{(r)}[n]$ (sendo r um número inteiro) que é tido como zero em valores não existentes em $x[n]$, isso é

$$x_{(r)}[n] = \begin{cases} x[n/r], & \text{se } n \text{ for um múltiplo de } r \\ 0, & \text{se } n \text{ não for um múltiplo de } r. \end{cases} \tag{2.9}$$

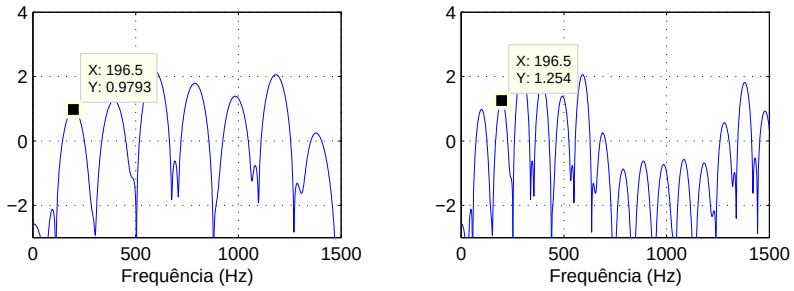
O mesmo resultado será obtido se estas compressões do espectro forem implementadas como decimações do sinal na frequência, uma vez que estamos lidando com sinais discretos também no domínio da frequência.

Desta forma, este algoritmo se beneficia do fato de estarmos lidando com sinais harmônicos, pois a cada compressão do espectro o valor da amostra referente à frequência fundamental será somado às demais parciais harmônicas deste sinal. Após algumas iterações do método, o valor da amostra referente à frequência fundamental possuirá uma amplitude maior do que a amplitude das demais parciais harmônicas, diminuindo a tendência de ocorrência de erros de oitavas na detecção do *pitch*.

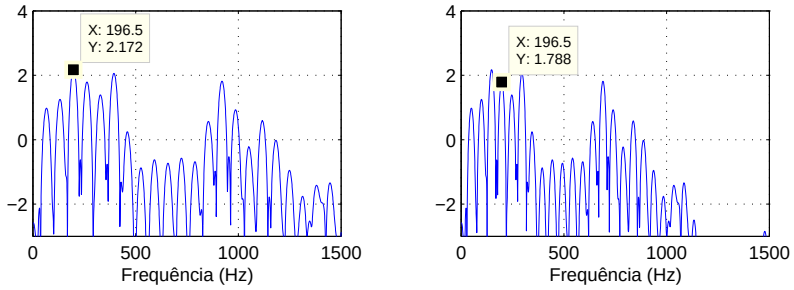
Este método também se mostra eficaz em casos onde a frequência fundamental

encontra-se suprimida, pois, após algumas iterações, aparecerá um pico na amostra referente à frequência fundamental.

A Figura 2.6 apresenta o espectro de $\log|X(e^{j\omega r})|$ para valores de $r = 1$, 2, 3 e 4, i.e., o espectro do sinal sem compressão e suas duas primeiras compressões. Já a Figura 2.7 apresenta o somatório das r primeiras compressões, para $r = 3$ e 4. Podemos ver que, neste exemplo, foram necessárias quatro iterações do método para se obter o *pitch* do sinal. Estas figuras representam um trecho sonoro de um sinal de voz cantada executando a vogal *a* em um *pitch* igual a 196,5 Hz, que vem a ser a nota musical sol na terceira oitava.

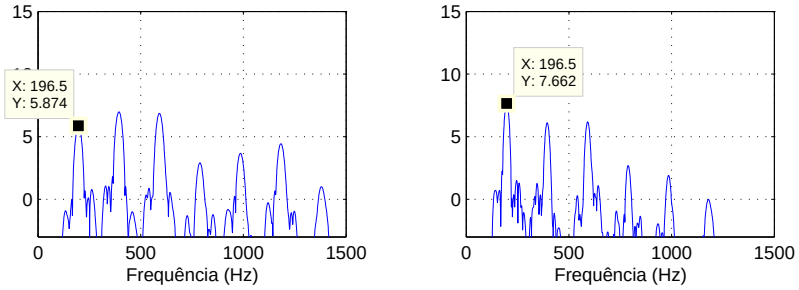


(a) $r = 1$ (b) $r = 2$



(c) $r = 3$ (d) $r = 4$

Figura 2.6: Representação de diferentes compressões do espectro, com valores de r = 1, 2, 3 e 4.



(a) somatório em $r = 3$ (b) somatório em $r = 4$

Figura 2.7: Somatório das r primeiras compressões do espectro, com valores de r = 3 e 4.

2.4.1 Limitações quanto ao número de iterações

A cada iteração do algoritmo logHPS, o sinal é comprimido por uma taxa r, i.e., o tamanho do sinal (em amostras) N passa a ser r vezes menor. Desta forma, este algoritmo possui limitações quanto ao seu número total de iterações K, pois à medida que comprimimos o espectro de um sinal, menor se torna a sua frequência máxima f_M. Como estamos lidando com sinais de voz, esta limitação tomará como referência a frequência máxima de *pitch* produzida por um ser humano, que é igual a 1100 Hz.

Pelo teorema da amostragem, sabemos que a frequência máxima de um sinal digital f_M é igual a metade de sua taxa de amostragem f_s [40]. Como a resolução frequencial no domínio da DFT (*Discrete Fourier Transform*) é igual a f_s/N, e a frequência máxima deverá ser igual a 1100 Hz, a relação abaixo deverá ser obedecida:

$$f_M \leq \frac{N}{2K} \cdot \frac{f_s}{N}. \tag{2.10}$$

Simplificando, temos

$$f_M \leq \frac{f_s}{2K}. \tag{2.11}$$

Desta forma, podemos obter o valor máximo para o número de iterações K pela relação

$$K \leq \frac{f_s}{2f_M}. \tag{2.12}$$

Portanto, para um sinal com taxa de amostragem igual a 8 kHz o número máximo de iterações K deverá ser igual a 3, pois $\frac{8000}{2 \cdot 3} = 1333{,}3$ Hz. Se K for igual a 4, a relação da equação (2.12) não será obedecida, pois $\frac{8000}{2 \cdot 4} = 1000$ Hz, menor do que o *pitch* máximo produzido por voz humana.

Para um sinal com taxa de amostragem igual a 44,1 kHz o valor máximo de K será igual a 20, pois $\frac{44100}{2 \cdot 20} = 1102{,}5$ Hz.

Após algumas simulações, é possível perceber que este método não apresenta bons resultados para sinais com frequência de amostragem iguais 8 kHz, pois apenas 3 iterações nem sempre são suficientes para se obter o *pitch* correto. Por isso, os sinais de 8 kHz foram simulados com mais de 3 iterações, resultando, assim, na incapacidade do método de detectar *pitches* muito altos nessa taxa de amostragem.

Por outro lado, 20 iterações para sinais de 44,1 kHz se mostraram muito além do necessário. Após algumas simulações, chegou-se ao valor empírico de 10 iterações para este tipo de sinal. Desta forma, o *pitch* continua sendo detectado devidamente, e o custo computacional reduz-se significativamente.

2.4.2 Resolução Frequencial

Outro aspecto ao qual devemos atentar é a resolução frequencial. Como o número de amostras do sinal será reduzido a cada iteração deste método, a resolução frequencial piorará, pois o menor intervalo de frequência discernível IF passará a ser igual a $\frac{f_s}{N/r}$, ou ainda $\frac{r \cdot f_s}{N}$; ao final das K iterações, teremos como intervalo $\frac{K \cdot f_s}{N}$.

Esta piora na resolução frequencial resulta em uma detecção de *pitch* menos precisa. Uma solução para esta questão é o aumento do tamanho da DFT (N) através da utilização de *zero padding*[1].

Para determinar o quanto de *zero padding* deve ser realizado, é necessário calcular o tamanho do sinal para que, após K compressões, o erro na detecção permaneça dentro de uma faixa admissível. Neste trabalho, optou-se por um intervalo de frequência discernível máximo admissível $\mathrm{IF_M}$ de 10 Hz. Desta forma, tem-se um erro máximo de ± 5 Hz na determinação do *pitch*.

Com isso, pode-se estabelecer a relação

$$\mathrm{IF} = \frac{K \cdot f_s}{N}, \tag{2.13}$$

tal que o tamanho da DFT em função do intervalo será

$$N = \frac{K \cdot f_s}{\mathrm{IF}}. \tag{2.14}$$

Para determinar o tamanho que a DFT deverá possuir antes das compressões, de forma que após as K iterações seja mantida a resolução frequencial melhor que em um intervalo frequencial igual a $\mathrm{IF_M}$, deve-se respeitar a relação

$$N \geq \frac{K \cdot f_s}{\mathrm{IF_M}}, \tag{2.15}$$

pois quanto maior o valor de N, menor será o intervalo frequencial, e assim, melhor a resolução frequencial.

É importante lembrar que, para aumentar o desempenho computacional, N deve ser sempre uma potência de dois. Assim, deve-se adotar para N a primeira potência de dois maior ou igual à relação acima.

Conforme visto na Seção 2.4.1, um sinal com taxa de amostragem igual a 44,1 kHz deveria ter um número máximo de 20 iterações. Para este número de iterações, o tamanho da DFT, considerando $\mathrm{IF_M}$ igual a 10 Hz, deveria ser igual a 88.200 amostras, ou melhor, 131.072 amostras para que fosse igual a uma potência de dois. Com isso, pode-se entender o porquê de se terem adotado somente 10 iterações: assim

[1] *Zero padding* consiste em adicionar zeros nos extremos do sinal no domínio do tempo com o intuito de aumentar a resolução da representação discreta do espectro.

teremos uma janela de 44.100 amostras, ou 65.536 em potência de dois, reduzindo o custo computacional pela metade.

2.4.3 Determinando o *pitch*

Muitas vezes, o máximo global da função representará o *pitch* do sinal logo nas primeiras iterações e manterá este valor até o final; outras vezes, o *pitch* correto só aparecerá nas últimas iterações. Situações como estas não gerariam erro, pois bastaria que fosse verificado o máximo global da função após as K iterações e eleger a frequência relacionada a ele como o *pitch* encontrado. Contudo, existem situações em que o máximo global da função representa o *pitch* do sinal nas primeiras iterações, mas termina em valores que não indicam o verdadeiro *pitch*.

Uma forma bastante eficaz encontrada para solucionar está questão é a verificação do máximo global de cada iteração. Logo no início, elege-se o primeiro máximo global como sendo o possível *pitch*. A cada iteração, é verificado se o máximo encontrado é um múltiplo do possível *pitch* ou vice-versa (considerando um pequeno erro de precisão de ± 5 %). Caso seja um múltiplo, elege-se o menor entre eles, pois o maior é compreendido como um harmônico do menor. Caso não seja um múltiplo, é eleito o maior entre eles.

Desta forma, o erro deste método é reduzido significativamente.

2.5 Ajustes Finais

Cada um dos métodos aqui descritos apresenta alguns erros em determinados *frames* de certos exemplos de sinais de áudio. A próxima seção fará uma comparação detalhada entre eles.

Porém, é possível fazer um ajuste simples nestes resultados, que eliminará erros ocorridos em intervalos de poucos *frames* consecutivos como os que podem ser vistos na Figura 2.8, que ilustra a evolução na detecção do *pitch* ao longo dos *frames* de um sinal.

Este ajuste final consiste em detectar picos (ou vales) no vetor que contém a detecção do *pitch*, no qual cada elemento representa uma janela do sinal. Como cada janela de 1024 amostras, em uma taxa de amostragem de 44,1 kHz, representa 23,2 ms do sinal, é razoável supor que este pico (ou vale) está associado a um erro na detecção, pois é impossível um ser humano alterar o *pitch* desta forma por tão pouco tempo.

A Figura 2.9 ilustra o sinal da Figura 2.8 após este ajuste final.

O maior pico (ou vale) admissível neste trabalho é de 5 *frames*, que representam 116,1 ms. Abaixo deste valor, o algoritmo irá encará-lo como errôneo, e implemen-

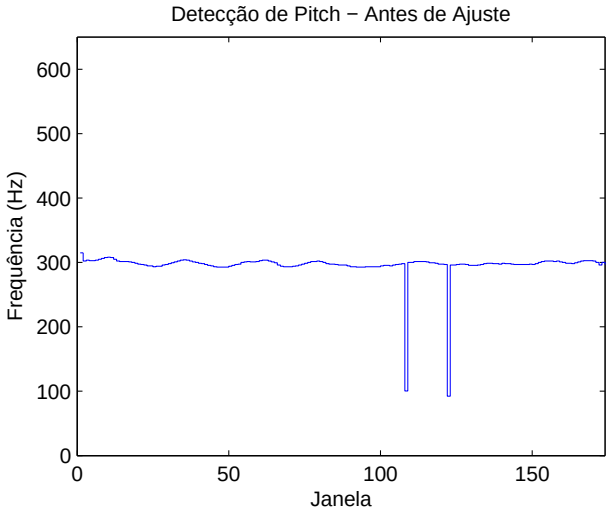


Figura 2.8: Evolução na detecção do *pitch* ao longo dos *frames* de um sinal - antes do ajuste final.

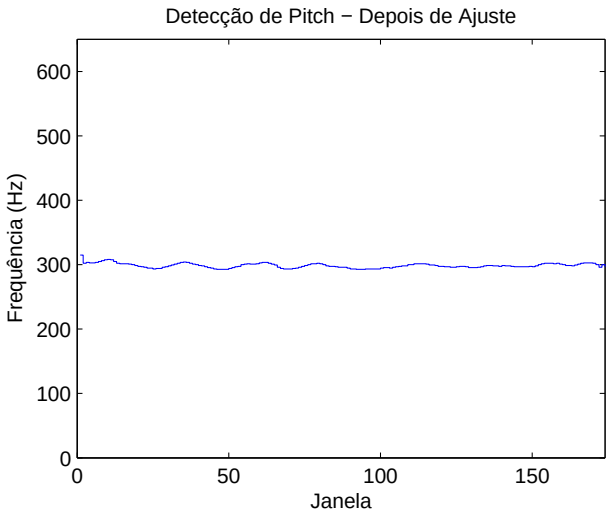


Figura 2.9: Evolução na detecção do *pitch* ao longo dos *frames* de um sinal - após o ajuste final.

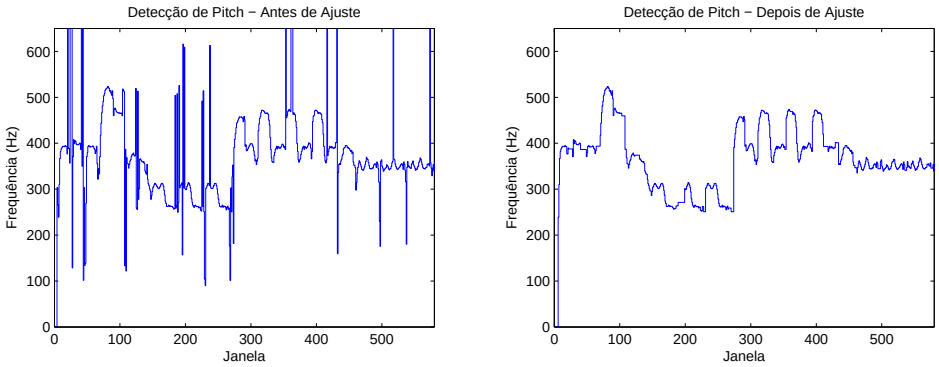


Figura 2.10: Evolução do *pitch* ao longo de um sinal de voz cantada antes e depois do ajuste final.

tará o devido ajuste.

A Figura 2.10 exemplifica um sinal complexo, em termos de harmonicidade, antes e depois deste ajuste final. Nela podemos ver o quanto este ajuste é significativo na detecção do *pitch*.

2.6 Comparação entre os métodos

Foram realizados testes com diferentes sinais utilizando cada um dos três métodos citados acima. Vejamos a comparação entre os três métodos para dois sinais diferentes. O primeiro é um sinal de *humming*[2] com taxa de amostragem igual a 8 kHz. O segundo é um sinal de vocalise em que o cantor emite uma única nota vibrato[3]; neste a taxa de amostragem é igual a 44,1 kHz.

As Figuras 2.11 e 2.12 mostram a evolução do *pitch* ao longo do tempo utilizando os métodos YIN, Cepstral e logHPS para os sinais de *humming* e vocalise, respectivamente.

Nestas figuras, vemos que os algoritmos YIN e Cepstral podem assumir valores de *pitch* muito altos e irreais em frames surdos do sinal. Isto acontece pois, na ausência de um *pitch*, o sinal não apresenta um pico significativo. Como estes algoritmos procuram pelo máximo global do sinal, em sinais surdos eles o encontram nas regiões mais próximas de zero, pois, nos três casos, é a região de maior energia, como pode ser visto nas Figuras 2.3 e 2.4.

Como os algoritmos YIN e Cepstral operam no domínio do tempo, quanto mais próximo de zero estiver o mínimo ou máximo, respectivamente, da função, maior será o valor do *pitch*. Este comportamento justifica os picos de *pitch* encontrados nas Figuras 2.11 e 2.12. Para solucionar este erro, seria necessária a utilização de um algoritmo para detectar se o sinal é sonoro ou surdo (VUS - *voiced/unvoiced/silence*)

[2] *Hum* em inglês é sussurro, *humming* é o ato de cantarolar uma melodia, sem emitir palavras.
[3] Vibrato é uma técnica musical que consiste na oscilação da frequência de uma nota.

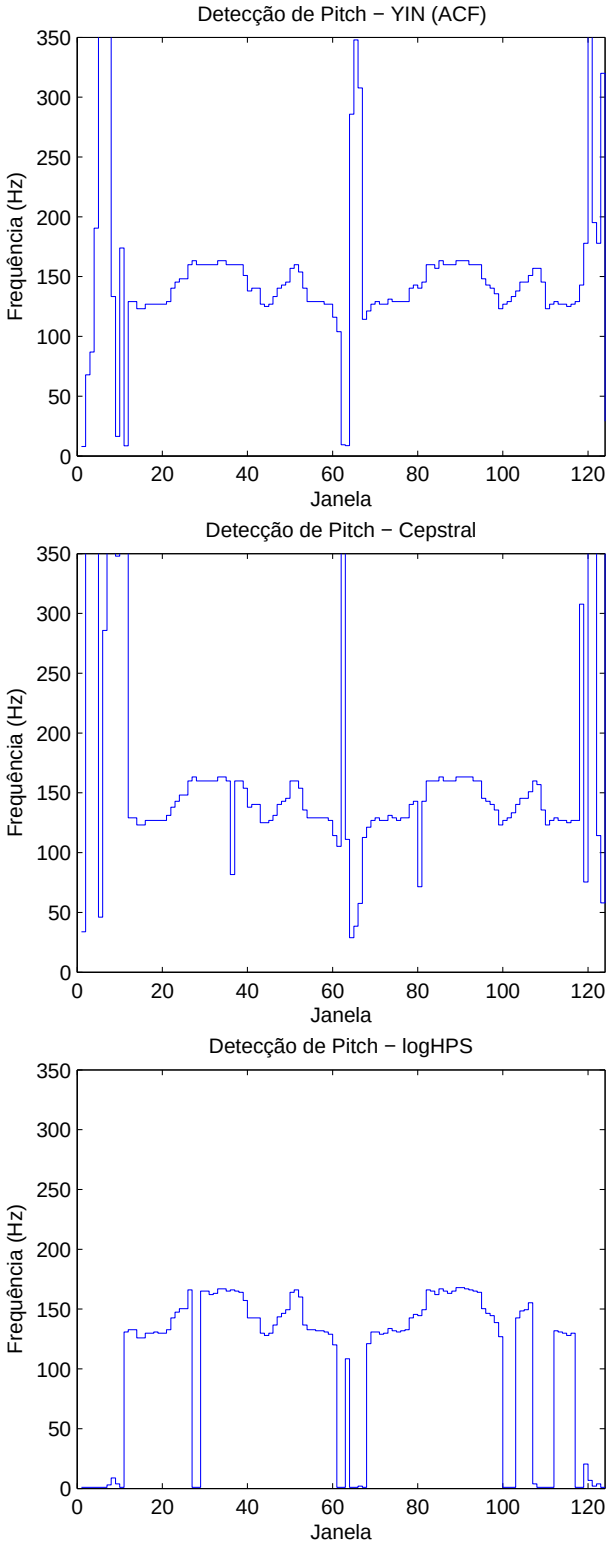


Figura 2.11: Evolução do *pitch* ao longo do tempo para um sinal de *humming* pelo método YIN, Cepstral e logHPS, respectivamente.

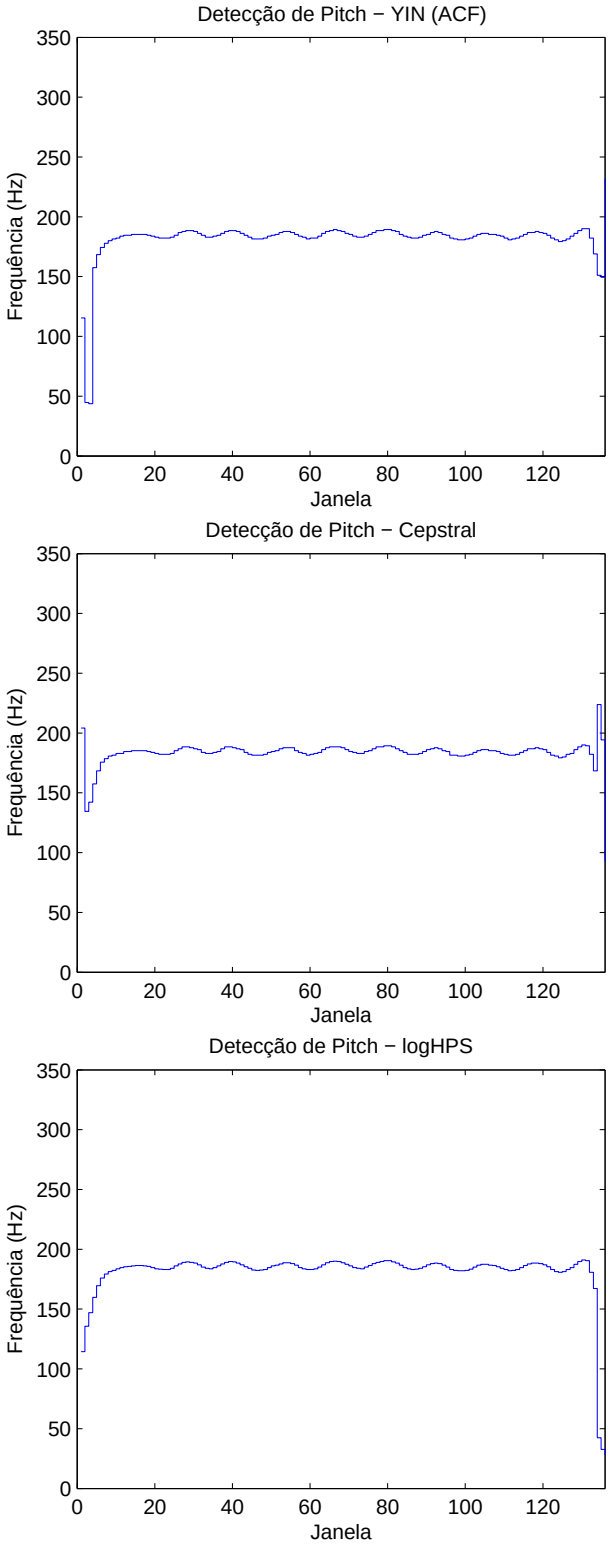


Figura 2.12: Evolução do *pitch* ao longo do tempo para um sinal de vocalise pelo método YIN, Cepstral e logHPS, respectivamente.

[41] [37].

Já o algoritmo logHPS se beneficia do fato de trabalhar no domínio da frequência, pois quanto mais próximo do zero estiver o máximo da função, menor será o valor do *pitch*. Com isto, em sinais surdos, onde há ausência de *pitch*, ao encontrar um pico próximo de zero o *pitch* relacionado também será próximo de zero, tornando-se desnecessário o uso de algoritmos para detectar se o sinal é sonoro ou surdo.

2.6.1 Diferença Média

Para efeito de comparação da detecção do *pitch* nos trechos sonoros destes sinais, foi calculada, após os ajustes finais apresentados na Seção 2.5, a diferença relativa percentual média entre os métodos para os dois sinais aqui analisados. Para tal, arbitrou-se como base de comparação o método logHPS, como podemos ver na Tabela 2.1.

Tabela 2.1: Diferença relativa percentual média da detecção de *pitch* entre os métodos para os sinais de *humming* e vocalise, tomando como base o método logHPS.

Sinal de *Humming*	
YIN em relação ao logHPS (%)	Cepstral em relação ao logHPS (%)
1,9391	1,9578
Sinal de Vocalise	
YIN em relação ao logHPS (%)	Cepstral em relação ao logHPS (%)
0,4363	0,45473

A máxima diferença relativa percentual média foi de 1,9578 %, um indício de que todos estes métodos funcionam de forma comparável em trechos sonoros do sinal.

2.6.2 Análise Subjetiva

Realizou-se ainda outra comparação através de uma análise subjetiva.

Esta análise consistiu na criação de um sinal puramente senoidal, com sua frequência fundamental f_0 igualada ao *pitch* detectado em cada intervalo de *frame* de cada um destes dois sinais testados. Este sinal senoidal foi somado ao sinal original, reproduzido e analisado, com a finalidade de se perceber algum erro aparente na detecção.

O resultado desta análise subjetiva foi positivo para todos os métodos, levando em consideração os trechos sonoros. Não foi possível perceber quaisquer variações entre o *pitch* das notas reproduzidas pelos sinais originais e o sinal senoidal. Porém, nos trechos surdos, notou-se uma clara distinção entre estes sinais para os métodos

YIN e Cepstral, o que já não ocorreu com o método logHPS devido às considerações mencionadas acima.

2.6.3 Custo Computacional

Por fim, estes três métodos foram comparados em termos de custo computacional, calculando-se a média do tempo levado, em cada *frame* do sinal, para a execução dos mesmos. A Tabela 2.2 apresenta os valores, em segundos, do tempo médio para cada um destes métodos. Estes valores foram obtidos em um PC de 64 bits com processador de dois núcleos de 2 GHz e memória de 4 GBytes. Estes algoritmos foram desenvolvidos e executados no *software* Matlab®.

Tabela 2.2: Custo computacional médio para execução dos três métodos nos sinais de *humming* e vocalise.

Sinal de *Humming*		
YIN (s)	Cepstral (s)	logHPS (s)
0,061236	0,00029343	0,0013722
Sinal de Vocalise		
YIN (s)	Cepstral (s)	logHPS (s)
0,061881	0,00032952	0,012264

Vemos que o tempo médio de processamento do método Cepstral é muito inferior ao dos demais, e que o método YIN apresenta o maior custo computacional. Ambos apresentam valores muito similares para os dois sinais em questão (*humming* e vocalise). Já o método logHPS teve um aumento no custo computacional de uma ordem de grandeza de um sinal para o outro, devido ao aumento no número de iterações (ver Seção 2.4.1), porém ainda permanecendo mais rápido do que o método YIN.

2.7 Conclusão

Este capítulo descreveu três diferentes algoritmos de detecção de *pitch*, todos eles comuns em processamento de sinais de voz cantada. Para o algoritmo Logaritmo do Produto do Espectro Harmônico (logHPS), foram propostas algumas melhorias em sua implementação. Foi possível, também, realizar algumas comparações e destacar vantagens e desvantagens de cada método.

Todos os algoritmos apresentaram resultados bastante satisfatórios nos trechos sonoros dos sinais. Porém, somente o logHPS apresentou em trechos surdos resultados tratáveis diretamente, o que torna dispensável o uso de algoritmos de detecção de trechos sonoros e surdos.

Quanto ao custo computacional, o logHPS obteve o segundo melhor desempenho, ficando atrás do Cepstral. Porém, se considerarmos que o Cepstral necessita de um algoritmo de detecção de trechos sonoros e surdos para funcionar adequadamente, possivelmente o seu custo computacional total será igual ou maior que o custo do logHPS.

Por estas razões, optou-se por utilizar, nesta pesquisa, o algoritmo logHPS.

Capítulo 3

Envoltória Espectral em Sinais de Voz

3.1 Introdução

Um dos objetivos principais da síntese de um determinado sinal de áudio é a produção do timbre desejado, definido pelas amplitudes relativas de seus harmônicos no espectro e sua evolução ao longo do tempo. Isso não vale somente para sinais de voz, mas para qualquer sinal harmônico encontrado na natureza.

A envoltória espectral é um dos parâmetros mais importantes na realização da síntese aditiva através da inversa da DFT, pois contém a informação do timbre do sinal a ser sintetizado, já que molda a amplitude de cada parcial harmônica do sinal sintético.

Porém, não só a etapa de síntese faz uso da envoltória espectral. Esta também apresenta grande relevância na análise da vogal contida em um sinal de voz; isso porque os métodos de detecção de vogal que serão apresentados no Capítulo 4 tomam como base a posição das ressonâncias da envoltória espectral no espectro de módulo, chamadas de formantes.

Por estas razões, a Seção 3.2 descreverá o conceito de envoltória espectral, o qual será amplamente utilizado nesta dissertação, tanto na etapa de análise quanto na de síntese. Também será descrito o modelo LPC e como ele pode ser utilizado na estimação da envoltória espectral de um sinal.

Uma vez obtida a envoltória espectral, a Seção 3.3 descreverá o conceito de formantes de um sinal de voz; também serão descritas formas de se localizar os primeiros formantes no espectro e a relevância dos mesmos na determinação de cada vogal contida em um sinal de voz.

Por fim, com base no conceito de formantes, a Seção 3.4 apresentará um estudo para definir qual a ordem ideal para o modelo LPC para a estimação da envoltória

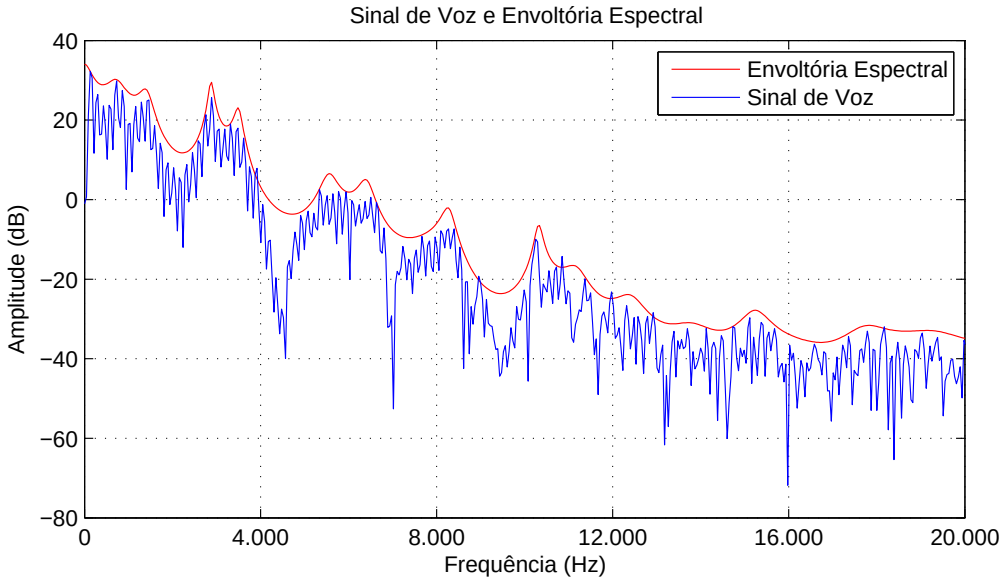


Figura 3.1: Espectro de módulo de um sinal qualquer (em azul) e sua envoltória espectral (em vermelho).

espectral.

3.2 Envoltória Espectral

Conforme será visto no Capítulo 6, o método de síntese aditiva utilizado neste trabalho consiste na inclusão dos harmônicos de um sinal diretamente em seu espectro de módulo. Para que isso seja feito, faz-se necessário um parâmetro que delimite a amplitude de cada pico (parcial harmônica) de forma que seja incorporado o timbre desejado. A modelagem da energia de cada parcial harmônica do espectro de módulo do sinal sintetizado é determinada pela envoltória espectral [14].

A envoltória espectral vem a ser uma curva suave que delimite a amplitude dos picos do espectro de módulo de um sinal qualquer, como pode ser visto na Figura 3.1, onde o espectro de módulo do sinal está representado pela cor azul e a sua envoltória pela cor vermelha.

Diversos métodos para estimar envoltórias espectrais podem ser encontrados na literatura, como a Envoltória Espectral por Cepstrum e Envoltória Espectral pelo Cepstrum Discreto [42].

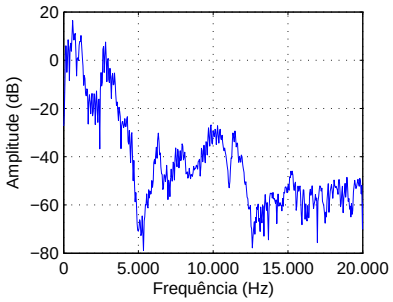
Neste trabalho de mestrado, optou-se pela extração da envoltória espectral através da utilização do modelo AR (Autorregressivo), que também podemos chamar de LPC (codificação por predição linear, do inglês, *Linear Predictive Coding*) pelo fato de estarmos lidando com sinais de voz [43] [42] [29], conforme descrito abaixo.

3.2.1 Modelo Autorregressivo

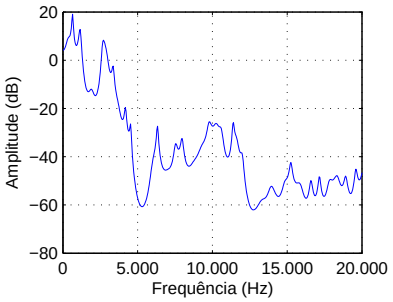
O modelo autorregressivo estima um determinado sinal com a saída de um filtro digital só polos excitado por ruído branco; com isso, a resposta na frequência do filtro será semelhante ao espectro do sinal desejado. O modelo autorregressivo utiliza um filtro só-polos porque um sinal de áudio contém muitos picos no espectro de módulo.

Para fazer com que a resposta deste filtro seja igual ao espectro do sinal desejado, é necessário que sua ordem seja de várias dezenas (considerando uma frequência de amostragem de 44,1 kHz).

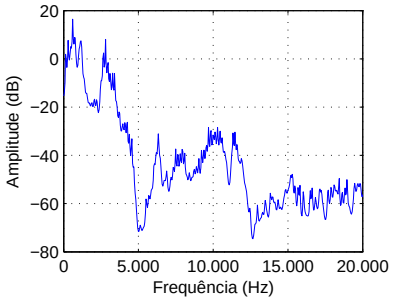
A Figura 3.2 (a) apresenta o espectro de um trecho de 1024 amostras do sinal de áudio de um cantor executando a vogal *a* em um *pitch* de 200 Hz discretizado com uma taxa de amostragem f_s igual a 44,1 kHz. As estimativas desse espectro ilustradas nas Figuras 3.2 (b), (c), (d) são as respostas da magnitude dos filtros criados pelo modelo autorregressivo com ordens p iguais a 100, 300 e 500, respectivamente.



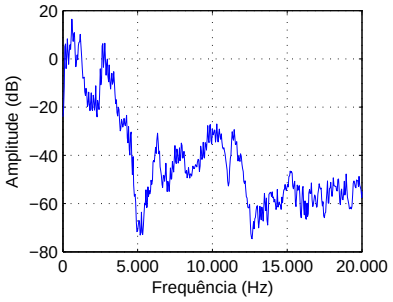
(a) sinal de voz



(b) resposta do filtro AR com $p = 100$



(c) resposta do filtro AR com $p = 300$



(d) resposta do filtro AR com $p = 500$

Figura 3.2: Sinal de voz cantada executando a vogal *a* em um *pitch* igual a 200 Hz a uma taxa de amostragem f_s igual a 44,1 kHz em uma janela de 1024 amostras, ilustrado na Figura (a). Resposta do filtro gerado pelo modelo AR com $p = 100$, 300 e 500, ilustrado nas Figuras (b), (c), (d).

Através destas figuras podemos ver que quanto maior a ordem do filtro gerado pelo modelo AR maior será a resolução do espectro estimado pela resposta deste filtro. Vemos que, para a taxa de amostragem igual a 44,1 kHz, a ordem $p = 500$

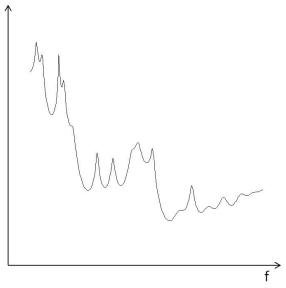


Figura 3.3: Resposta em frequência a um impulso unitário de um filtro que simule o trato vocal de um cantor qualquer. Pode, também, ser chamada de envoltória espectral do trato vocal.

apresenta uma boa estimativa do sinal original.

3.2.2 Codificação por Predição Linear

A codificação por predição linear (LPC) é amplamente utilizada na transmissão e compressão de sinais de voz. Ela parte da prerrogativa de que a resposta do filtro não precisa modelar completamente o espectro do sinal, mas deve representar apenas o sistema físico que modifica um sinal de excitação na produção do som. Desta forma, cria-se um modelo fonte-filtro.

Este filtro deverá conter as ressonâncias obtidas através da reverberação da onda sonora pelo corpo de um instrumento musical, como um violão ou uma flauta. Como estamos lidando com sinais de voz, o filtro deverá representar as resonâncias do trato vocal, que vem a ser o meio por onde o ar que sai dos pulmões se propaga. A Figura 3.3 ilustra a resposta em frequência a um impulso unitário de um filtro que simule o trato vocal de um cantor qualquer durante a emissão de uma vogal cantada.

Neste modelo, a caixa acústica do violão, o corpo da flauta ou o nosso trato vocal são responsáveis por modelar o timbre do sinal de saída por modificação de um sinal de entrada, que pode ser a reverberação da corda, para o primeiro caso, ou o sopro do ar, nos dois últimos.

Como entrada do sistema (fonte de excitação) será utilizado um trem de pulsos que trazem consigo informação de amplitude e *pitch* do sinal, conforme ilustrado na Figura 3.4. Para obter uma aproximação do sinal originalmente desejado, realiza-se a convolução no domínio do tempo desta fonte de excitação com o filtro que representa o modelo físico. Desta forma, o sinal resultante (ilustrado pela Figura 3.5) possuirá o timbre, o *pitch* e a amplitude desejados.

Abaixo, é narrada uma breve explicação da modelagem do trato vocal através do modelo LPC. Uma explicação mais aprofundada pode ser obtida em [29].

O modelo LPC se propõe a prever uma amostra do sinal $s[n]$ através de uma

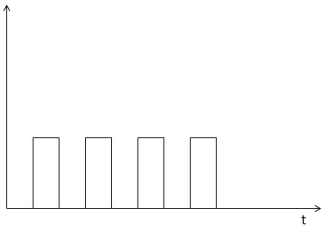


Figura 3.4: Trem de pulsos utilizado como fonte de excitação, representado no domínio do tempo.

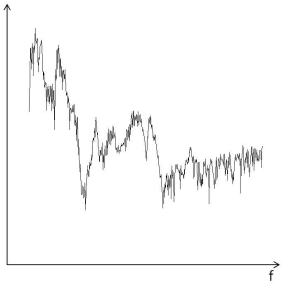


Figura 3.5: Sinal (representado no domínio da frequência) resultante da filtragem da fonte de excitação com o filtro que representa o modelo físico.

combinação linear das p amostras anteriores, sendo p a ordem do modelo, através de

$$\hat{s}[n] = \sum_{i=1}^{p} a_i s[n-i], \tag{3.1}$$

onde $\hat{s}[n]$ é a aproximação de $s[n]$ e a_i são os coeficientes de predição linear, que serão obtidos pela minimização da função-erro dada por $e[n] = s[n] - \hat{s}[n]$.

Uma vez obtidos os coeficientes a_i, podemos criar o filtro só-polos cuja resposta de magnitude será uma boa aproximação da função de transferência do trato vocal; este filtro tem transferência dada por

$$H(z) = \frac{1}{1 - \sum_{i=1}^{p} a_i z^{-i}}. \tag{3.2}$$

Uma característica do modelo LPC que pode ser destacada é o fato de ele não preservar o nível DC do sinal original na envoltória espectral obtida através dele, como pode ser visto na Figura 3.6.

Com isso, deve-se utilizar uma heurística que venha nivelar o nível DC da envoltória espectral estimada com o nível DC do sinal original.

O método utilizado neste trabalho primeiramente procura pelo maior pico do espectro de módulo do sinal original. Feito isso, toda a envoltória espectral é dividida pela razão entre o valor da envoltória espectral na posição deste pico e o valor do

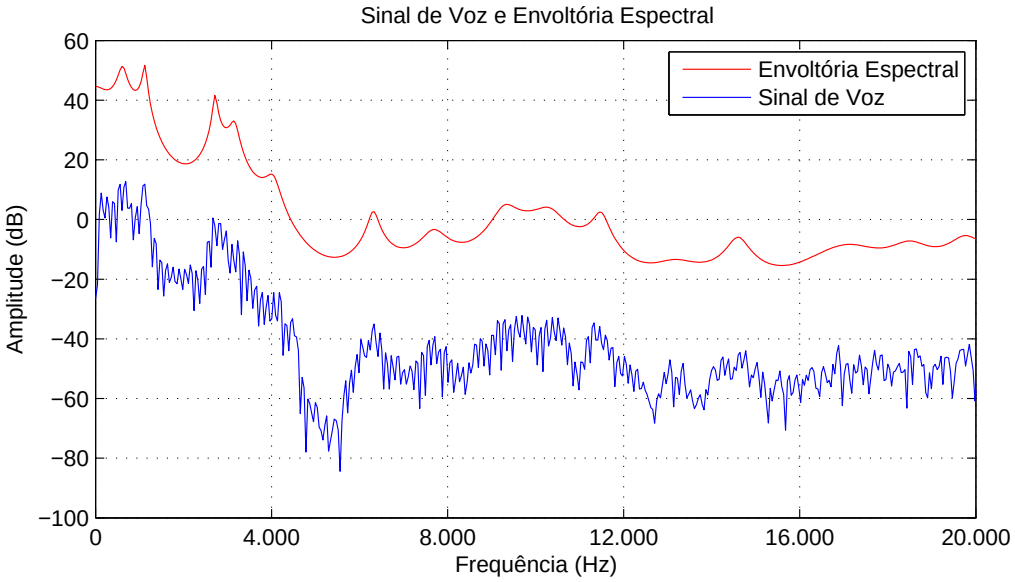


Figura 3.6: Espectro de módulo de um sinal qualquer (em azul) e sua envoltória espectral (em vermelho) sem ajuste do nível DC.

pico em si. Desta forma, a envoltória espectral se posicionará de forma a envolver os picos do sinal original, como podemos visualizar na Figura 3.1.

3.3 Formantes Vocais

Ao analisarmos a envoltória espectral de trechos correspondentes a vogais de diferentes sinais de voz, podemos encontrar um comportamento comum a todos eles: a presença de certos picos na formação da envoltória do espectro. Cada pico corresponde a uma frequência de ressonância relacionada a diferentes componentes do trato vocal. Essas frequências de ressonância são chamadas de formantes. A Figura 3.7 ilustra um sinal de voz cantada e a sua envoltória espectral, onde podemos identificar seus formantes em cada pico da envoltória.

Conforme descrito acima, é possível modelar o trato vocal através de filtros digitais, de forma que os formantes seriam as frequências de ressonância destes filtros. Como também foi visto, o *pitch* é determinado pela excitação à entrada do filtro na forma de pulsos periódicos. Com isso, pode-se dizer que os formantes independem do *pitch* produzido, assim como filtros independem da frequência do sinal de entrada. Na verdade, ao modularmos o *pitch*, o trato vocal sofre pequenas alterações, o que acarreta pequenas modulações também nos formantes. É importante preservar essa quase-invariância dos formantes ao se realizar modificações artificiais em sinais de voz, para que a naturalidade da voz seja preservada.

Um exemplo em que os formantes de um sinal de voz são alterados impropriamente se dá quando se tenta fazer modificações de *pitch* através da mudança da

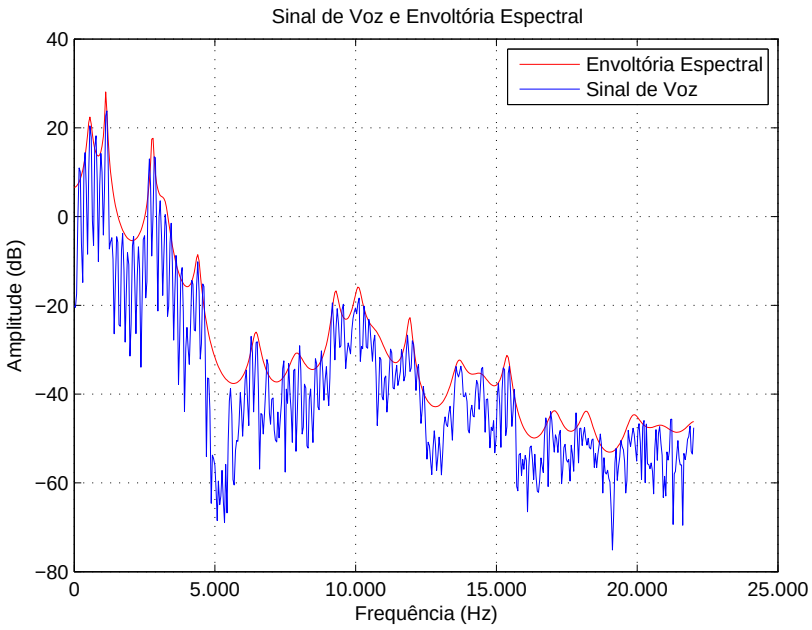


Figura 3.7: Sinal de voz (em azul) e sua envoltória espectral (em vermelho).

frequência de amostragem, i.e., alteração da velocidade de reprodução do sinal. Esta operação seria a forma mais simplista de alteração de *pitch*, mas não preserva o timbre do sinal original, pois realiza a compressão ou expansão de todo o espectro, e assim da envoltória espectral. Isto significa dizer que os formantes não se encontrarão em suas posições originais, resultando em um sinal robótico e claramente sintético. Esse exemplo nos permite compreender a importância dos formantes na construção dos timbres.

Os formantes são muito úteis também na detecção de vogal, pois os três primeiros formantes contribuem decisivamente com a formação das vogais; os demais só apresentam relevância na definição do timbre [44] [45] [46].

A Tabela 3.1 possui os valores médios para os três primeiros formantes das vogais *a*, *e*, *i*, *o* e *u* do português brasileiro, valores estes retirados de Rauber [1]. A primeira coluna desta tabela possui o símbolo IPA (*International Phonetic Alphabet*) do fonema, a segunda possui uma palavra na língua portuguesa que exemplifique o uso deste fonema; as demais colunas mostram o valor médio dos três primeiros formantes. Como Rauber apresentou duas tabelas, uma para locutores masculinos e outra para femininos, a Tabela 3.1 apresenta a média das duas tabelas apresentadas por ela.

Para efeito de comparação, a Tabela 3.2 possui os valores dos três primeiros formantes de fonemas correspondentes na língua inglesa. Por este motivo, esta tabela possuirá uma coluna com uma palavra na língua inglesa que exemplifique o uso deste fonema e outra coluna com a vogal aproximada correspondente a este fonema na língua portuguesa. Estes valores foram retirados de Peterson e Barney

Tabela 3.1: Valores médios da frequência dos formantes de cinco vogais, com exemplos na língua portuguesa. Retirado de Rauber [1].

Símbolo	Exemplo na Língua Portuguesa	F_1	F_2	F_3
a	ata	770	1.536	2.463
ɛ	pé	550	2.095	2.763
i	giz	295	2.461	3.075
ɔ	pó	586	1.031	2.513
u	tu	317	857	2.700

[2].

Tabela 3.2: Valores médios da frequência dos formantes de cinco vogais, com exemplos na língua inglesa. Retirado de Peterson e Barney[2].

Símbolo	Exemplo na Língua Inglesa	Vogal Correspondente em Português	F_1	F_2	F_3
ʌ	*but*	*a*	520	1.190	2.390
ɛ	*bet*	*e*	530	1.840	2.480
i	*beet*	*i*	270	2.290	3.010
ɔ	*bought*	*o*	570	840	2.410
u	*boot*	*u*	300	870	2.240

Devido a esta relação existente entre os formantes e as vogais, os métodos de detecção de vogal estudados no Capítulo 4 desta dissertação utilizam a informação dos primeiros formantes de um sinal de voz. Por isso, o primeiro passo para a detecção de vogal deverá ser a identificação destes formantes.

Para localizar no espectro a posição dos três primeiros formantes, foram estudados dois métodos: a simples localização dos picos na envoltória espectral e a identificação das frequências dos polos do filtro do modelo LPC, utilizado na extração da envoltória.

3.3.1 Localização dos Picos na Envoltória Espectral

Um método intuitivo para detecção dos formantes é o simples ato de localizar os picos da envoltória espectral. Para isso, basta que seja feita a identificação dos máximos locais da função que representa esta envoltória.

Uma desvantagem deste método é o fato de nem sempre um formante chegar a formar um pico na envoltória. Isso pode acontecer quando um formante de baixa

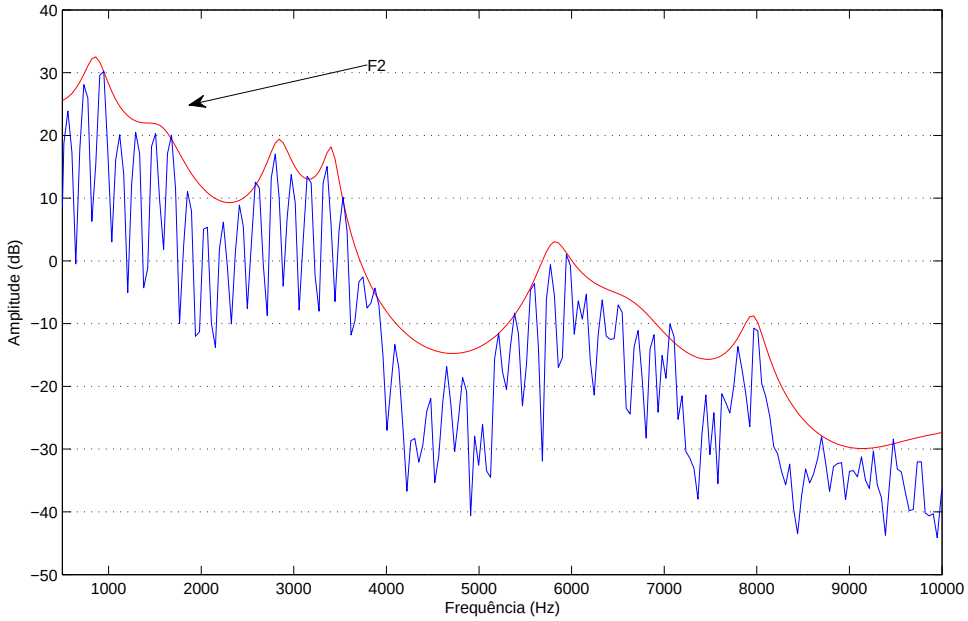


Figura 3.8: Exemplo em que um formante não chega a formar um máximo local, indicado pela seta.

amplitude se encontra próximo de outro de maior amplitude. A Figura 3.8 ilustra um caso em que o segundo formante (indicado na figura pela seta) não forma um máximo local, impedindo sua identificação como formante.

Este erro fará com que um quarto ou quinto formante seja identificado entre os três primeiros, o que acarretará em um parâmetro falso para a identificação da vogal emitida.

3.3.2 Polos do Modelo LPC

Conforme visto na Seção 3.2, a envoltória espectral será estimada pela resposta em frequência do filtro gerado pelos coeficientes do modelo LPC [42].

O número de polos produzidos é igual ao número de coeficientes LPC, que varia de acordo com a ordem p do modelo, sendo que cada polo possui um equivalente conjugado que pode ser descartado. Cada um dos $p/2$ polos relevantes pode ser associado a um formante da envoltória espectral. Normalmente, a ordem do modelo será pelo menos duas vezes maior do que o número de formantes de um sinal de voz. Deste modo, cada formante estará relacionado a um polo deste filtro, porém nem todo polo representará um formante [47].

Os polos com amplitudes mais próximas do círculo unitário são os que mais provavelmente representam formantes, pois apresentam uma ressonância maior, e tendem a produzir, assim, picos na envoltória espectral [47].

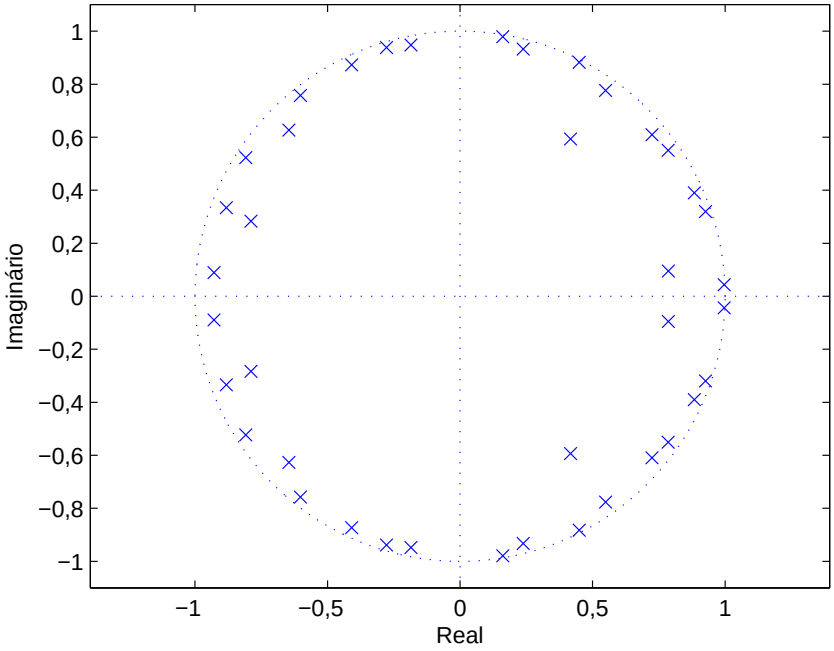


Figura 3.9: Localização dos polos no círculo unitário do exemplo apresentado na Tabela 3.3.

A frequência relacionada a cada polo pode ser obtida através de

$$f = \theta \cdot \frac{f_s}{2\pi}, \tag{3.3}$$

onde f_s é a frequência de amostragem e θ o ângulo do polo em questão.

A Figura 3.9 apresenta o círculo unitário contendo os polos de um modelo LPC de ordem p igual a 40 para um sinal de voz cantada contendo a vogal i a um *pitch* igual a 293 Hz em uma taxa de amostragem de 44,1 kHz, em que os quatro primeiros formantes estão indicados na Figura 3.10. Através da Tabela 3.3 podemos visualizar os cinco primeiros polos deste exemplo, com suas frequências e amplitudes. Nela, podemos ver que o segundo polo, cuja amplitude é menor que a dos demais, não representa um formante.

Porém, o uso deste critério nem sempre garante que os polos com amplitudes maiores serão formantes do sinal de voz. Isso porque muitas vezes todos os polos se encontram muito próximos do círculo unitário, sem apresentarem uma diferença significativa entre eles, que justificaria a distinção entre polos que representam formantes ou não.

Por esta razão, este método não será utilizado na detecção dos primeiros formantes, mas somente na obtenção dos primeiros polos da envoltória espectral de um sinal de áudio. Dependendo da aplicação, diferentes estratégias serão utilizadas para determinar quais os três primeiros formantes dentre estes primeiros polos.

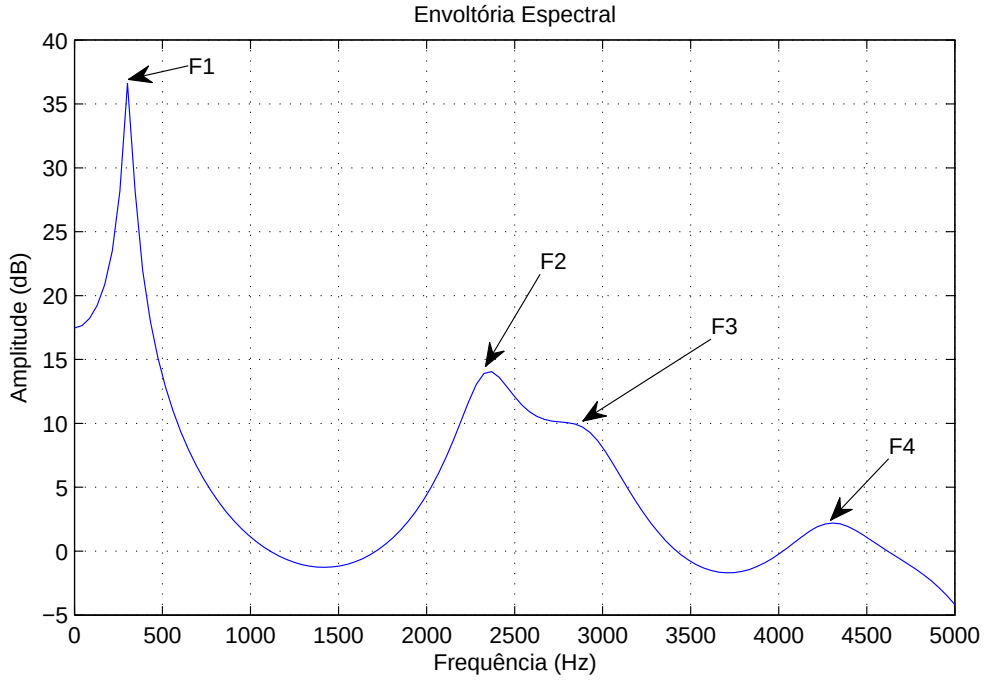


Figura 3.10: Localização dos quatro primeiros formantes na envoltória espectral do exemplo apresentado na Tabela 3.3.

Tabela 3.3: Frequência e amplitude dos cinco primeiros polos de sinal de voz cantada contendo a vogal i em um *pitch* igual a 293 Hz em uma taxa de amostragem de 44,1 kHz. A quarta coluna indica qual formante é representado por este polo, caso aplicável.

Nº do Polo	Frequência dos Polos	Amplitude dos Polos	Formante
1	304,63	0,99754	F_1
2	847,83	0,79132	
3	2.334	0,9798	F_2
4	2.918,6	0,9664	F_3
5	4.293,2	0,9590	F_4

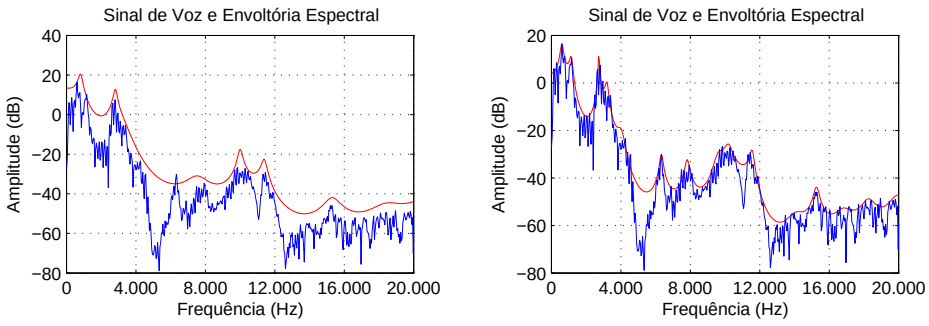
Essas estratégias serão descritas no Capítulo 4.

3.4 Ordem do Modelo LPC

Conforme descrito na Seção 3.2.1, a ordem p do modelo AR determina o nível de detalhes do sinal original contidos na resposta em frequência do filtro por ele gerado; quanto maior o valor de p, maior será este nível de detalhes. Foi demonstrado que uma ordem $p = 500$ representa uma boa estimativa para o sinal original. Porém, como o intuito deste capítulo é a obtenção da envoltória espectral de um sinal, a ordem utilizada para o modelo LPC deverá ser bem menor que 500.

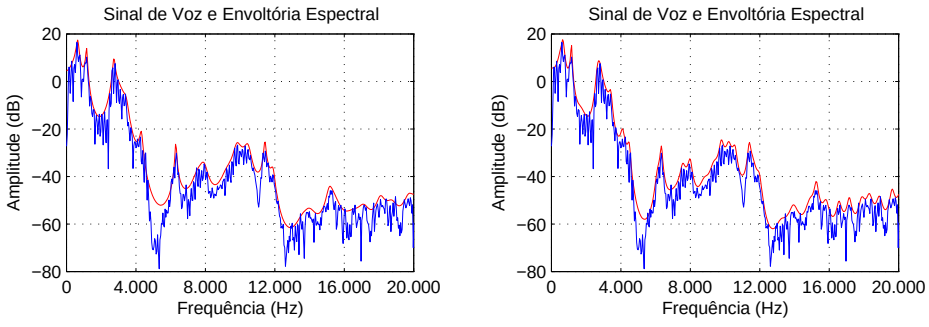
Devemos ter em mente que a ordem do modelo LPC não deverá ser muito baixa a ponto de perdermos informações de alguns formantes e nem muito alta a ponto de não mais modelarmos a envoltória espectral, mas sim o sinal em si.

A Figura 3.11 apresenta a envoltória espectral do mesmo sinal estudado na Figura 3.2, que representa um sinal de áudio de um cantor executando a vogal *a* em um *pitch* de 200 Hz a uma taxa de amostragem f_s igual a 44,1 kHz em uma janela de 1024 amostras. São apresentas envoltórias espectrais para p igual a 20, 40, 60 e 80 nas Figuras 3.11 (a), (b), (c) e (d), respectivamente.



(a) $p = 20$

(b) $p = 40$



(c) $p = 60$

(d) $p = 80$

Figura 3.11: Sinal de voz (em azul) e envoltória espectral estimada pelo modelo LPC (em vermelho) com ordem p igual a 20, 40, 60 e 80.

Através destas figuras, podemos ver que o modelo LPC com ordem p igual a 20

não representa uma boa estimativa para a envoltória espectral do sinal, pois muitos formantes não são representados por ela. Podemos verificar isso ao notarmos que o terceiro pico desta envoltória, supostamente o terceiro formante, se encontra em aproximadamente 7.500 kHz e, de acordo com a Tabela 3.1, o terceiro formante possui uma média máxima de 3.010 Hz.

O modelo LPC com ordem p igual a 80 começa modelar nuances do sinal original, e não mais simplesmente a sua envoltória. Podemos verificar esta afirmação ao notarmos muitos picos inerentes ao próprio sinal na faixa de frequência de 16 a 20 kHz.

Já os modelos LPC com ordens p iguais a 40 e 60 mostraram-se muito similares e eficazes; a partir de $p = 40$ já é possível identificar os três primeiros formantes corretamente, pois vemos que o primeiro pico se encontra em aproximadamente 600 Hz, o segundo pico em 1.120 Hz e o terceiro em 2.700 Hz, que vêm a ser valores plausíveis para os formantes da vogal *a*, de acordo com a Tabela 3.1.

Desta forma, optou-se por utilizar nesta dissertação o modelo LPC com ordem p igual a 40, pois é um valor moderado e que ainda dá boa representatividade aos formantes de um sinal de áudio a uma taxa de amostragem f_s igual a 44,1 kHz.

3.5 Conclusão

Neste capítulo, foi descrito o conceito de envoltória espectral, o qual é essencial na realização da síntese aditiva através da inversa da DFT (pois carrega a informação do timbre a ser sintetizado) e também de grande importância na detecção da vogal emitida pelo sinal de voz através do conceito de formantes vocais.

Foram relatados o modelo autorregressivo e o modelo LPC, destacando-se as diferenças conceituais entre eles. Nesta linha, mostrou-se como adquirir uma estimativa para a envoltória espectral através do modelo LPC, e demonstrou-se que este modelo com ordem p igual a 40 pode ser utilizado com uma taxa de amostragem de 44,1 kHz.

Outro importante aspecto tratado neste capítulo foi o dos formantes de um sinal de voz, descrevendo a sua definição e relevância na determinação da vogal emitida.

Foi exposta a ideia de que os três primeiros formantes contêm a informação da vogal emitida, assim como apresentada uma tabela contendo as vogais e a posição frequencial média de seus primeiros formantes, tanto para vogais do português brasileiro quanto para vogais da língua inglesa.

Para auxiliar na detecção de vogal apresentada no próximo capítulo, foram descritas duas formas de se determinar os primeiros formantes de um espectro: a localização dos picos na envoltória espectral e dos polos do modelo LPC; ambos apresentaram falhas em alguns casos específicos.

A partir dos polos do modelo LPC, o Capítulo 4 apresentará diferentes estratégias para determinar quais os três primeiros formantes de um sinal de voz no qual deseja-se detectar a vogal emitida.

Capítulo 4

Detecção de Vogais

4.1 Introdução

Sinais de voz cantada, diferentemente de sinais de instrumentos musicais, além da melodia, trazem consigo uma mensagem nos diversos vocábulos contidos em um idioma, o que possibilita a compreensão da letra de uma determinada música.

Neste ponto da dissertação, considera-se que a melodia do sinal de referência já foi detectada pelo método de detecção de *pitch* escolhido no Capítulo 2. Porém, esta melodia pode ser executada com a utilização de inúmeros fonemas diferentes. Desta forma, ainda na fase de análise do sinal de referência, métodos de detecção de fonema devem ser executados para que o sinal sintetizado possua, além da mesma melodia, os mesmos fonemas que o sinal de referência.

Conforme mencionado anteriormente, a síntese de consoantes não será objeto de estudo nesta dissertação, que trata somente da síntese de vogais, onde está contida essencialmente a melodia a ser sintetizada.

Primeiramente, neste capítulo será descrita a construção de um banco de dados contendo, estatisticamente, a posição dos três primeiros formantes das vogais *a*, *e*, *i*, *o* e *u*[1]. Este banco de dados será utilizado nos métodos de detecção de vogal por probabilidade e por proximidade, os quais serão propostos nas Seções 4.3 e 4.4, respectivamente. Na Seção 4.5, é apresentada uma alternativa para se preservar a vogal emitida pelo sinal de referência através da substituição dos primeiros polos da envoltória espectral na criação do sinal sintético.

Após a descrição de alguns ajustes, será apresentada, por fim, uma avaliação de cada método com o intuito de eleger o algoritmo de detecção de vogal mais adequado a este trabalho.

[1]Restringimo-nos a esses cinco fonemas orais por simplicidade. O desenvolvimento a seguir serve como uma prova de conceito, e pode ser estendido para situações mais gerais.

4.2 Banco de Dados com Parâmetros de Vogais

Conforme mencionado acima, os métodos de detecção de vogal por probabilidade e por proximidade dependem de uma base de dados que contenha informações previamente coletadas a respeito das vogais. Desta forma, faz-se necessária a criação deste banco de dados, que chamaremos de Banco de Dados de Probabilidades de Vogais, e que deverá conter alguns parâmetros que, posteriormente, serão utilizados na detecção da vogal, conforme será visto nas próximas seções deste capítulo.

Este banco será desenvolvido a partir de outro banco de dados que contenha amostras de áudio para todas as vogais, em diferentes *pitches* e de timbres diferentes, i.e., sinais de diferentes cantores. Este banco de dados será chamado de Banco de Dados de Amostras de Vogais.

Este segundo banco de dados foi criado a partir de trechos sonoros de aproximadamente um segundo de duração para cada uma das vogais *a*, *e*, *i*, *o* e *u* da língua portuguesa. Foram obtidos sinais de dois cantores masculinos, um lírico e outro popular, com uma extensão de *pitch* que varia de 110 a 415 Hz, e uma cantora popular com uma extensão de *pitch* que varia de 175 a 700 Hz. Cada trecho sonoro foi agrupado e concatenado em um único sinal de áudio por vogal, resultando, assim, em cinco sinais diferentes com diversas amostras para cada vogal.

A variedade de *pitches* contornará a modulação dos formantes originada pela mudança de altura, tornando estes dados mais confiáveis para diversos valores de *pitch*.

Os sinais do Banco de Dados de Amostras de Vogais serão janelados em *frames* de 1024 amostras. Em cada *frame*, serão determinados os formantes pelo método descrito na Seção 3.3.2, a partir da identificação dos quatro primeiros polos do modelo LPC. Estes polos serão comparados com a Tabela 3.1, para se aproximar às vogais do português brasileiro, ou comparados com a Tabela 3.2, para se aproximar às vogais da língua inglesa; assim, por proximidade, serão extraídos os três primeiros formantes dentre estes quatro polos. Esta forma de determinação dos três primeiros formantes só é possível porque, nesta etapa, a vogal é conhecida.

Este método pode ser ilustrado pelo exemplo da Figura 4.1, que possui a frequência de cada um dos quatro primeiros polos de uma envoltória espectral de um sinal de voz cantada executando a vogal *a*, localizados em 545,85 Hz; 1371,9 Hz; 2985,5 Hz; 3645,6 Hz. Nela, podemos ver que os três primeiros polos foram eleitos como os três primeiros formantes, através da aproximação de cada polo a cada um dos três primeiros formantes da vogal *a*, segundo a Tabela 3.1.

Um possível erro deste método se daria quando um polo viesse a ser o mais próximo para dois formante diferentes, conforme ilustrado na Figura 4.2, que também vem a ser referente a um sinal contendo a vogal *a*, porém com as frequências

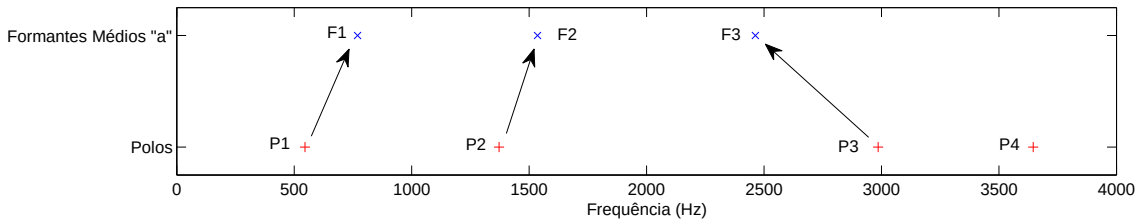


Figura 4.1: Aproximação de cada um dos quatro primeiros polos de um sinal de voz cantada contendo a vogal *a* aos três primeiros formantes da vogal *a*, segundo a Tabela 3.1.

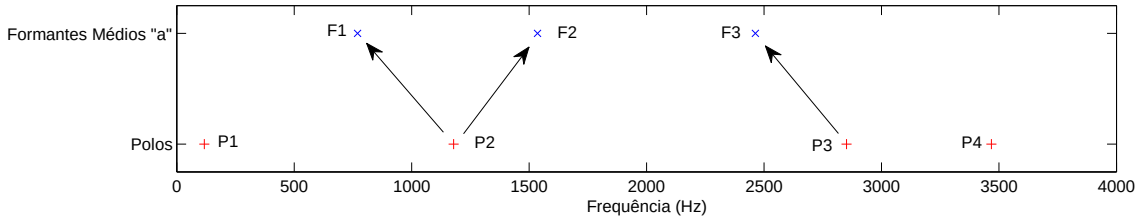


Figura 4.2: Erro no método de aproximação de cada um dos quatro primeiros polos de um sinal de voz cantada contendo a vogal *a* aos três primeiros formantes da vogal *a*, segundo a Tabela 3.1.

dos polos iguais a 117,27 Hz; 1178,5 Hz; 2851,2 Hz; 3467,7 Hz. Neste caso, esta amostra (janela) será simplesmente descartada na geração deste banco de dados.

De posse dos formantes, é gerado um histograma para cada um dos três primeiros formantes de cada uma das vogais *a*, *e*, *i*, *o* e *u*. Esses histogramas possuirão as probabilidades das posições no espectro de frequência de cada formante para cada vogal.

Uma vez gerados os histogramas, aproxima-se uma curva normal (gaussiana) de modo a suavizar os valores de cada histograma. Esta gaussiana é dada através de

$$f(x,\mu,\sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)}, \tag{4.1}$$

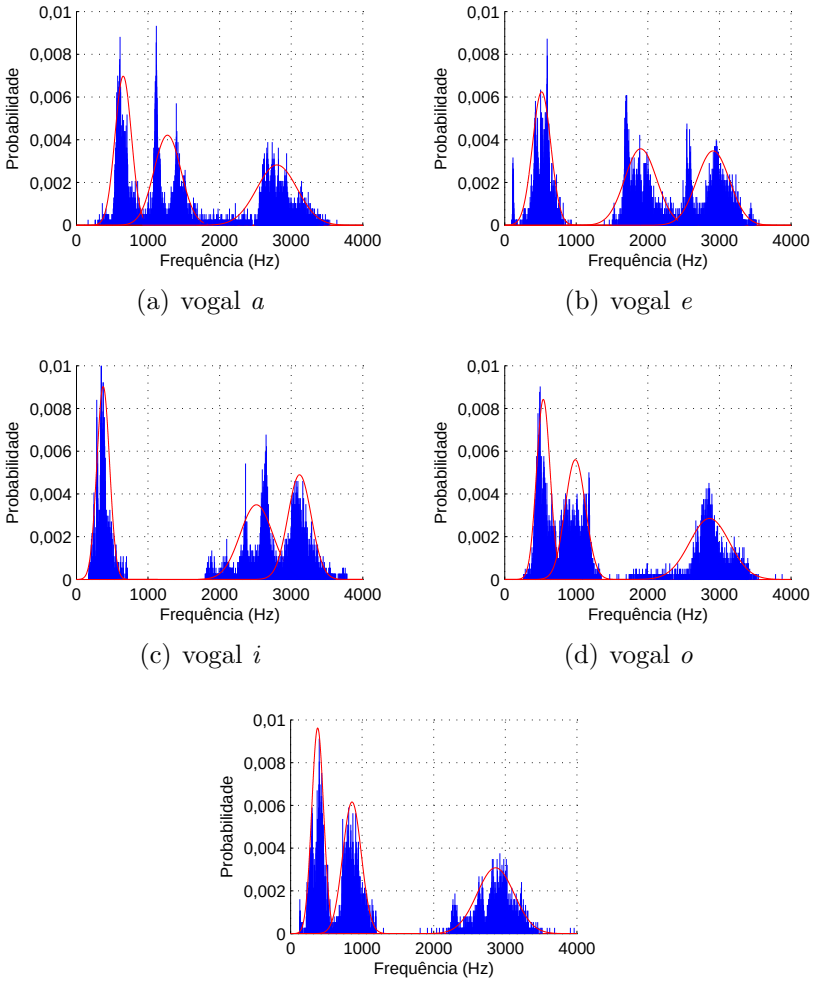
com área total igual a 1 e média e desvio padrão dados por

$$\mu = \frac{1}{n}\sum_{i=1}^{n} x_i, \tag{4.2}$$

$$\sigma = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (x_i - \mu)^2}. \tag{4.3}$$

A Figura 4.3 apresenta os resultados obtidos na criação do Banco de Dados de Probabilidades de Vogais com base no português brasileiro (Tabela 3.1), ilustrando os histogramas e as curvas normais a eles ajustadas para cada formante de cada

vogal.



(a) vogal *a*

(b) vogal *e*

(c) vogal *i*

(d) vogal *o*

(e) vogal *u*

Figura 4.3: Histograma e distribuição normal dos três primeiros formantes das vogais *a*, *e*, *i*, *o* e *u*.

De forma a simplificar o Banco de Dados de Probabilidades de Vogais, serão armazenados nele somente os valores de média e desvio padrão de cada gaussiana. A partir destes resultados, pode-se definir a Tabela 4.1, que vem a ser uma atualização da Tabela 3.1, contendo os valores médios de cada formante obtidos através das gaussianas geradas.

Pode-se dizer que estes resultados validam os valores obtidos por Rauber [1], pois a maioria dos formantes apresentou valores bem próximos aos valores apresentados por ela.

Tabela 4.1: Valores médios da frequência dos formantes de cada vogal da língua portuguesa, retirados das gaussianas produzidas para o Banco de Dados de Probabilidades de Vogais.

Vogal	F_1	F_2	F_3
a	653,8	1.272,3	2.801,5
e	516,9	1.900,1	2.912,3
i	374,4	2.515,5	3.117,5
o	540,2	987,6	2.863,0
u	378,8	859,1	2.864,6

4.3 Detecção de Vogal por Probabilidade

O primeiro método de detecção da vogal reproduzida em um sinal de voz proposto nesta dissertação opera através da verificação da probabilidade de cada um dos N_F primeiros formantes corresponder a um dos formantes de uma determinada vogal. Para que isso seja possível, utiliza-se o Banco de Dados de Probabilidades de Vogais produzido na seção anterior, que possui uma distribuição normal para cada formante de cada vogal.

Para a definição dos primeiros formantes de um *frame* de um sinal, o método seleciona os N_P primeiros polos da envoltória espectral e determina a probabilidade de cada polo estar relacionado a cada um dos N_F primeiros formantes, sendo $N_P > N_F$. O algoritmo fará esta verificação utilizando as gaussianas de cada uma das cinco vogais. As frequências dos polos com maior probabilidade serão definidas como os N_F primeiros formantes.

Nos casos em que a frequência de um mesmo polo for definida como dois formantes diferentes, esta vogal será descartada na detecção deste *frame*. Por exemplo, considere um sinal contendo a vogal *o*. No momento em que for testada a probabilidade deste sinal conter a vogal *i*, é bem provável que um terceiro polo seja o mais provável tanto para o segundo quanto para o terceiro formantes (verificar estas informações na Figura 4.3). Este erro descarta a possibilidade de que a vogal presente no sinal seja a vogal *i*.

Após determinados os formantes mais prováveis para cada vogal, somam-se as probabilidades destes formantes e, assim, elege-se a vogal cuja soma das probabilidades dos formantes é máxima.

Uma desvantagem deste método é o fato de que este algoritmo sempre retornará as vogais de forma discreta, i.e., somente poderá resultar em *a*, *e*, *i*, *o* ou *u*. Quaisquer valores intermediários entre estas vogais serão aproximados para a vogal mais próxima.

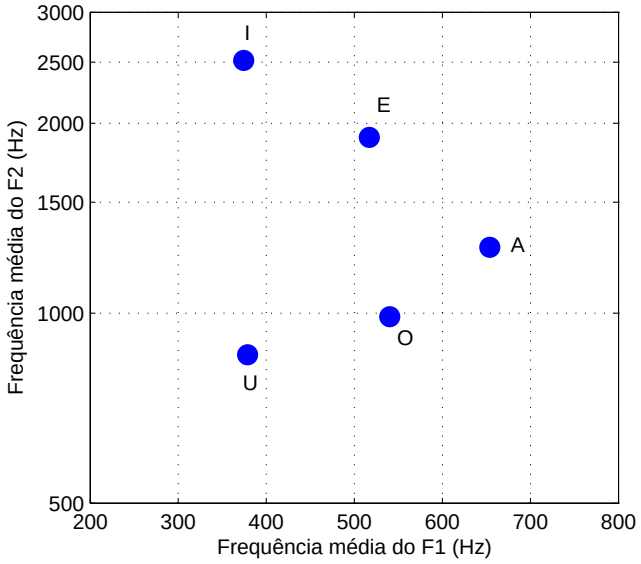


Figura 4.4: Gráfico contendo a posição média, segundo Tabela 4.1, dos dois primeiros formantes no plano $F_1 \times F_2$.

4.4 Detecção de Vogal por Proximidade

Os primeiros formantes de um sinal de voz podem ser encarados como pontos em uma reta, plano ou espaço, se utilizarmos somente o primeiro, os dois primeiros ou os três primeiros formantes, respectivamente, com cada formante associado a uma dimensão do espaço. Desta forma, é proposto um método em que a vogal é estimada através do cálculo da menor distância entre os formantes de um sinal de referência e os formantes do sinal alvo.

A Figura 4.4 nos mostra a posição dos dois primeiros formantes da Tabela 4.1 no plano cartesiano: o eixo das abscissas possui os valores do primeiro formante e o eixo das ordenadas os valores do segundo formante.

Para este método, os formantes são estimados conforme descrito na Seção 4.3, adquiridos os N_P primeiros polos da envoltória espectral; através do Banco de Dados de Probabilidades de Vogais, estimam-se por probabilidade os N_F primeiros formantes.

De posse dos primeiros formantes do sinal, calcula-se a menor distância entre o ponto representado pelos N_F primeiros formantes de um *frame* do sinal de referência (sinal de entrada) e o ponto representado pelos N_F primeiros formantes de cada vogal do sinal alvo. É importante que sejam utilizados os N_F primeiros formantes de um sinal alvo que se encontre aproximadamente no mesmo *pitch* do sinal de referência, para que os formantes sejam os mais próximos possíveis. As envoltórias espectrais do sinal alvo, bem como o valor dos três primeiros formantes de cada envoltória, estarão armazenadas no banco de dados de síntese, que será descrito no Capítulo 6.

Ao considerarmos dois formantes, a distância entre eles deve ser calculada como a distância entre os pontos $m(F_{1i},F_{2i})$ e $n(F_{1ii},F_{2ii})$ em um plano, que é dada por

$$d = \sqrt{(F_{1i} - F_{1ii})^2 + (F_{2i} - F_{2ii})^2}. \tag{4.4}$$

Caso consideremos a utilização de três formantes, deve ser calculada a distância entre os pontos $m(F_{1i},F_{2i},F_{3i})$ e $n(F_{1ii},F_{2ii},F_{3ii})$ através de

$$d = \sqrt{(F_{1i} - F_{1ii})^2 + (F_{2i} - F_{2ii})^2 + (F_{3i} - F_{3ii})^2}. \tag{4.5}$$

Este cálculo deverá ser realizado para os formantes de cada uma das cinco possíveis vogais existentes no banco de dados de síntese. A vogal eleita deverá ser a que apresentar o menor valor de d.

Este método apresenta a mesma desvantagem do método de detecção de vogal por probabilidade descrito na Seção 4.3: o algoritmo retornará as vogais de forma discreta. Desta forma, quaisquer valores intermediários entre estas vogais serão aproximados para a vogal mais próxima.

4.5 Substituição dos Primeiros Polos

Conforme visto anteriormente, os três primeiros formantes contêm as informações da vogal que foi emitida por um cantor, ou locutor. Vimos, também, que a frequência dos polos do filtro gerado pelos coeficientes do modelo LPC de um determinado sinal, dependendo de sua ordem, são iguais às frequências dos formantes da envoltória espectral deste sinal.

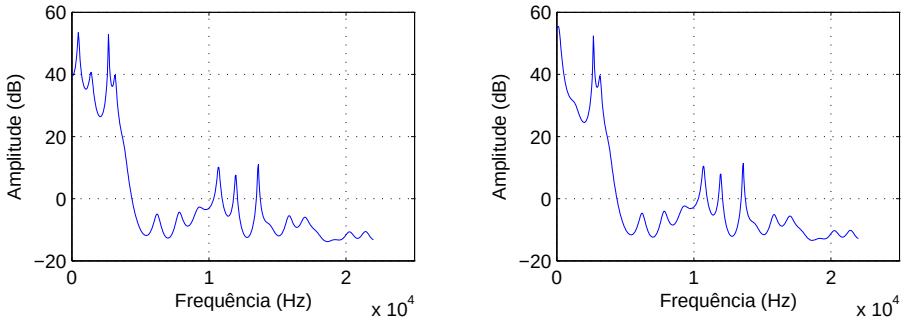
Nesta dissertação, foram utilizados sinais com taxa de amostragem igual a 44,1 kHz e ordem p do modelo LPC igual a 40. Para este valor de ordem, segundo descrito na Seção 3.3.2, temos 20 polos que podem estar relacionados a formantes de um sinal.

De forma empírica foi possível perceber que, para este valor de p, três dos quatro primeiros polos, i.e., dos quatro polos que possuem as frequências mais baixas, serão os três primeiros formantes do sinal.

Com isso, é proposto o método de substituição dos primeiros polos, onde, em vez de detectar a vogal contida no sinal de referência, substitui-se os N_P primeiros polos da envoltória espectral dos *frames* do sinal alvo, contida no banco de dados de síntese, pelos N_P primeiros polos da envoltória espectral dos *frames* do sinal de referência, onde $1 \leq N_P \leq 4$ e, assim, realiza-se a síntese com a envoltória espectral resultante desta substituição. Para que isso seja feito, basta identificar quais os coeficientes do modelo LPC relacionados aos N_P primeiros polos e substituí-

los pelos coeficientes dos N_P polos desejados e, assim, refazer o filtro cuja resposta em frequência será a nova envoltória espectral.

A Figura 4.5 apresenta, primeiramente, a envoltória espectral original de um determinado sinal, seguida da envoltória espectral deste mesmo sinal com os dois primeiros polos substituídos pelos dois primeiros polos de um sinal de referência. Nesta figura, pode-se perceber que os dois primeiros formantes da envoltória original foram substituídos pelos formantes do sinal de referência.



(a) Sem substituição de polo (b) Com substituição de polo

Figura 4.5: Envoltória espectral de um sinal sem a substituição de seus polos e envoltória espectral deste sinal com os dois primeiros polos substituídos, respectivamente.

Com o fim de analisar os resultados deste método, realizou-se uma análise subjetiva com diferentes sinais de voz cantada, para valores de N_P iguais a 1, 2 e 3.

Ao realizarmos esta substituição com $N_P = 1$, i.e., substituindo somente o primeiro polo, não foi possível identificar a vogal emitida pelo sinal de referência no sinal sintetizado. Com $N_P = 2$, a vogal foi reconhecida para a maioria dos sinais, porém o sinal sintetizado apresentava algumas características do timbre do sinal de referência. Já com $N_P = 3$, a vogal era sempre muito bem representada, porém mais características do timbre do sinal de referência se mostravam contidas no sinal sintetizado.

Este método define muito bem a vogal contida no sinal de referência, e com a vantagem de preservar a vogal de forma contínua, i.e., quaisquer vogais com valores intermediários às cinco vogais estudadas serão fielmente sintetizadas.

Porém, uma grande desvantagem é o fato de introduzir, no sinal sintetizado, características do timbre do sinal de referência. Isto nos mostra que os três primeiros formantes não carregam somente a informação da vogal contida em um sinal, mas também características relevantes acerca do timbre desse sinal.

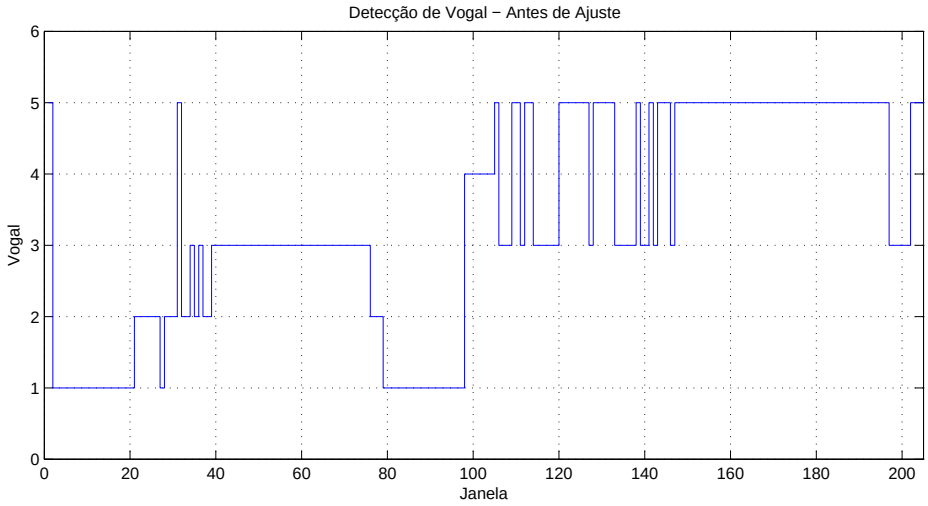


Figura 4.6: Evolução na detecção da vogal ao longo dos *frames* de um sinal - antes do ajuste final.

4.6 Ajustes Finais

De forma semelhante ao que foi descrito na Seção 2.5, os resultados dos métodos de detecção de vogal por probabilidade e por proximidade frequentemente apresentam erros de curta duração, onde ocorre a detecção de uma determinada vogal por um período curto de poucas janelas. Conforme visto anteriormente, foram utilizadas janelas de 1024 amostras em uma taxa de amostragem de 44,1 kHz, o que representa 23,2 ms do sinal.

A Figura 4.6 ilustra um exemplo onde podem ser vistos picos, ou vales, espúrios na detecção da vogal; convencionamos que 1 representa a vogal *a*, 2 a vogal *e*, 3 a vogal *i*, 4 a vogal *o* e 5 a vogal *u*. Estes picos, ou vales, de curta duração não devem ser considerados, pois seria impossível para um ser humano variar tão rapidamente a vogal emitida.

O maior pico (ou vale) admissível neste trabalho, para a detecção de vogal, é de 3 *frames*, que representam 69,7 ms. Caso seja detectado um pico (ou vale) dentro deste intervalo, o algoritmo irá encará-lo como errôneo, e fará com que a vogal da janela em questão seja considerada igual à vogal da janela anterior.

Podemos ver na Figura 4.7 o exemplo da Figura 4.6 após a realização do ajuste final, onde os picos, ou vales, de curta duração não são mais encontrados.

4.7 Definindo Vogais Intermediárias

Conforme mencionado anteriormente, os métodos de detecção de vogal por probabilidade e por proximidade retornam valores discretos para representar as vogais, pois será detectado somente se se trata da vogal *a*, *e*, *i*, *o* ou *u*. Desta forma, se o

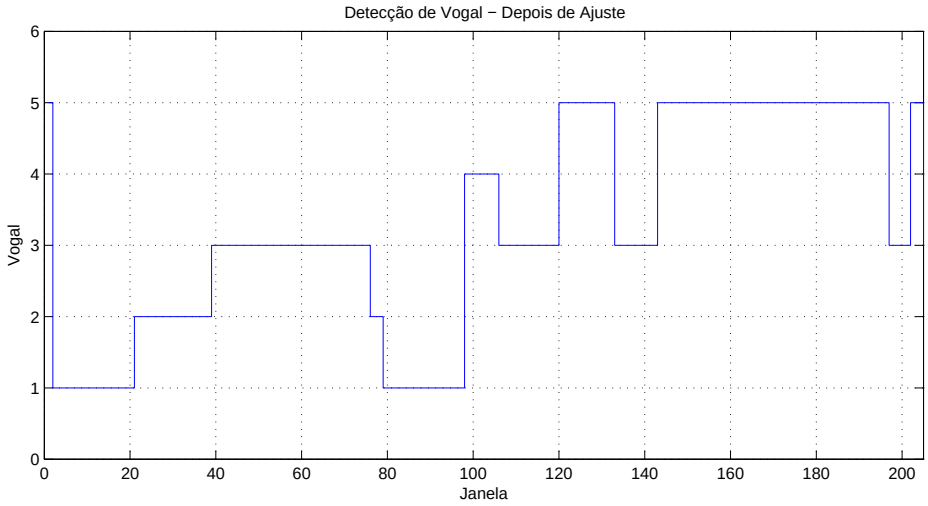


Figura 4.7: Evolução na detecção da vogal ao longo dos *frames* de um sinal - depois do ajuste final.

sinal de referência possuir uma vogal com um valor intermediário, esta vogal deverá ser aproximada para uma das cinco vogais aqui apresentadas, conforme visto nas Seções 4.3 e 4.4.

Uma forma de tornar este resultado contínuo é através da detecção não só da vogal mais provável, ou próxima, mas também da detecção da segunda vogal mais provável, ou próxima, dependendo do método. De posse das duas vogais mais prováveis, ou próximas, realiza-se o *morphing* entre elas. A técnica de *morphing* utilizada encontra-se descrita mais adiante, no Capítulo 5.

Para que seja realizado o *morphing*, é necessário determinar a proporção de cada uma destas duas vogais eleitas no sinal resultante. Esta proporção é representada pelo parâmetro α, que varia linearmente de 1 a 0, onde 1 resulta na vogal mais provável (ou próxima) e 0 na segunda vogal mais provável (ou próxima).

No método de detecção de vogal por probabilidade, a proporção da vogal mais provável será determinada através da probabilidade total de cada uma das duas vogais mais prováveis, e será dada por

$$\alpha = \frac{P_1}{P_1 + P_2}, \tag{4.6}$$

onde P_1 e P_2 representam a probabilidade da primeira e da segunda vogais mais prováveis, respectivamente.

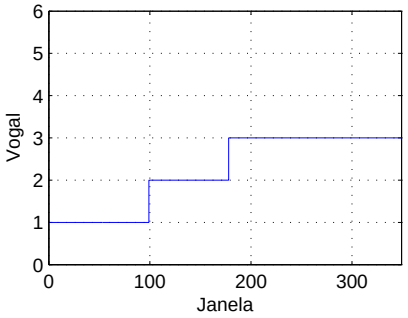
Já no método de detecção de vogal por proximidade, a proporção da vogal mais próxima será determinada através da distância da primeira vogal dividida pela distância total das duas vogais mais próximas. Como a vogal mais próxima

apresenta uma menor distância, esta relação será dada por

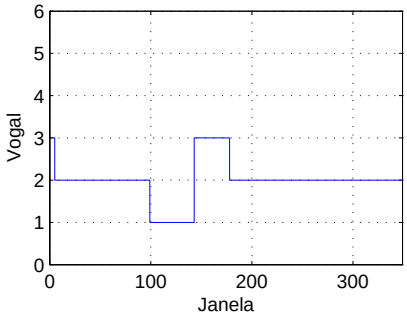
$$\alpha = 1 - \frac{d_1}{d_1 + d_2}, \tag{4.7}$$

onde d_1 e d_2 representam a distância da primeira e da segunda vogais mais próximas, respectivamente.

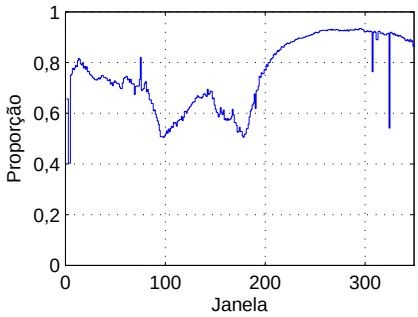
A Figura 4.8 apresenta um exemplo em que o sinal de referência executa a sequência de vogais *a*, *e* e *i*, variando de forma contínua entre elas, i.e., com uma variação suave, interligando uma vogal na outra. A detecção das primeiras e segundas vogais mais prováveis se encontra nos dois primeiros gráficos, respectivamente, onde os algarismos 1, 2 e 3 representam as vogais *a*, *e* e *i*, respectivamente. O terceiro gráfico possui o valor da proporção da primeira vogal frente à segunda, para que seja realizado o *morphing* devido entre elas.



(a) Primeira vogal eleita



(b) Segunda vogal eleita



(c) Proporção

Figura 4.8: Amostra de um cantor seguindo a sequência de vogais *a*, *e* e *i* de forma contínua ao longo do tempo, onde 1 representa a vogal *a*, 2 a vogal *e* e 3 a vogal *i*. A detecção da vogal mais provável se encontra em (a) e a segunda mais provável em (b). A proporção da vogal mais provável é apresentada em (c).

Nesta figura, podemos observar o desenvolvimento gradual da proporção entre as duas vogais mais prováveis, de forma a representar os valores intermediários entre cada duas vogais.

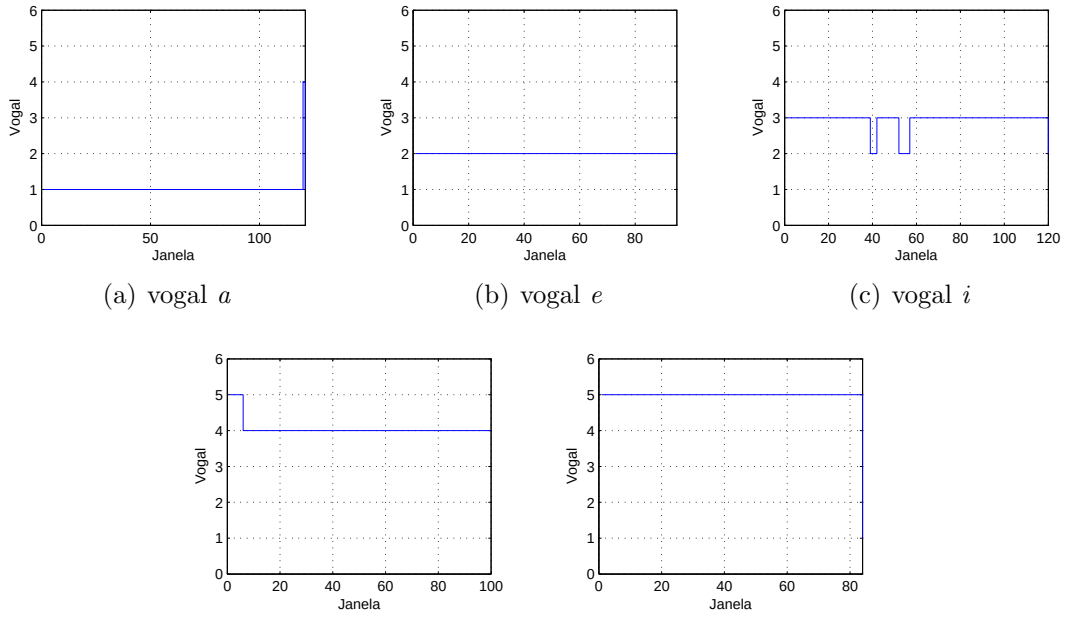
4.8 Avaliação dos Resultados

4.8.1 Detecção de Vogal por Probabilidade

O método de detecção de vogal por probabilidade é bastante eficaz quando se tem um vasto Banco de Dados de Amostras de Vogais, contendo diversos cantores (ou locutores) emitindo todas as vogais e em variados *pitches*. Como não havia um banco de dados como esse disponível, e não havia tempo hábil para a criação de um, este método não se mostrou muito robusto para todas as faixas de *pitch*.

Conforme descrito na Seção 4.5, foram utilizados, para a detecção dos formantes, os quatro primeiros polos do modelo LPC, $N_P = 4$. Através de uma análise subjetiva, foi possível determinar que para o método de detecção de vogal por probabilidade, os resultados foram melhores ao serem utilizados somente os dois primeiros formantes ($N_F = 2$) na análise de probabilidade.

Desta forma, foi possível obter uma boa estimativa de vogais em sinais com determinados valores de *pitch*, como na Figura 4.9, que representa a detecção de vogal ao longo do tempo para cinco sinais diferentes, cada um com uma das vogais *a*, *e*, *i*, *o* e *u*, respectivamente. Todos estes sinais possuem um *pitch* igual a 196 Hz.



(a) vogal *a* (b) vogal *e* (c) vogal *i*

(d) vogal *o* (e) vogal *u*

Figura 4.9: Detecção de vogal por probabilidade para cinco sinais distintos, onde cada sinal possui uma vogal diferentes, e todos em um *pitch* igual a 196 Hz. O algarismo 1 representa a vogal *a*, 2 a vogal *e*, 3 a vogal *i*, 4 a vogal *o* e 5 a vogal *u*.

Neste exemplo, podem-se perceber alguns erros típicos, como na detecção da vogal *a*, que o algoritmo pode entender como sendo a vogal *o*. Isso acontece porque o primeiro e o terceiro formantes destas vogais são muito próximos, diferenciando-se

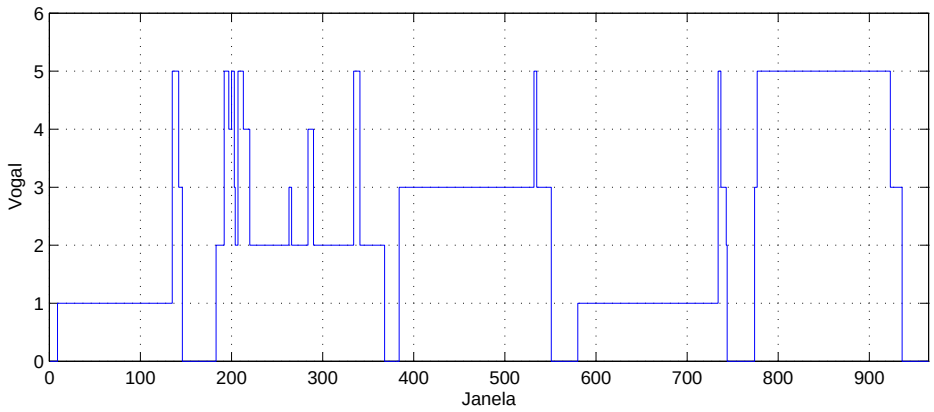


Figura 4.10: Detecção de vogal por probabilidade para uma amostra de um cantor seguindo a sequência de vogais *a*, *e*, *i*, *o* e *u* em um *pitch* de aproximadamente 300 Hz. O algarismo 1 representa a vogal *a*, 2 a vogal *e*, 3 a vogal *i*, 4 a vogal *o* e 5 a vogal *u*.

as duas somente pelo segundo formante.

Ao testar este método em outros valores de *pitch*, pode-se ver um número maior de erros, como no caso da Figura 4.10, em que um cantor executa, de forma sequencial, as vogais *a*, *e*, *i*, *o* e *u* em um *pitch* de aproximadamente 300 Hz. Quanto maior o valor do *pitch*, mais difícil se torna a tarefa de detectar a vogal emitida, pois a distinção entre os formantes das diferentes vogais se torna menos evidente. De maneira subjetiva, podemos notar que a diferença percebida entre a vogal *o* e a vogal *a*, por exemplo, torna-se sutil em valores altos de *pitch*, pois quanto maior o *pitch* maior é a dificuldade para um cantor de executar com perfeição as diferentes vogais. Neste exemplo, podemos ver a presença de alguns picos espúrios que não foram reparados pelo ajuste final apresentado na Seção 4.6, pois possuem duração maior do que três *frames*.

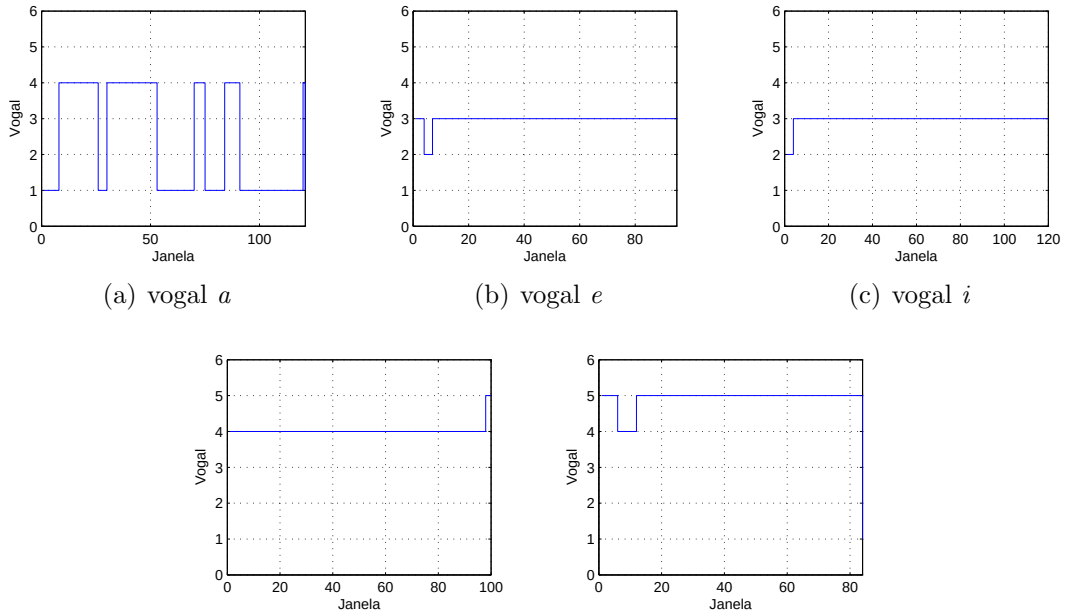
4.8.2 Detecção de Vogal por Proximidade

O método de detecção de vogal por proximidade apresentou resultados semelhantes aos obtidos pelo método de probabilidade, porém com um número maior de erros.

De forma semelhante aos demais métodos, foram utilizados os $N_P = 4$ primeiros polos do modelo LPC como os possíveis primeiros formantes. Através de uma análise subjetiva dos sinais sintetizados, foi possível determinar que o método de detecção de vogal por proximidade tem um melhor desempenho ao serem utilizados os três primeiros formantes na detecção ($N_F = 3$), desta forma trabalhando com três dimensões.

A Figura 4.11 apresenta os resultados na detecção de vogal para cinco sinais distintos, cada um contendo uma das cinco vogais diferentes, e todos em um valor

de *pitch* igual a 196 Hz. Estes sinais são os mesmos da Figura 4.9.



(a) vogal *a* (b) vogal *e* (c) vogal *i*

(d) vogal *o* (e) vogal *u*

Figura 4.11: Detecção de vogal por proximidade para cinco sinais distintos, onde cada sinal possui uma vogal diferentes, e todos em um *pitch* igual a 196 Hz. O algarismo 1 representa a vogal *a*, 2 a vogal *e*, 3 a vogal *i*, 4 a vogal *o* e 5 a vogal *u*.

Pode-se perceber que os resultados são um tanto inferiores em comparação aos do método de probabilidade. Vemos um grande erro na detecção das vogais *a* e *e*, que foram confundidas com as vogais *o* e *i*, respectivamente. Estes erros são facilmente compreendidos ao verificarmos a proximidade dos pontos que representam os formantes de cada vogal no plano $F_1 \times F_2$, apresentado na Figura 4.4.

A Figura 4.12 apresenta o resultado da detecção de vogal por proximidade do sinal utilizado na Figura 4.10. Nesta figura, podemos observar uma queda na qualidade dos resultados devido ao aumento do *pitch*, tal qual ocorreu no método de detecção por probabilidade. A detecção de vogal em sinais com alto valor de *pitch* também se mostra uma tarefa difícil para o método de detecção por proximidade, pois a distinção entre os formantes das diferentes vogais se torna menos evidente, como discutido na Seção 4.8.1. Mais uma vez, podemos ver a presença de alguns picos espúrios que não foram reparados pelo ajuste final apresentado na Seção 4.6, pois possuem duração maior do que três *frames*.

4.8.3 Substituição dos Primeiros Polos

Conforme descrito na Seção 4.5, o método de substituição dos primeiros polos apresentou bons resultados na representação da vogal presente no sinal de referência.

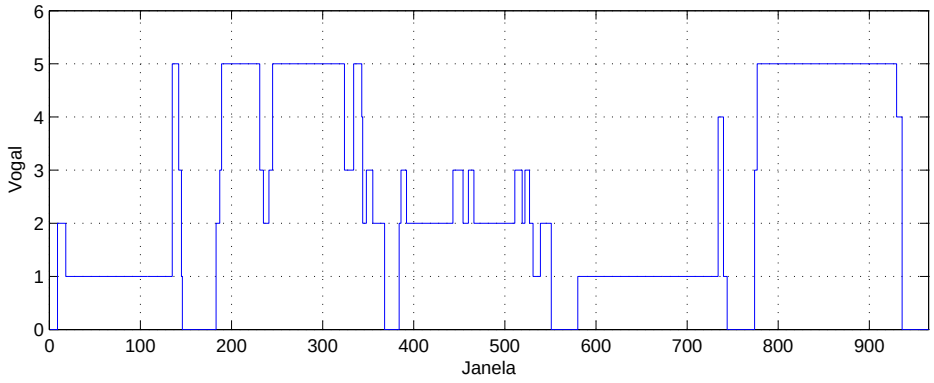


Figura 4.12: Detecção de vogal por proximidade para uma amostra de um cantor seguindo a sequência de vogais *a*, *e*, *i*, *o* e *u* em um *pitch* de aproximadamente 300 Hz. O algarismo 1 representa a vogal *a*, 2 a vogal *e*, 3 a vogal *i*, 4 a vogal *o* e 5 a vogal *u*.

Porém, além da vogal, o sinal sintetizado também apresentou características do timbre do sinal de referência.

Desta forma, este método não será utilizado neste trabalho, pois um dos objetivos desta dissertação é criar sinais sintéticos com o timbre de um determinado cantor alvo. Portanto, nenhuma alteração deverá ser feita no timbre almejado.

4.9 Conclusão

Inicialmente, este capítulo descreveu o desenvolvimento de um banco de dados que contém os valores estatísticos dos três primeiros formantes de cada vogal, para que seja utilizado pelas técnicas de detecção de vogal por probabilidade e por proximidade.

A partir das informações dos formantes e do banco de dados de vogais, foram propostas duas técnicas de detecção de vogal, uma que seleciona a vogal de maior probabilidade (método de detecção de vogal por probabilidade) e outra que seleciona a vogal com menor distância entre formantes (método de detecção de vogal por proximidade).

A terceira técnica proposta não almeja a detecção da vogal, mas faz uso da substituição dos polos da envoltória espectral que representam os primeiros formantes do sinal de voz para que seja sintetizada a vogal presente no sinal de referência. Apesar de essa técnica apresentar bons resultados na vogal sintetizada, ela carrega características relevantes acerca do timbre do sinal de referência. Como a preservação do timbre do sinal alvo é um dos objetivos deste trabalho, esta técnica não será utilizada.

Após avaliação dos resultados dos dois primeiros métodos apresentados de de-

tecção de vogal, chegou-se à conclusão de que os dois métodos apresentam bons resultados em sinais com valores de *pitch* medianos. Em *pitches* muito baixos, os resultados apresentam alguns erros relevantes. Porém, em *pitches* muito altos, a determinação da vogal torna-se ainda mais difícil, devido ao aumento da proximidade entre as diferentes vogais, conforme mencionado anteriormente.

Entre estes dois métodos, a detecção por probabilidade se mostrou mais confiável e robusta do que a por proximidade. Desta forma, para que seja feita a detecção da vogal emitida pelo sinal de referência, e seja utilizada na etapa de síntese, o método de detecção de vogal por probabilidade se mostrou mais apropriado para a execução deste trabalho.

Uma possível maneira de melhorar os resultados deste método seria através da criação de um novo Banco de Dados de Probabilidades de Vogais a partir de um Banco de Dados de Amostras de Vogais mais variado, i.e., contendo sinais de diversos cantores e locutores diferentes, com uma variedade maior de *pitch* e para todas as possíveis variações das vogais, i.e., vogais abertas, fechadas e nasaladas. Com isso, aumentar-se-ia a confiabilidade dos valores estatísticos de cada formante para cada vogal.

Capítulo 5

Morphing de Envoltórias Espectrais

5.1 Introdução

Neste ponto da dissertação, a fase de análise encontra-se concluída, pois foram detectados o *pitch* do sinal de referência ao longo do tempo (a melodia presente no sinal) e as vogais executadas por ele também ao longo do tempo. Desta forma, as características responsáveis pela compreensão da música executada pelo sinal de referência encontram-se disponíveis para a etapa de síntese.

Neste capítulo, será descrito o conceito de *morphing*, nome utilizado na literatura para referir-se a técnicas que venham a mesclar parâmetros de dois sinais diferentes, de forma a obter um novo sinal com características intermediárias, conforme será descrito na Seção 5.2.

O *morphing* será amplamente utilizado na etapa de síntese desta dissertação, pois será responsável por minimizar mudanças bruscas entre cada duas janelas sintetizadas através de sua aplicação em um determinado intervalo de janelas. Também será utilizado na obtenção de valores intermediários de vogais, para que não sejam reproduzidas somente as vogais *a*, *e*, *i*, *o* e *u*, obtidas pelo método de detecção de vogal descrito no capítulo anterior.

A Seção 5.3 apresentará com maiores detalhes a motivação de se utilizar o *morphing* nesta dissertação, descrevendo como o *morphing* poderá aperfeiçoar os resultados obtidos neste trabalho.

Já as Seções 5.4 e 5.5 descreverão duas diferentes técnicas para a realização do *morphing*, apresentando uma análise crítica sobre elas.

5.2 Definição

A palavra *morphing* vem a ser o gerúndio, na língua inglesa, do verbo *to morph*, que pode ser traduzido por transformar. *Morph* também pode ser compreendido como o substantivo metamorfose.

Este termo é muito utilizado na literatura para indicar técnicas de transformação de sinais, normalmente ao longo do tempo, onde um sinal começará com uma determinada característica e terminará com outra através de transformações suaves entre elas, como por exemplo um sinal que comece com um timbre de violão e termine com o de uma flauta, variando suavemente de um para o outro ao longo do tempo.

O *morphing* também pode ser utilizado para se obter, a partir de dois sinais distintos, um novo sinal com timbre, duração, *pitch* e/ou intensidade intermediários. Conforme citado no primeiro capítulo desta dissertação, o sintetizador apresentado por Haken *et al.* chamado de *Continuum* realiza o *morphing* em três dimensões, onde o eixo x corresponde ao *pitch*, o eixo y ao timbre e o z à dinâmica [18].

5.3 Motivação

Como a proposta desta dissertação não é transformar um sinal existente, mas sim criar um sinal totalmente sintético com determinadas características de um sinal de referência, o uso de *morphing* não parece necessário em um primeiro momento.

Contudo, todo o processamento aqui proposto, tanto de análise quanto de síntese, é realizado em janelas de 1024 amostras, com sobreposição de 512 amostras a uma taxa de amostragem igual a 44,1 kHz, o que resulta em janelas de 23,2 ms. Desta forma, qualquer variação brusca entre janelas soará artificial ao ouvido humano, pois não seria possível um ser humano realizar variações significativas em tão pouco tempo.

As Seções 2.5 e 4.6 apresentaram melhorias para esta questão na fase de análise. Porém, conforme será detalhado no Capítulo 6, variações bruscas entre duas janelas sintetizadas são comumente encontradas ao final do processamento devido à variação das envoltórias espectrais de uma janela para outra.

Por isso, deverá ser estabelecido um intervalo de n janelas ao longo do qual o timbre deverá variar suavemente através do *morphing* entre os timbres extremos. Terminado este intervalo, um novo *morphing* deverá ser feito para o próximo intervalo de n janelas até chegar ao final do sinal, conforme ilustrado na Figura 5.1. Isso tornará suaves as transições ao longo do tempo no sinal sintetizado.

Outra utilidade do *morphing* de timbres para este trabalho é a determinação de vogais intermediárias, i.e., entre os valores das vogais *a*, *e*, *i*, *o* e *u* da língua portuguesa. Conforme descrito na Seção 4.7, o algoritmo de detecção de vogal de-

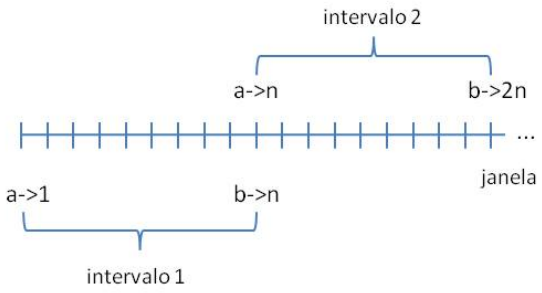


Figura 5.1: Intervalo de janelas para que seja feita uma transição suave de timbres entre *frames* através do *morphing*.

senvolvido somente retorna uma destas cinco vogais. Com o fim de obter valores contínuos entre estas vogais, o algoritmo de detecção de vogal retornará, também, a segunda vogal mais provável de forma que, através do *morphing* do timbre destas vogais, se obtenha um fonema vocálico intermediário e mais próximo da vogal executada pelo sinal de referência.

Os detalhes de implementação do *morphing* no algoritmo proposto por esta dissertação serão apresentados no Capítulo 6.

Conforme descrito no Capítulo 3, a informação de timbre de um sinal pode ser encontrada em sua envoltória espectral. Com isso, o *morphing* de timbres realizado nesta dissertação será feito através da manipulação da envoltória espectral dos dois sinais a serem combinados através de uma interpolação entre tais envoltórias.

Vejamos, abaixo, duas formas de interpolação de envoltórias espectrais.

5.4 Interpolação Ingênua

A forma mais intuitiva de realizarmos o *morphing* de dois timbres, do sinal fonte e do sinal alvo, é através da interpolação dos vetores que contêm as envoltórias espectrais. Para isso, utiliza-se um parâmetro α, chamado fator de *morphing* ou interpolação, variando de forma linear de 1 a 0, onde 1 resulta no sinal fonte e 0 no sinal alvo. Assim, 0,5 deve resultar em um sinal com um timbre perceptivamente a meio caminho entre os dois.

Segundo Caetano e Rodet, este tipo de interpolação pode ser chamada de ingênua (em inglês, *naïve*), pois desconsidera parâmetros relacionados à percepção humana dos timbres, contidos em sua envoltória espectral [20].

Esta interpolação é obtida simplesmente através de

$$m[n] = \alpha \cdot f[n] + (1 - \alpha) \cdot a[n], \tag{5.1}$$

onde $f[n]$ vem a ser a envoltória espectral do sinal fonte, $a[n]$ a envoltória espectral

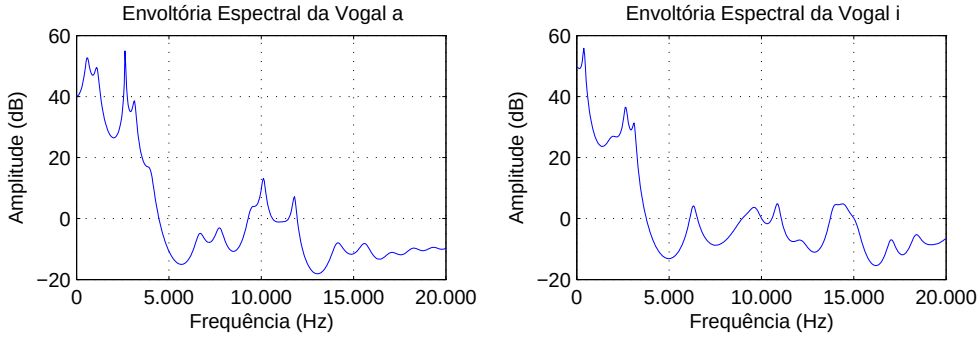


Figura 5.2: A esquerda, a envoltória espectral de um sinal de voz cantada executando a vogal *a* em um *pitch* de 196 Hz em uma janela de 2,3 ms. A direita, a envoltória espectral de um sinal de voz cantada executando a vogal *i* em um *pitch* de 196 Hz em uma janela de 2,3 ms.

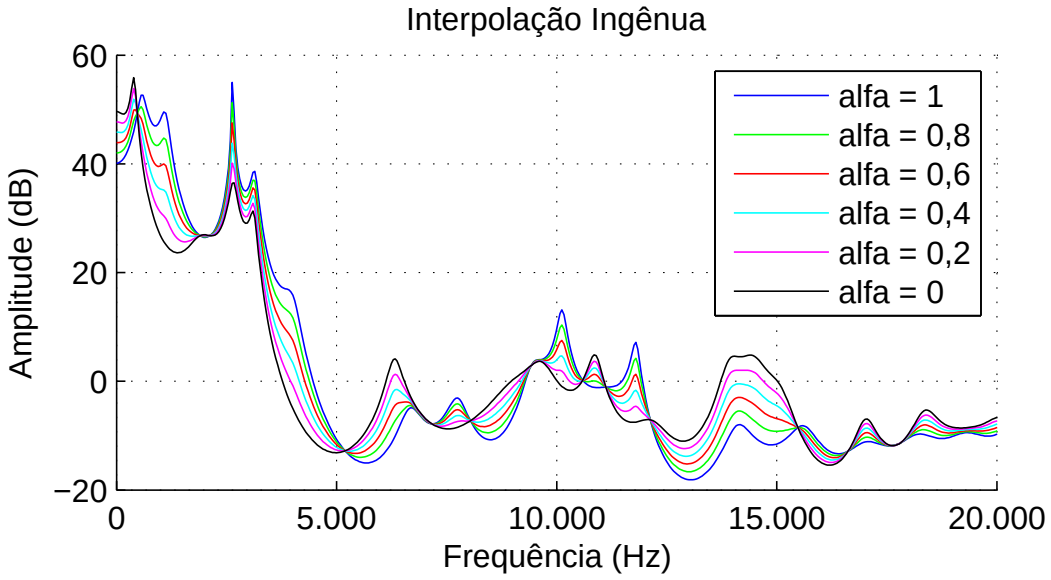


Figura 5.3: Interpolação ingênua dos dois sinais apresentados na Figura 5.2 para valores de α iguais a 1; 0,8; 0,6; 0,4; 0,2 e 0.

do sinal alvo e $m[n]$ a envoltória espectral resultante da interpolação ingênua.

Para efeito de teste, foram realizados *morphings* através da interpolação ingênua de dois sinais diferentes, um onde o cantor executa a vogal *a* e outro a vogal *i*, ambos em um *pitch* de 196 Hz em uma janela de 2,3 ms. Suas envoltórias espectrais estão ilustradas na Figura 5.2.

Foi simulada a interpolação ingênua entre estas duas envoltórias para valores de α iguais a 1; 0,8; 0,6; 0,4; 0,2 e 0, como pode ser visto na Figura 5.3.

Uma grande desvantagem deste método de *morphing* está na produção de novos formantes na envoltória espectral, pois nos valores intermediários de α ele tende a manter os formantes das envoltórias espectrais do sinal fonte e do sinal alvo em suas posições frequenciais originais, apenas com uma diferenciação em suas amplitudes.

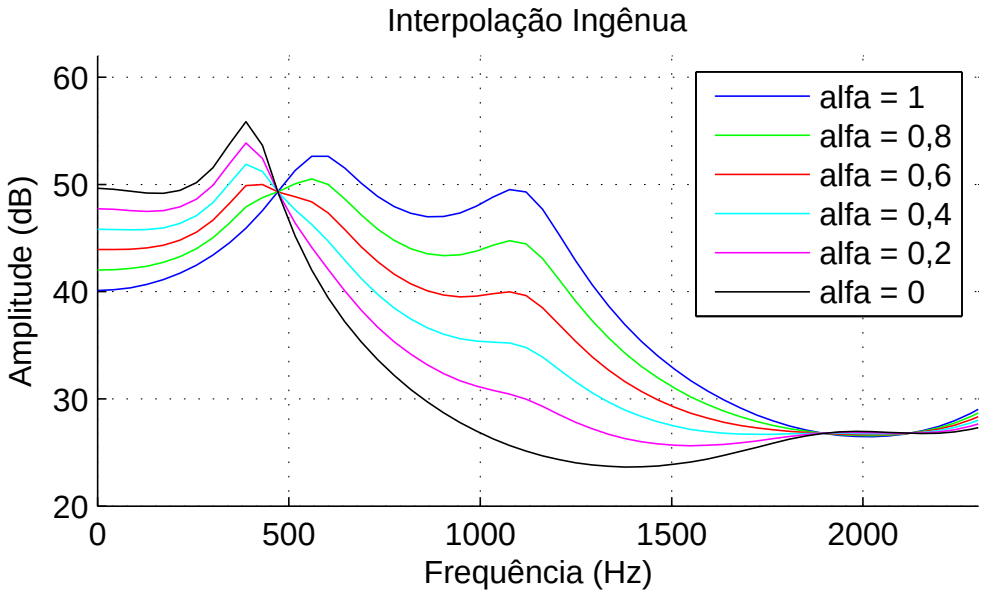


Figura 5.4: Ampliação dos dois primeiros formantes dos sinais da Figura 5.3. Interpolação ingênua dos dois sinais apresentados na Figura 5.2 para valores de α iguais a 1; 0,8; 0,6; 0,4; 0,2 e 0.

Este fato pode ser visto na Figura 5.4, que vem a ser uma ampliação dos dois primeiros formantes dos sinais da Figura 5.3.

Nesta figura, podemos verificar que em α igual a 0,8 e 0,6, por exemplo, a envoltória apresenta levemente duas corcovas nas frequências dos primeiros formantes do sinal fonte e do sinal alvo, como se fosse um polo em cada frequência dos formantes do sinal fonte e do sinal alvo, o que duplicaria, no pior caso, o número de formantes no sinal resultante.

5.5 Interpolação dos Coeficientes LPC

Caetano e Rodet apresentaram um método de *morphing* que preserva o número de formantes e estima suas posições frequenciais em valores intermediários de α [20] [23] [19].

Este método, em vez de realizar a interpolação dos vetores que contêm as envoltórias espectrais do sinal fonte e alvo, executa a interpolação dos coeficientes do modelo LPC destes sinais, que vêm a ser os coeficientes do filtro cuja resposta em frequência estima suas envoltórias espectrais, conforme visto no Capítulo 3. Para a realização desta interpolação, também faz-se uso do fator de interpolação α, que varia de forma linear de 1 a 0, onde 1 resulta no sinal fonte e 0 no sinal alvo.

De forma similar a equação (5.1), a interpolação é feita através de

$$m_{\mathrm{LPCcoef}}[n] = \alpha \cdot f_{\mathrm{LPCcoef}}[n] + (1-\alpha) \cdot a_{\mathrm{LPCcoef}}[n], \tag{5.2}$$

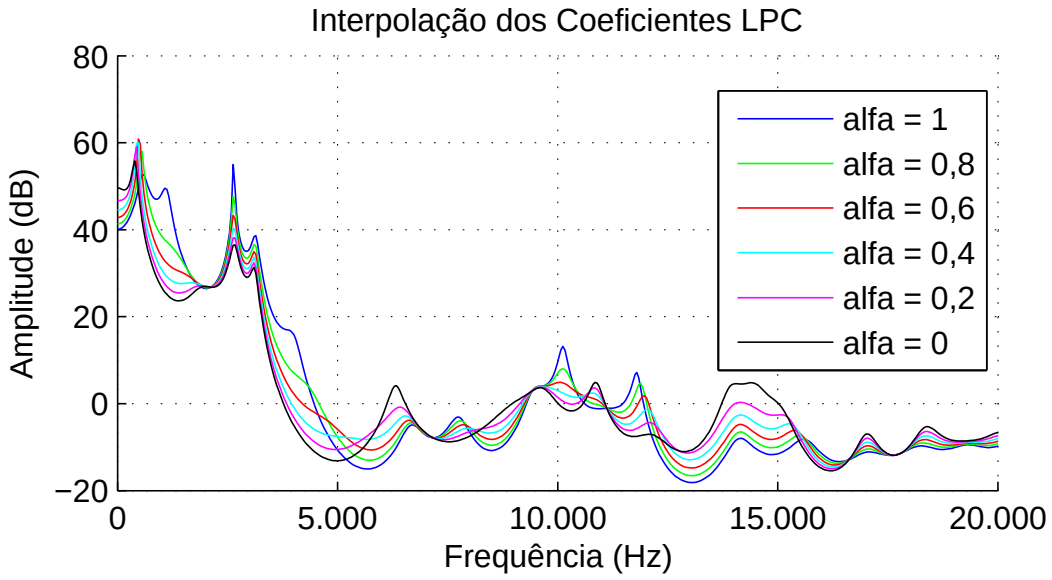


Figura 5.5: Interpolação dos coeficientes LPC dos dois sinais apresentados na Figura 5.2 para valores de α iguais a 1; 0,8; 0,6; 0,4; 0,2 e 0.

onde $f_{\mathrm{LPCcoef}}[n]$ vêm a ser os coeficientes do modelo LPC do sinal fonte, $a_{\mathrm{LPCcoef}}[n]$ os coeficientes do modelo LPC do sinal alvo e $m_{\mathrm{LPCcoef}}[n]$ os coeficientes do modelo LPC resultantes da interpolação.

De forma análoga à simulação apresentada na Figura 5.3, a Figura 5.5 ilustra um exemplo de *morphing* através da interpolação dos coeficientes do modelo LPC dos exemplos apresentados na Figura 5.2 para valores de α iguais a 1; 0,8; 0,6; 0,4; 0,2 e 0.

Como pode ser visto nesta figura, a técnica de *morphing* através da interpolação dos coeficientes LPC não cria novos formantes nos sinais resultantes em valores intermediários de α, mas faz com que os formantes sejam gradualmente deslocados na frequência de forma a se posicionarem em frequências intermediárias entre os respectivos formantes do sinal fonte e do sinal alvo.

Isso pode ser melhor visualizado na Figura 5.6, que vem a ser uma ampliação dos dois primeiros formantes dos sinais da Figura 5.5.

Ao analisarmos o primeiro formante desta figura, vemos claramente que ele é transportado frequencialmente do primeiro formante do sinal fonte para o primeiro formante do sinal alvo, de acordo com o valor de α. Desta forma, não são gerados formantes irreais, que propiciariam timbres claramente sintéticos, mas são produzidos timbres perceptivamente intermediários entre o sinal fonte e o sinal alvo.

Por isso, optou-se por utilizar esta técnica de *morphing* para este trabalho de mestrado.

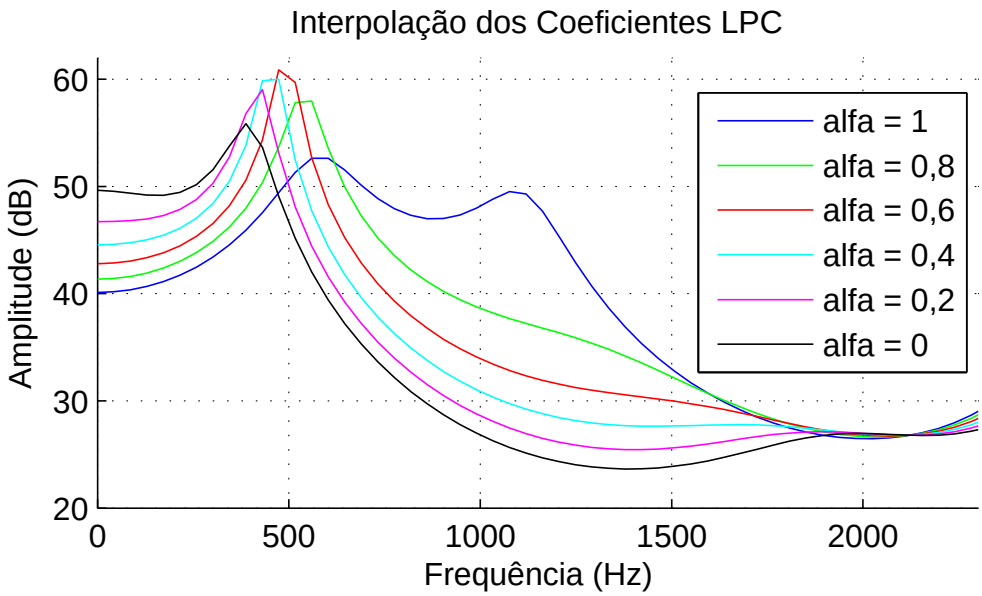


Figura 5.6: Ampliação dos dois primeiros formantes dos sinais da Figura 5.5. Interpolação dos coeficientes LPC dos dois sinais apresentados na Figura 5.2 para valores de α iguais a 1; 0,8; 0,6; 0,4; 0,2 e 0.

5.6 Conclusão

Este capítulo expôs o conceito de *morphing*, o qual vem a ser responsável, nesta dissertação, por suavizar o resultado da síntese e também obter valores intermediários entre as vogais *a*, *e*, *i*, *o* e *u*.

Foram descritos dois métodos para a realização do *morphing* entre os timbres dos sinais fonte e alvo: a interpolação ingênua e a interpolação dos coeficientes LPC.

O primeiro método realiza a interpolação dos vetores que contêm os valores discretos das envoltórias espectrais. Este método não se mostrou eficaz, pois mantém a posição frequencial dos formantes dos dois sinais e apenas ajusta suas amplitudes, o que pode acarretar em um acréscimo no número de formantes presentes no sinal gerado, o que normalmente resultará em um sinal perceptivelmente sintético.

Já o método de interpolação dos coeficientes LPC realiza o deslocamento da posição frequencial de cada formante à medida que se transforma o timbre do sinal fonte no sinal alvo. Com isso, o número de formantes presentes nos sinais sempre será preservado, resultando em um timbre final perceptivelmente intermediário entre os dois originais.

Desta forma, o *morphing* através da interpolação dos coeficientes LPC mostrou-se o método mais apropriado para o desenvolvimento desta dissertação, pois resulta em uma envoltória espectral cujo timbre associado soará natural e com as características desejadas dos sinais fonte e alvo.

Capítulo 6

Algoritmo de Síntese

6.1 Introdução

Os sinais de áudio encontrados na natureza são produzidos através da perturbação do meio no qual eles se propagam. Isso pode ser verificado através de um simples estalar de dedos, o qual produz uma perturbação que gera uma onda mecânica que se propaga pelo ar e chega aos nossos ouvidos, permitindo-nos identificar o som produzido por tal ação. Conforme mencionado anteriormente, a ressonância desta onda mecânica em um meio físico produz um determinado timbre, pois este meio opera como um filtro que modela as frequências do espectro de um sinal.

A síntese de áudio possui a finalidade de criar artificialmente um sinal com um determinado timbre, que pode ser uma imitação de um timbre encontrado na natureza ou um totalmente novo.

Conforme descrito no Capítulo 1, o objetivo deste trabalho é recriar o timbre de um determinado cantor para que ele aparente ter executado uma determinada melodia contida em um sinal de referência (sinal de entrada). Este capítulo descreve técnicas para se criar artificialmente um sinal que contenha o timbre de um determinado cantor previamente estudado (sinal alvo) através da síntese de áudio.

Inicialmente, a Seção 6.2 descreverá o conceito de síntese aditiva e a razão de ter sido escolhida como a técnica de síntese mais adequada a este trabalho. Esta seção também introduzirá a técnica de síntese aditiva que realiza a criação do sinal sintético no domínio da frequência, e o transforma para o domínio do tempo através da inversa da transformada de Fourier discreta (iDFT).

Normalmente, as técnicas de processamento de sinais de áudio que lidam com a transformada de Fourier exigem uma atenção especial quanto à manipulação do espectro de fase, o qual, na maioria das vezes, não é facilmente compreendido. Por esta razão, a Seção 6.2.2 possui a finalidade de descrever com detalhes a criação do espectro de fase sintético.

A Seção 6.3 descreve a criação do banco de dados de síntese, o qual será utilizado para o armazenamento das envoltórias espectrais dos sinais alvo, de forma que sejam simulados os seus timbres no sinal resultante da síntese.

Na Seção 6.4 é apresentada a implementação do algoritmo desenvolvido neste trabalho de mestrado, seguido da Seção 6.5, que descreve a avaliação de tal algoritmo. Para que seja feita esta avaliação, foram realizados quatro experimentos que revelam os pontos positivos e negativos dos métodos aqui propostos.

6.2 Síntese Aditiva

Inúmeros métodos para sintetizar um sinal de áudio podem ser encontrados na literatura, como a síntese por amostras (em inglês, *sampling synthesis*) [15], a síntese FM (*frequency modulation synthesis*) [48] e a síntese granular (*granular synthesis*) [49].

Conforme visto anteriormente, uma importante característica de sinais quase periódicos e harmônicos é o fato de seus espectros de módulo se constituírem, principalmente, de picos (máximos locais) que representam as parciais harmônicas do sinal, como no caso de sinais de voz. Uma técnica de síntese que tira proveito desta característica é a síntese aditiva.

Sabe-se que uma senoide é representada na frequência por dois impulsos unitários localizados nas frequências $\pm f_0$, onde f_0 vem a ser a frequência de oscilação desta senoide. Desta forma, temos que

$$\cos(2\pi f_0 t) \overset{\mathcal{F}}{\longleftrightarrow} \frac{1}{2}\left[\delta(f - f_0) + \delta(f + f_0)\right], \tag{6.1}$$

conforme demonstrado por Oppenheim e Willsky [50]. A Figura 6.1 ilustra o espectro de módulo de um cosseno com uma frequência de oscilação igual a f_0.

A partir deste conceito, a síntese aditiva constitui-se na soma de diversas senoides, sendo a primeira referente à frequência fundamental e as demais às suas parciais harmônicas, formando assim o timbre desejado através da formação dos picos no espectro de módulo [12]. Este processo é repetido a cada *frame* do sinal, para que haja uma variação da melodia e da intensidade do sinal ao longo do tempo.

Para a realização desta soma, cria-se um banco de osciladores contendo o número de senoides desejadas, i.e., o número desejado de parciais, onde cada senoide é ajustada em termos de módulo (A), frequência (f) e fase (ϕ), de acordo com a equação

$$y = A \operatorname{sen}(2\pi f t + \phi). \tag{6.2}$$

Quanto maior o número de senoides, maior a semelhança entre um sinal sintetizado e um sinal real; por isso, o ideal seria a criação de todas as senoides necessárias

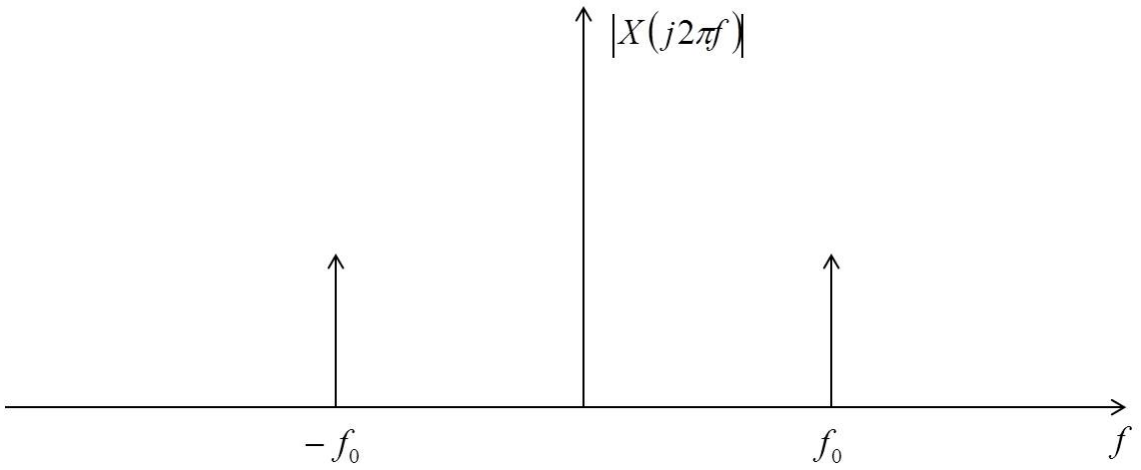


Figura 6.1: Espectro de módulo de um cosseno com frequência de oscilação igual a f_0.

para preencher o espectro audível. Porém, ao aumentarmos o número de parciais, também será aumentado o custo computacional do processo.

Para a realização da síntese, este trabalho utilizará a técnica de síntese aditiva através da inversa da DFT [14], que será detalhada na Seção 6.2.1. A ideia básica deste método é criar diretamente na frequência os picos gerados por cada senoide do banco de osciladores, de forma a preencher todo o espectro do sinal com as suas parciais harmônicas.

6.2.1 Síntese aditiva usando a inversa da DFT

O método de síntese aditiva através da inversa da DFT se dá através da construção do espectro de cada *frame* diretamente no domínio da frequência. Após esta criação, realiza-se a inversa da transformada de Fourier discreta (iDFT) para se obter o sinal no domínio do tempo.

Conforme mencionado acima, é necessário definir módulo, frequência e fase de cada parcial, os quais podemos visualizar no espectro de módulo e fase de um sinal sintético ilustrados na Figura 6.2. Com este elevado número de variáveis, este método proporciona uma grande liberdade de se alterar o timbre como se desejar; porém, possui a desvantagem de tornar a síntese uma ferramenta complexa e que, sem um método para modelar tais variáveis, dificilmente apresentará bons resultados. Para que cada variável seja determinada de forma controlada, facilitando a obtenção de um resultado favorável, vejamos abaixo como se obter a frequência, o módulo e a fase de cada parcial.

O primeiro passo para a criação do espectro de módulo deverá ser a determinação da frequência de cada parcial harmônica, para que assim seja feito o devido posicionamento de cada pico no espectro. De forma simples, o primeiro pico deverá ser po-

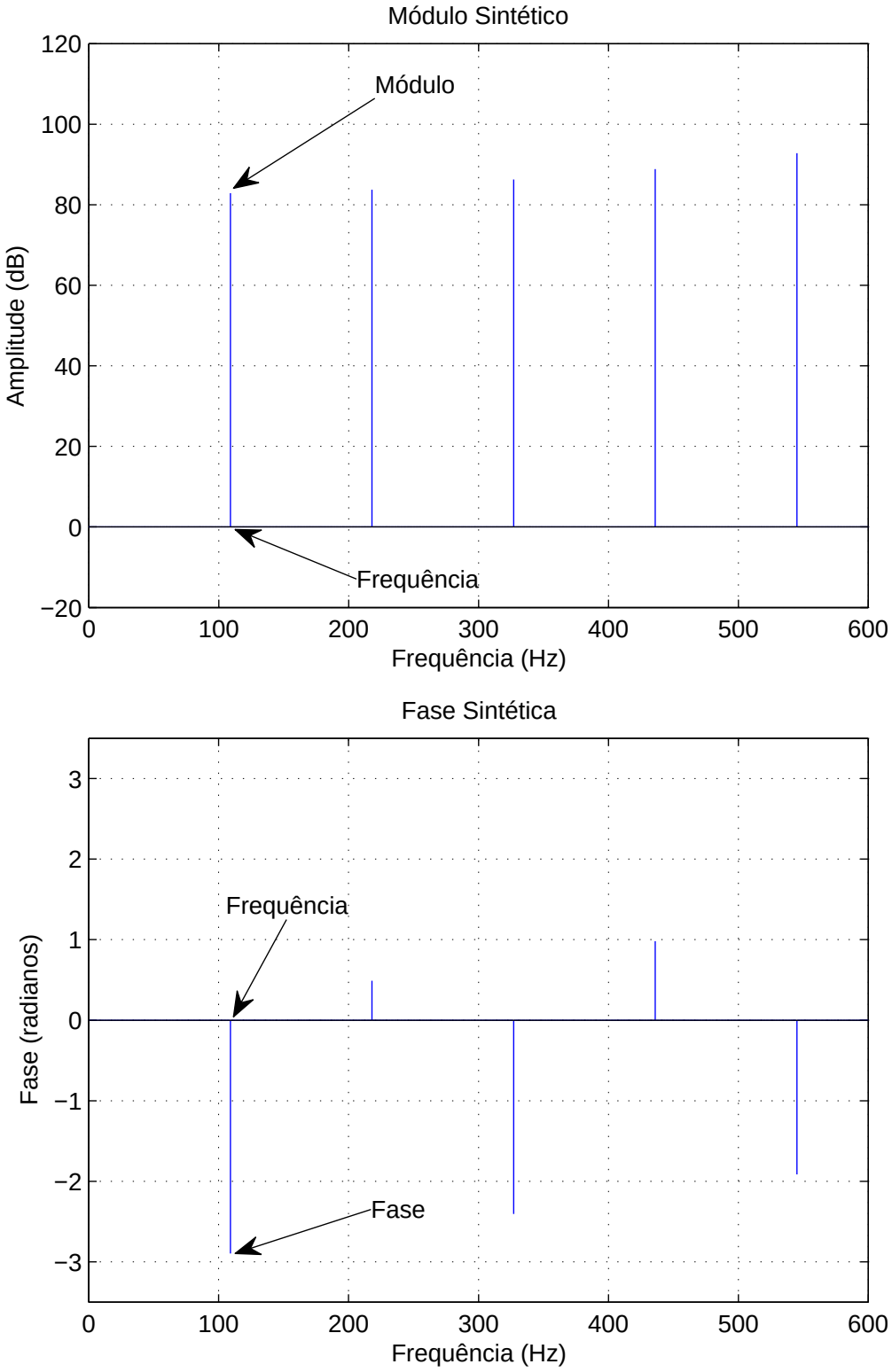


Figura 6.2: Espectro de módulo e fase de parciais harmônicas de um sinal sintetizado.

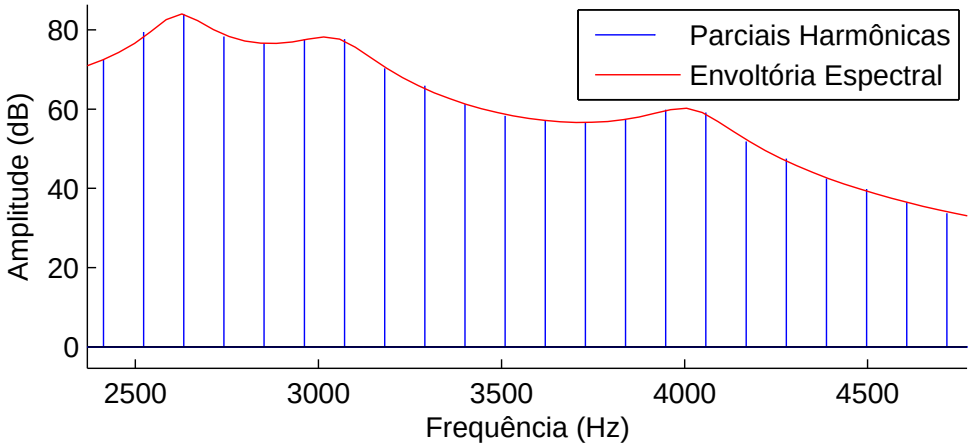


Figura 6.3: Espectro de módulo de um sinal sintetizado. Em azul: as parciais harmônicas. Em vermelho: a envoltória espectral.

sicionado na frequência relacionada ao *pitch* desejado; as demais parciais harmônicas seguirão uma progressão aritmética com razão r igual a frequência fundamental, i.e., o intervalo entre cada parcial deverá ser igual ao valor do *pitch*.

O segundo passo deverá ser a definição da amplitude de cada pico. Para tal, utiliza-se a envoltória espectral de forma que a amplitude de uma parcial localizada na frequência f_i será igual ao módulo da envoltória espectral em f_i, como foi visto no Seção 3.2.

A Figura 6.3 ilustra a disposição no espectro de módulo das parciais harmônicas de um sinal sintetizado com um *pitch* igual a 109 Hz executando a vogal *a*. Nela podemos ver, em vermelho, a envoltória espectral utilizada para determinar a amplitude de cada parcial, em azul.

Neste ponto, a parte positiva do espectro de módulo de um sinal sintético já se encontra devidamente construída. Para determinar sua parte negativa, basta espelharmos a parte positiva em relação ao eixo das ordenadas, visto que o espectro de módulo é um sinal par. Um exemplo de espectro de módulo sintetizado completo de um sinal de voz executando a vogal *a* em um *pitch* de 100 Hz pode ser visto na Figura 6.4.

O terceiro passo deverá ser a construção do espectro de fase, discutida na Seção 6.2.2.

Uma vez construído o espectro complexo (de módulo e de fase), utiliza-se a inversa da DFT para obter o sinal no tempo [14]. Com esta técnica de síntese aditiva o custo computacional reduz-se consideravelmente, permitindo facilmente adicionar todas as parciais possíveis ao espectro.

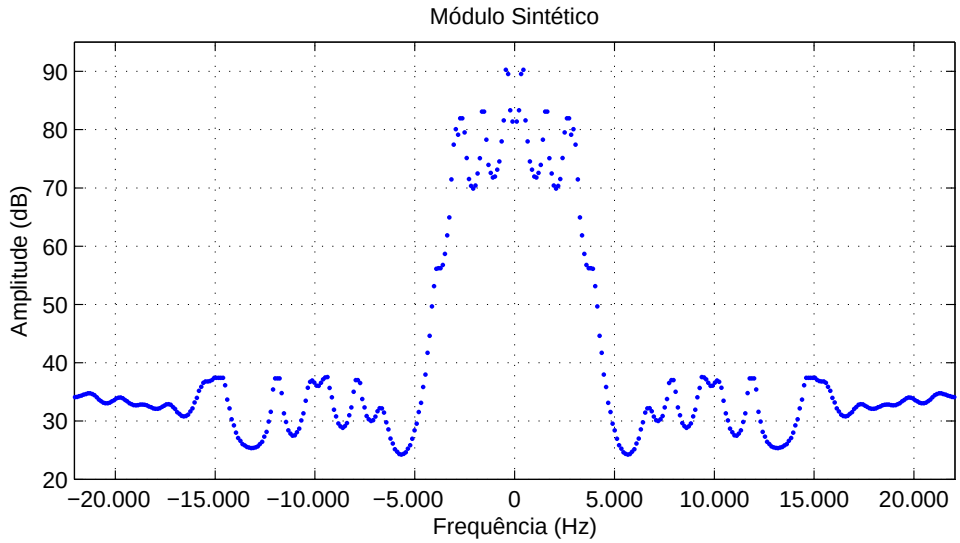


Figura 6.4: Espectro de módulo de um sinal de voz executando a vogal *a* em um *pitch* de 100 Hz, parte positiva e parte negativa.

6.2.2 Espectro de Fase

De forma semelhante ao espectro de módulo, será criada primeiramente a parte positiva do espectro de fase; a parte negativa será obtida pelo espelhamento da parte positiva em relação ao eixo das ordenadas e das abscissas, visto que o espectro de fase é um sinal impar, como pode ser visto na Figura 6.5, que ilustra um espectro de fase sintético de um sinal com *pitch* igual a 146 Hz a uma taxa de amostragem de 44,1 kHz.

A fase de um sinal, ao contrário de seu módulo, nem sempre é facilmente compreendida. Pode-se dizer que a fase ϕ de uma senoide indica em qual momento do seu período ela estará na origem dos tempos. A Figura 6.6 ilustra um cosseno com fase 0, fase $\pi/2$ e fase π, respectivamente.

Para facilitar a compreensão do espectro de fase, o Apêndice A faz uma análise dos espectros de frequência, relembrando a dualidade entre sinais senoidais e exponenciais, assim como série e transformada de Fourier.

Cálculo da Fase

Não podemos nos esquecer de que desejamos realizar a síntese através da soma de várias cossenoides, a fim de compor todas as parciais do sinal. O fato de estarmos lidando com cossenoides nos é favorável, pois vimos que ao representarmos um cosseno de f_i Hz no domínio da frequência temos um impulso, no espectro de módulo, localizado em f_i Hz. Vimos, também, que em seu espectro de fase teremos, em f_i Hz, a fase que esta cossenoide possui no início da janela de observação, conforme descrito nas seções acima, podendo variar de $-\pi$ a π.

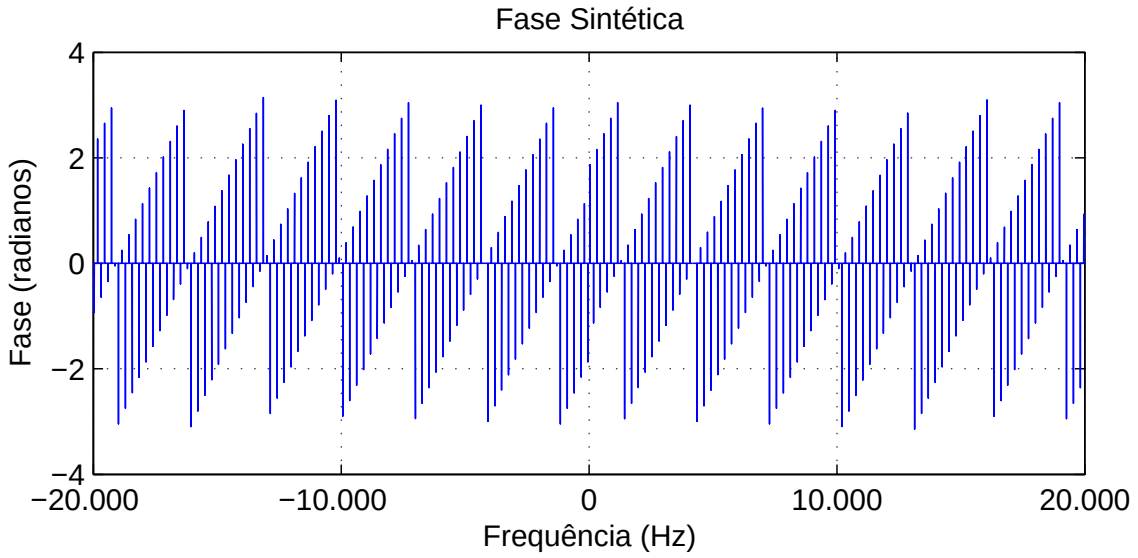
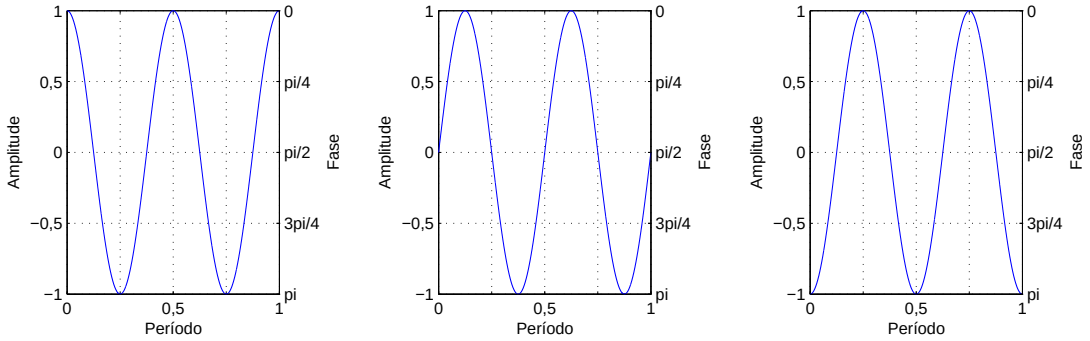


Figura 6.5: Espectro de fase sintético, criado para um *pitch* de 146 Hz a uma taxa de amostragem de 44,1 kHz.



(a) $\phi = 0$ (b) $\phi = \pi/2$ (c) $\phi = \pi$

Figura 6.6: Cosseno com fase 0, $\pi/2$ e π, da esquerda para a direita.

Conforme será visto ainda neste capítulo, as janelas artificiais geradas pela síntese serão sobrepostas e somadas (em inglês, *overlap and add*) para se obter o sinal final. Para que esta sobreposição e soma não interfira na amplitude do sinal final, cada janela sintética será multiplicada por uma janela de Hann, como pode ser visto na Figura 6.7. Esta figura apresenta a multiplicação de uma janela de Hann de 1024 amostras por um cosseno de frequência igual a 370 Hz e uma taxa de amostragem de 44,1 kHz.

Com isso, para que seja feito o ajuste de fase, é preciso calcular a fase da cossenoide em questão ao final de cada *hop*[1], e assim associar este valor à fase inicial da cossenoide do *frame* seguinte. A Figura 6.8 ilustra as três primeiras janelas resultantes da síntese de um cosseno de amplitude aproximada de 0,18, frequência igual

[1] *hop* é a parcela da janela que não será sobreposta pela janela seguinte.

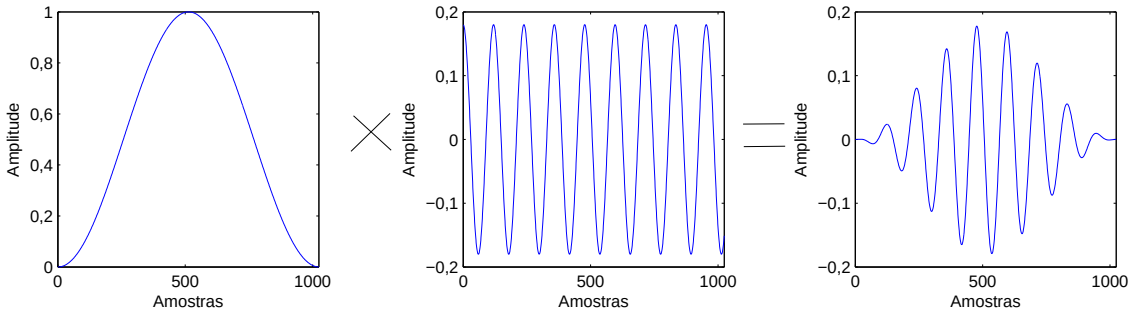


Figura 6.7: A esquerda, temos uma janela de Hann de 1024 amostras multiplicada por um *frame* de um cosseno de frequência igual a 370 Hz, taxa de amostragem de 44,1 kHz e amplitude aproximada de 0,18. A direita, temos o resultado desta multiplicação.

a 70 Hz e taxa de amostragem de 44,1 kHz; cada janela possui 1024 amostras e um *hop* de 50 %, i.e., 512 amostras. Portanto, fica claro que a fase inicial da janela seguinte deve ser igual à fase da janela atual após um *hop*, onde neste caso será na amostra 513. Estas cossenoides foram geradas diretamente na frequência através da criação do espectro de módulo, semelhante à Figura 6.1, e do espectro de fase, conforme será descrito ao longo dessa seção.

Portanto, para determinar a fase inicial de cada cossenoide de um *frame*, é necessário calcular a defasagem de cada cossenoide entre cada dois *frames* consecutivos. Como estamos utilizando janelas de Hann, a sobreposição deverá ser fixa em 50 % (conforme visualizado na Figura 6.8), i.e., o valor do *hop* é igual à metade do tamanho da janela (N), assim

$$hop = \frac{N}{2}, \tag{6.3}$$

assim como foi feito na etapa de análise.

Sendo a taxa de amostragem f_s e a frequência da cossenoide em questão f_i, temos a quantidade de amostras por período dada por

$$\text{amostras por período} = \frac{f_s}{f_i}. \tag{6.4}$$

Podemos, assim, calcular a quantidades de períodos existentes em um *hop* dividindo o número de amostras em um *hop* pelo número de amostras por período.

$$\text{PPH} = \frac{hop\ (\text{amostras/hop})}{f_s/f_i\ (\text{amostras/período})}, \tag{6.5}$$

$$\text{PPH} = \frac{f_i \cdot hop}{f_s}, \tag{6.6}$$

sendo PPH o número de períodos por *hop*.

Se o número de períodos por *hop* for um número inteiro, temos que a fase desta se-

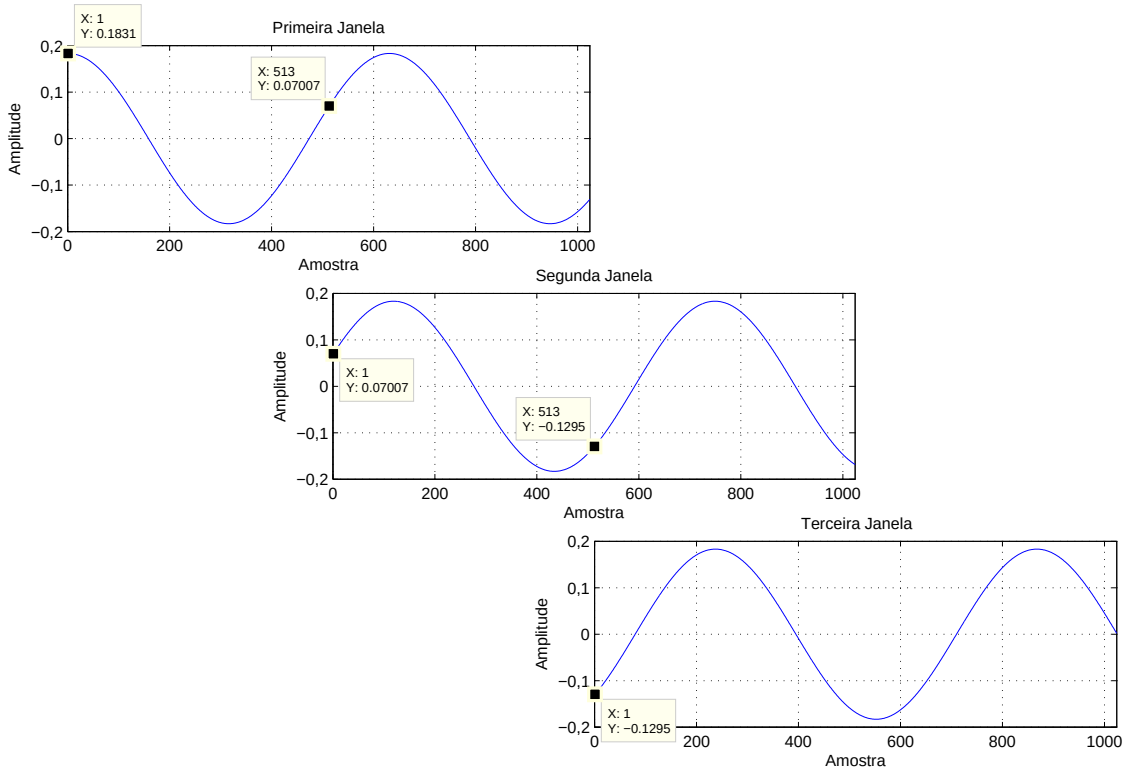


Figura 6.8: Três primeiras janelas sintetizadas de um sinal, onde cada janela possui 1024 amostras e um *hop* de 50 %, i.e., 512 amostras.

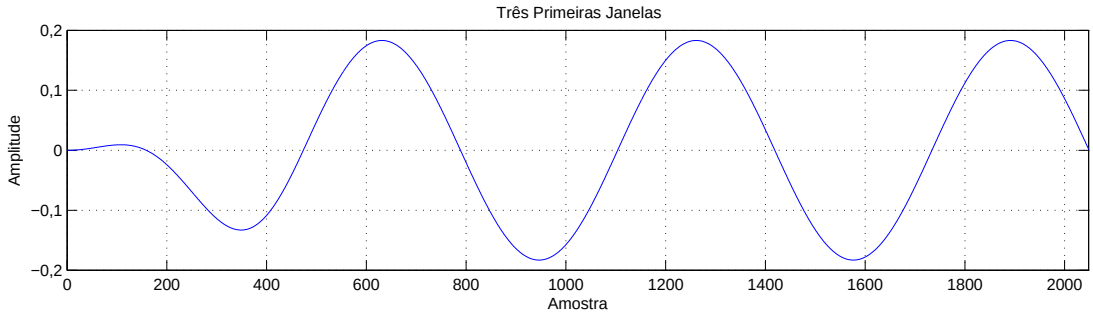


Figura 6.9: Cosseno sintético após a sobreposição e soma das três primeiras janelas de um sinal, ilustradas na Figura 6.8. As primeiras 512 amostras encontram-se minimizadas porque cada janela foi multiplicada por uma janela de Hann.

noide no *frame* seguinte será igual à fase do *frame* atual, pois teremos PPH períodos completos em um *hop*. Porém, se PPH não for inteiro, temos que a fase no *frame* seguinte será igual à fase do *frame* atual somada a uma defasagem.

A defasagem entre *frames* está justamente na parte não inteira de PPH, por isso ela é dada por

$$\text{defasagem} = \text{PPH} - fix(\text{PPH}), \tag{6.7}$$

onde a função *fix* retorna o arredondamento de um número real em direção a zero, tornando-o um número inteiro. Como $0 \leq \text{defasagem} < 1$ e a fase passa a se repetir após 2π, basta multiplicarmos a defasagem por 2π para obtermos a diferença da fase seguinte pela atual.

Com isso, o cálculo da fase de uma determinada senoide será dado por

$$\phi_{n+1,i} = \phi_{n,i} + 2\pi \cdot \text{defasagem}, \tag{6.8}$$

onde n representa o *frame* em questão e i o *bin* da DFT, i.e., a frequência da senoide desejada. No primeiro *frame* a ser sintetizado, cada senoide poderá ter fase aleatória, pois o importante é a continuidade destas senoides de um *frame* para outro. Com isso, arbitrou-se neste trabalho que todas as cossenoides possuirão fase igual a zero no primeiro *frame*.

Caso a equação (6.8) resulte em um valor de $\phi_{n+1,i}$ maior que π, basta subtrair 2π de $\phi_{n+1,i}$ quantas vezes forem necessárias, até que tenhamos $-\pi \leq \phi_{n+1,i} \leq \pi$. Como a fase é cíclica com um período de 2π, pois representa o ângulo de inclinação entre uma linha da origem ao ponto complexo z e o eixo real do plano complexo, conforme descrito na Seção A.1, a não realização desta subtração não resultará em um resultado errôneo, mas somente servirá para efeito de organização dos valores e geração de gráficos.

A Figura 6.9 ilustra a sobreposição e soma das três janelas vistas na Figura 6.8; com isso, podemos ver que a síntese resulta em um cosseno sem nenhuma descontinuidade de fase entre janelas. Nesta figura, podemos observar que a amplitude das primeiras 512 amostras encontra-se reduzida; isso se deu pelo fato de todas as janelas terem sido multiplicadas por uma janela de Hann. Como a primeira metade da primeira janela não foi sobreposta a nenhuma outra, ela permaneceu com a amplitude determinada pela janela de Hann, conforme visto na Figura 6.7.

Uma desvantagem de se construir o sinal na frequência é o fato de o domínio da DFT ser atemporal, i.e., ao transformarmos o sinal construído para o domínio do tempo teremos na janela valores fixos para o módulo A, frequência f e fase ϕ de cada parcial.

Com a sobreposição e soma, o módulo A e a frequência f de cada parcial terão seus valores gradualmente transferidos de uma janela para a outra, eliminando o

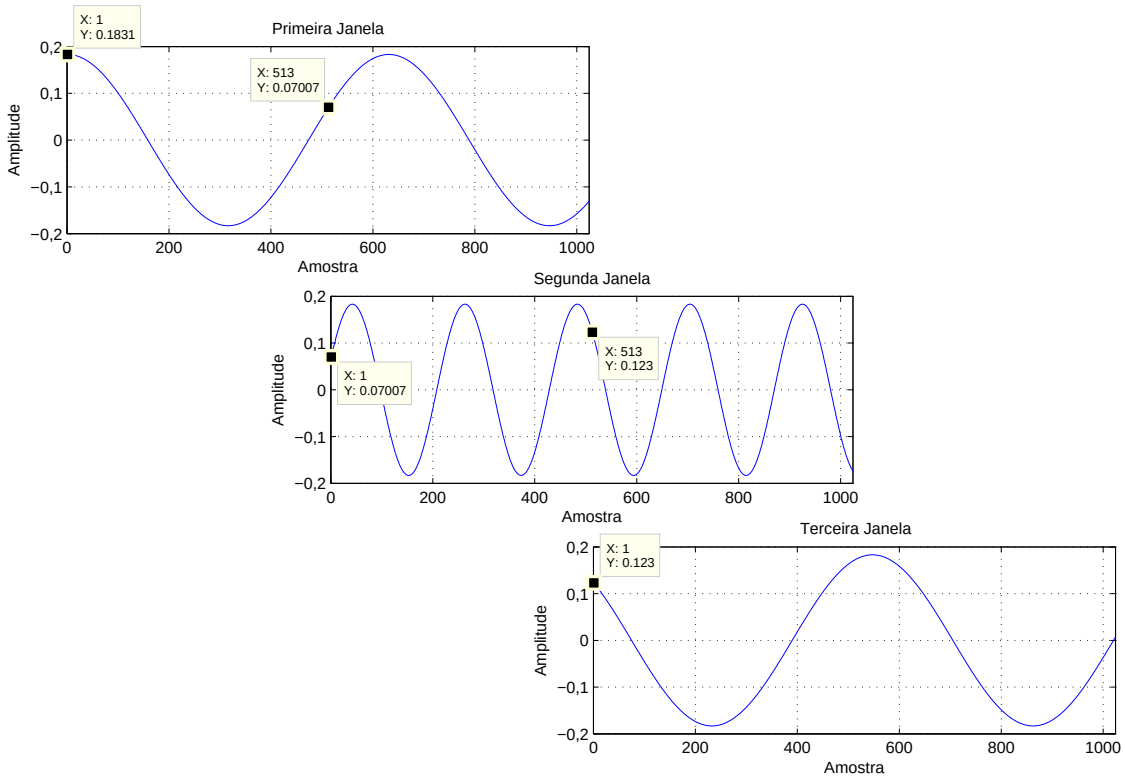


Figura 6.10: Três primeiras janelas sintetizadas de um sinal, onde cada janela possui 1024 amostras e um *hop* de 50 %, i.e., 512 amostras. A primeira e a terceira janelas possuem uma cossenoide de frequência igual a 70 Hz; já a segunda, possui uma frequência igual a 200 Hz.

problema descrito acima [14]. Porém, este fato não ocorrerá com a fase ϕ de cada parcial, pois seu valor será fixo no início de cada janela.

Nas Figuras 6.8 e 6.9, vimos um exemplo em que a frequência não sofria alteração entre janelas; deste modo, o sinal sobreposto não sofreu nenhuma descontinuidade. Para efeito de teste, vejamos, na Figura 6.10, um exemplo em que a primeira e a terceira janelas possuam uma cossenoide de frequência igual a 70 Hz, e a segunda janela, uma cossenoide de frequência 200 Hz, sendo todas de amplitude aproximada de 0,18 e frequência de amostragem de 44,1 kHz. Segundo a Seção 2.5, sabemos que este caso não representa uma situação real, mas a utilizaremos para exemplificar o cálculo de fase.

Neste caso, o simples casamento da fase do início da janela atual com a da janela anterior no final do *hop* não garante a suavidade esperada nas transições. A Figura 6.11, que ilustra a sobreposição e soma das três janelas da Figura 6.10, mostra que para frequências distintas ocorrem artefatos no sinal resultante. Isso só seria evitado interpolando-se a frequência de uma janela para outra.

Na presente aplicação, as situações em que se precisa preservar a continuidade entre janelas correspondem quase sempre a transições sutis de frequência, e portanto o procedimento simples descrito anteriormente deve bastar.

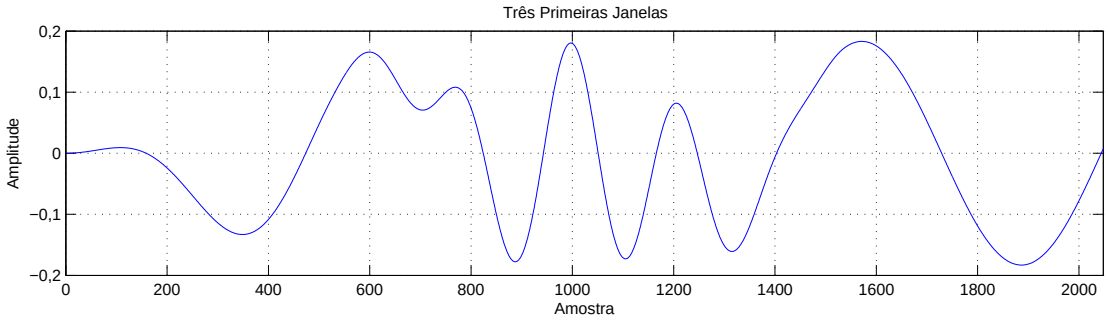


Figura 6.11: Cosseno sintético após a sobreposição e soma das três primeiras janelas de um sinal, ilustradas na Figura 6.10. As primeiras 512 amostras encontram-se minimizadas pois cada janela foi multiplicada por uma janela de Hann.

6.3 Banco de Dados de Síntese

Conforme vimos anteriormente, o timbre de um sinal de voz é criado através da variação da amplitude de cada parcial em seu espectro de módulo. Vimos também que, para a técnica de síntese utilizada nesta dissertação, o parâmetro utilizado para determinar a amplitude das parciais de cada janela, e assim determinar o timbre do sinal, é a envoltória espectral. Portanto, faz-se necessária a criação de um banco de dados contendo as envoltórias espectrais do cantor alvo do qual deseja-se simular o timbre.

Para a criação deste banco de dados, foram selecionadas amostras de áudio de dois cantores alvo: uma voz masculina de um tenor de canto lírico e uma voz feminina de uma soprano de canto popular. As durações destas amostras de áudio variaram de 0,5 a 1 segundo.

Para o cantor lírico, foi selecionada uma amostra de áudio para cada semitom, variando da nota Lá 2 ($pitch$ = 110 Hz) até a nota Sol$^{\#}$ 4 ($pitch$ = 415,3 Hz). Isso foi repetido para cada uma das cinco vogais *a*, *e*, *i*, *o* e *u*. Estas amostras foram retiradas do banco de dados Real World Computing Database (RWC), referência RWC-MDB-I-2001-W12, No.47 [51]. Para a cantora popular, também foi selecionada uma amostra de áudio para cada semitom, porém, variando da nota Fá 3 ($pitch$ = 174,61 Hz) até a nota Fá 5 ($pitch$ = 698,46 Hz). Isso também foi repetido para cada uma das cinco vogais *a*, *e*, *i*, *o* e *u*. Estas amostras foram gravadas utilizando um microfone Shure SM 58. Todas as amostras utilizadas possuem uma taxa de amostragem igual a 44,1 kHz.

Este banco de dados foi gerado pelo *software* Matlab® através da criação de uma *struct* com os campos abaixo:

1. matriz que contém a envoltória espectral;

2. valor numérico que determine a nota musical, e assim o *pitch* relacionado a ela;

3. valor numérico que indique a vogal emitida ($a = 1$, $e = 2$, $i = 3$, $o = 4$ e $u = 5$);

4. vetor contendo a frequência dos três primeiros formantes desta envoltória.

Para adquirir as envoltórias espectrais, foi feito o janelamento destas amostras de áudio com janelas de Hann de 1024 amostras e 50 % de sobreposição. Estimou-se a envoltória espectral de cada janela através do modelo LPC descrito na Seção 3.2.2.

Com o intuito de obter uma diversidade suficiente de envoltórias, foram armazenadas dez envoltórias espectrais na matriz do primeiro campo da *struct* acima, sendo o vetor de cada envoltória salvo em uma diferente linha desta matriz. Para evitar possíveis transitórios presentes no início e final de cada sinal, estas dez envoltórias foram retiradas de janelas localizadas no meio de cada trecho de áudio. Para generalizar a envoltória, convencionou-se que a primeira linha de cada matriz será sempre a média das dez envoltórias aqui extraídas. O último campo da *struct* deverá conter a frequência dos três primeiros formantes da primeira envoltória espectral dentre as dez extraídas.

O armazenamento das envoltórias espectrais nas linhas da matriz do primeiro campo da *struct* pode ser feito de duas maneiras: através do armazenamento de cada elemento do vetor que contém a envoltória, e do armazenamento dos coeficientes do modelo LPC que geram o filtro cuja resposta em frequência seja igual a envoltória.

A vantagem de se armazenar cada elemento do vetor que contém a envoltória é uma execução mais rápida da síntese, pois torna desnecessário calcular a resposta em frequência a um impulso unitário de cada filtro, como será no armazenamento dos coeficientes do modelo LPC. Porém, como o *morphing* visto no Capítulo 5 é feito utilizando-se tais coeficientes, armazená-los torna o método de síntese mais maleável, pois permite que sejam realizados *morphings* entre diferentes envoltórias espectrais, de forma a tornar o resultado da síntese mais natural. Outra vantagem de se armazenar os coeficientes do modelo LPC é a diminuição da memória necessária para o armazenamento do banco de dados. Isso pode ser facilmente compreendido ao notarmos que, para uma janela de 1024 amostras, o armazenamento de todos os elementos do vetor que contém a envoltória espectral necessitaria de memória para armazenar as 1024 amostras para cada envoltória espectral. Já o armazenamento dos coeficientes do modelo LPC de ordem 40, que resulta em 41 coeficientes, necessitaria de memória para armazenar somente 41 amostras para cada envoltória espectral, o que representaria aproximadamente 4 % da memória requerida pelo primeiro método.

Com isso, o banco de dados de síntese foi criado armazenando-se os coeficientes do modelo LPC relacionados às envoltórias espectrais.

Este banco vem a ser um vetor da *struct* descrita nesta seção, do qual cada

elemento está relacionado a uma amostra de áudio, que se relaciona a uma nota musical e a uma vogal.

6.4 Implementação do Algoritmo

Neste ponto desta dissertação, vimos a teoria de como realizar a síntese de um sinal de voz cantada. Porém, existem ainda alguns detalhes de implementação importantes a serem discutidos. Ao longo desta seção, serão apresentados alguns detalhes acerca do algoritmo implementado neste trabalho de mestrado.

6.4.1 *Loop* de Análise

A partir de um sinal de referência de taxa de amostragem igual a 44,1 kHz e que contenha somente uma linha melódica executada por um cantor de referência, será executada a análise deste sinal, dividindo-o em janelas de Hann de 1024 amostras com um *hop* de 50 %. Em seguida, será realizado um *loop* que passará por todas as janelas deste sinal, detectando o *pitch* e as duas vogais mais prováveis de cada janela, utilizando os métodos descritos nos Capítulos 2 e 4, respectivamente.

Ao final deste *loop* de análise, serão realizados os ajustes finais descritos nas Seções 2.5 e 4.6 para tentar evitar detecções errôneas de *pitch* e vogal.

6.4.2 *Loop* de Síntese

Após a definição da melodia a ser sintetizada, através da detecção do *pitch*, e das vogais executadas, inicia-se o *loop* de síntese, que passará pelo mesmo número de janelas obtidas pela etapa de análise. Deste modo, para cada janela do sinal de referência será criada uma janela sintética com a mesma taxa de amostragem de 44,1 kHz e 1024 amostras com um *hop* de 50 %.

Tratamento da Envoltória Espectral

Pelo fato de a síntese ser realizada janela a janela, é possível que ocorram descontinuidades entre elas, o que geraria efeitos pouco naturais no sinal sintetizado final para os ouvidos humanos. Com o fim de suavizar estas possíveis descontinuidades, logo no início do *loop* são definidos um ponto a e um ponto b que demarcam um intervalo de 10 janelas, conforme a Figura 6.12, que ilustra sobre um eixo as janelas de um sinal, no exemplo cinquenta e duas. Vê-se que, para que não haja descontinuidade entre um intervalo e outro, sempre o ponto b coincide com o ponto a do próximo intervalo, de forma que $a \leq \text{intervalo} < b$. O último intervalo será, na maioria das vezes, menor que os demais, pois o número de janelas em geral não

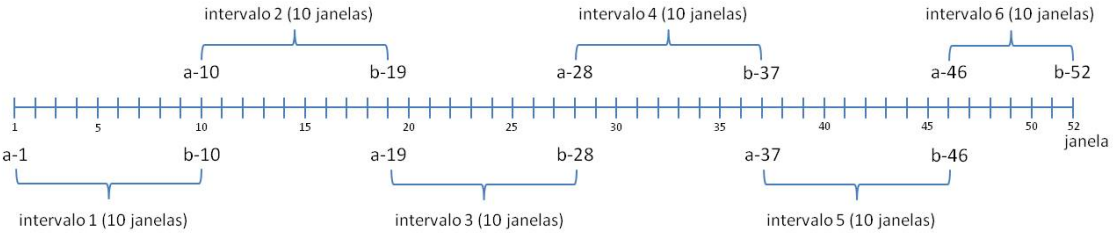


Figura 6.12: Representação das janelas de um sinal com 52 *frames*. Esta figura ilustra um intervalo a e b a cada 10 janelas, coincidindo-se a última janela de um intervalo com a primeira do próximo.

coincidirá com um múltiplo do número de janelas por intervalo. Caso haja trechos surdos no sinal de referência, definidos por janelas em que o *pitch* estimado resulta igual a zero, este algoritmo entenderá o início deste trecho como o ponto b e seu fim como o novo ponto a.

Para que este intervalo suavize o sinal sintetizado, o algoritmo aplica o *morphing* da envoltória espectral do ponto a com o ponto b, conforme descrito no Capítulo 5. O contador i do *loop* de síntese nos indica qual janela está sendo sintetizada; com isso, pode-se determinar o fator de *morphing* α como sendo

$$\alpha = 1 - \frac{i - a}{N_J}, \tag{6.9}$$

onde N_J é o número de janelas em cada intervalo, que no nosso caso é igual a 10; para a taxa de amostragem utilizada, isso resulta em 232,2 ms, um intervalo de tempo razoável para haver uma alteração significativa do sinal de referência. De acordo com esta equação, no início de cada intervalo, onde i é igual a, α sempre será igual a 1, que resultará na envoltória espectral da própria janela $i = a$. À medida que i se aproxima de b, o resultado do *morphing* retornará uma envoltória espectral cada vez mais próxima da envoltória do ponto b.

Conforme descrito na Seção 4.7, o algoritmo de detecção de vogal retorna as duas vogais mais prováveis e o fator de *morphing* que define a proporção da vogal mais provável frente à segunda mais provável. Para se adquirir uma envoltória espectral que resulte em uma vogal intermediária, também se deve realizar o *morphing* das envoltórias de cada uma destas duas vogais mais prováveis.

Porém, este *morphing* de vogais deve ser realizado antes do *morphing* da envoltória espectral do ponto a com o ponto b, pois ele definirá as envoltórias nestes pontos. Para facilitar a compreensão da execução do algoritmo, a Figura 6.13 ilustra a sequência de *morphings* definida para se produzir a envoltória espectral que será utilizada na síntese de cada janela.

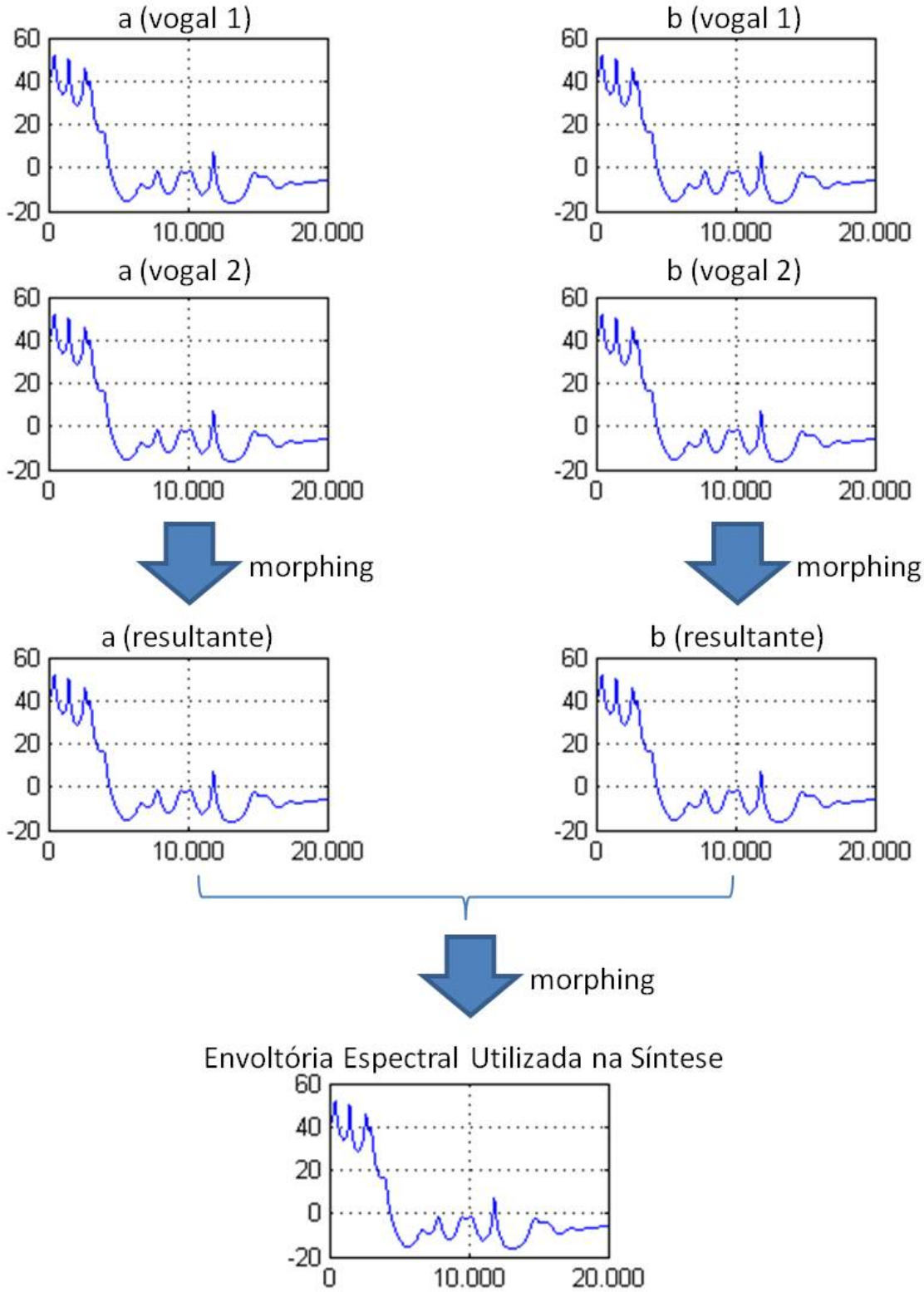


Figura 6.13: Diagrama de blocos que ilustra a sequência de *morphings* definida para se produzir a envoltória espectral que será utilizada na síntese de cada janela.

Realização da Síntese

Uma vez definida a envoltória espectral da janela em questão, realiza-se a síntese, descrita nas Seções 6.2.1 e 6.2.2.

Para que se pudesse determinar a fase de cada parcial, conforme Seção 6.2.2, e assim produzir o espectro de fase, criou-se um vetor para armazenar as fases de cada harmônico de uma janela para a outra, do qual cada índice corresponderá a um harmônico. Desta forma, mesmo que o *pitch* mude de uma janela para a outra, e assim mude a posição frequencial de cada harmônico, a fase no início de uma janela será igual à fase do final do *hop* da janela anterior para todos os harmônicos presentes no sinal.

Como o sinal criado é discreto, deve-se levar em consideração a resolução frequencial utilizada na detecção do *pitch*, conforme descrito na Seção 2.4.2, para que o algoritmo seja capaz de recriar exatamente o *pitch* detectado na etapa de análise. Portanto, o algoritmo de síntese criará uma janela no domínio da frequência com 2^{16} amostras (65.536 amostras), conforme definido na Seção 2.4.2 para sinais com taxa de amostragem igual a 44,1 kHz.

Desta forma, haverá um *bin* da DFT que corresponderá exatamente à frequência encontrada na etapa de análise, além de que todas as parciais harmônicas criadas também corresponderão exatamente a *bins* da DFT criada.

Após gerados os espectros de módulo e de fase, será aplicada a inversa da transformada de Fourier discreta (iDFT) para se obter o sinal sintético no domínio do tempo. Contudo, como este sinal foi criado com uma janela de 2^{16} amostras, ele também possuirá 2^{16} amostras no domínio do tempo; mas como se trata de um sinal periódico, basta considerarmos somente suas primeiras 1024 amostras, que é o tamanho de janela utilizado desta dissertação.

Ajuste de Amplitude

Neste ponto, a síntese da janela em questão encontra-se finalizada. Porém, é necessário verificar a amplitude do *frame* sintetizado com relação ao *frame* de referência.

Para tal, calcula-se a energia da janela do sinal original e a energia da janela do sinal sintetizado. A energia de um sinal qualquer é dada por

$$E = \sum_{n=1}^{N} x_n^2, \tag{6.10}$$

onde x_n é o sinal e N o número de amostras deste sinal.

Define-se, então, a relação entre as energias das janelas

$$r = \sqrt{\frac{E_r}{E_s}}, \tag{6.11}$$

sendo E_r a energia da janela de referência e E_s a energia da janela sintetizada. Desta forma, a energia da janela sintetizada pode ser ajustada multiplicando-se suas amostras pela relação r.

$$s2_n = r \cdot s1_n, \tag{6.12}$$

onde $s1_n$ é a janela sintetizada antes do ajuste e $s2_n$ é a janela sintetizada depois do ajuste de amplitude.

Através deste método, o sinal sintético tenta manter a amplitude do sinal de referência em cada uma de suas janelas.

Uma possível solução mais sofisticada seria analisar e igualar a intensidade percebida destas janelas. Porém, como a janela sintetizada é construída com o mesmo valor de *pitch* da janela de referência, e assim seus espectros possuem aproximadamente o mesmo número de parciais, o ajuste de amplitude através da energia do sinal representa uma boa aproximação para a intensidade percebida.

6.4.3 Sobreposição e Soma de Janelas

Após a finalização do *loop* de síntese, onde todas as janelas foram devidamente construídas, o algoritmo realiza a sobreposição e soma de todas as janelas, conforme mencionado na Seção 6.2.2.

Para que se possa somar as janelas sintetizadas com uma sobreposição de 50 % do tamanho de cada janela, as janelas sintéticas são multiplicadas por janelas de Hann, ilustrada na Figura 6.7, as quais permitem uma perfeita sobreposição sem que a amplitude do sinal resultante seja corrompida, conforme visto nas Figuras 6.8, 6.9, 6.10 e 6.11.

Deste modo, o sinal sintetizado encontra-se devidamente construído, finalizado e pronto para ser reproduzido por um sistema de áudio.

6.5 Avaliação do Método

Nesta seção, serão apresentados alguns experimentos realizados com o fim de avaliar o desempenho do algoritmo desenvolvido neste trabalho de mestrado. Os sinais resultantes serão aferidos, de forma subjetiva, principalmente quanto à naturalidade da síntese de voz cantada. Ao final da avaliação de cada experimento, será apresentada uma tabela contendo uma avaliação feita pelo autor para cada item analisado. Estas tabelas foram geradas aos moldes da avaliação feita por Fonseca e Ferreira em

[30], onde a nota dada pode variar de "− − −" (muito ruim), passando por "0", até "+ + +" (muito bom).

Foram realizados testes com os sinais utilizados na criação do Banco de Dados de Amostras de Vogais para avaliar, principalmente, os métodos de detecção de vogal e a escolha do número de formantes ideal para cada método, conforme discutido no Capítulo 4. Porém, como estes exemplos tratam de sinais de voz cantada que contêm somente vogais em *pitches* constantes, eles podem ser considerados como menos relevantes na avaliação do método como um todo, e assim, não serão descritos nesta seção.

Para a realização e avaliação dos experimentos de forma completa, foram utilizados os seis sinais empregados nos testes do trabalho de Fonseca e Ferreira [30], pois vêm a ser sinais de áudio de músicas populares contendo somente a voz dos cantores, i.e., sem nenhum instrumento musical, executando melodias não triviais, que são:

1. Tom's Diner - Suzanne Vega
2. Amazing Grace - LeAnn Rimes
3. Frozen - Madonna
4. The Rhythm of the Night - Corona
5. Bohemian Rhapsody - Fugees
6. Relax (take it easy) - Mika

Todos os sinais resultantes (sintetizados) podem ser encontrados em `http://gpa.lps.ufrj.br/dissertacaolmonteiro`, tanto os referentes à avaliação dos métodos de detecção de vogal quanto os que serão descritos abaixo. Os sinais originais podem ser encontrados em `http://www.estg.ipleiria.pt/~nuno.fonseca/papers/dafx2010/`, publicado por Fonseca e Ferreira [30].

6.5.1 Ressíntese do Sinal de Referência

Descrição do Experimento

O primeiro experimento realizado vem a ser a ressíntese do sinal de referência. Isso porque, ao invés de se utilizar envoltórias espectrais do banco de dados de síntese, realizou-se a síntese com a envoltória espectral do próprio sinal de referência, estimada a cada janela do sinal. Espera-se obter como saída um sinal praticamente idêntico ao sinal de entrada (referência), inclusive com o seu próprio timbre.

Objetivo do Experimento

O objetivo deste experimento é de analisar:

1. a síntese da melodia detectada pelo método de detecção de *pitch*;
2. a capacidade de o método de síntese utilizado produzir o timbre desejado;
3. a continuidade do sinal sintetizado;
4. o comportamento deste método em sinais que contenham trechos surdos, como a presença de consoantes e a respiração do cantor;
5. o comportamento do método em sinais onde o cantor adiciona uma quantidade significativa de ar à sua voz;
6. o comportamento do método em sinais com os efeitos de *reverb*[2] e *delay*[3].

Avaliação do Resultado

Para todos os seis sinais, a melodia (retirada do método de detecção de *pitch*) e o timbre foram muito bem reconhecidos no sinal sintético. Este resultado nos permite, mais uma vez, validar o método de detecção de *pitch* discutido no Capítulo 2. Também podemos dizer que estes resultados provaram a capacidade de se reproduzir os timbres desejados através da envoltória espectral correta e da síntese aditiva através da iDFT.

Além disso, pode-se perceber uma boa continuidade do sinal sintetizado, não aparentando ter problemas com relação à sobreposição e soma das janelas do sinal sintetizado.

Como foram utilizadas as envoltórias espectrais do sinal de referência, este experimento nos mostrou que as consoantes podem sim ser representadas no sinal resultante de uma síntese aditiva, porém com transitórios bastante suaves. Somente a síntese do fonema da língua portuguesa representado pela letra *s* apresentou resultados ruins, onde aparentou resultar no fonema da língua portuguesa representado pela letra *z*, com exceção do sinal Frozen, onde as consoantes *s* encontraram-se simplesmente eliminadas do sinal sintetizado. O motivo destes comportamentos é o fato de que o fonema da letra *s* assemelha-se muito a um sinal ruidoso, que seria impossível de ser representado somente por picos (harmônicos) no espectro de módulo.

[2] O *reverb* é um efeito utilizado em sinais de áudio que simula a reverberação de uma onda sonora em diferentes ambientes, como em uma sala vazia ou em um estádio.

[3] O *delay* (em português, atraso) é um efeito utilizado em sinais de áudio que simula o eco através da reprodução do mesmo sinal com atrasos no tempo.

Já em trechos surdos do sinal de referência representados pela respiração do cantor, podem-se notar alguns erros referentes à detecção do *pitch*; como consequência, temos no sinal sintetizado a presença de alguns picos de alta frequência indesejados, como nos sinais Tom's Diner e Bohemian Rhapsody.

A maior falha do sistema proposto detectada neste experimento vem a ser o comportamento do mesmo frente aos efeitos de *reverb* e *delay* presentes em um sinal de áudio, assim como em sinais em que o cantor adiciona uma quantidade significativa de ar à sua voz durante o canto. Isso acontece porque a síntese aditiva só representa sinais harmônicos através da criação de picos no espectro de módulo, o que o torna incapaz de representar o ar presente na voz de um cantor, ou a reverberação do sinal de áudio em um ambiente qualquer. Estes erros podem ser observados em todos os sinais, com exceção do Frozen, que por apresentar poucos efeitos na voz (sendo o sinal de referência mais "limpo"), produziu o melhor resultado sob o ponto de vista da percepção.

A presença de *delay* no sinal de referência, por possuir sobreposições do próprio sinal com um atraso temporal, produz outra falha no método: a presença de mais de um *pitch* a ser detectado e sintetizado na mesma janela; o que não faz parte do escopo dos métodos aqui propostos. Estes artefatos podem ser encontrados nos sinais The Rhythm of the Night e Relax (take it easy).

Este experimento inicial nos permite observar que a criação do espectro somente a partir das parciais harmônicas produz sinais com pouco "brilho", pois omite as demais componentes do espectro presentes em um sinal real.

A Tabela 6.1 apresenta uma análise subjetiva dos resultados descritos acima.

Tabela 6.1: Avaliação dos resultados do experimento 1, que pode variar de "− − −" (muito ruim), passando por "0", até "+ + +" (muito bom). Caso um determinado tipo de avaliação não se aplique, utiliza-se "n/a".

Avaliação		Toms	Grace	Frozen	Night	Bohem	Relax
1	Detecção de *pitch*	+ + +	+ + +	+ + +	+ + +	+ + +	+ + +
2	Timbre	+ + +	+ + +	+ + +	+ + +	++	+
3	Continuidade	++	++	+ + +	+ + +	+ + +	++
4	Em trechos surdos	−−	+	+ + +	+	−−	−−
5	Em presença de ar	−−	n/a	n/a	n/a	−	−−
6	Em *reverb* e *delay*	−	−	n/a	+	0	− − −

6.5.2 Sintetizar Somente a Vogal *a*

Descrição do Experimento

O segundo experimento consiste na utilização do banco de dados de síntese para adquirir as envoltórias espectrais a serem sintetizadas. Serão testados tanto o timbre masculino quanto o feminino contidos no banco de dados (ver Seção 6.3). Porém, o método de detecção de vogal ainda não será utilizado, realizando-se a síntese somente com a vogal *a*.

Objetivo do Experimento

O objetivo deste experimento é analisar os timbres contidos no banco de dados de síntese.

Avaliação do Resultado

Após analise subjetiva dos sinais resultantes, podem-se reconhecer os timbres contidos no banco de dados de síntese em cada sinal sintetizado. Como a maioria dos sinais de referência utilizados nestes experimentos possui uma média de *pitch* elevada, por serem cantados por mulheres, a utilização do timbre da voz feminina de canto popular na síntese soou mais natural do que a voz masculina de canto lírico, como era de se esperar.

A Tabela 6.2 apresenta uma análise subjetiva dos resultados descritos acima.

Tabela 6.2: Avaliação dos resultados do experimento 2, que pode variar de "− − −" (muito ruim), passando por "0", até "+ + +" (muito bom).

Avaliação		Toms	Grace	Frozen	Night	Bohem	Relax
1	Timbres do tenor	+ + +	++	+ + +	++	++	0
2	Timbres da soprano	+ + +	+ + +	+ + +	+ + +	+ + +	+

6.5.3 Execução Completa do Algoritmo

Descrição do Experimento

O terceiro experimento consiste na execução do algoritmo com todos os passos descritos na Seção 6.4, i.e., de forma semelhante à do segundo experimento, porém com a utilização dos resultados da detecção de vogal. Para efeito de teste, este experimento fará a detecção de vogal através de dois Banco de Dados de Probabilidades de Vogais, um tendo sido criado utilizando-se a Tabela 3.1, para se aproximar às

vogais do português brasileiro, e outro utilizando-se a Tabela 3.2, para se aproximar às vogais da língua inglesa.

Objetivo do Experimento

O objetivo deste experimento é de:

1. analisar a síntese das vogais detectadas pelo método de detecção de vogal;
2. verificar se há diferença ao se utilizar o Banco de Dados de Probabilidades de Vogais do português brasileiro ou do inglês.

Avaliação do Resultado

Mais uma vez foi possível observar que a presença de *reverb*, *delay* ou de uma quantidade significativa de ar na voz exerce influência no método aqui proposto, de forma a denegrir os seus resultados. Mais uma vez foi possível notar que quanto mais "limpo" (i.e., com menor presença destes efeitos) for o sinal, melhor os resultados, como no caso do sinal Frozen, que por possuir poucos efeitos na voz, as vogais do sinal sintetizado seguiram as do sinal de referência (original).

Após serem avaliados os sinais gerados a partir da detecção da vogal pelo Banco de Dados de Probabilidades de Vogais do português brasileiro e do inglês, verificou-se que não há diferença significativa entre eles. Ao ouvir e comparar estes resultados cautelosamente é possível perceber algumas diferenças nas vogais sintetizadas com os diferentes bancos de dados, porém, nada que justifique uma escolha decisiva entre eles.

A Tabela 6.3 apresenta uma análise subjetiva dos resultados descritos acima.

Tabela 6.3: Avaliação dos resultados do experimento 3, que pode variar de "$---$" (muito ruim), passando por "0", até "$+++$" (muito bom).

Avaliação		Toms	Grace	Frozen	Night	Bohem	Relax
1	Vogal - português	0	$+$	$++$	$+$	$-$	$-$
2	Vogal - inglês	0	$++$	$++$	$+$	$-$	$-$

6.5.4 *Morphing* do Algoritmo Completo com a Ressíntese

Descrição do Experimento

O quarto e último experimento consiste em realizar o *morphing* entre os sinais obtidos através da síntese completa realizada no experimento 6.5.3 e da ressíntese realizada no experimento 6.5.1, de forma que o sinal resultante comece com o sinal

em 6.5.3 e termine com o sinal em 6.5.1. Para realizar este *morphing*, será utilizada a técnica descrita no Capítulo 5.

Objetivo do Experimento

O objetivo deste experimento é de analisar o *morphing* suave entre dois timbres.

Avaliação do Resultado

O resultado deste experimento foi bem favorável, visto que para todos os exemplos o sinal sintetizado inicia com um timbre do banco de dados de síntese e finaliza com o timbre do sinal de referência, variando linearmente de um para o outro, conforme desejado.

6.6 Conclusão

Conforme discutido neste capítulo, o fato de os sinais de voz serem sinais harmônicos, ou quase harmônicos, e assim serem formados por diversos picos em seu espectro de módulo, torna a síntese aditiva uma opção bastante atrativa, visto que consiste na criação do sinal através de um somatório de senoides, cujo espectro de módulo vem a ser um pico em sua frequência fundamental. Foi descrita, então, a técnica de síntese aditiva cuja premissa é a produção do sinal no domínio da frequência, seguido da inversa da DFT para se obter o sinal no domínio do tempo; tal técnica permite uma redução do custo computacional perante as técnicas tradicionais de síntese aditiva, as quais utilizam bancos de osciladores.

Também neste capítulo, foi descrita a criação do banco de dados de síntese, o qual é utilizado para o armazenamento das envoltórias espectrais, que trazem a informação do timbre a ser recriado pela síntese. Este banco de dados foi criado para dois sinais alvo diferentes: um cantor masculino de canto lírico e uma cantora feminina de canto popular, através de várias amostras de áudio de tais cantores.

Após a criação do banco de dados de síntese, foi apresentado o conceito e execução de todo o algoritmo criado neste trabalho de mestrado, detalhando todas as nuances necessárias para a sua implementação. Em seguida, foram descritos quatro experimentos, com a finalidade de validar a síntese executada neste trabalho.

A partir destes experimentos, podemos perceber que, quanto mais "limpo" for o sinal de referência, i.e., sem efeitos de *reverb*, *delay* e sem a adição de ar à voz do cantor durante o canto, melhor é o resultado do sinal sintetizado. Nessa situação, o *pitch* e a vogal são detectados de forma mais precisa e sintetizados de forma mais clara.

Com base nestes experimentos, também foi possível perceber que o uso de envoltórias espectrais na síntese aditiva através da iDFT permitiu a reprodução dos timbres dos sinais alvo tornando o timbre sintético reconhecível. Porém, uma desvantagem deste método foi a criação de sinais sem muito "brilho", uma vez que só foram criados os harmônicos de cada sinal; desta forma, todas as demais componentes presentes no espectro de um sinal de voz cantada foram omitidas. Uma possível maneira de se contornar esta questão seria a modelagem do sinal residual e ruído presentes em sinais de voz, seguido da adição de suas componentes no espectro sintetizado.

Capítulo 7

Conclusões

A síntese de voz ainda é um campo de processamento de sinais de áudio com vasta perspectiva de expansão. A criação artificial da fala, ou canto, tal qual como produzidos por um ser humano, vem a ser uma tarefa desafiadora, mesmo com as avançadas tecnologias que possuímos nos tempos atuais.

Esta dissertação de mestrado buscou contribuir, mesmo que de forma modesta, com o desenvolvimento de tal área de estudo, implementando técnicas descritas na literatura, propondo-lhes melhorias e analisando os seus resultados.

O algoritmo apresentado nesta dissertação visa, a partir de um sinal de referência, a sintetizar um sinal totalmente novo que contenha toda a melodia e os fonemas vocálicos presentes neste sinal de referência, mas com um timbre de um determinado cantor alvo; assim, através do sinal sintetizado, teremos a impressão de que a música presente no sinal de referência foi executada pelo cantor alvo.

Para se obter os parâmetros responsáveis por identificar a música executada no sinal de referência, algumas técnicas de análise de sinais de voz foram descritas e desenvolvidas ao longo deste texto, para assim determinar o *pitch* e o fonema a serem sintetizados.

Após analisar algumas técnicas de detecção de *pitch*, optou-se pela técnica Logaritmo do Produto do Espectro Harmônico (logHPS). Detalhes importantes sobre sua implementação foram apresentados pelo autor nesta dissertação, como a limitação do número de iterações do método e a qualidade da resolução frequencial. O autor também apresentou uma forma de se reduzir erros na estimação do *pitch* buscando por possíveis valores de *pitch* (candidatos ao *pitch*) a cada iteração do método. Comparando esta técnica de detecção de *pitch* com as técnicas YIN e Cepstral, verificou-se que todas as técnicas estudadas apresentam bons resultados na maioria dos sinais em trechos sonoros; porém, em trechos surdos, somente o logHPS apresentou resultados consistentes, dispensando a utilização de técnicas de detecção de trechos sonoros e surdos. Ainda assim, erros eventuais foram identificados nos experimentos realizados para avaliação do método; tais erros variam de acordo com a complexi-

dade do sinal de referência, não em termos de harmonia, mas devido à presença excessiva da respiração do cantor ou de efeitos de *delay*. Este último mostrou-se prejudicial pelo fato de ocasionar muitas vezes a detecção de dois ou mais valores de *pitch* simultâneos; essa condição não foi originalmente prevista neste trabalho, visto que seriam analisados sinais de voz isolados dos demais instrumentos presentes em uma música.

Em sinais de voz, o timbre de um determinado cantor e a melodia executada por ele (informação de *pitch*) encontram-se, basicamente, representadas através das vogais; as consoantes contribuem muito pouco com tais parâmetros. Por isso, este trabalho restringiu-se à análise e síntese de vogais.

Com o intuito de detectar as vogais presentes em um sinal de voz cantada, foram propostos, pelo autor desta dissertação, dois métodos: por probabilidade e por proximidade. Amostras de sinais sintetizados pelos dois métodos, assim como os resultados dos experimentos, podem ser encontradas em `http://gpa.lps.ufrj.br/dissertacaolmonteiro`. Através destas amostras, podemos ver que ambos apresentaram resultados bons em valores de *pitch* medianos; porém, em *pitches* muito baixos ou altos, os métodos apresentaram um número de falhas razoável. Não é difícil compreender as falhas obtidas em tais faixas de *pitch*, uma vez que os próprios cantores possuem dificuldade para executar com perfeição as diferentes vogais em tais alturas, onde não há, por exemplo, uma diferença significativa entre a vogal *a* e a vogal *o*. Avaliando os resultados de forma subjetiva, o método de detecção de vogal por probabilidade apresentou melhores resultados frente ao método por proximidade.

Através dos experimentos realizados para validação do método, observou-se que o algoritmo de detecção de vogal também apresentou melhores resultados em sinais "limpos", i.e., sem a presença de efeitos na voz, como o *reverb*, o *delay* e a adição de ar à voz do cantor durante o canto.

Uma possível extensão a este trabalho seria a construção de um Banco de Dados de Probabilidades de Vogais separado por faixas de *pitch*, visto que a posição frequencial dos três primeiros formantes variam, mesmo que não muito, de acordo com a altura da nota produzida; estas pequenas variações já vêm a ser suficientes para se obter valores equivocados na detecção da vogal emitida. Para a criação de um Banco de Dados de Probabilidades de Vogais mais robusto, também seria importante a obtenção de um Banco de Dados de Amostras de Vogais mais amplo e heterogêneo, possuindo um número maior de amostras de cantores diferentes e em *pitches* diferentes. Também seria interessante criar este banco de dados para todos os fonemas de vogais presentes no português brasileiro. Outra extensão seria a inclusão dos fonemas referentes às consoantes, com técnicas para análise e síntese dos mesmos.

Também foi proposto pelo autor desta dissertação um método para se sintetizar

as vogais do sinal de referência através da substituição dos primeiros polos da envoltória espectral. Apesar de este método permitir uma perfeita representação da vogal contida no sinal de referência, viu-se que os primeiros polos também carregam consigo detalhes do timbre do sinal de referência. Como um dos objetivos desta dissertação é de recriar o timbre de um determinado cantor alvo, este método não foi utilizado nesta dissertação.

A técnica de *morphing* utilizada neste trabalho, a qual realiza a interpolação dos coeficientes LPC, obteve grande êxito em sua implementação, visto que os sinais obtidos através desta técnica possuem timbres perceptivamente entre os sinais fonte e alvo. Como a técnica de síntese utilizada faz uso de envoltórias espectrais para modelar o timbre desejado, e as envoltórias espectrais foram estimadas através do modelo LPC, esta técnica de *morphing* mostrou-se ideal para o uso nessa dissertação. Visto que os timbres desejados foram reconhecidos no sinal sintetizado, o modelo LPC mostrou-se robusto para a aplicação requerida por esta dissertação, onde foi utilizado na estimação e *morphing* de envoltórias espectrais. Com isso, não foi necessário o estudo de outras técnicas, como por exemplo do modelo LSF (*Line Spectral Frequencies*).

Com os experimentos realizados, foi possível perceber que a técnica de síntese aditiva utilizada garante uma perfeita execução da melodia detectada pelo algoritmo de detecção de *pitch*, assim como os timbres dos cantores alvo, cujas envoltórias espectrais encontram-se armazenadas no banco de dados de síntese, mostraram-se reconhecíveis na síntese. Como as vogais não deixam de ser variações de um timbre, elas também foram sintetizadas com êxito através desta técnica de síntese, lembrando que as falhas nas vogais se deveram à etapa de análise, e não à de síntese.

Porém, um sinal produzido somente por seus harmônicos se mostrou ser um sinal com pouco "brilho", pois todas as demais componentes presentes em um espectro de voz são omitidas na síntese aditiva. Como extensão a este trabalho, seria proveitoso estudar a modelagem e síntese do sinal residual, assim como de ruídos presentes em sinais reais de voz cantada.

Referências Bibliográficas

[1] RAUBER, A. S. "An acoustic description of Brazilian Portuguese oral vowels", *Diacrítica, Ciências da Linguagem*, v. 22, n. 1, pp. 229–238, 2008. Disponível em: <http://www.nupffale.ufsc.br/rauber/Rauber_diacritica_22_1.pdf>.

[2] PETERSON, G. E., BARNEY, H. L. "Control Methods Used in a Study of the Vowels", *Journal of the Acoustical Society of America*, v. 24, n. 2, pp. 175–184, mar. 1952. Disponível em: <http://asadl.org/jasa/resource/1/jasman/v23/i1/p148_s3?bypassSSO=1>.

[3] ESQUEF, P. A. A., BISCAINHO, L. W. P. "DSP Techniques for Sound Enhancement of Old Recordings". In: Perez-Meana, H. M. (Ed.), *Advances in Audio and Speech Signal Processing: Technologies and Applications*, 1 ed., Igi Global, cap. 4, pp. 93–130, National Polytechnic Institute of Mexico, 2007. ISBN: 1599041324. doi: 10.4018/978-1-59904-132-2.ch004.

[4] BOSI, M., GOLDBERG, R. E. *Introduction to Digital Audio Coding and Standards*. 1 ed. Boston / Dordrecht / London, Kluwer Academic Publishers, 2003.

[5] "Dennis Klatt's History of Speech Synthesis". 1987. http://www.cs.indiana.edu/rhythmsp/ASA/Contents.html.

[6] COOK, P. R. "Voice physics and neurology". In: Cook, P. R. (Ed.), *Music, cognition, and computerized sound: an introduction to psychoacoustics*, MIT Press, pp. 105 – 116, Cambridge, MA, USA, 1999. ISBN: 0-262-03256-2.

[7] FLANAGAN, J. L., GOLDEN, R. M. "Phase Vocoder", *Bell System Technical Journal*, pp. 1493–1509, nov. 1966. Disponível em: <"'http://www.ee.columbia.edu/~dpwe/e6820/papers/FlanG66.pdf"'>.

[8] SCHAFER, R. W., RABINER, L. R. "Design and Simulation of a Speech Analysis- Synthesis System Based on Short-Time Fourier Analysis", *Audio and Electroacoustics, IEEE Transactions on*, v. 21, n. 3, pp. 165–174,

jun. 1973. ISSN: 0018-9278. Disponível em: <http://ieeexplore.ieee.org/iel6/8337/26098/01162474.pdf?arnumber=1162474>.

[9] PORTNOFF, M. R. "Implementation of the Digital Phase Vocoder Using the Fast Fourier Transform", *Acoustics, Speech and Signal Processing, IEEE Transactions on*, v. 24, n. 3, pp. 243 – 248, jun. 1976. ISSN: 0096-3518. Disponível em: <http://ieeexplore.ieee.org/iel6/29/26118/01162810.pdf?arnumber=1162810>.

[10] ALLEN, J. B. "Short Term Spectral Analysis, Synthesis, and Modification by Discrete Fourier Transform", *Acoustics, Speech and Signal Processing, IEEE Transactions on*, v. 25, n. 3, pp. 235 – 238, jun. 1977. ISSN: 0096-3518. Disponível em: <http://ci.nii.ac.jp/naid/10021135333/en/>.

[11] BEAUCHAMP, J. W. "Analysis and Synthesis of Musical Instrument Sounds". In: Beauchamp, J. W. (Ed.), *Analysis, Synthesis, and Perception of Musical Sounds: The Sound of Music*, Modern Acoustics and Signal Processing, Springer New York, cap. 1, pp. 1–89, University of Illinois at Urbana, USA, ago. 2007. ISBN: 978-0-387-32576-7. doi: 10.1007/978-0-387-32576-7_1. Disponível em: <http://www.springerlink.com/content/l32t21366807775g/fulltext.pdf>.

[12] SMITH, J. O., SERRA, X. "PARSHL: An Analysis/Synthesis Program for Non-Harmonic Sounds Based on a Sinusoidal Representation". In: *International Computer Music Conference*, University of Illinois at Champaign/Urbana, USA, 1987. Disponível em: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.142.9147&rep=rep1&type=pdf>.

[13] MCAULAY, R. J., QUATIERI, T. F. "Speech Analysis/Synthesis Based on a Sinusoidal Representation", *Acoustics, Speech and Signal Processing, IEEE Transactions on*, v. 34, n. 4, pp. 744 – 754, ago. 1986. ISSN: 0096-3518. Disponível em: <http://ieeexplore.ieee.org/iel6/29/26199/01164910.pdf?arnumber=1164910>.

[14] RODET, X., DEPALLE, P. "Spectral Envelopes and Inverse FFT Synthesis". In: *AES Convention:93*, San Francisco, California, USA, out. 1992. Disponível em: <http://www.aes.org/e-lib/online/download.cfm/6740.pdf?ID=6740>.

[15] MASSIE, D. C. "Wavetable Sampling Synthesis". In: Kahrs, M., Brandenburg, K. (Eds.), *Applications of Digital Signal Processing to Au-*

dio and Acoustics, v. 437, *The International Series in Engineering and Computer Science*, Springer US, cap. 8, pp. 311–341, New York, Boston, Dordrecht, London, Moscow, 2002. ISBN: 978-0-306-47042-4. doi: 10.1007/0-306-47042-X_8.

[16] HAKEN, L., FITZ, K., CHRISTENSEN, P. "Beyond Traditional Sampling Synthesis: Real-Time Timbre Morphing Using Additive Synthesis". In: Beauchamp, J. W. (Ed.), *Analysis, Synthesis, and Perception of Musical Sounds: The Sound of Music*, Modern Acoustics and Signal Processing, Springer New York, cap. 3, pp. 122–144, University of Illinois at Urbana, USA, 2007. ISBN: 978-0-387-32576-7. doi: 10.1007/978-0-387-32576-7_3. Disponível em: <http://www.springerlink.com/content/h689311603512773/fulltext.pdf>.

[17] TELLMAN, E., HAKEN, L., HOLLOWAY, B. "Timbre Morphing of Sounds with Unequal Numbers of Features", *Journal of the Audio Engineering Society*, v. 43, n. 9, pp. 678–689, set. 1995. ISSN: 0004-7554. Disponível em: <http://www.aes.org/e-lib/online/download.cfm/7931.pdf?ID=7931>.

[18] HAKEN, L., FITZ, K., TELLMAN, E., et al. "A Continuous Music Keyboard Controlling Polyphonic Morphing Using Bandwidth-Enhanced Oscillators". In: *International Computer Music Conference*, v. 1997, Thessaloniki, Greece, set. 1997. Disponível em: <http://quod.lib.umich.edu/cgi/p/pod/dod-idx?c=icmc;idno=bbp2372.1997.100>.

[19] CAETANO, M. *Morphing Isolated Quasi-Harmonic Acoustic Musical Instrument Sounds Guided by Perceptually Motivated Features*. Tese de D.Sc., Institut de Recherche et Coordination Acoustique/Musique, Paris, França, nov. 2011.

[20] CAETANO, M., RODET, X. "Evolutionary Spectral Envelope Morphing by Spectral Shape Descriptors". In: *International Computer Music Conference*, Montreal, QC, Canada, ago. 2009. Disponível em: <"'http://recherche.ircam.fr/anasyn/caetano/publications.html"'>.

[21] CAETANO, M., RODET, X. "Automatic Timbral Morphing of Musical Instrument Sounds by High-Level Descriptors". In: *International Computer Music Conference*, New York / Stony Brook, USA, jun. 2010. Disponível em: <http://articles.ircam.fr/textes/Caetano10a/>.

[22] CAETANO, M., RODET, X. "Independent Manipulation of High-Level Spectral Envelope Shape Features For Sound Morphing by Means of Evoluti-

onary Computation". In: *DAFx-10*, set. 2010. Disponível em: <"'http://recherche.ircam.fr/anasyn/caetano/publications.html"'>.

[23] CAETANO, M., RODET, X. "Sound Morphing by Feature Interpolation". In: *ICASSP*, pp. 161–164, 2011. Disponível em: <"'http://recherche.ircam.fr/anasyn/caetano/publications.html"'>.

[24] BONADA, J., LOSCOS, A., CANO, P., et al. "Spectral Approach to the Modeling of the Singing Voice". In: *111th AES Convention*, New York, set. 2001. Disponível em: <http://www.mtg.upf.edu/files/publications/aes2001-bonada.pdf% textgreater .

[25] BONADA, J., LOSCOS, A. "Sample-based singing voice synthesizer by spectral concatenation". In: *Proceedings of the Stockholm Music Acoustics Conference (SMAC 03)*, Stockholm, Sweden, ago. 2003. Disponível em: <http://www.mtg.upf.edu/files/publications/SMAC2003-aloscos.pdf>.

[26] KENMOCHI, H., OHSHITA, H. "VOCALOID - Commercial singing synthesizer based on sample concatenation". In: *Special Session - Synthesis of Singing Challenge*, Belgium, ago. 2007. Disponível em: <http://www.let.uu.nl/~Gerrit.Bloothooft/personal/SSC/Yamaha/VOCALOID_Interspeech.pdf>.

[27] "Synthesis of Singing Challenge". 2007. Disponível em: <http://www.interspeech2007.org/Technical/synthesis_of_singing_challenge.php.Acessadoemmar\unhbox\voidb@x\setbox\z@\hbox{c}\accent24code2010.>.

[28] MAYOR, O., BONADA, J., JANER, J. "KaleiVoiceCope: Voice Transformation From Interactive Installations To Video-Games". In: *AES 35th International Conference: Audio for Games*, London, UK, fev. 2009. Disponível em: <http://mtg.upf.edu/files/publications/Kaleivoicecope_aes35.pdf>.

[29] PAIVA, R. C. D. D. *Transformações em Sinais de Voz: Morphing e Modificação de Pitch*. Dissertação de M.Sc., Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ - Brasil, fev. 2008.

[30] FONSECA, N., FERREIRA, A. "Singing Voice Resynthesis Using Vocal Sound Libraries". In: *International Conference on Digital Audio Effects (DAFx-10)*, Graz, Austria, set. 2010. Disponível em: <http://dafx10.iem.at/proceedings/papers/FonsecaFerreira_DAFx10_P22.pdf>.

[31] DELLER, J. J. R., HANSEN, J. H. L., PROAKIS, J. G. *Discrete-Time Processing of Speech Signals*. 2 ed. Nova York, Wiley-IEEE Press, 1999.

[32] DUBNOWSKI, J., SCHAFER, R., RABINER, L. "Real-Time Digital Hardware Pitch Detector", *IEEE Transactions on Acoustics Speech and Signal Processing*, v. 24, n. 1, pp. 2–8, 1976. Disponível em: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1162765>.

[33] CHEVEIGNE, A. D., KAWAHARA, H. "YIN, a fundamental frequency estimator for speech and music", *Journal of the Acoustical Society of America*, v. 111, out. 2002. doi: 10.1121/1.1458024. Disponível em: <http://asadl.org/jasa/resource/1/jasman/v111/i4/p1917_s1?isAuthorized=no>.

[34] BOGERT, B., HEALY, M., TUKEY, J. "The quefrency alanysis of time series for echoes: Cepstrum, Pseudo-Autocovariance, Cross-Cepstrum and Saphe Cracking". In: *Proc. Symp. on Time Series Analysis*, pp. 209–243, 1963.

[35] YNOGUTI, C. A., VIOLARO, F. "On The Use Of Principal Component Analysis Over Mel Cepstral Coefficients", *Revista Científica Periódica - Telecomunicações*, v. 5, n. 2, pp. 13–17, dez. 2002. ISSN: 1516-2338.

[36] OPPENHEIM, A. V., SCHAFER, R. W. "From Frequency to Quefrency: A History of the Cepstrum", *IEEE Signal Processing Magazine*, v. 21, n. 5, pp. 95–100, 2004. Disponível em: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1328092>.

[37] RABINER, L. R., SCHAFER, R. W. "Algorithms for Estimating Speech Parameters". In: *Theory and Applications of Digital Speech Processing*, 1 ed., cap. 10, pp. 578–662, Upper Saddle River, New Jersey, Pearson, 2010. ISBN: 0-13-603428-4. Disponível em: <http://books.google.com.br/books?id=W9leOgAACAAJ>.

[38] SCHROEDER, M. R. "Period Histogram and Product Spectrum: New Methods for Fundamental Frequency Measurement", *Journal of The Acoustical Society of America*, v. 43, n. 4, pp. 829–834, abr. 1968. doi: 10.1121/1.1910902.

[39] OPPENHEIM, A. V., WILLSKY, A. S., NAWAB, S. H. "The Discrete-Time Fourier Transform". In: *Signals & Systems*, 2 ed., cap. 5, pp. 358–422, Upper Saddle River, NJ, USA, Prentice-Hall, 1997.

ISBN: 81-203-1246-5. Disponível em: <http://www.amazon.com/Signals-Systems-2nd-Alan-Oppenheim/dp/0138147574>.

[40] OPPENHEIM, A. V., WILLSKY, A. S., NAWAB, S. H. "Sampling". In: *Signals & Systems*, 2 ed., cap. 7, pp. 514–581, Upper Saddle River, NJ, USA, Prentice-Hall, 1997. ISBN: 81-203-1246-5. Disponível em: <http://www.amazon.com/Signals-Systems-2nd-Alan-Oppenheim/dp/0138147574>.

[41] CAO, Y., SRIDHARAN, S., MOODY, M. "Voiced/Unvoiced/Silence Classification of Noisy Speech in Real Time Audio Signal Processing". In: *Audio Engineering Society Convention 5r*, mar. 1995. Disponível em: <http://www.aes.org/e-lib/browse.cfm?elib=7721>.

[42] RODET, X., SCHWARZ, D. "Spectral Envelopes and Additive + Residual Analysis/Synthesis". In: Beauchamp, J. W. (Ed.), *Analysis, Synthesis, and Perception of Musical Sounds: The Sound of Music*, Modern Acoustics and Signal Processing, Springer New York, cap. 5, pp. 175–227, University of Illinois at Urbana, USA, 2007. ISBN: 978-0-387-32576-7. doi: 10.1007/978-0-387-32576-7_5. Disponível em: <http://www.springerlink.com/content/r4505428m17j4484/fulltext.pdf>.

[43] ATAL, B. S., HANAUER, S. L. "Speech Analysis and Synthesis by Linear Prediction of the Speech Wave", *Journal of the Acoustical Society of America*, v. 50, n. 2B, pp. 637–655, abr. 1971. doi: 10.1121/1.1912679. Disponível em: <http://courses.cs.tamu.edu/rgutier/cpsc689_s07/atalHanauer1971lpc.pdf>.

[44] FLANAGAN, J. *Speech Analysis Synthesis and Perception*. Kommunikation und Kybernetik in Einzeldarstellungen. Warren Township, New Jersey, Springer-Verlag, 1972. ISBN: 9780387055619. Disponível em: <http://books.google.com.br/books?id=i-kgAQAAIAAJ>.

[45] LOSCOS, A. *Spectral Processing of the Singing Voice*. Tese de D.Sc., Universitat Pompeu Fabra, Barcelona, Espanha, 2007.

[46] RABINER, L. R., SCHAFER, R. W. "Fundamentals of Human Speech Production". In: *Theory and Applications of Digital Speech Processing*, 1 ed., cap. 3, pp. 67–123, Upper Saddle River, New Jersey, Pearson, 2010. ISBN: 0-13-603428-4. Disponível em: <http://books.google.com.br/books?id=W9leOgAACAAJ>.

[47] RABINER, L. R., SCHAFER, R. W. "Linear Predictive Analysis of Speech Signals". In: *Theory and Applications of Digital Speech Processing*, 1 ed., cap. 9, pp. 473–577, Upper Saddle River, New Jersey, Pearson, 2010. ISBN: 0-13-603428-4. Disponível em: <http://books.google.com.br/books?id=W9leOgAACAAJ>.

[48] DODGE, C., JERSE, T. A. "Synthesis Using Distortion Techniques". In: *Computer Music: Synthesis, Composition and Performance*, 2nd ed., cap. 5, pp. 115–168, Dartmouth College and Boeing Company, Macmillan Library Reference, 1997. ISBN: 0-02-864682-7. Disponível em: <http://dl.acm.org/citation.cfm?id=549805>.

[49] DODGE, C., JERSE, T. A. "Granular Synthesis". In: *Computer Music: Synthesis, Composition and Performance*, 2nd ed., cap. 8, pp. 262–276, Dartmouth College and Boeing Company, Macmillan Library Reference, 1997. ISBN: 0-02-864682-7. Disponível em: <http://dl.acm.org/citation.cfm?id=549805>.

[50] OPPENHEIM, A. V., WILLSKY, A. S., NAWAB, S. H. "The Continuous-Time Fourier Transform". In: *Signals & Systems*, 2 ed., cap. 4, pp. 284–357, Upper Saddle River, NJ, USA, Prentice-Hall, 1997. ISBN: 81-203-1246-5. Disponível em: <http://www.amazon.com/Signals-Systems-2nd-Alan-Oppenheim/dp/0138147574>.

[51] "Real World Computing Database (RWC)". 2001. Disponível em: <http://staff.aist.go.jp/m.goto/RWC-MDB/>.

[52] OPPENHEIM, A. V., WILLSKY, A. S., NAWAB, S. H. "Signals & Systems". In: *Signals & Systems*, 2 ed., cap. 1, pp. 1–73, Upper Saddle River, NJ, USA, Prentice-Hall, 1997. ISBN: 81-203-1246-5. Disponível em: <http://www.amazon.com/Signals-Systems-2nd-Alan-Oppenheim/dp/0138147574>.

[53] OPPENHEIM, A. V., WILLSKY, A. S., NAWAB, S. H. "Fourier Series Representation of Periodic Signals". In: *Signals & Systems*, 2 ed., cap. 3, pp. 177–283, Upper Saddle River, NJ, USA, Prentice-Hall, 1997. ISBN: 81-203-1246-5. Disponível em: <http://www.amazon.com/Signals-Systems-2nd-Alan-Oppenheim/dp/0138147574>.

Apêndice A

Interpretação dos Espectros de Frequência

Sabemos que senos e cossenos vêm a ser os sinais periódicos mais triviais representados pela matemática. Sabemos também que a transformada de Fourier nos permite representar sinais diversos através de uma soma infinitesimal de senoides.

Toda senoide necessita de três parâmetros para serem devidamente representadas, que são o módulo, a frequência e a fase. Uma forma comum de representar um sinal senoidal se dá através de exponenciais complexas, que permitem-nos lidar mais facilmente com as informações de módulo e fase deste sinal.

Na maioria das vezes, um sinal visto no domínio da frequência pela transformada de Fourier vem a ser um sinal complexo, o qual pode ser visualizado por seu espectro de módulo e seu espectro de fase.

Visto que a transformada de Fourier é dada na forma exponencial, recordemos a dualidade entre sinais senoidais e exponenciais complexas.

A.1 Sinais Senoidais e Exponenciais

Um número complexo é composto por parte real e parte imaginária, e pode ser representado nas coordenadas retangulares conforme

$$z = a + jb, \tag{A.1}$$

onde j vem a ser o número imaginário $\sqrt{-1}$, z o número complexo e a e b números reais que se referem às partes real e imaginária de z, respectivamente.

Um número complexo pode ser representado com o auxílio da exponencial complexa $e^{j\theta}$, onde e é o número neperiano, que vem a ser aproximadamente 2,718. Esta exponencial possui módulo unitário, portanto, traça o círculo unitário no plano complexo, sendo θ o ângulo de inclinação entre uma linha da origem ao ponto complexo

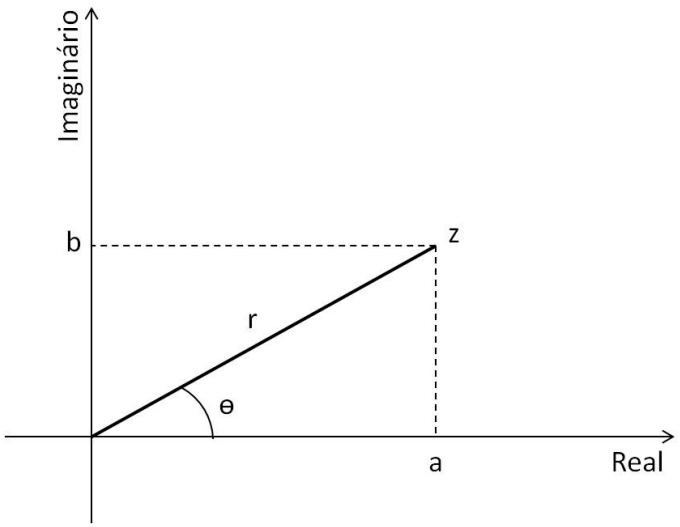


Figura A.1: Plano complexo, onde o eixo horizontal representa a parte real e o eixo vertical a parte imaginária de um sinal complexo. Nesta figura, podemos ver as coordenadas retangulares e polares do número complexo z.

z e o eixo real; desta forma, um número complexo qualquer pode ser representado nas coordenadas polares por

$$z = re^{j\theta}, \tag{A.2}$$

onde r refere-se ao módulo e θ à fase do número complexo z. O plano complexo com suas variáveis pode ser visto na Figura A.1.

Esta representação do número complexo na coordenada polar através da exponencial complexa do número neperiano e foi provada pelo matemático Euler através da expansão da série de Taylor da função exponencial e^y, sendo y um número imaginário; desta forma, temos a equação de Euler como sendo

$$e^{j\theta} = \cos\theta + j\,\mathrm{sen}\,\theta, \tag{A.3}$$

a qual permite representar senos e cossenos através de exponenciais complexas [52].

Através da Figura A.1 pode-se, por trigonometria simples, determinar a transformação de coordenadas retangulares para coordenadas polares

$$r = \sqrt{a^2 + b^2}, \tag{A.4}$$

$$\theta = \mathrm{arctg}\,\frac{b}{a}, \tag{A.5}$$

e a transformação inversa, de coordenadas polares para coordenadas retangulares

$$a = r\cos\theta, \tag{A.6}$$

$$b = r\,\mathrm{sen}\,\theta. \tag{A.7}$$

A partir das equações (A.4) e (A.5), vemos que o conjugado[1] do número complexo z na forma polar é simplesmente $re^{-j\theta}$. Com isso, temos que o conjugado da equação de Euler é igual a

$$e^{-j\theta} = \cos\theta - j\,\mathrm{sen}\,\theta. \tag{A.8}$$

Somando-se as equações (A.3) e (A.8), temos

$$e^{j\theta} + e^{-j\theta} = \cos\theta + \cos\theta + j\,\mathrm{sen}\,\theta - j\,\mathrm{sen}\,\theta, \tag{A.9}$$

e assim,

$$\cos\theta = \frac{e^{j\theta} + e^{-j\theta}}{2}. \tag{A.10}$$

Ao subtrairmos a equação (A.3) pela equação (A.8), temos

$$e^{j\theta} - e^{-j\theta} = \cos\theta - \cos\theta + j\,\mathrm{sen}\,\theta - (-j\,\mathrm{sen}\,\theta), \tag{A.11}$$

e assim,

$$\mathrm{sen}\,\theta = \frac{e^{j\theta} - e^{-j\theta}}{2j}. \tag{A.12}$$

Inserindo os parâmetros módulo (A), frequência (f) e fase (ϕ) na equação (A.10), temos

$$r\cos(\omega_0 t + \phi) = \frac{re^{j(\omega_0 t+\phi)} + re^{-j(\omega_0 t+\phi)}}{2}, \tag{A.13}$$

ou ainda

$$r\cos(\omega_0 t + \phi) = \frac{r}{2}\left(e^{j\phi}e^{j\omega_0 t} + e^{-j\phi}e^{-j\omega_0 t}\right), \tag{A.14}$$

onde ω_0 é a frequência angular dada por $2\pi f_0$.

A.2 Série de Fourier

Definida a dualidade entre sinais senoidais e exponenciais, vejamos a série de Fourier em sua forma exponencial. O intuito principal da série de Fourier é representar um sinal periódico qualquer através de uma soma de senos e cossenos. A equação abaixo representa o par de equações desta série em sua forma exponencial para um sinal contínuo e periódico:

$$X[k] = \frac{1}{T}\int_T x(t)e^{-jk\omega_0 t}dt, \tag{A.15}$$

$$x(t) = \sum_{k=-\infty}^{\infty} X[k]e^{jk\omega_0 t}, \tag{A.16}$$

[1] O conjugado de um número complexo $z = a + jb$ vem a ser ele mesmo com o negativo de sua parte imaginária, i.e., $\bar{z} = a - jb$.

onde T vem a ser o período fundamental do sinal $x(t)$ e $X[k]$ os coeficientes complexos da série de Fourier exponencial, indexada pela variável inteira k; $X[0]$ indica o nível DC do sinal, $X[1]$ é o coeficiente da frequência fundamental ω_0, $X[2]$ o coeficiente do segundo harmônico ($2\omega_0$), $X[3]$ do terceiro harmônico ($3\omega_0$) e assim por diante [53].

Como os coeficientes $X[k]$ são números complexos, pode-se determinar o seu módulo $|X[k]|$ e fase $\angle X[k]$, e assim representar $X[k]$ como sendo $|X[k]|\, e^{j\angle X[k]}$, conforme visto em A.1. Substituindo este valor na equação (A.16), temos

$$x(t) = \sum_{k=-\infty}^{\infty} |X[k]|\, e^{j\angle X[k]} e^{jk\omega_0 t}. \tag{A.17}$$

Analisando as equações (A.3), (A.14) e (A.17), podemos verificar que um sinal periódico pode ser criado pela soma de senos e cossenos, cujas informações de módulo e fase encontram-se descritas em $X[k]$, cada k indicando um harmônico da frequência fundamental deste sinal.

A.3 Transformada de Fourier

De maneira bastante resumida, podemos descrever a transformada de Fourier como sendo uma generalização da série de Fourier que permite representar sinais periódicos e não periódicos na frequência. Para tal, considera-se que o período do sinal em questão é infinito, i.e., no infinito ele começaria a se repetir [50]. Vejamos, abaixo, o par transformada e inversa da transformada de Fourier:

$$X(j\omega) = \int_{-\infty}^{\infty} x(t) e^{-j\omega t} dt, \tag{A.18}$$

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\omega) e^{j\omega t} d\omega. \tag{A.19}$$

De forma análoga à vista na seção anterior, cada coeficiente da transformada inversa de Fourier $X(j\omega)$ (equação (A.19)), que agora será denominada como o sinal $x(t)$ no domínio da frequência, é um sinal complexo que possui um módulo e uma fase diretamente atrelada a este módulo, i.e., $|X(j\omega)|\, e^{j\angle X(j\omega)}$.

A partir da equação (A.19), podemos observar que um sinal no domínio do tempo pode ser criado por uma soma infinitesimal de cada elemento deste sinal no domínio da frequência, cada elemento correspondendo a uma frequência do espectro, com seus módulo e fase, $|X(j\omega)|\, e^{j\angle X(j\omega)}$.

Assim, pode-se, reforçar o conceito da transformada de Fourier, onde um si-

nal pode ser criado pela soma de infinitas senoides em intervalos infinitesimais de frequência, onde, para cada frequência infinitesimal, tem-se exatamente um valor de módulo e um valor de fase atrelados a tal senoide.

A.4 Transformada de Fourier Discreta

Um sinal discreto $x[n]$ com duração de N amostras pode ser representado pela transformada de Fourier discreta (DFT, do inglês *Discrete Fourier Transform*)

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-jk\frac{2\pi}{N}n}, \tag{A.20}$$

cuja inversa é dada por

$$x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{jk\frac{2\pi}{N}n}. \tag{A.21}$$

Se $x[n]$ decorreu da amostragem do sinal x(t) a uma taxa f_s em Hertz que atende ao critério de Nyquist [52], então a raia (em inglês, *bin*) k da DFT representa a frequência em Hertz $f_k = k\frac{f_s}{N}$ do espectro de $x(t)$. Para cada raia k, tem-se o valor do sinal na frequência $X[k]$ que é proporcional a $X(jk\frac{2\pi}{N}f_s)$, para $0 \leq k \leq N-1$. Como $X[k]$ é um sinal complexo, pode-se extrair, para cada raia k, a informação do módulo $|X[k]|$ e fase $\angle X[k]$, que vêm a ser os parâmetros necessários para determinar a senoide representada por tal raia da DFT.