ESTUDO PARA SIMPLIFICAÇÃO DO CODIFICADOR DE VOZ MELP

Marcelo Mamede Ventura

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: Sergio Lima Netto

Rio de Janeiro
Setembro de 2012

ESTUDO PARA SIMPLIFICAÇÃO DO CODIFICADOR DE VOZ MELP

Marcelo Mamede Ventura

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. Sergio Lima Netto, Ph.D.

Prof. Eduardo Antonio Barros da Silva, Ph.D.

Prof. José Antonio Apolinário Junior, D.Sc.

RIO DE JANEIRO, RJ – BRASIL
SETEMBRO DE 2012

Mamede Ventura, Marcelo

Estudo para Simplificação do Codificador de Voz MELP/Marcelo Mamede Ventura. – Rio de Janeiro: UFRJ/COPPE, 2012.

XI, 57 p.: il.; 29, 7cm.

Orientador: Sergio Lima Netto

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

Referências Bibliográficas: p. 51 – 54.

1. processamento de sinais. 2. codificador de voz. 3. MELP. I. Lima Netto, Sergio. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

À minha mãe, dedico este trabalho.

Agradecimentos

Meus sinceros agradecimentos:

- ao Prof. Sergio Lima Netto pela inestimável ajuda, senso prático e enorme paciência durante todo o período do mestrado;
- à Morgana Corrêa Campos Alves, minha esposa, pelo companheirismo e pela paciência;
- à Marinha do Brasil pela oportunidade de realizar, em tempo parcial, este trabalho. Em especial, gostaria de agradecer ao Capitão-de-Fragata (EN) Italo Ramella por todo o incentivo, confiança e suporte prestados. Também ao amigo Capitão-de-Corveta (EN) José Francisco de Andrade Júnior, que em parte influenciou na escolha do tema;
- a Thiago de Moura Prego pela ajuda na realização dos testes subjetivos, fornecendo o *setup* de testes, base de vozes e indicação de artigos relevantes; e
- a todas as pessoas que doaram voluntariamente parte do seu tempo para a realização dos testes subjetivos presentes neste trabalho.

Marcelo Mamede Ventura

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

ESTUDO PARA SIMPLIFICAÇÃO DO CODIFICADOR DE VOZ MELP

Marcelo Mamede Ventura

Setembro/2012

Orientador: Sergio Lima Netto

Programa: Engenharia Elétrica

Apresentam-se, nesta dissertação, alguns esquemas que visam à melhoria do codificador de voz MELP. A principal delas foi obtida na rotina de cálculo do *pitch*, resultando em uma significativa redução da complexidade computacional, medida por meio do decréscimo no tempo de execução do codificador em aproximadamente 13%, sem perda significativa na qualidade da voz reconstruída. Foi também avaliada a influência na qualidade da voz: da ordem dos coeficientes LP, do modelamento de pulso, e da quantização nas etapas de cálculo final do *pitch* e de modelamento do pulso. Por fim, destaca-se a importância da realização de avaliações subjetivas da qualidade da voz reconstruída em conjunto com os procedimentos de automatização proporcionados pela ferramenta PESQ.

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

STUDY FOR SIMPLIFICATION OF THE MELP VOCODER

Marcelo Mamede Ventura

September/2012

Advisor: Sergio Lima Netto

Department: Electrical Engineering

In this work, we present some schemes for improvement of the MELP vocoder. The main improvement was made on the routine for pitch calculation, which resulted on a great reduction on the computational complexity, measured by a decrease of about 13% in the vocoder running time, without significant loss of quality of the decoded voice. There were also evaluated the influence on the voice quality of: the order of the LP coefficients, the pulse shaping stage, and the quantization in the final pitch calculation and in the pulse shaping. At the end, it is highlighted the importance of subjective evaluations of voice quality in conjunction with the automation procedures provided by the PESQ tool.

Sumário

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

O espectro de freqüências pode ser considerado um recurso natural limitado, o que implica a necessidade de seu uso eficiente. Nesse sentido, os codificadores de voz prestam seu auxílio, realizando uma "compressão"da voz digitalizada e obtendo uma representação mais eficiente do sinal para envio pelo canal.

Alguns canais de comunicação são muito restritivos e exigem uma alta taxa de compressão para transmissão da voz digitalizada, o que demanda codificadores de baixa taxa. Por exemplo, em canais de 3 kHz, os modems para HF (*high-frequency*) operam em geral na faixa de 75 bps a 9600 bps, tipicamente em 1200 bps [1]. Mesmo taxas de 2400 ou 4800 bps são possíveis somente em condições favoráveis de transmissão. Isso ocorre devido às condições de propagação ionosférica nos canais de HF, sujeitos a fenômenos de dispersão, absorção e múltiplos percursos, sofrendo portanto de desvanecimento seletivo [2]. Dessa forma, há no canal de HF baixas taxas de transmissão e uma necessidade de representação eficiente da voz digitalizada.

Além disso, os codificadores de voz facilitam o emprego de códigos corretores de erro e criptografia.

Em um ambiente onde a voz digitalizada é tratada como dados, é possível transmitir simultaneamente voz e dados. Assim, uma maior compressão da voz resulta em mais espaço de *payload* para dados.

Essas características somadas tornam os codificadores de baixa taxa muito interessantes para as aplicações militares, razão pela qual o desenvolvimento dessa classe de codificadores está intimamente relacionada a esse tipo de aplicação.

A proposta deste trabalho é realizar um estudo do codificador de voz MELP (*Mixed-excitation Linear Prediction*), que surgiu como um substituto ao codificador LPC-10, descrito nas normas STANAG 4198 [3] e FED-STD 1015 [4]. Os testes comparativos do MELP com o LPC-10 e outros codificadores indicaram que, dentre outros requisitos favoráveis, o MELP apresenta uma excelente relação entre taxa e qualidade da voz decodificada, resultando na sua normatização e amplo emprego pelas forças de segurança, em particular no âmbito da OTAN (Organização do Tratado

do Atlântico Norte).

O contexto militar principal das aplicações e os interesses comerciais talvez sejam os motivos pelos quais o MELP seja pouco considerado nas pesquisas realizadas no Brasil e tão pouca literatura seja produzida em nossa língua. Contudo, hoje encontramos o codificador de voz MELP sendo utilizado em famílias de rádios modernos e exportáveis para o Brasil, os quais são empregados pelas forças de segurança nacionais [5]. Além disso, existe uma tendência de maior participação do Brasil em operações conjuntas com outras Forças Armadas no âmbito da ONU (Organização das Nações Unidas), como a MINUSTAH (Haiti) [6] e a UNIFIL (Líbano) [7], o que demanda ampliar o entendimento dessas tecnologias.

1.1 Proposta do Trabalho

Este trabalho realiza um estudo do codificador de voz MELP, implementando alguns esquemas que visam a simplificação do algoritmo de codificação de voz sem perda de qualidade perceptível da voz decodificada. A simplificação que resultou no melhor resultado foi implementada na rotina de cálculo do *pitch*, e permitiu obter uma significativa redução da complexidade computacional, medida por meio do decréscimo no tempo de execução do codificador em aproximadamente 13%, sem perda significativa na qualidade da voz reconstruída. Foi também avaliada a influência na qualidade da voz: da ordem dos coeficientes LP, do modelamento de pulso, e da quantização nas etapas de cálculo final do *pitch* e de modelamento do pulso, que implicaram resultados menos expressivos. O trabalho destaca a importância da realização de avaliações subjetivas da qualidade da voz reconstruída em conjunto com os procedimentos de automatização proporcionados pela ferramenta PESQ.

As idéias e estudos propostos são baseados na forma como os algoritmos são descritos no *draft* FED-STD [8] e na MIL-STD 3005 [9], e são independentes de uma implementação particular.

1.2 Organização da Dissertação

O Capítulo 2 apresenta uma visão geral do codificador de voz MELP, abordando de forma sucinta a sua história e os esquemas utilizados no codificador e no decodificador. Também são apresentadas as formas de avaliação objetiva e subjetiva da qualidade da voz utilizadas neste trabalho.

O Capítulo 3 propõe um esquema de simplificação do algoritmo de codificação de voz a partir de modificação na primeira etapa de estimativa do *pitch*. São apresentados a metodologia e os resultados objetivos e subjetivos da avaliação da qualidade da voz decodificada resultantes da modificação.

O Capítulo 4 complementa o estudo, apresentando outros melhoramentos ou tentativas de melhoramentos analisados. São avaliadas as influências na qualidade da voz: da ordem dos coeficientes LP, do modelamento de pulso, e da quantização nas etapas de cálculo final do *pitch* e de modelamento do pulso. A partir dos resultados obtidos no estudo da avaliação dos limiares dos parâmetros intensidade de vozeamento de baixa freqüência e do *peakness*, destaca-se a necessidade de acompanhar os procedimentos de automatização proporcionados pela ferramenta PESQ por meio de avaliações subjetivas da qualidade da voz reconstruída.

Por fim, o Capítulo 5 corresponde ao fechamento do trabalho, apresentando conclusões e propostas para futuras pesquisas.

Capítulo 2

Visão Geral do Codificador de Voz MELP

2.1 Introdução

O objetivo deste capítulo é apresentar uma visão geral do codificador de voz MELP, iniciando na Seção 2.2 com um breve histórico do seu surgimento e desenvolvimento.

Em seguida, na Seção 2.3, são descritas em linhas gerais as características do MELP.

A Seção 2.4 apresenta o codificador e a Seção 2.5, o decodificador.

A Seção 2.6 apresenta uma introdução teórica sobre as avaliações objetivas e subjetivas, utilizadas respectivamente como ferramenta guia na otimização dos parâmetros do codificador de voz e como instrumento para respaldar os resultados obtidos.

Por fim, na Seção 2.7, temos as conclusões do capítulo.

2.2 História do Codificador

O codificador MELP (*mixed excitation linear prediction*) surgiu na década de 90 do século XX, tendo como embrião as idéias de Allan McCree. Em sua tese de doutorado pelo Instituto de Tecnologia da Georgia (Estados Unidos da América), Allan McCree propôs um novo modelo de vocoder LPC para codificação de voz a baixa taxa [10] [11].

Os avanços na área de codificação de voz observados na década de 90 sugeriam que o padrão corrente em 2400 bps estava se tornando obsoleto [12] [13] [14]. Por conta disso, o Departamento de Defesa Americano fomentou em 1996 a criação de um consórcio, *Digital Voice Processing Consortium* (DVPC), cujo objetivo era selecionar um novo codificador de 2400 bps, para uso tanto militar quanto civil.

A meta do consórcio era compilar os requisitos dos vários usuários e definir um processo de seleção que atendesse aos critérios estabelecidos. Foram fixadas medidas de desempenho em ambientes ruidosos e calmos, bem como requisitos de *hardware* [15].

Nesse contexto, o codificador MELP foi proposto como candidato ao novo padrão [16] e, ao final do processo, eleito vencedor [17] [18].

Posteriormente, em 1999, os EUA normatizaram o codificador na MIL-STD-3005 (*Military Standard*) [9].

Em 2001, a OTAN também realizou um processo de avaliação, teste e seleção de um codificador de voz (*NATO Narrow Band Voice Coder*), que viria a ser padronizado no padrão STANAG 4591 (*Standardization Agreement*). Essa escolha buscava alto desempenho das comunicações em ambientes acústicos severos (p.ex. aeronaves supersônicas, helicópteros e veículos terrestres) e canais suscetíveis a erros, reconhecendo que os codificadores LPC10e (STANAG 4198) e CVSD (STANAG 4209), operando respectivamente em 2400 bps e 16 kbps, não mais representavam o estado-da-arte. Além disso, esses codificadores apresentavam desempenho substancialmente degradado nos ambientes acústicos típicos das operações exercidas pela OTAN. Também eram requeridas as taxas de 1200 e 2400 bps [19].

A taxa de 1200 bps no MELP é obtida por meio de uma eficiente quantização vetorial de blocos de três quadros consecutivos de parâmetros (super quadro) [20]. Processo similar permitiu também a redução a 600 bps [21].

2.3 Visão Geral

Essencialmente, o MELP é um codificador de voz paramétrico, que trabalha a uma taxa de 2400 bps. Sua versão mais moderna oferece também as opções de 1200 e 600 bps, obtidas basicamente por meio de quantização vetorial de blocos de parâmetros.

O grande mérito do MELP foi superar as limitações do antigo codificador de baixa taxa LPC10 (estabelecido na norma americana *Federal Standard* 1015) [4], que também opera a 2400 bps, conferindo melhor qualidade de áudio e maior robustez ao ruído ambiente. Essa melhoria de qualidade foi possível graças ao uso de um modelo mais sofisticado de produção da voz, ao custo de uma maior complexidade computacional.

A Figura 2.1 apresenta o diagrama em blocos do decodificador de voz do MELP. Nessa figura, pode ser observado que o MELP mantém alguma similaridade com o LPC10, mas introduz características novas. Essas características, representadas pelos blocos em cinza na Figura 2.1, propõem-se a tornar a voz sintetizada mais natural e, portanto, de melhor qualidade.

Em relação ao LPC10, os principais melhoramentos introduzidos pelo MELP

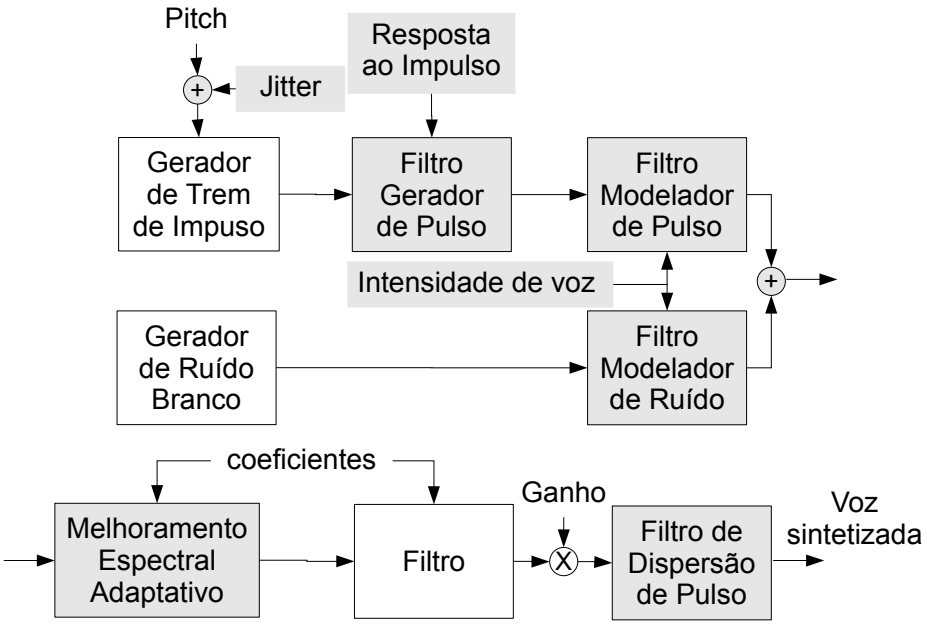

Figura 2.1: Diagrama de blocos do MELP.

foram:

- mistura de excitações: o codificador permite combinar excitações periódicas e ruidosas, associadas respectivamente às classificações vozeadas e não vozeadas do LPC;

- pulsos aperiódicos: essa melhoria permite a geração de trens de impulso aperiódicos, de forma a modelar as transições (vozeado / não-vozeado) e variações de pitch, caracterizando um terceiro tipo de excitação: vozeado com *jitter*;

- modelamento pela "magnitude"de Fourier: tendo em vista que as excitações reais não são trens de impulso ideais e suas formas reais possuem informações relevantes sobre o sinal de voz, é possível realizar uma filtragem que resulta em um sinal de excitação mais próximo do original. Esse procedimento de modelagem de pulso é baseado no cálculo das magnitudes de Fourier do sinal de erro de predição;

- melhoramento espectral adaptativo: aumenta a qualidade perceptual do sinal sintetizado, ressaltando as características espectrais originais baseadas nos coeficientes de predição linear; e

- dispersão de pulso: torna o sinal de voz sintetizado mais natural, tendo em vista que a geração da excitação é realizada a partir de um banco de filtros de

largura de banda fixa.

2.4 Codificador

As rotinas de cálculo de cada um dos parâmetros bem como os procedimentos de quantização serão apresentados a seguir, tendo [9] e [22] como as principais referências.

A idéia desta seção e da seguinte é apresentar ao leitor uma visão geral introdutória do algoritmo MELP, frisando as principais etapas e características. Para implementar o codificador é necessário recorrer à norma MIL-STD-3005 [9], que apresenta de forma pormenorizada cada uma das etapas, e inclui também os coeficientes dos filtros e dicionários utilizados no MELP.

2.4.1 Remoção das Baixas freqüências

O primeiro passo do codificador é remover as baixas freqüências presentes no sinal de entrada, a partir da aplicação de um filtro passa-altas *Chebychev* tipo II de 4a. ordem, com freqüência de corte de 60 Hz e faixa de rejeição de 30 dB. A saída desse filtro, para fins de descrição das demais etapas do codificador, será considerada o sinal de entrada.

Os sinais de voz de entrada mais recentes são armazenados em um *buffer*, e a última amostra do quadro corrente, indicada na Figura 2.2, representa uma referência importante para diversos cálculos realizados no codificador.

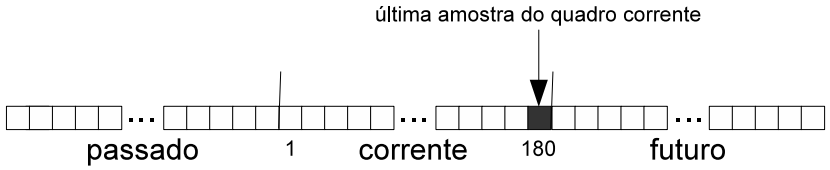

Figura 2.2: Última amostra do quadro corrrente.

2.4.2 Estimação dos Parâmetros Intensidade de Voz

O parâmetro intensidade de vozeamento (*voicing strength*) está associado a quanto um determinado quadro é vozeado e, portanto, ao seu grau de periodicidade. No MELP esse parâmetro é estimado em cinco bandas de freqüências distintas, resultando, portanto, em cinco parâmetros intensidade de vozeamento (vs_i, i = 1, 2, 3, 4 e 5), correspondendo, respectivamente, às faixas de freqüência de 0 a 500 Hz, de 500 a 1000 Hz, de 1 a 2 kHz, de 2 a 3 kHz e de 3 a 4 kHz.

Esses filtros são do tipo IIR, implementados por meio de um filtro de *Butterworth* de 6a. ordem. A opção pelo IIR deve-se à baixa complexidade computacional em comparação com o FIR, aliado ao fato de a não linearidade da resposta em freqüência ser de menor importância nesse caso [22].

O processo de cálculo dos parâmetros vs_i está ilustrado na Figura 2.3.

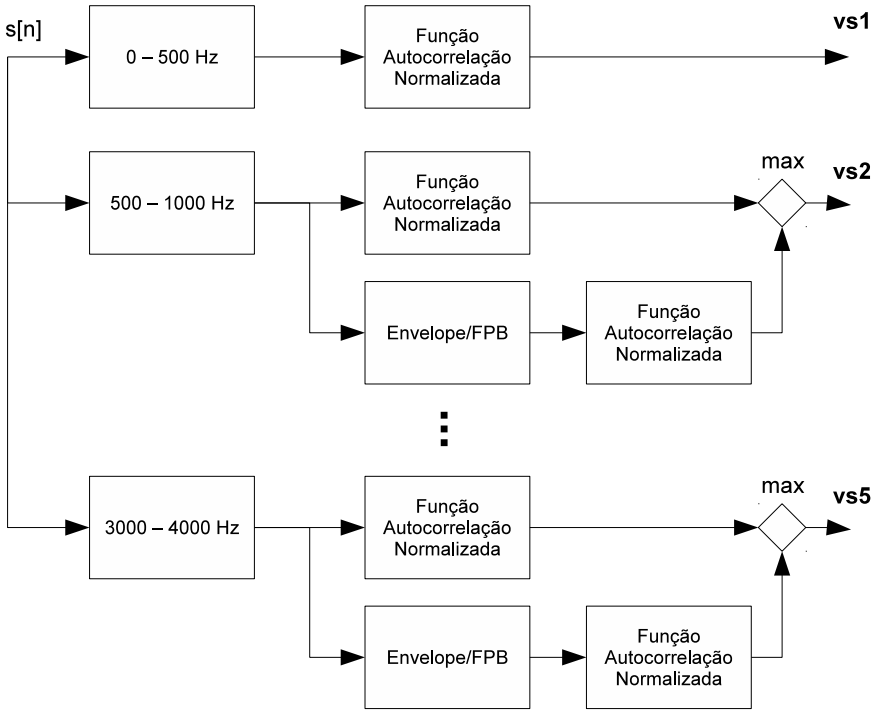

Figura 2.3: Cálculo dos parâmetros intensidade de vozeamento.

Uma primeira estimativa do valor da intensidade de vozeamento para baixas freqüências (0 - 500 Hz), ou seja, vs_1, é obtida por meio do valor $r(P_2)$ resultante das rotinas de estimativa do *pitch*, e serão vistas com mais detalhe no Capítulo 3. Esse valor pode sofrer modificação em decorrência de outras características do sinal, conforme será mostrado a seguir nesta subseção.

A estimativa da intensidade de vozeamento para as demais quatro bandas é obtida da seguinte forma:

1. Cálculo da autocorrelação normalizada do sinal resultante do filtro passa-faixa correspondente;

$$r(\tau) = \frac{c_\tau(0,\tau)}{\sqrt{c_\tau(0,0)c_\tau(\tau,\tau)}} \tag{2.1}$$

com

$$c_\tau(m,n) = \sum_{k=-\lfloor \tau/2 \rfloor - 80}^{-\lfloor \tau/2 \rfloor + 79} s_{k+m} s_{k+n} \tag{2.2}$$

onde o termo $\lfloor \tau/2 \rfloor$ na Equação (2.2) corresponde a um valor inteiro obtido por truncamento;

2. Cálculo da autocorrelação normalizada da envoltória do sinal filtrado pelo passa-faixa correspondente, no qual a envoltória é obtida por meio de uma retificação do sinal (valores absolutos das amostras), seguida de uma filtragem passa-baixa; e

3. O valor de vs_i corresponde ao maior valor obtido dos procedimentos 1 e 2 acima. Dessa forma, o parâmetro intensidade de vozeamento é determinado pela comparação entre o resultado da autocorrelação do sinal passa-banda e aquele obtido pela envoltória do sinal, escolhendo-se o maior deles.

O procedimento acima é realizado para todos os 4 filtros restantes, resultando nos parâmetros vs_2, vs_3, vs_4 e vs_5.

Um valor denominado *peakness* é utilizado para auxiliar na determinação final dos valores de vs_i para $i = 1$, 2 e 3, sendo calculado por

$$p = \frac{\sqrt{\frac{1}{160} \sum_{n=-80}^{79} e^2[n]}}{\frac{1}{160} \sum_{n=-80}^{79} |e[n]|} \tag{2.3}$$

onde $e[n]$ representa o sinal de erro da predição linear obtido com a janela da Figura 2.4. O *peakness* é um parâmetro auxiliar, não transmitido no quadro do MELP. Esse parâmetro corresponde à razão entre as normas L2 e L1 do erro de predição linear, e está associado à presença de amostras com amplitudes altas (picos) em relação à amplitude média das amostras. O cálculo do *peakness* é útil tanto para determinar se um quadro é vozeado ou não-vozeado quanto para detectar transições de sonoridade entre quadros consecutivos (ex.: de não-vozeado para vozeado), pois entre as transições são esperados valores altos do parâmetro p. Uma boa explanação e a apresentação de exemplos sobre esse parâmetro podem ser vistas em [22].

O valor de *peakness* pode alterar os parâmetros de intensidade de vozeamento correspondente às três faixas de freqüência mais baixas, conforme a seguinte regra:

- Se $p > 1{,}34$, então $vs_1 = 1$
- Se $p > 1{,}60$, então $vs_2 = 1$ e $vs_3 = 1$

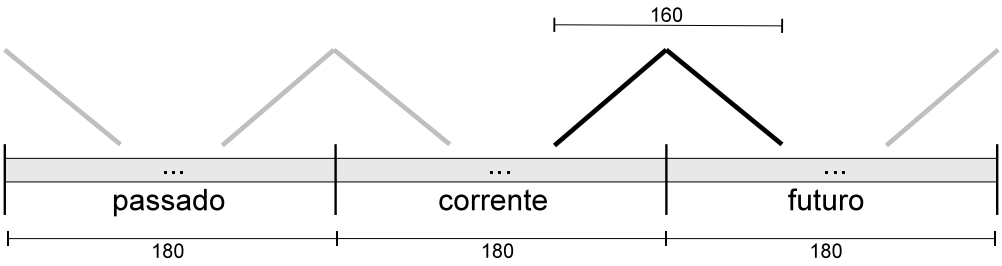

Figura 2.4: Janela para cálculo do *peakness*.

O MELP, portanto, combina as medidas de autocorrelação e *peakness* para decidir a respeito da classificação vozeado/não-vozeado e ajustar os valores de vs_1, vs_2 e vs_3.

Quantização:

Cada um dos parâmetros de intensidade de vozeamento são quantizados com 1 bit. O vs_1 é quantizado conjuntamente com o valor do *pitch* final em uma palavra código de 7 bits.

- Se $vs_1 \leq 0{,}6$, o quadro é considerado não-vozeado e um código com todos os bits iguais a zero é enviado, ou seja, $qvs_i = 0$ $(i = 1, 2, ..., 5)$, onde o valor qvs_i corresponde ao valor quantizado de vs_i;

- Do contrário ($vs_1 > 0{,}6$), o quadro é considerado vozeado.

  Se $vs_i > 0{,}6$, então $qvs_i = 1$, senão $qvs_i = 0$, para $i = 2$, 3, 4 e 5; e

- Por fim, caso qvs_2, qvs_3 e $qvs_4 = 0$, então $qvs_5 = 0$, ou seja, é forçado para bit 0.

Tem-se, portanto, a seqüência $(qvs_2, qvs_3, qvs_4, qvs_5)$ representada por 4 bits.

2.4.3 Ganho

O ganho é calculado duas vezes por quadro, resultando nos valores G_1 e G_2. O cálculo de G_1 é realizado com a janela centralizada 90 amostras antes da última amostra do quadro corrente. O cálculo de G_2 é realizado com a janela centralizada na última amostra do quadro corrente.

Conforme será visto na Seção 2.5, o decodificador do MELP realiza uma interpolação que combina os dois parâmetros (G_1 e G_2).

O comprimento da janela utilizada é o mesmo para os dois valores e depende do valor do *pitch*, conforme procedimento a seguir:

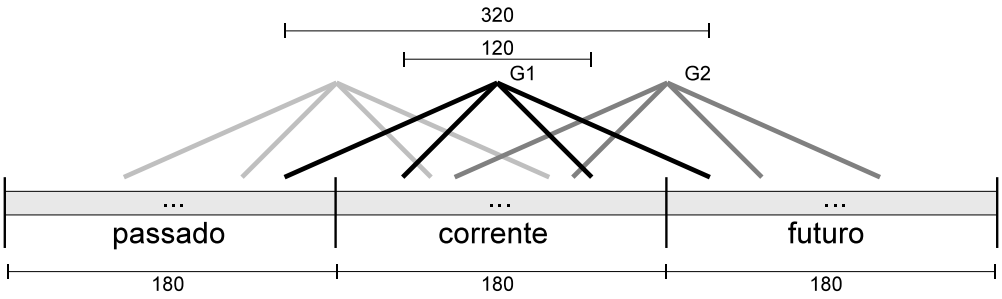

Figura 2.5: Janelas para cálculo dos ganhos.

- Quando $vs_1 > 0{,}6$, o comprimento da janela (N) corresponde ao menor múltiplo de P2 (estimativa do *pitch*) maior que 120 amostras. Caso o comprimento exceda 320 amostras, divide-se o valor por dois;
- Quando $vs_1 \leq 0{,}6$, o comprimento da janela (N) é de 120 amostras.

Dessa forma, o comprimento da janela está contido no intervalo $[120, 320]$.

O ganho referente ao valor RMS do sinal $s[n]$ considerado na janela correspondente, medido em dB, é calculado por:

$$G = 10 \log_{10}(0{,}01 + \frac{1}{N} \sum_{n} s^2[n]), \tag{2.4}$$

onde N é o tamanho da janela. O termo "0,01"evita que o argumento do logaritmo fique muito próximo de zero. Caso a medida do ganho em dB seja menor que zero, é assumido o valor $G = 0$.

Quantização:

O ganho G_2 é quantizado com 5 bits, sendo utilizada quantização uniforme no intervalo de 10 a 77 dB.

O ganho G_1 é quantizado com 3 bits, sendo utilizado um algoritmo adaptativo dependente do valor de G_2.

Se o valor de G_2 do quadro corrente diferir de até 5 dB do valor de G_2 correspondente ao quadro anterior e o valor de G_1 for até 3 dB do valor médio de G_2 dos quadros corrente e anterior, então o quadro é estável (*steady-state*) e um código especial (tudo zero) é enviado para indicar que o decodificador deve utilizar como G_1 a média dos valores de G_2 dos quadros corrente e anterior.

Do contrário, o quadro representa uma transição e G_1 será quantizado uniformemente, variando no intervalo 6 dB abaixo do menor valor de G_2 para os quadros corrente e anterior e 6 dB acima do máximo valor dos G_2. Esse intervalo é limitado

aos valores de 10 a 77 dB.

2.4.4 Estimação do Pitch

O cálculo do *pitch* é realizado em várias etapas e será descrito com mais detalhes no Capítulo 3, tendo em vista que nessa etapa foram realizadas simplificações que resultaram em uma redução da complexidade computacional.

Importante mencionar que em uma das etapas do cálculo do *pitch* encontra-se uma estimativa denominada P2 e que esse valor é utilizado na determinação da intensidade de vozeamento, no cálculo do ganho e no próprio algoritmo de cálculo do *pitch* final.

No algoritmo final do *pitch*, pode vir a ser utilizado um valor correspondente à média do valor do *pitch* de quadros anteriores, denominado P_{avg}.

- Se $r(P3) > 0{,}8$ e $G_2 > 30$ dB, então P3, que corresponde ao candidato a *pitch* final, é colocado em um *buffer* contendo os três maiores valores do *pitch* (p_i, $i = 1$, 2 e 3). O valor de $r(P3)$ corresponde ao resultado da aplicação da função de autocorrelação normalizada; e

- Do contrário, os três valores do *pitch* no *buffer* são atualizados de acordo com

$$p_i = 0{,}95p_i + 0{,}05P_{default}, \tag{2.5}$$

  para $i = 1, 2$ e 3, onde $P_{default} = 50$ amostras.

O valor de P_{avg} atualizado é:

$$P_{avg} = \frac{1}{3}\sum_{i=1}^{3} p_i. \tag{2.6}$$

Quantização:

O valor final do *pitch* é quantizado em uma escala logarítmica de 99 níveis e um quantizador uniforme cobrindo a faixa de 20 a 160 amostras. Esses valores de *pitch* são então mapeados em uma palavra código de 7 bits usando uma tabela, que pode ser obtida em [9]. O valor final do *pitch* é quantizado em conjunto com o parâmetro intensidade de vozeamento de baixa freqüência, vs_1.

2.4.5 Analise de Predição Linear

A análise predição linear (*linear prediction* - LP) utiliza uma janela de *Hamming* de 200 amostras (25 ms) centrada na última amostra do quadro corrente e, portanto, utiliza amostras tanto do quadro corrente como do quadro "futuro"(próximo quadro

armazenado no *buffer*). Essa análise possui ordem 10 e é implementada por meio do algoritmo de recursão de *Levinson-Durbin*.

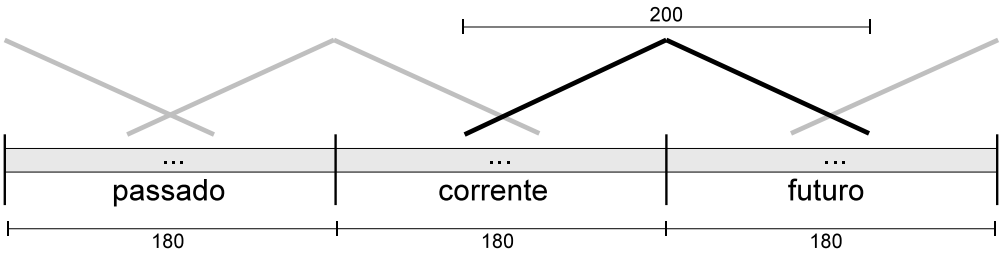

Figura 2.6: Janela para análise LP.

Aos coeficientes de predição aplica-se uma expansão de banda dos coeficientes por um fator 0,994 (15 Hz), de forma que cada coeficiente é multiplicado por $0,994i$, para $i = 1, 2, ..., 10$. Esse procedimento evita que os filtros de síntese com pólos localizados muito próximos do círculo unitário (marginalmente estáveis) se tornem instáveis após quantização. Outra vantagem citada em [22] é o fato de que a técnica de expansão de banda reduz a duração da resposta ao impulso, o que resultaria em maior robustez face aos erros introduzidos pelo canal.

Quantização:

Os 10 coeficientes de predição são convertidos em componentes LSF (*Linear Spectral Frequency*), e dispostos em um vetor de 10 elementos. Em seguida, um procedimento força que os componentes LSF (f_i, para i = 1, ..., 10) estejam em ordem ascendente com separação mínima de 50 Hz, verificando-se os pares adjacentes e corrigindo caso necessário. Dessa forma, as freqüências LSF são ordenadas de tal forma que $f_i < f_{i+1}$. Essa verificação por pares é realizada quantas vezes for necessária.

Em seguida, um algoritmo (descrito em [9]) assegura a separação mínima de 50 Hz entre os componentes de freqüência ($d = f_{i+1} - f_i < 50$ Hz). Essa rotina é executada 10 vezes.

O procedimento acima resulta em um vetor LSF, e a ele se aplica um quantizador vetorial MSQV (*Multi-Stage Vector Quantizer*). O dicionário desse quantizador possui quatro estágios com 128, 64, 64 e 64 níveis, respectivamente.

O vetor quantizado, $\hat{f}$, corresponde à soma dos vetores selecionados pelo procedimento de busca em cada um dos estágios.

A busca no dicionário visa encontrar o vetor que minimiza a distância euclidiana

(d^2) ponderada entre os vetores não quantizados e quantizados, conforme:

$$d^2(f, \hat{f}) = \sum_{i=1}^{10} w_i (f_i - \hat{f}_i) \tag{2.7}$$

onde

$$w_i = \begin{cases} P(f_i)^{0,3} & , 1 \leq i \leq 8, \\ 0,64 P(f_i)^{0,3} & , i = 9, e \\ 0,16 P(f_i)^{0,3} & , i = 10, \end{cases}$$

f_i é a i-ésima componente do vetor LSF não quantizado e $P(f_i)$ corresponde ao espectro de potência do inverso do filtro de predição, avaliado na freqüência f_i.

O procedimento para assegurar ordem ascendente e separação mínima é também aplicado ao vetor LSF quantizado. Esse vetor resultante é utilizado no cálculo da magnitude de Fourier.

2.4.6 *Flag* Aperiódico

Esse parâmetro define se o decodificador utilizará pulsos aperiódicos (*jittery voiced*) ao invés de periódicos. Na condição aperiódica, o *flag* assume o valor "1". Essa decisão é determinada pelo parâmetro vs_1.

Em resumo, tem-se a seguinte condição:

$$af = \begin{cases} 1 & , \text{se } vs_1 < 0{,}5, e \\ 0 & , \text{caso contrário.} \end{cases}$$

O caso $af = 1$ também ocorre nos quadros de transição, onde a excitação encontra-se em um estado intermediário entre a periodicidade e a aleatoriedade.

O quadro do codificador MELP reserva um bit para o *flag* aperiódico.

2.4.7 Magnitude de Fourier

Essa etapa é utilizada nos modos vozeados ou vozeados com *jitter*, e a sua função é obter um sinal de excitação mais próximo da excitação original. Nos casos reais, não temos como excitação um trem de impulsos, mas sim um trem de pulsos, em que os pulsos apresentam em sua forma informações referentes ao sinal de voz original.

Para obter essas informações relativas ao formato dos pulsos, o codificador realiza uma medida das magnitudes de Fourier das 10 primeiras freqüências harmônicas do *pitch* do resíduo da predição linear resultante dos coeficientes de predição quantizados. Portanto, este procedimento é executado após o processo de cálculo e quantização dos parâmetros LSF.

Utiliza-se uma janela de amostragem de 200 amostras, como a da Figura 2.6,

para a análise LP, aplicada ao erro de predição linear, e calcula-se uma FFT (*Fast Fourier Transform*) com 512 pontos. A seguir são apresentadas, de forma resumida, as etapas desse processo:

1. Cálculo dos 10 coeficientes de predição quantizados a partir do vetor LSF quantizado;

2. Obtém-se o sinal resíduo com os coeficientes LP calculados;

3. Janela de *Hamming* de 200 amostras é aplicada, e o sinal é preenchido com zeros (*zero-padded*) até 512 pontos e a FFT é realizada;

4. Os valores complexos da FFT são transformados em valores reais (magnitudes);

5. As freqüências harmônicas são encontradas por meio de um algoritmo que busca os valores de pico (*peak-picking algorithm*). Esse algoritmo mensura o máximo dentro do intervalo centrado na estimativa inicial de cada harmônico do *pitch* e com comprimento definido pelo valor truncado de $\frac{12}{\hat{P}3}$. A estimativa inicial da localização da i-ésima harmônica é $\frac{512i}{\hat{P}3}$ e a magnitude dos harmônicos procurados são limitados ao menor de 10 ou $\frac{P3}{4}$;

6. Essas magnitudes são normalizadas para ter um valor RMS de 1,0. Caso menos que 10 harmônicos sejam encontrados, as magnitudes restantes são definidas como "1,0";

7. As 10 magnitudes encontradas são quantizadas por meio de um quantizador vetorial de 8 bits. A busca no dicionário é realizada utilizando a distância euclidiana ponderada, com pesos fixos que enfatizam as baixas freqüências sobre as altas freqüências. Tem-se então novamente a Equação (2.7), mas com os pesos definidos por:

$$w_i = \left[\frac{117}{25 + 75(1 + 1,4(\frac{f_i}{1000})^2)^{0,69}} \right]^2 . \tag{2.8}$$

2.4.8 Empacotamento e Correção de Erros

Após obtidos os parâmetros do codificador do MELP, o quadro é montado de acordo com a Tabela 2.1.

Os 41 bits correspondentes aos coeficientes LSF, aos ganhos (G_1 e G_2), ao parâmetro $pitch/vs_1$ (quantizados conjuntamente) e ao sincronismo são transmitidos tanto para os quadros vozeados quanto para os não-vozeados.

Tabela 2.1: *Tabela de Alocação de Bits do MELP.*

Parâmetros	Vozeado	Não-vozeado
Line Spectral Frequencies	25	25
Magnitude de *Fourier*	8	-
Ganho (2 por quadro)	8	8
Pitch, intensidade de vozeamento (baixa freqüência)	7	7
Intensidade de Vozeamento (*Bandpass voicing*)	4	-
Flag aperiódico	1	-
Código Corretor de Erro	-	13
Bit de sincronismo	1	1
Total de Bits / 22,5 ms (quadro)	54	54

Nas situações em que temos somente quadros vozeados, têm-se também as magnitudes de Fourier, os parâmetros intensidade de vozeamento relacionados aos filtros passa-faixa ("bandpass voicing") e o *flag* aperiódico, que totalizam 13 bits.

Nos quadros não-vozeados, esses 13 bits são utilizados nos códigos detectores/corretores de erros.

Para detecção/correção de erros são utilizados três códigos de *Hamming* (7, 4) e um código de *Hamming* (8, 4). O primeiro, (7, 4), é capaz de corrigir um erro simples de bit, ao passo que o segundo, (8, 4), é capaz de detectar dois erros de bits.

O código (8, 4) é utilizado para os 4 bits mais significativos (MSBs) do primeiro índice do MSVQ, sendo alocados nas posições que seriam dos parâmetros intensidade de vozeamento.

Os 3 bits restantes do primeiro índice MSVQ (quantização LSF), mais um bit reserva (cujo valor é zero), são cobertos pelo código (7, 4), com os 3 bits de paridade resultantes escritos nos MSBs dos índices de quantização vetorial de Fourier.

Os 4 MBSs da palavra código de G_2 são protegidos por 3 bits de paridade escritos nos próximos 3 bits das magnitudes de Fourier.

Por fim, o LSB do índice do segundo ganho e 3 bits da palavra código de G_1 são protegidos por 3 bits de paridade escrito nos 2 LSBs da magnitude de Fourier e do *flag* aperiódico.

O diagrama das etapas realizadas pelo codificador pode ser observado na Figura 2.7. As setas verticais indicam a ordem de cálculo, conforme sugerido em [9]. Na horizontal constam as entradas e saídas dos blocos.

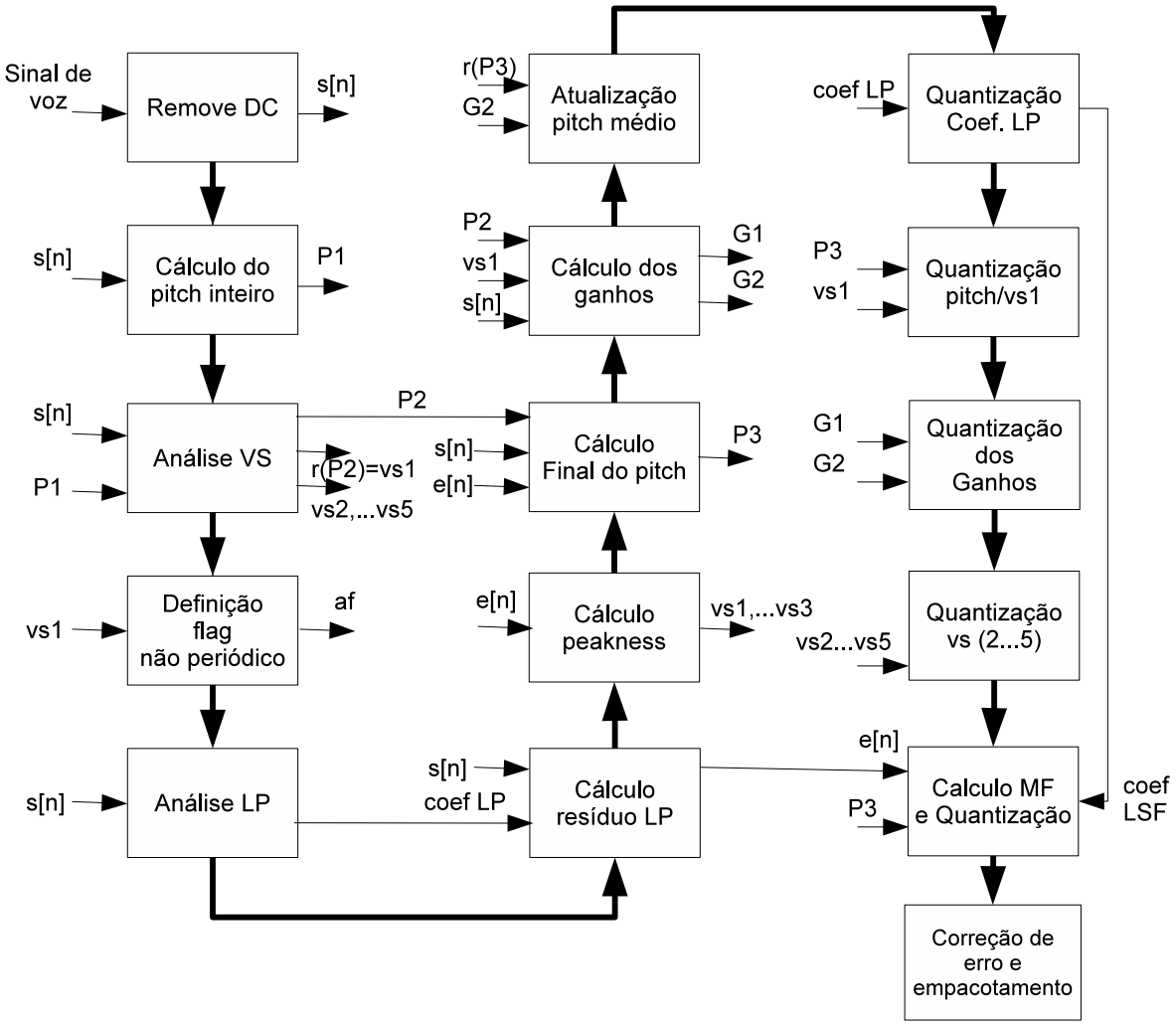

Figura 2.7: Diagrama das etapas do codificador.

2.5 Decodificador

Uma vez recebidos, os bits são "desempacotados"e atribuídos aos respectivos parâmetros após as verificações e correções de erros de bits eventualmente impostos pelo canal.

Um diagrama com as etapas do decodificador está presente na Figura 2.8.

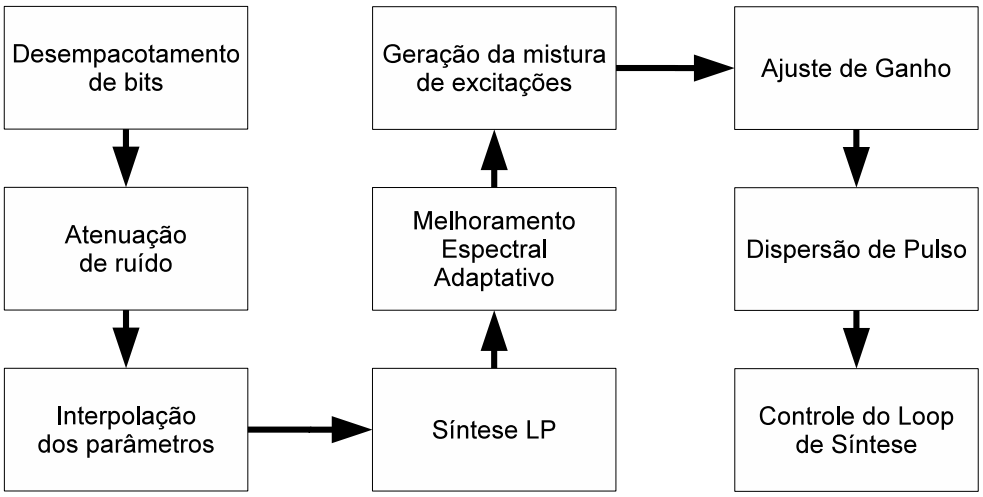

Figura 2.8: Diagrama das etapas do decodificador.

O *pitch* é o primeiro parâmetro decodificado, uma vez que ele contém a informação referente ao modo, já que os parâmetros são diferentes nos casos vozeado e não-vozeado.

- Se todos os bits $pitch/vs_1$ são "0"ou somente um bit é igual a "1", tem-se o modo não-vozeado;
- Se dois bits $pitch/vs_1$ possuem valor "1", tem-se a condição *frame erasure*. Essa condição ocorre quando se detecta um erro que não pode ser corrigido, e resulta, na prática, em repetição do quadro anterior. Nesse caso, a condição $G_1 = G_2$ é imposta, de forma a não permitir ganho na transição; e
- Do contrário, tem-se o modo vozeado e o valor do *pitch* é decodificado utilizando a tabela presente em [9], mencionada na Subseção 2.4.4.

No caso em que não há a condição *frame erasure*, os demais parâmetros são decodificados.

Os componentes LSFs são verificados quanto à ordem ascendente e separação mínima, como ocorre no codificador.

No modo não-vozeado, alguns parâmetros assumem valores padrão: *pitch* = 50 amostras; *jitter* = 25%; $vs_i = 0$ (i = 1, 2, 3, 4 e 5); e magnitudes de Fourier = 1. A definição desses valores é importante, pois o decodificador do MELP utiliza interpolação linear dos parâmetros baseada no período do *pitch* durante a síntese da voz, conforme será visto mais adiante.

No modo vozeado, temos: $vs_1 = 1$; *jitter* = 25%, se $af = 1$ (*flag* aperiódico), senão *jitter* = 0%; vs_i recebem os valores 1 ou 0 correspondentes aos bits 1 e 0, respectivamente. A exceção ocorre no caso 0001 para qvs_i (i = 2, 3, 4 e 5), pois nesse caso temos a imposição de $vs_5 = 0$.

No caso especial em que todos os bits do parâmetro G_1 possuem valores nulos, alguns erros em G_2 podem ser detectados e corrigidos, o que melhora o desempenho do codificador na presença de ruídos no canal. Um pseudo-algoritmo para decodificação dos parâmetros G_1 e G_2 pode ser consultado em [9].

Para síntese da voz, o decodificador MELP efetua procedimentos como ajustes no ganho, interpolação dos parâmetros e filtragens que visam melhorar a qualidade e naturalidade do resultado final, conforme será visto nas subseções a seguir.

2.5.1 Atenuação do Ruído

Os ganhos G_1 e G_2 sofrem uma pequena atenuação dos seus valores na presença de sinais de entrada correspondentes aos "silêncios". Nessas situações, aplica-se uma regra de subtração espectral, que tem a sua origem em uma simplificação do caso invariante em freqüência do método *Smoothed Spectral Subtraction Noise Supression* [9].

Inicialmente, deve haver uma atualização da estimativa do valor de ruído de fundo, conforme procedimento a seguir:

- Se $G_1 > G_n + C_{up}$, então $G_n = G_n + C_{up}$
- Se $G_1 < G_n - C_{down}$, então $G_n = G_n - C_{down}$
- Do contrário, $G_n = G_1$

onde C_{up} = 0,0337435 e C_{down} = 0,135418. Dessa forma, a estimativa de ruído se move para cima 3 dB/s ou para baixo 12 dB/s para uma taxa de atualização de ganho de 88,9 atualizações por segundo. A estimativa do ruído está limitada ao intervalo de 10 a 80, para garantir que a atenuação seja aplicada apenas aos "silêncios". Não há atualização no caso de quadros repetidos.

Essa estimativa do ruído de fundo é utilizada também no procedimento de "Melhoria Espectral Adaptativa", que será visto adiante.

Tem-se então que:

$$G_1 = G_1 - G_{att} \tag{2.9}$$

sendo G_{att} um valor positivo em dB, calculado por

$$G_{att} = -10\log_{10}(1 - 10^{0,1(G_n+3-G_1)}), \tag{2.10}$$

onde os valores de G_n e G_1 estão em dB. Essa correção está limitada ao valor máximo de 6 dB para evitar flutuações e distorções no sinal.

Os mesmos procedimentos apresentados acima são repetidos, dessa vez para o parâmetro G_2.

2.5.2 Interpolação de Parâmetros

O decodificador utiliza um sofisticado processo de interpolação linear dos parâmetros, que visa suavizar as transições entre os quadros. São interpolados os parâmetros ganho, LSFs, *jitter*, magnitudes de Fourier, coeficientes dos filtros modeladores de pulso e ruído e coeficiente *tilt* do filtro de melhoramento espectral.

A interpolação é realizada com base nos parâmetros do quadro corrente e do passado, sendo feita a cada instante n_0 pertencente ao quadro corrente ($n_0 \in [0, 179]$).

O fator de interpolação é definido por

$$\alpha = \frac{n_0}{180}. \tag{2.11}$$

O parâmetro interpolado, que será chamado de I, é calculado de acordo com

$$I = (1 - \alpha)I_p + \alpha I_c. \tag{2.12}$$

onde I_p representa o parâmetro do quadro passado e I_c o parâmetro do quadro corrente.

Especificamente no caso dos ganhos, a interpolação é realizada de acordo com:

$$G = \begin{cases} (1-\alpha)G_{2p} + \alpha G_1 & , n_0 < 90 \\ (1-\alpha)G_1 + \alpha G_2 & , 90 \leq n_0 < 180 \end{cases} \tag{2.13}$$

onde G_{2p} representa G_2 do quadro passado.

2.5.3 Geração da Excitações Mistas

O sinal de excitação mista é formado pela soma das excitações filtradas dos pulsos e do ruído. No caso dos pulsos, tem-se a excitação e_p obtida pela inversa da

Transformada Discreta de Fourier:

$$e_p = \frac{1}{T}\sum_{k=0}^{T-1} M(k)e^{2\pi nk/T} \tag{2.14}$$

com $n = 0, 1, ..., T-1$.

O valor T é o período do *pitch*, obtido por:

$$T = T_0(1 + jitter.x), \tag{2.15}$$

onde T_0 é o valor do *pitch* interpolado e x é uma variável aleatória uniformemente distribuída no intervalo $[-1, 1]$. O período do *pitch* obtido é arredondado para o valor inteiro mais próximo, limitado ao intervalo 20 a 160.

Os $M(k)$ $(k = 1, 2, ..., 10)$ da Equação (2.14) são os valores interpolados das magnitudes de Fourier. Todos os valores de $M(k)$ são reais e simétricos, de forma que e_p é real e as magnitudes obedecem:

$$M(T-k) = M(k) \tag{2.16}$$

para $k = 1, 2, ..., L$, onde

$$L = \begin{cases} \frac{T}{2} & \text{para T par,} \\ \frac{T-1}{2} & \text{para T ímpar.} \end{cases}$$

Para o termo DC, tem-se que $M(0) = 0$.

O pulso é multiplicado pela raiz quadrada do *pitch* para resultar em um valor RMS unitário, que é então multiplicado por 1000 (valor nominal do sinal).

O ruído, por sua vez, é obtido por meio de um gerador de números aleatórios com valor RMS 1000, no intervalo de -1732 a 1732.

As excitações de pulsos e de ruído são aplicadas, respectivamente, aos filtros modeladores de pulso e de ruído, conforme apresentado na Figura 2.1, e posteriormente somadas, resultando na mistura de excitações.

A resposta desses filtros é controlada pelos parâmetros intensidade de vozeamento (*voicing strenghts*), vs_i $(i = 1, 2, ..., 5)$.

Cada um desses filtros é composto por um banco de cinco filtros, que cobrem as faixas de freqüência de 0 a 500 Hz, de 500 a 1000 Hz, de 1 a 2 kHz, de 2 a 3 kHz e de 3 a 4 kHz, que são as mesmas faixas de freqüência utilizadas no codificador. Contudo, ao contrário do que ocorre no codificador, temos nesse caso filtros FIR com 31 coeficientes. A referência [22] apresenta como razões para a escolha de filtros FIR: fase linear, implicando atraso de grupo constante com a freqüência; relativa facilidade em alterar a resposta em freqüência; e facilidade de realizar as

interpolações, sem preocupação com instabilidades.

Os coeficientes desses filtros de síntese (h_i) são apresentados na Tabela A-I encontrada em [9].

A resposta em freqüência desse filtros modeladores (filtros de síntese) é o resultado dos cinco filtros em paralelo, expressos por:

$$h_p[n] = \sum_{i=1}^{5} vs_i h_i[n], e \tag{2.17}$$

$$h_n[n] = \sum_{i=1}^{5} (1 - vs_i) h_i[n]. \tag{2.18}$$

Pode ser observado que o ganho dos dois filtros permanece constante, pois quando o ganho de um deles é alto, o do outro é baixo, e vice-versa.

2.5.4 Melhoramento Espectral Adaptativo

A mistura de sinais de excitação é aplicada à entrada do filtro de melhoramento espectral adaptativo, cuja função de transferência é descrita por

$$H(z) = (1 - \mu z^{-1}) \frac{1 + \sum_{i=1}^{10} a_i \beta^i z^{-i}}{1 + \sum_{i=1}^{10} a_i \alpha^i z^{-i}}, \tag{2.19}$$

$$\text{onde} \begin{cases} \alpha = 0{,}5ps \\ \beta = 0{,}8ps. \end{cases}$$

Esse filtro é idêntico ao de pós-processamento utilizado no codificador CELP [22] [23], e a sua função, como o próprio nome sugere, é melhorar a qualidade subjetiva (perceptual) do sinal sintetizado, acentuando as características perceptuais originais.

Esse filtro é composto por um filtro pólo/zero de ordem 10 e um filtro de compensação de *tilt* de 1a. ordem. Seus coeficientes são gerados pela expansão de banda da função de transferência do filtro de predição linear, $A(z)$, correspondentes aos valores LSF interpolados.

Para obter μ, temos inicialmente $\mu = max(0{,}5k_1; 0)$, seguido de uma interpolação e então multiplicação por ps, que representa a probabilidade do sinal, estimada por:

$$ps = \frac{G_{int} - G_n - 12}{18} \tag{2.20}$$

O parâmetro k_1, correspondente ao primeiro coeficiente de reflexão, é derivado do coeficiente LP a_1.

2.5.5 Síntese de Predição Linear

Utiliza um filtro de síntese, cujos coeficientes LP são obtidos a partir dos valores LSF interpolados enviados pelo codificador.

2.5.6 Ajuste do Ganho

Uma vez que a excitação é gerada em um nível arbitrário, é necessário ajustar a saída do filtro de síntese. Esse ajuste é realizado calculando um fator de escala, S_g, para cada período de *pitch* de comprimento T, conforme:

$$S_g = \frac{10^{G/20}}{\sqrt{\frac{1}{T}\sum_n \hat{s}^2[n]}}, \tag{2.21}$$

onde G corresponde ao ganho interpolado, em dB. No denominador, tem-se o valor RMS do sinal sintetizado $\hat{s}$.

Para evitar descontinuidades na voz sintetizada, esse fator de escala é interpolado linearmente entre os valores anteriores e correntes das 10 primeiras amostras do período do pitch.

2.5.7 Dispersão do Pulso

O filtro de dispersão de pulso é obtido por meio de um filtro FIR de ordem 65, derivado de um pulso triangular espectralmente plano. Os coeficientes desse filtro podem ser obtidos em [9].

Esse filtro tem a função de melhorar a naturalidade da voz sintetizada, tendo em vista que a geração da excitação é realizada a partir de um banco de filtros de largura de banda fixa.

De acordo com [14], nas regiões que não contêm os formantes, o filtro melhora o casamento entre as formas de onda da voz original e a sintetizada a partir dos filtros passa-faixa.

2.5.8 Controle do *Loop* de Síntese

Após o processamento de cada período do *pitch*, o decodificador atualiza n_0 adicionando T ao número de amostras no período que acabou de ser sintetizado.

Se $n_0 < 180$, a síntese para o quadro corrente continua a partir da etapa de interpolação dos parâmetros.

Do contrário, o decodificador armazena em um *buffer* o restante das amostras no período corrente que estendem além do final do quadro corrente e subtrai 180 de n_0 para produzir o valor inicial do próximo quadro.

2.6 Avaliação da Qualidade de Voz

O presente trabalho fez uso de métodos para avaliações da qualidade de voz como guia para orientar o processo de otimização do algoritmo, ou para ratificar os resultados obtidos com as otimizações propostas.

No primeiro caso, utilizou-se a avaliação objetiva proporcionada pelo algoritmo *Perceptual Evaluation of Speech Quality* (PESQ), definido na recomendação P.862 da União Internacional de Telecomunicações (*International Telecommunication Union* - ITU) [24].

No segundo caso, visando ratificar os resultados obtidos, foram realizadas avaliações por meio dos testes *Absolute Category Rating (ACR)* e *Comparison Category Rating* (CCR), baseados na recomendação ITU-T P.800 [25].

2.6.1 Avaliação Objetiva

O PESQ permite estimar a nota *Mean Opinion Score* (MOS) de um sinal de voz com banda estreita, tipicamente presente em banda telefônica. O MOS é uma avaliação subjetiva, descrita na ITU-T P.800, que representa a média das avaliações de um grupo de pessoas, que atribuem valores inteiros de "1"a "5"às frases decodificadas pelo *vocoder* em estudo. Na escala MOS, "5"corresponde à melhor nota e "1"à pior. O MELP possui MOS tipicamente de 3 a 3,2.

Dessa forma, o PESQ permite estimar objetivamente, por meio de um algoritmo implementado em um computador, valores MOS que seriam obtidos por meio de experimentos demorados e custosos, realizados com um grupo de ouvintes em condições particulares.

O algoritmo realiza uma comparação entre o sinal de voz original com o sinal degradado, correspondente à saída de um sistema de telecomunicações. O resultado obtido pelo PESQ pode ser mapeado na escala de avaliação subjetiva MOS por meio da Equação (2.22) [23]:

$$MOS = 0,999 + \frac{4}{1 + e^{-1,4945PESQ+4,6607}} \tag{2.22}$$

O PESQ estima a nota MOS de forma aceitável quando o sinal de voz é afetado pelos seguintes processos ou degradações: filtragem; atraso variável; codificação de baixa taxa de bits; e erros de canal [26].

2.6.2 Avaliação Subjetiva

Os testes subjetivos realizados se basearam na recomendação ITU-T P.800 [25], mais especificamente nos tipos de testes *Absolute Category Rating (ACR)* e *Comparison*

Category Rating (CCR). O primeiro deles objetiva estabelecer uma nota absoluta para a qualidade do áudio, de acordo com uma escala de referência (MOS).

Já o segundo fornece uma medição relativa por meio da comparação direta entre a versão original e a modificada.

2.7 Conclusão

As Seções 2.2 e 2.3 apresentam, respectivamente, a origem do MELP e as suas principais características. Mostrou-se que o MELP pode ser visto como uma evolução do LPC10, e que ele permite uma melhor representação da voz, a custo de uma maior complexidade computacional. Decorre dessa melhor representação um maior MOS.

A Seção 2.4 apresentou a estrutura do codificador e os procedimentos para obtenção dos parâmetros. Na Seção 2.5 foram vistos os procedimentos para decodificação que permitem a reconstrução do sinal de voz original. As seções relativas à codificação e à decodificação permitem ao leitor um primeiro contato com o codificador MELP, apresentando quais são os parâmetros utilizados e, de forma geral, como esses parâmetros são calculados e utilizados. O detalhamento do cálculo do *pitch* foi deixado para o Capítulo 3, no qual são descritas a forma original e a proposta de simplificação.

Por fim, na Seção 2.6 foram apresentadas as formas de avaliação objetiva e subjetiva da qualidade da voz. A avaliação objetiva, realizada por meio da ferramenta PESQ, foi utilizada ao longo de todo o trabalho como guia nos procedimentos de simplificação propostos. A avaliação subjetiva foi utilizada no capítulo 3 para ratificar os resultados obtidos.

Capítulo 3

Aceleração do Cálculo de *Pitch*

3.1 Introdução

Este capítulo descreve as alterações implementadas na rotina de cálculo do *pitch*, que resultaram em uma simplificação do algoritmo do codificador de voz MELP.

Na Seção 3.2 é apresentada, de forma geral, o algoritmo de cálculo do *pitch*, conforme consta em [9].

A Seção 3.3 descreve as modificações propostas. Primeiramente, mostra-se a forma "padrão"do cálculo da função de correlação normalizada, bem como o uso das recursões que permitem reduzir o número de operações. Em seguida, são apresentadas as modificações propriamente ditas, implementadas por meio do uso de decimações, exclusivamente na primeira etapa do cálculo do *pitch*.

As alterações realizadas resultam em uma família de versões modificadas do codificador, cada uma com complexidade computacional e qualidade próprias. Na Seção 3.4, é apresentado o procedimento utilizado para realização de uma avaliação objetiva, que serviu para selecionar uma versão, dentre as várias disponíveis, para ser submetida às avaliações subjetivas.

A Seção 3.5 descreve as avaliações subjetivas realizadas: *Absolute Category Rating* (ACR) e *Comparison Category Rating* (CCR).

Na Seção 3.6 são apresentadas as conclusões deste capítulo.

3.2 Descrição do Algoritmo para Cálculo do Pitch

Conforme apresentado no Capítulo 2, no MELP todo o processamento se inicia com uma filtragem destinada a retirar eventual componente DC presente no sinal. Essa filtragem é obtida por meio de um filtro passa-altas de *Chebychev* tipo II, com freqüência de corte de 60 Hz e banda de rejeição de 30 dB.

Para fins práticos, considera-se o sinal de voz a saída desse filtro.

Um diagrama em blocos do cálculo do *pitch* é apresentado na Figura 3.1.

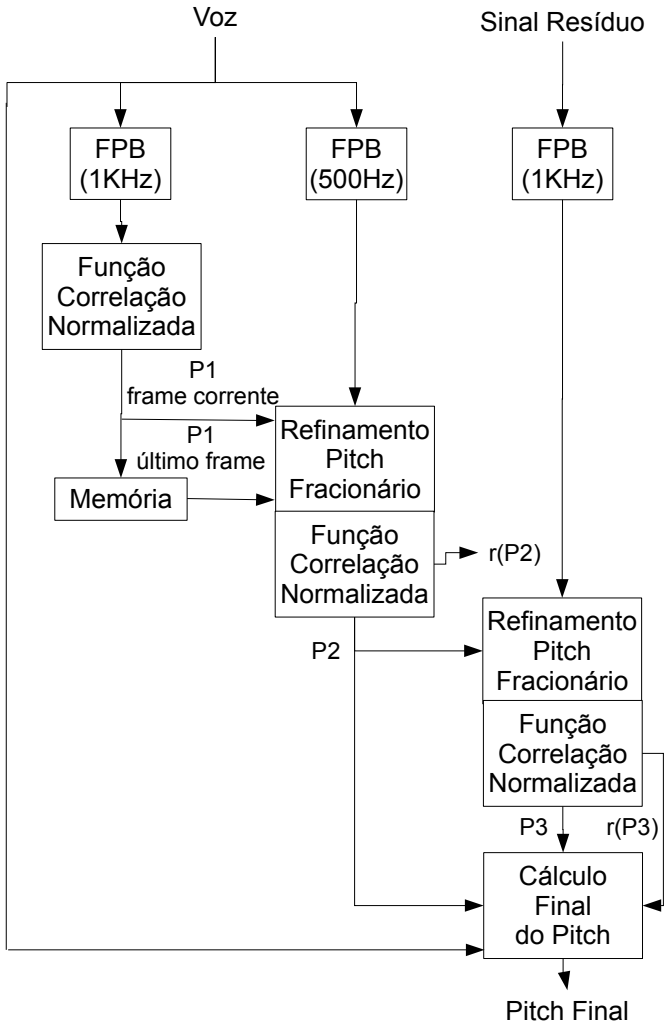

Figura 3.1: Diagrama de blocos do decodificador MELP.

Nos cálculos do *pitch*, são consideradas as 160 últimas amostras do quadro corrente e as primeiras 160 amostras do próximo quadro, conforme apresentado na Figura 3.2.

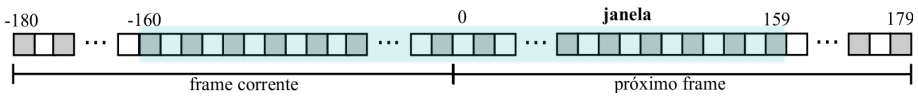

Figura 3.2: Janela de amostragem.

A partir de então, é realizada uma filtragem do sinal de voz por meio de um filtro passa-baixas com freqüência de corte de 1 kHz. O resultado dessa filtragem corresponde ao sinal s_k utilizado pela função de correlação normalizada determinada

por

$$r(\tau) = \frac{c_\tau(0,\tau)}{\sqrt{c_\tau(0,0)c_\tau(\tau,\tau)}}, \tag{3.1}$$

com

$$c_\tau(m,n) = \sum_{k=-\lfloor\tau/2\rfloor-80}^{-\lfloor\tau/2\rfloor+79} s_{k+m}s_{k+n}. \tag{3.2}$$

O termo $\lfloor\tau/2\rfloor$ na Equação (3.2) corresponde a um valor inteiro obtido por truncamento. A normalização utilizada na Equação (3.1) visa compensar a variação de energia ao longo do tempo [9] [22].

Nas Equações (3.1) e (3.2), o valor de τ está associado ao *pitch*, e está compreendido no intervalo $40 \leq \tau \leq 160$. Isso indica que, considerando a freqüência de amostragem de 8 kHz, a busca do *pitch* está restrita inicialmente aos valores, em freqüência, que correspondem a 50 Hz $\leq$ *pitch* $\leq$ 200 Hz.

No intervalo considerado, o valor de τ que maximiza $r(\tau)$, descrita em (3.1), resulta em uma primeira estimativa inteira do *pitch*, denominada P1.

Posteriormente, uma etapa de refinamento do *pitch* é realizada, visando melhorar a estimativa inicial. Neste sentido, o procedimento verifica se o valor de máxima correlação está localizado no intervalo $[\tau-1;\tau]$ ou $[\tau;\tau+1]$. Essa verificação ocorre avaliando se $c_\tau(0,\tau-1) > c_\tau(0,\tau+1)$, pois tal situação indica que o valor máximo está em $[\tau-1;\tau]$ e, nesse caso, $\tau = (\tau-1)$. Caso contrário, a etapa seguinte é executada diretamente, consistindo no uso das Equações (3.1) e (3.2), com um novo sinal $\bar{s}_k$, saída de um filtro passa-baixas com freqüência de corte de 500 Hz.

Essa nova busca, contudo, é realizada somente para 5 amostras em torno do candidato a P1 do quadro corrente e do quadro anterior, ou seja, é efetuada no intervalo $[P1-5; P1+5]$.

Os dois valores obtidos, que maximizam a função de correlação normalizada nos intervalos considerados, são usados para determinar dois valores para a variável:

$$\eta = \frac{C1-C2}{C3+C4}, \tag{3.3}$$

onde

$$C1 = c_\tau(0,\tau+1)c_\tau(\tau,\tau), \tag{3.4}$$

$$C2 = c_\tau(0,\tau)c_\tau(\tau,\tau+1), \tag{3.5}$$

$$C3 = c_\tau(0,\tau+1)(c_\tau(\tau,\tau) - c_\tau(\tau,\tau+1)), \tag{3.6}$$

$$C4 = c_\tau(0,\tau)(c_\tau(\tau+1,\tau+1) - c_\tau(\tau,\tau+1)). \tag{3.7}$$

A variável η, por fim, é utilizada no cálculo da correlação fracionária:

$$r(\tau+\eta) = \frac{(1-\eta)c_\tau(0,\tau) + \eta c_\tau(0,\tau+1)}{\sqrt{c_\tau(0,0)((1-\eta)^2 c_\tau(\tau,\tau) + 2\eta(1-\eta)c_\tau(\tau,\tau+1) + \eta^2 c_\tau(\tau+1,\tau+1))}} \tag{3.8}$$

Dessa forma, dentre os dois valores $(\tau+\eta)$ obtidos, aquele que resulta no maior valor para (3.8) é escolhido como um candidato a *pitch* fracionário, denominado P2.

Na Equação (3.3), os valores podem eventualmente ficar fora do intervalo $[0, 1]$, mas são limitados entre $[-1, 2]$. O *pitch* fracionário está limitado entre 20 e 160 [9].

O valor de P2 é também utilizado posteriormente na determinação da intensidade de vozeamento, no cálculo do ganho e no próprio cálculo do *pitch* final.

A etapa seguinte utiliza novamente a Equação (3.1) para realizar uma busca levando em conta 5 amostras em torno do candidato P2, $[P2-5, P2+5]$, seguido de um refinamento do *pitch* para o valor que maximiza a função de correlação normalizada no intervalo considerado. O sinal utilizado desta vez é o sinal resíduo obtido por um filtro passa-baixas com freqüência de corte de 1 kHz. O procedimento resulta no candidato a *pitch* P3.

O cálculo final do *pitch* é então realizado de acordo com o algoritmo apresentado na Figura 3.3. Na etapa de dupla verificação do *pitch*, um procedimento analisa e corrige caso estejam sendo considerados múltiplos do *pitch* atual. O valor de D_{th} é um limiar utilizado nessa etapa.

Como pode ser percebido pela Figura 3.3, o procedimento final eventualmente pode produzir novos valores para P3 e r(P3). Além disso, nota-se que o tempo total de processamento realizado na etapa de cálculo do *pitch* depende dos parâmetros de um dado quadro. Eventualmente, o algoritmo pode decidir ignorar todos os cálculos realizados e optar por um *pitch* médio, P_{avg}, atualizado ao longo do processo de codificação dos diversos quadros.

3.3 Modificações Propostas

Nesta seção são apresentadas as modificações que resultaram na versão simplificada do algoritmo. Essas alterações foram implementadas na etapa do cálculo da primeira estimativa inteira do *pitch*, ou seja, no cálculo de P1.

Inicialmente é apresentada a forma "padrão", conforme descrita na norma [9], mas com uma implementação que explora um padrão de recursividade, e que resulta em uma implementação mais eficiente em termos de redução do número de operações.

Em seguida, é apresentada a explanação da versão modificada, onde se fez uso de decimações.

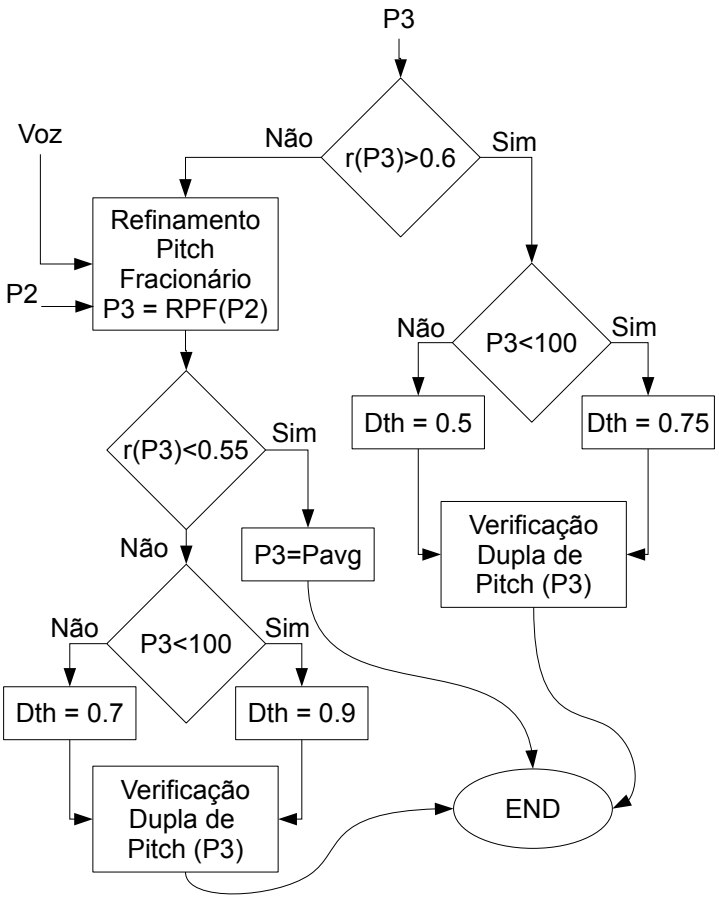

Figura 3.3: Algoritmo original para determinação do *pitch* final.

3.3.1 Computação Padrão com Uso de Recorrência

Na prática, o cálculo para obtenção de P1 por meio da Equação (3.1) pode ser feito para $r^2(\tau)$, evitando assim a realização de uma raiz quadrada em cada etapa do cálculo:

$$r^2(\tau) = \frac{c_\tau(0,\tau)c_\tau(0,\tau)}{c_\tau(0,0)c_\tau(\tau,\tau)}. \tag{3.9}$$

Dessa forma, a raiz quadrada pode ser efetuada apenas uma vez, após determinado o valor de τ que maximiza $r^2(\tau)$ no intervalo [40,160]. Sendo assim, para a busca do primeiro τ são realizadas 319 operações matemáticas (160 multiplicações e 159 adições) para cada um dos três diferentes $c_\tau[.]$ em (3.9), resultando em 960 operações para determinar o máximo valor de $r^2(\tau)$.

Além disso, podem ser utilizadas algumas relações de recorrência, permitindo o uso de valores prévios de $c_\tau(0,0)$ ou de $c_\tau(\tau,\tau)$ para determinar $c_{\tau-1}(0,0)$ ou $c_{\tau-1}(\tau-1,\tau-1)$, respectivamente, assumindo que os cálculos estão sendo realizados do maior valor de τ para o menor, ou seja, $\tau = 160, 159, \ldots, 41, 40$.

O desenvolvimento a seguir visa apresentar essas fórmulas de recorrência. Tem-se que o numerador e os fatores do denominador de $r(\tau)$ são:

$$c_\tau(0,\tau) = c[0,\tau,\tau] = \sum_{k=-\lfloor\tau/2\rfloor-80}^{-\lfloor\tau/2\rfloor+79} s_k s_{k+\tau}, \tag{3.10}$$

$$c_\tau(0,0) = c[0,0,\tau] = \sum_{k=-\lfloor\tau/2\rfloor-80}^{-\lfloor\tau/2\rfloor+79} s_k^2, \tag{3.11}$$

$$c_\tau(\tau,\tau) = c[\tau,\tau,\tau] = \sum_{k=-\lfloor\tau/2\rfloor-80}^{-\lfloor\tau/2\rfloor+79} s_{k+\tau}^2. \tag{3.12}$$

Os valores de τ variam de 40 a 160 (ou de 160 a 40). Para cada τ são calculados $c[0,\tau,\tau]$, $c[0,0,\tau]$ e $c[\tau,\tau,\tau]$, utilizados na obtenção de $r(\tau)$. Por exemplo, para $\tau = 160$: $\lfloor\tau/2\rfloor = \lfloor 160/2\rfloor = 80$, tem-se

$$c_{160}(0,160) = c[0,160,160] = \sum_{k=-160}^{-1} s_k s_{k+160}, \tag{3.13}$$

$$c_{160}(0,0) = c[0,0,160] = \sum_{k=-160}^{-1} s_k^2, \tag{3.14}$$

$$c_{160}(160,160) = c[160,160,160] = \sum_{k=-160}^{-1} s_{k+160}^2 = \sum_{k=0}^{159} s_k^2. \tag{3.15}$$

Para $\tau = 159$: $\lfloor \tau/2 \rfloor = \lfloor 159/2 \rfloor = 79$, tem-se

$$c[0, 159, 159] = \sum_{k=-159}^{0} s_k s_{k+159}, \tag{3.16}$$

$$c[0, 0, 159] = \sum_{k=-159}^{0} s_k^2 = s_0^2 - s_{-160}^2 + \sum_{k=-160}^{-1} s_k^2 = s_0^2 - s_{-160}^2 + c[0, 0, 160], \tag{3.17}$$

$$c[159, 159, 159] = \sum_{k=-159}^{0} s_{k+159}^2 = \sum_{k=0}^{159} s_k^2 = c[160, 160, 160]. \tag{3.18}$$

Para $\tau = 158$: $\lfloor \tau/2 \rfloor = \lfloor 158/2 \rfloor = 79$, tem-se

$$c[0, 158, 158] = \sum_{k=-159}^{0} s_k s_{k+158}, \tag{3.19}$$

$$c[0, 0, 158] = \sum_{k=-159}^{0} s_k^2 = c[0, 0, 159], \tag{3.20}$$

$$\begin{aligned} c[158, 158, 158] \;&=\; \sum_{k=-159}^{0} s_{k+158}^2 = \sum_{k=-1}^{158} s_k^2 = \\ & s_{-1}^2 - s_{159}^2 + \sum_{k=-1}^{158} s_k^2 = s_{-1}^2 - s_{159}^2 + c[159, 159, 159]. \end{aligned} \tag{3.21}$$

A representação gráfica do procedimento pode ser visto na Figura 3.4.
Esses resultados mostram que existe uma relação de recorrência entre os $c[.]$.
Considerando valores ímpares de τ, têm-se:

$$c_\tau(0,0) = c_{\tau+1}(0,0) + s^2_{-\lfloor \tau/2 \rfloor + 79} - s^2_{-\lfloor \tau/2 \rfloor - 81}, \tag{3.22}$$

$$c_{\tau-1}(0,0) = c_\tau(0,0), \tag{3.23}$$

e

$$c_\tau(\tau,\tau) = c_{\tau+1}(\tau+1,\tau+1), \tag{3.24}$$

$$c_{\tau-1}(\tau-1,\tau-1) = c_\tau(\tau,\tau) + s^2_{-\lfloor \tau/2 \rfloor + 79 + \tau} - s^2_{-\lfloor \tau/2 \rfloor - 81 + \tau}. \tag{3.25}$$

Explorando essas recorrências, realizam-se apenas 319 operações para calcular $c[0, \tau, \tau]$, 4 operações para corrigir $c[0, 0, \tau - 1]$ ou $c[\tau - 1, \tau - 1, \tau - 1]$ (note que, para

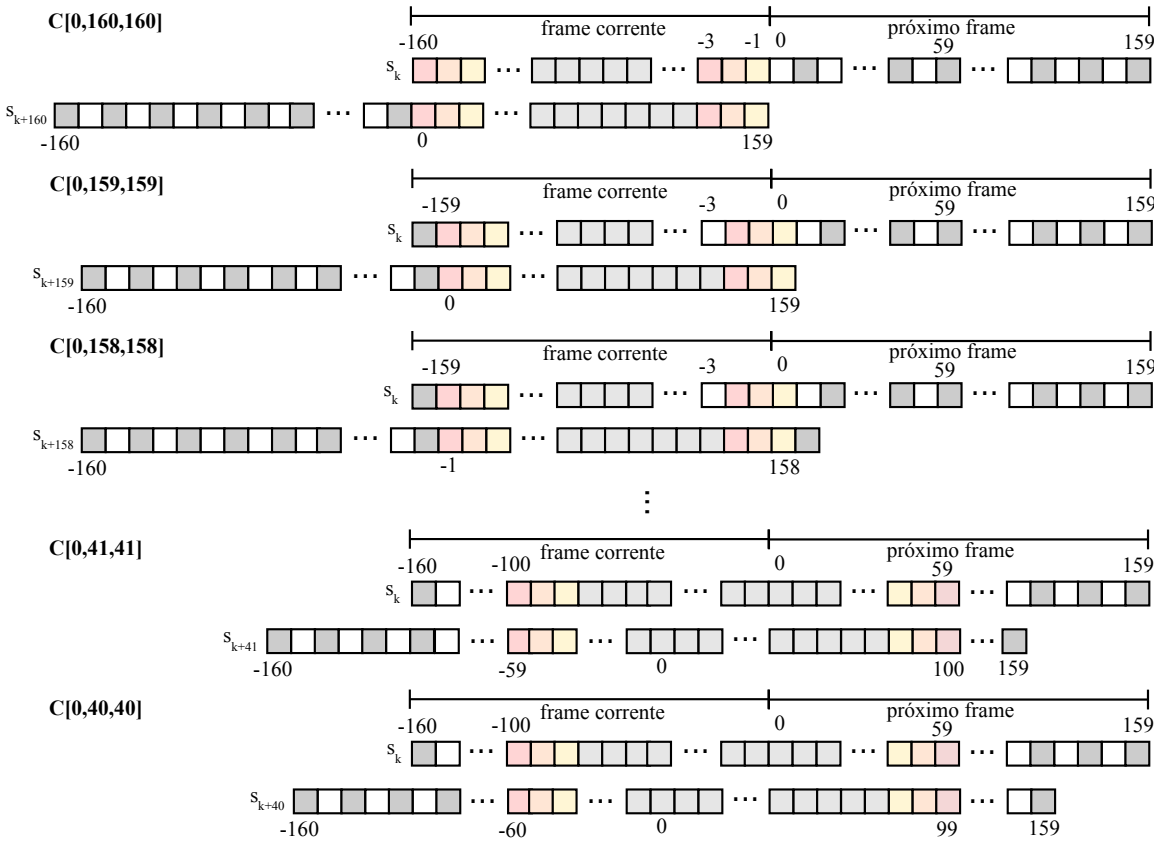

Figura 3.4: Correlação.

um dado τ, a correção é necessária apenas para um dos $c[.]$ do denominador de (3.9)) e 3 operações para calcular $r^2(\tau)$, correspondendo a um total de 326 operações.

Dessa forma, para encontrar P1 em um dado quadro, o algoritmo com uso de recorrência realiza $960 + (160 - 40) * 326 = 40.080$ operações, contra as $961 * 121 = 116.281$ operações efetuadas diretamente pela Equação (3.1).

3.3.2 Versão Modificada com Uso de Decimações

Considerando que o cálculo do *pitch* possui diversas etapas de refinamento até chegar ao seu valor final, a realização de decimação em k (na Equação (3.2)) e/ou em τ (no intervalo [40,160]) na busca de P1 poderia reduzir substancialmente a complexidade computacional, seguindo a estratégia empregada em [27] para o Codec ITU-T G.729. A amplitude das decimações levaria a uma relação de custo benefício entre tempo de processamento e qualidade da voz reconstruída.

Sendo D_k e D_τ os fatores de decimação de k e τ, respectivamente, a Equação (3.26) fornece a quantidade aproximada do número de operações requeridas na etapa de estimação do parâmetro P1:

$$N(D_k, D_\tau) = 2(320 - D_k) + \lfloor \frac{121}{D_\tau} \rfloor (2\lfloor \frac{160}{D_k} \rfloor + 2) + (\lfloor \frac{121}{D_\tau} \rfloor - 1)(4D_t + (D_k - 1)\lceil \frac{D_\tau}{2} \rceil)). \tag{3.26}$$

A diminuição da complexidade advinda do uso das decimações é avaliada nas seções seguintes.

3.4 Avaliação Objetiva e Seleção de Amostra

Os fatores de decimação D_k e D_τ introduzidos na seção anterior foram implementados na versão de referência do MELP, variando cada fator de 1 a 10, resultando em 100 diferentes configurações. Cada configuração corresponde a uma complexidade computacional distinta, estimada pela Equação (3.26) e uma qualidade de voz particular, estimada nessa etapa pelo algoritmo PESQ (*Perceptual Evaluation of Speech Quality*) [24].

Para avaliar a relação entre a complexidade computacional e a qualidade de voz, quando usados os fatores de decimação D_k e D_τ, foi utilizada uma base de dados consistindo em 40 sinais de falantes da Língua Inglesa americana (20 homens e 20 mulheres), tendo cada amostra uma duração média de 4,15 s, sendo realizada a codificação/decodificação por meio de cada uma das 100 diferentes versões do MELP. Para maiores detalhes relativos à base de vozes utilizada, consultar o Anexo A deste trabalho.

O resultado desse procedimento pode ser visto na Figura 3.5, onde são apresentados os valores PESQ-MOS e suas correspondentes complexidades computacionais estimadas. A figura sugere que, para um dado nível de decimação, é possível obter um sinal com o mesmo nível de qualidade da voz do original, mas com o número de operações significativamente reduzido. Os pontos preenchidos e conectados do gráfico definem um feixe côncavo que representa a melhor relação custo-benefício entre boa qualidade de voz x complexidade.

A redução da complexidade computacional pode ser traduzida, em termos mais práticos, em menor tempo de processamento. Nesse contexto, foi realizada uma medição no tempo de processamento (*time profiling*) por meio da ferramenta *DTrace*, disponível nos sistemas operacionais baseados em Unix (Solaris, FreeBSD e MAC OS). Esse procedimento, realizado nos pontos constituintes do feixe côncavo da Figura 3.5, resultou nos pontos apresentados na Tabela 3.1 e na Figura 3.6. Um maior detalhamento sobre essa ferramenta computacional utilizada pode ser obtida no Anexo B.

Esses resultados sugerem que o uso da decimação dos parâmetros na etapa de estimação de P1 resultou em um significativo impacto no tempo total de proces-

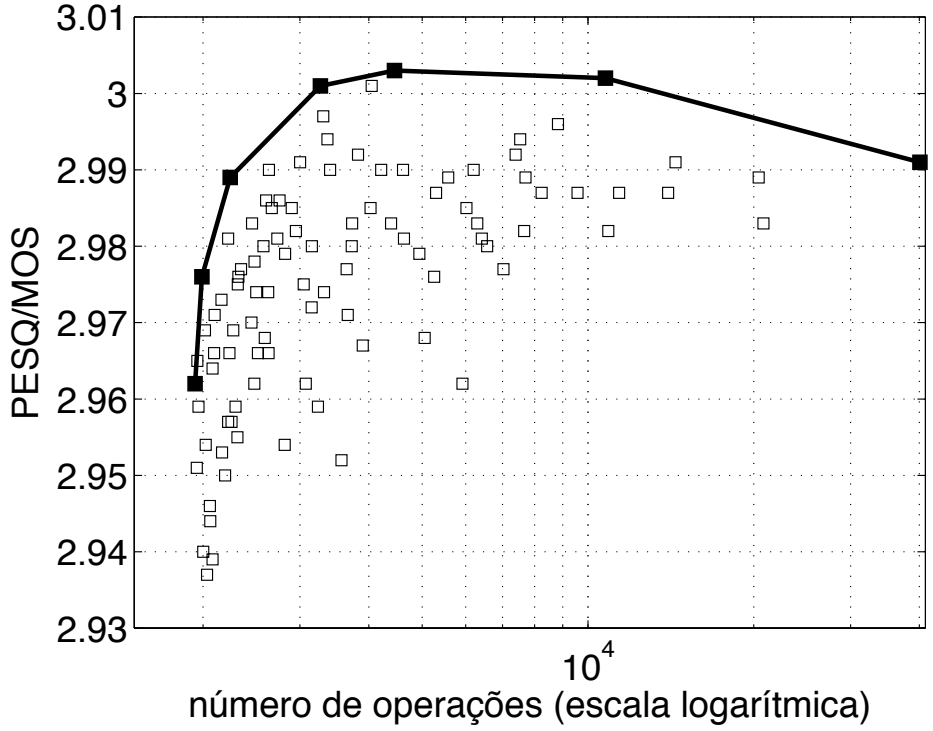

Figura 3.5: Número de operações × valores PESQ-MOS das versões do MELP modificadas.

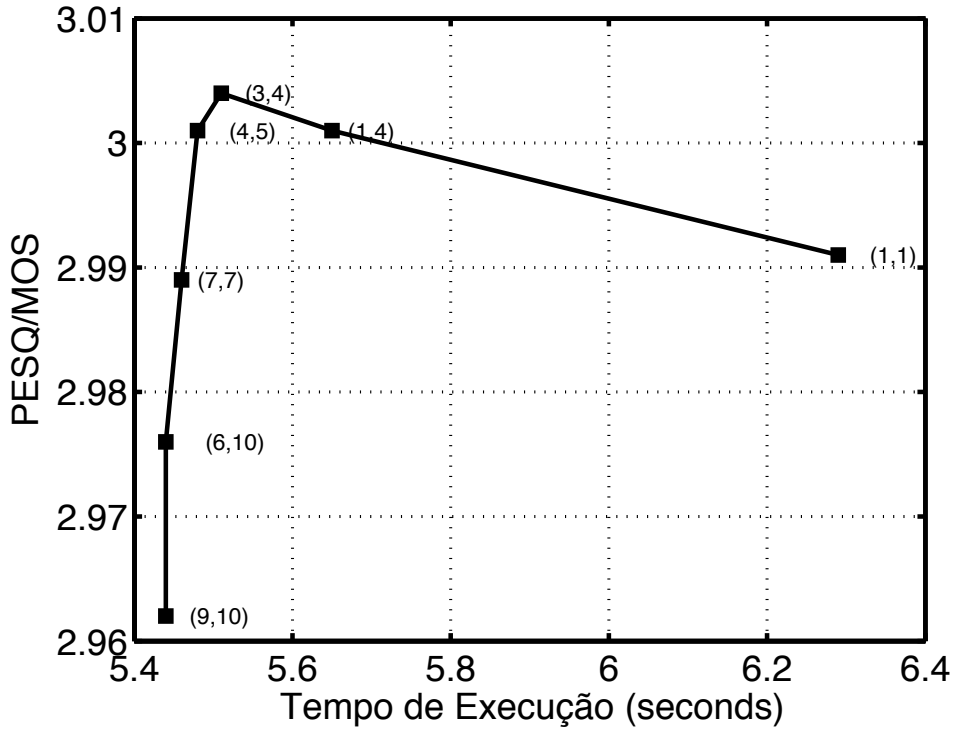

Figura 3.6: Tempo de execução × PESQ-MOS para o feixe côncavo da Fig. 3.5.

Tabela 3.1: *Tempo de execução × PESQ-MOS para o feixe côncavo da Fig. 3.5.*

D_k	D_τ	Tempo (s)	PESQ-MOS	% redução
9	10	5.44 ± 0.02	2.962	13.5
6	10	5.46 ± 0.02	2.976	13.2
7	7	5.45 ± 0.03	2.989	13.4
4	5	5.48 ± 0.02	3.001	12.9
3	4	5.51 ± 0.02	3.003	12.4
1	4	5.65 ± 0.03	3.002	10.1
1	1	6.29 ± 0.02	2.991	0.0

samento, permitindo uma redução de 13% e mantendo-se uma qualidade de voz aceitável.

A tabela mostra que a partir de 13% o ganho no tempo de execução é pouco expressivo, estando os valores dentro do intervalo de incerteza.

3.5 Avaliação Subjetiva

De forma a confirmar, em termos de qualidade do sinal decodificado, os resultados obtidos por meio dos procedimentos descritos nas seções anteriores, foram realizadas avaliações subjetivas de uma versão modificada selecionada.

Para realizar a avaliação subjetiva e confirmar a hipótese levantada na seção anterior, foram considerados dois tipos de testes, o *Absolute Category Rating* (ACR) e o *Comparison Category Rating* (CCR), descritos na recomendação P.800 [25].

Em ambos os testes foram utilizadas uma versão MELP original e sua versão decimada, com D_k e D_τ iguais a 4 e 5, respectivamente.

Uma vez que os testes subjetivos devem ser realizados com uma base de vozes na mesma língua dos ouvintes, no caso o português do Brasil, foram utilizadas frases nesse idioma. O mesmo banco de vozes foi utilizado nos dois testes subjetivos.

3.5.1 *Absolute Category Rating* - ACR

Neste teste, 32 sinais de fala, usando a Língua Portuguesa do Brasil, foram codificados e decodificados, por meio da versão original do MELP e de sua versão modificada. Estes sinais tinham duração entre 2 e 3 segundos e contemplavam diferentes falantes masculinos e femininos. Cada um dos sinais foi avaliado por 20 ouvintes "destreinados", que atribuíam uma nota MOS (*mean opinion score*) de acordo com a escala numérica apresentada na Tabela 3.2.

A média das notas para cada um dos 32 sinais é vista na Tabela 3.3. O valor oficial esperado para o MOS do MELP é em torno de 3,2. Os resultados da Tabela 3.3

Tabela 3.2: *Escala MOS*

Escala MOS	Significado
5	Excelente
4	Bom
3	Regular
2	Ruim
1	Pobre

mostram uma pequena polarização, que tem sua explicação mais provável na dificuldade de se estabelecer uma referência por meio de valores de MOS conhecidos. Tal polarização, até mesmo mais acentuada, pode ser observada também em outros trabalhos semelhantes [28].

Tabela 3.3: *MOS*

MELP original	3,51 $\pm$ 0,31
MELP modificado	3,50 $\pm$ 0,32

3.5.2 *Comparison Category Rating* - CCR

Nesta avaliação, os ouvintes realizam dois julgamentos por meio da resposta às perguntas: "Qual amostra apresenta melhor qualidade?" e "Quão melhor?". As respostas são quantificadas por meio da escala apresentada na Tabela 3.4. Os resultados obtidos para este teste estão apresentados no histograma da Fig. 3.7, onde as notas positivas favorecem o codificador modificado.

Tabela 3.4: *Escala de teste CCR.*

Escala CCR	Significado
3	Muito melhor
2	Melhor
1	Pouco melhor
0	Igual
-1	Pouco pior
-2	Pior
-3	Muito pior

De modo geral, os resultados de ambos os testes descritos nesta seção mostram uma certa equivalência na qualidade da voz resultante gerada pelas versões original

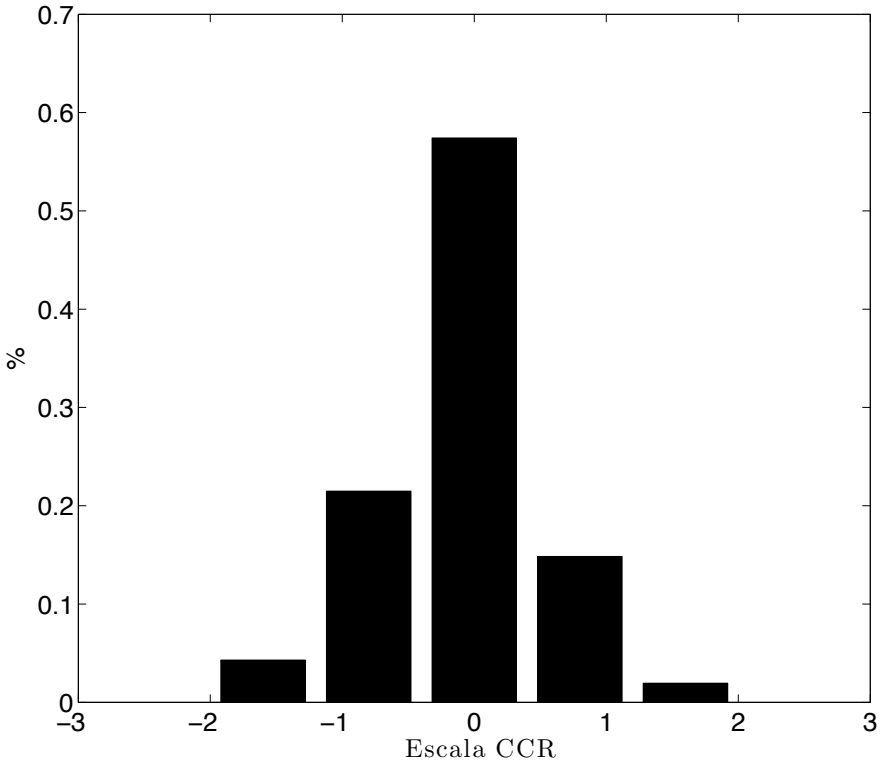

Figura 3.7: Histograma do teste CCR.

e simplificada do algoritmo MELP. Deste fato, pode-se concluir que a aceleração obtida para o algoritmo de codificação, como descrito neste capítulo, não provocou uma queda significativa de qualidade do sinal reconstruído.

3.6 Conclusão

Os testes subjetivos apresentados por este trabalho fornecem subsídios que permitem concluir que o esquema de aceleração proposto permitiu reduzir o tempo de execução em aproximadamente 13%, sem afetar significativamente a qualidade da voz reconstruída.

Em termos práticos, os testes subjetivos do tipo ACR resultaram em valores essencialmente equivalentes. Essa conclusão foi complementada e ainda reforçada pelos resultados dos testes tipo CCR, que indicaram que os ouvintes em geral perceberam os sinais originais e modificados como iguais.

Ressalta-se a importância da ferramenta PESQ como guia para seleção de uma versão para realização dos testes subjetivos, tendo em vista que seria impraticável a realização de testes subjetivos com as 100 versões de MELP obtidas.

Percebe-se que considerar mais valores de k e/ou τ não necessariamente resulta

em uma melhor qualidade de voz, sendo possível obter a mesma qualidade com menos esforço computacional, valendo-se do fato de que refinamentos adicionais do *pitch* são considerados pelo algoritmo do MELP em etapas posteriores ao cálculo de P1.

Os resultados obtidos neste capítulo foram publicados em [29].

Capítulo 4

Estudos Complementares

4.1 Introdução

Este capítulo apresenta alguns estudos complementares que visam observar a influência de algumas decisões de projeto do MELP, como escolha da ordem dos coeficientes LP, influência do uso desses coeficientes antes ou após a quantização vetorial (ou seja, com os coeficientes reconstruídos) e a influência da modelagem de pulso (cálculo dos parâmetros magnitudes de Fourier) na qualidade da voz. São apresentadas também algumas considerações finais referentes ao uso da ferramenta PESQ, decorrentes dos estudos dos limiares associados aos parâmetros intensidade de vozeamento de baixa freqüência e *peakness*.

A Seção 4.2 apresenta o estudo sobre a influência da ordem dos coeficientes LP na qualidade da voz sintetizada, avaliada por meio da ferramenta PESQ. O MELP utiliza exatamente o mesmo número de coeficientes do LPC10, ou seja, 10 coeficientes.

A Seção 4.3 analisa a influência, no cálculo das magnitudes de Fourier, do uso dos coeficientes LP reconstruídos a partir dos índices obtidos por meio da quantização vetorial, conforme estabelece a norma MIL-STD-3005, ou obtidos diretamente do algoritmo de predição linear. Em seguida, foi avaliada a influência da etapa correspondente à modelagem do trem de pulsos como um todo, fazendo uma comparação entre o MELP original e uma versão modificada, na qual a etapa de cálculo das magnitudes de Fourier foi eliminada.

Em seguida, na Seção 4.4, é avaliada a influência da utilização dos coeficientes LP antes ou depois da quantização vetorial (coeficientes reconstruídos) no cálculo sinal de resíduo utilizado na etapa final do cálculo do *pitch*, a exemplo do que foi realizado na primeira parte da seção anterior.

A Seção 4.5 apresenta relevantes considerações sobre a ferramenta PESQ, decorrentes dos estudos realizados para a avaliação dos limiares associados ao vs_1

(parâmetro de vozeamento de baixa freqüência) e ao parâmetro auxiliar *peakness*, ambos envolvido na definição da classificação dos quadros vozeados e não-vozeados. Essa seção mostra que o processo de automação proporcionado pela ferramenta PESQ deve ser sempre acompanhado da avaliação subjetiva da qualidade de voz reconstruída, de forma a garantir a consistência dos resultados.

Por fim, na Seção 4.6, são apresentadas as conclusões do capítulo.

4.2 Influência da Ordem dos Coeficientes na Qualidade da Voz Sintetizada

Da mesma forma que o padrão LPC10, o codificador de voz MELP trabalha com coeficientes LP de ordem 10. A presente seção visa avaliar essa escolha de ordem com base na avaliação objetiva da qualidade da voz sintetizada.

Nesse sentido, foram realizadas simulações variando a ordem dos coeficientes LP de 1 a 20. Para cada ordem, a codificação e a decodificação foram aplicadas a 60 sinais de voz, sendo calculado o valor PESQ/MOS de cada uma. Em seguida foi realizada uma média dos valores PESQ/MOS obtidos para cada ordem de coeficientes LP.

É importante destacar que os parâmetros obtidos no codificador foram transmitidos diretamente ao decodificador, sem que houvesse a etapa de quantização vetorial, pois do contrário seria necessário o desenvolvimento de pelo menos 18 quantizadores vetoriais específicos. Dessa forma, os resultados não consideram efeitos advindos da quantização vetorial, que, em princípio, resultariam em valores ligeiramente menores do PESQ/MOS.

Os resultados da simulação são apresentados na Figura 4.1.

Conforme esperado, a figura mostra uma grande degradação da qualidade da voz para ordens mais baixas. No início da curva, os valores PESQ/MOS aumentam a medida que cresce a ordem dos coeficientes LP.

Observa-se que o crescimento da qualidade da voz com a ordem é não-linear e que a partir da ordem 10 ocorre uma certa estabilização do PESQ/MOS. As variações dos valores PESQ/MOS a partir da ordem 10 são supostamente flutuações decorrentes da própria ferramenta PESQ.

Os resultados sugerem que a ordem 10 apresenta o melhor compromisso entre qualidade e complexidade, o que justificaria, portanto, sua utilização no MELP.

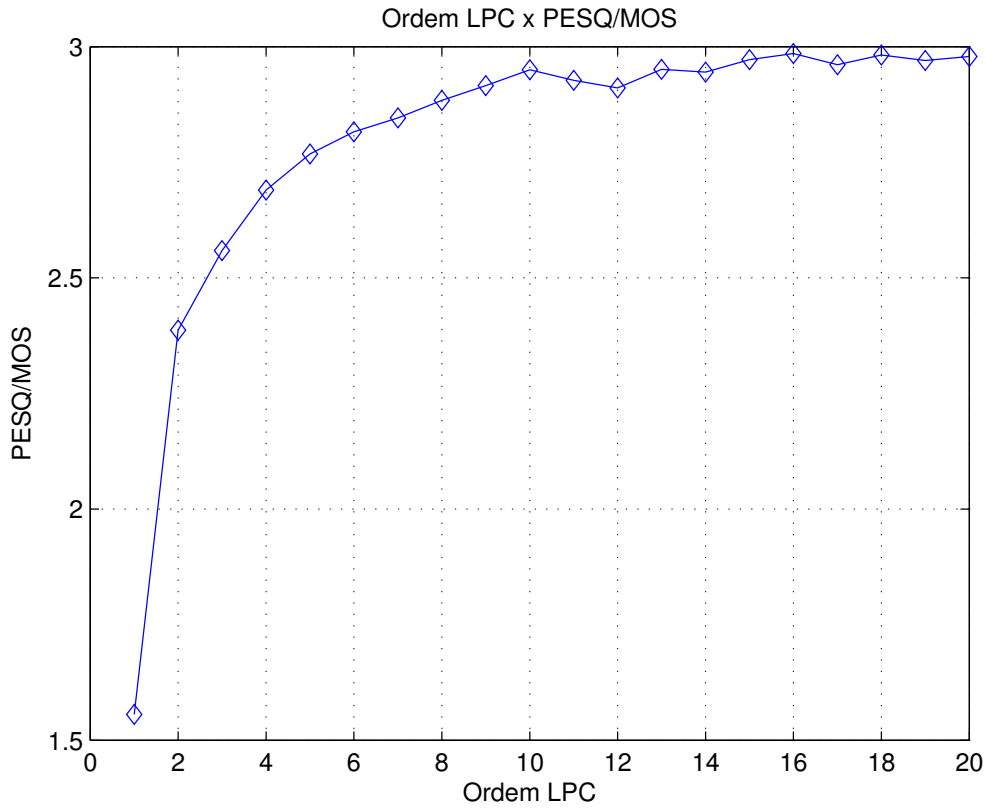

Figura 4.1: Relação entre ordem LPC e PESQ/MOS.

4.3 Estudo da Etapa Envolvendo a Modelagem de Fourier

Nesta seção é analisada a influência da utilização da quantização vetorial dos parâmetros associados à etapa de modelagem de Fourier, bem como da relevância da própria etapa como um todo.

De acordo com o procedimento descrito na MIL-STD-3005 [9], os coeficientes LP utilizados na etapa do cálculo das magnitudes de Fourier devem ser aqueles obtidos após a quantização vetorial. Isso significa incluir ainda no codificador uma etapa correspondente à reconstrução dos coeficiente LP a partir dos índices obtidos na quantização e, consequentemente, implica maior esforço computacional.

Inicialmente foi analisada a influência desse procedimento na qualidade da voz sintetizada. Nesta seção, a versão original corresponde àquela que segue a risca a recomendação da norma supracitada, ou seja, utiliza os coeficientes LP reconstruídos a partir dos índices da quantização vetorial. A versão modificada corresponde àquela onde os coeficientes LP utilizados são provenientes diretamente do algoritmo de predição linear, sem quantização. Busca-se, dessa forma, alguma forma de simplificação do algoritmo, por meio da eliminação da etapa envolvendo a reconstrução

dos coeficientes LP, caso essa alteração não implique degradação significativa da qualidade da voz reconstruída.

As duas implementações foram avaliadas por meio da ferramenta PESQ/MOS e os resultados encontram-se na Tabela 4.1. Os valores da tabela correspondem ao valor PESQ/MOS médio de 240 sinais de voz.

Tabela 4.1: *Influência da quantização dos coeficientes LP na modelagem de Fourier (240 sinais de voz).*

Versão	PESQ-MOS
MELP original	2,990 ± 0,030
MELP modificado	2,967 ± 0,029

Os resultados sugerem que a utilização dos coeficientes LP reconstruídos a partir dos índices da quantização vetorial, conforme previsto na norma, confere uma ligeira melhora na qualidade da voz sintetizada. Contudo, essa melhora, se considerada isoladamente, pode não ser significativa o suficiente para justificar o emprego do procedimento acima, por envolver mais etapas de cálculo e, consequentemente, implicar maior custo computacional.

Em seguida, adotou-se um passo mais radical, que foi desconsiderar totalmente a etapa de modelagem de Fourier e avaliar o impacto dessa decisão em termos de qualidade da voz. O resultado desse procedimento, aplicado a 240 sinais de voz, está apresentado na Tabela 4.2. A versão modificada corresponde ao algoritmo MELP sem a etapa de modelagem de Fourier.

Tabela 4.2: *Influência da modelagem de Fourier no MELP (240 sinais de voz).*

Versão	PESQ-MOS
MELP original	2,990 ± 0,030
MELP modificado	2,909 ± 0,027

Os resultados obtidos sugerem que os parâmetros "magnitude de Fourier", e o procedimento associado a eles, de fato melhoram a qualidade da voz, mas produzem, em termos de qualidade de voz, uma melhora apenas sutil.

Em consonância com os resultados apresentados, [28] conclui que magnitudes de Fourier seriam pouco significativas para a qualidade da voz e poderiam ser removidas, permitindo uma economia de 8 bits em cada quadro. No caso citado, a simplificação resultou em uma taxa de aproximadamente 1,92 kbp, onde foi realizada também uma simplificação adicional, onde os quatro bits associados aos parâmetros de intensidade de vozeamento foram reduzidos para 2 bits apenas.

4.4 Influência da Quantização Vetorial dos Coeficientes LP na Etapa do *Pitch*

Conforme visto no Capítulo 3 (Figura 3.1), a etapa final do cálculo do *pitch* utiliza o sinal resíduo obtido por meio dos coeficientes LP após filtragem passa-baixas de 1 kHz na função de correlação normalizada para obtenção do valor $P3$.

Ao contrário do procedimento estabelecido para determinação das magnitudes de Fourier, a norma MIL-STD-3005 [9] não prevê o uso do coeficientes obtidos após a quantização vetorial, como usualmente ocorre nos codificadores de voz.

A utilização dos coeficientes LP reconstruídos a partir dos índices resultantes da quantização vetorial, caso resultasse em alguma melhoria, não implicaria aumento da complexidade computacional, uma vez que, considerando a implementação original, essa reconstrução já é realizada na etapa de cálculo das magnitudes de Fourier.

Neste estudo, a versão original foi convencionada aquela que está de acordo com a norma MIL-STD-3005, ou seja, utiliza o sinal resíduo dos coeficientes LP antes da quantização vetorial. A versão modificada corresponde àquela com sinal de resíduo obtido por meio dos coeficientes LP reconstruídos, após a quantização vetorial.

As duas implementações foram avaliadas utilizando a ferramenta PESQ/MOS e os resultados encontram-se na Tabela 4.3. Os valores da tabela correspondem ao valor PESQ/MOS médio de 240 sinais de voz.

Tabela 4.3: *Influência da quantização dos coeficientes LP na determinação do pitch (240 sinais de voz).*

Versão	PESQ-MOS
MELP original	2,990 ± 0,030
MELP modificado	2,988 ± 0,028

Os resultados obtidos sugerem que a escolha dos coeficentes LP obtidos antes ou após a quantização vetorial não traz qualquer benefício em termos de qualidade da voz.

4.5 Estudo do Valor Ótimo para os Parâmetros vs_1 e *Peakness* e Considerações Relativas à Ferramenta PESQ

Diversas etapas do algoritmo MELP se baseiam em limiares que controlam as decisões realizadas pelo codificador. A hipótese inicial desta seção seria que o PESQ

poderia ser utilizado para definir os valores ótimos para os limiares, por meio de uma busca computacional exaustiva que maximizariam o valor PESQ. A idéia seria aplicar de forma ampla a metodologia implementada com sucesso no Capítulo 3, mas desta vez aplicada ao estudo dos valores dos limiares empregados MELP. Idealmente, o resultado final, composto por uma versão ou uma família de versões, seria então avaliado subjetivamente para ratificação.

Nesse sentido, o primeiro parâmetro escolhido foi o parâmetro vs_1. Junto com o parâmetro *peakness*, ele define se um determinado quadro é vozeado ou não-vozeado e também influencia na determinação da janela utilizada no cálculo dos ganhos. Nesses dois casos, o valor padrão do limiar de vs_1 é 0,6.

Para a realização desse estudo, o código do MELP foi alterado de forma a tornar o limiar de vs_1 um parâmetro variável, passado ao arquivo executável do codificador. O limiar variou no intervalo de 0 a 1 em passos de 0,02.

O resultado da simulação está apresentado na Figura 4.2. Cada ponto no gráfico corresponde à média do valor PESQ/MOS de 60 amostra de voz.

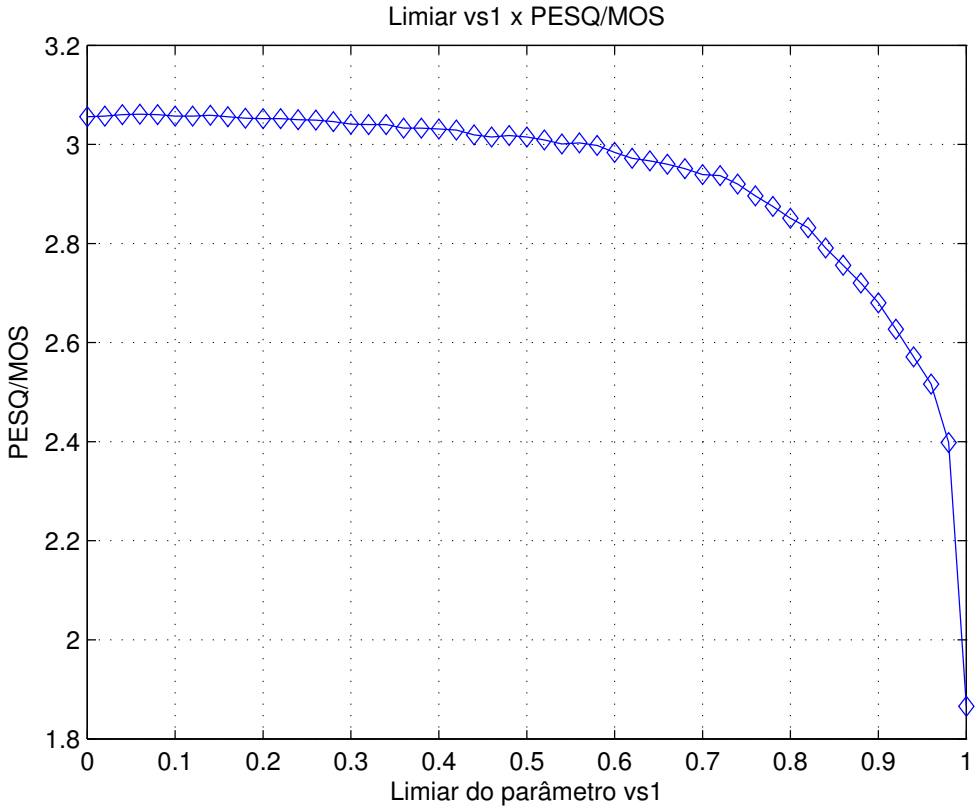

Figura 4.2: Relação entre limiar vs1 e PESQ/MOS.

De acordo com a figura, pelo menos em termos de avaliação objetiva realizada pela ferramenta PESQ, a melhor qualidade seria obtida com valores pequenos do limiar associado a vs_1.

Esse resultado é contra-intuitivo, pois sugere que o valor ótimo para vs_1 ocorreria em zero. Nessa situação, todos os quadros são considerados vozeados. Essa condição, contudo, não corresponde à realidade, pois ao se escutar dois sinais de voz decodificados na condição original e na condição com limiar próximo de zero, percebe-se que, embora seja perfeitamente inteligível a condição onde vs_1 está próximo de zero, a versão com o limiar na condição original apresenta qualidade superior. Os sinais de voz com limiar próximo de zero apresentam uma distorção perceptível, não captada pela ferramenta PESQ.

Resultado semelhante foi observado na tentativa de definição do parâmetro "ótimo"para o limiar do *peakness*.

O *peakness* é um parâmetro auxiliar utilizado no ajuste de valores das intensidades de vozeamento (vs_1, vs_2 e vs_3), e embora não seja enviado no quadro, influencia na decisão referente à classificação vozeado e não-vozeado.

Conforme visto no Capítulo 2, dois limiares estão associados ao *peakness*. O primeiro deles atua na alteração do valor de intensidade de vozeamento de baixa freqüência (vs_1), e o segundo na correção dos parâmetros de intensidade de vozeamento vs_2 e vs_3. Tem-se que:

- Se $p > 1{,}34$ (limiar 1 padrão), então $vs_1 = 1$; e
- Se $p > 1{,}60$ (limiar 2 padrão), então $vs_2 = 1$ e $vs_3 = 1$.

onde p corresponde ao valor *peakness* e vs_i ($i = 1$, 2, e 3) os valores de intensidade de vozeamento.

O estudo se concentrou no primeiro limiar, tendo em vista sua maior importância no algoritmo, haja vista que ele pode alterar a classificação vozeado ou não-vozeado definida previamente.

O código do MELP foi alterado de forma a tornar o limiar considerado um parâmetro arbitrário passado ao arquivo executável. A partir daí, o limiar variou no intervalo de 0 a 2, com passo de 0,02. Dessa forma, seria possível observar a variação da qualidade da voz por meio da ferramenta PESQ e ratificar a escolha original do limiar no algoritmo ou definir um novo valor "ótimo".

Os resultados obtidos são apresentados na Figura 4.3. Cada ponto no gráfico corresponde à média do valor PESQ/MOS de 240 sinais de voz.

Na base considerada, compreendendo 240 sinais de voz compostas em igual proporção por falas masculinas e femininas do inglês americano, os valores de *peakness* calculados estão no intervalo de 1,16 a 3,80, apresentando média 1,36 e desvio padrão 0,13.

Na figura, o patamar observado para valores menores do que 1,18 resulta do fato de que o menor valor do *peakness* para a base considerada é de 1,16, implicando que todos valores de limiar abaixo desse valor necessariamente devem resultam no

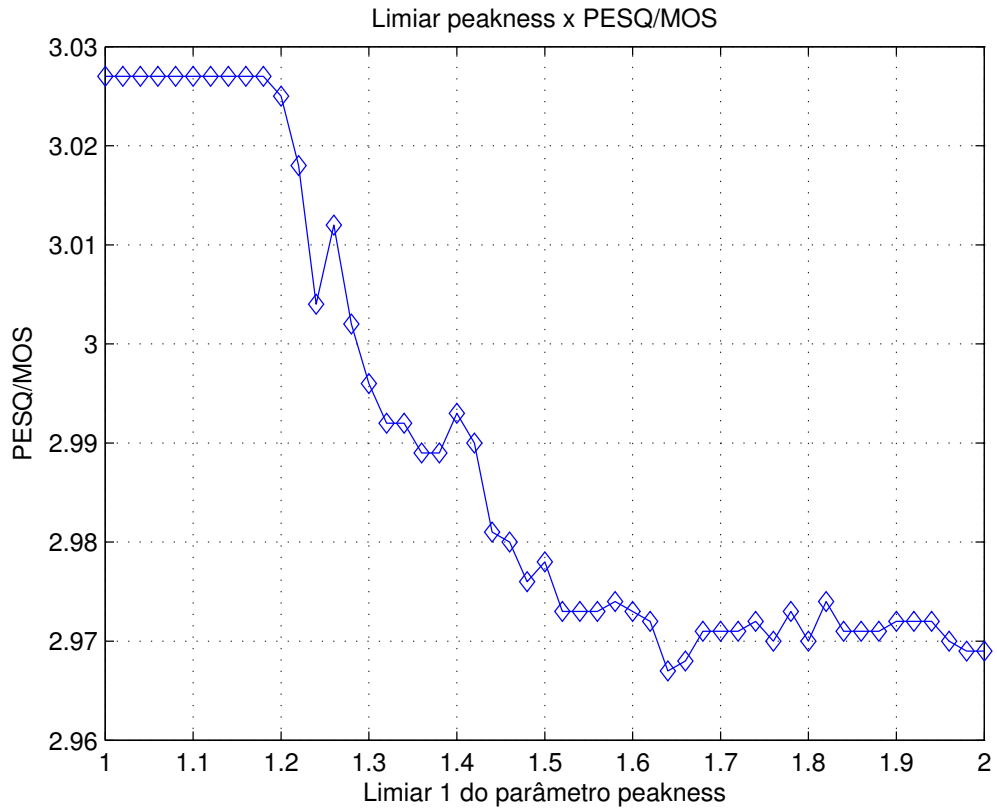

Figura 4.3: Relação entre o limiar 1 do *peakness* e o PESQ/MOS.

mesmo valor PESQ/MOS, que por sinal são os maiores observados. Essa condição é equivalente ao caso anterior com limiar de vs_1 igual a zero, pois também resulta em todos os quadros classificados como vozeados.

Embora a ferramenta seja de grande utilidade para orientar algumas decisões, conforme mostrado com sucesso principalmente no Capítulo 3, deve-se ter em mente que o processo de automatização proporcionado pelo PESQ demanda o acompanhamento de avaliações subjetivas, mesmo que informais, de forma a garantir a coerência dos resultados. A decisão final cabe sempre ao "ouvido humano", que afinal é o verdadeiro "usuário"e razão de ser dos codificadores de voz.

A ferramenta PESQ pode desempenhar um papel importante no desenvolvimento ou na melhoria de um codificador de voz, permitindo selecionar um conjunto limitado de versões potencialmente promissoras. Contudo, esse automatismo proporcionado pelo PESQ não dispensa a avaliação cuidadosa dos resultados pelo especialista.

4.6 Conclusão

Os resultados das Seções 4.2, 4.3 e 4.4 não representam constatações absolutas, mas devem ser vistos como indícios acerca do comportamento dos parâmetros ou proce-

dimentos, avaliados em um primeiro momento por avaliações subjetivas informais.

O resultado obtido no estudo da influência da ordem dos coeficientes LP, apresentado na Seção 4.2, está em perfeita conformidade com a escolha consagrada da ordem 10 para os codificadores paramétricos baseados no LPC. Ainda assim, seria pertinente realizar uma avaliação subjetiva do codificador de voz utilizando ordem 9 para os coeficientes LP, uma vez que o valor PESQ/MOS nesse caso está dentro da margem de flutuação do PESQ, conforme pode ser visto na Figura 4.1.

No caso da influência do modelamento do trem de pulsos, mostrado na Seção 4.3, o resultado está coerente com o apresentado em [28], implicando a possibilidade de descartar a etapa associada ao cálculo das magnitudes de Fourier com uma degradação muito pequena na qualidade de voz. Dessa forma, seria possível se beneficiar de uma redução da taxa ou, simplesmente, da complexidade computacional.

Os estudos apresentados nas Seções 4.3 e 4.4 sugerem que a utilização dos coeficientes LP obtidos diretamente da análise de predição linear, ou reconstruídos a partir dos coeficientes quantizados, produzem resultados equivalente em termos de qualidade da voz.

As principais conclusões referentes à Seção 4.5 decorrem da constatação na prática de peculiaridades da ferramenta PESQ, observadas durante o estudo de alguns limiares utilizados pelo MELP. Dos resultados decorre que é imprescindível que uma avaliação subjetiva informal seja sempre empregada em conjunto com os processos de automação proporcionados pelo PESQ, de forma a garantir a coerência dos resultados.

Capítulo 5

Conclusões

5.1 Contribuições do trabalho

Este trabalho apresentou um estudo do codificador de voz MELP, conforme descrito em [9]. A grande importância do MELP está em seu amplo emprego como codificador de baixa taxa, principalmente no âmbito das comunicações militares, que decorre da boa razão qualidade/taxa e do seu desempenho satisfatório em ambientes ruidosos.

O Capítulo 2 apresentou uma visão geral do MELP, destacando as razões pelas quais ele resulta em um melhor modelo para representação da voz em relação ao padrão LPC10. São descritos os procedimentos utilizados para codificação e decodificação da voz. Esse capítulo apresenta o método de avaliação objetiva da voz, realizada por meio da ferramenta PESQ [24], que foi muito utilizada nos Capítulos 3 e 4, e também os métodos subjetivos de avaliação de voz, realizada pelos testes dos tipos ACR e CCR [25], utilizados no Capítulo 3.

O Capítulo 3 apresenta uma descrição detalhada do procedimento de cálculo do *pitch* empregado no MELP, seguido de uma proposta de simplificação que resultou em uma redução de aproximadamente 13% da voz reconstruída, sem afetar significativamente a qualidade da voz [29], obtida por meio do uso de decimações dos parâmetros k e τ. Ressalta-se a importância da ferramenta PESQ como guia para seleção de uma versão para realização dos testes subjetivos, tendo em vista que seria impraticável a realização de testes subjetivos com as 100 versões de MELP obtidas. Nesse capítulo são também apresentados os resultados dos testes subjetivos dos tipos ACR e CCR, utilizados para ratificar os resultados obtidos.

O Capítulo 4 apresenta um estudo da influência na qualidade de voz: da ordem dos coeficientes LP, da etapa modelagem de pulso (cálculo das magnitudes de Fourier), e da quantização na etapa de cálculo final do *pitch* e de modelagem de pulso. O estudo da ordem dos coeficientes LP sugere que a ordem 10 resulta no melhor

compromisso ordem versus qualidade de voz. O estudo da modelagem de pulso, associado aos parâmetros magnitudes de Fourier, mostrou que a etapa poderia ser suprimida, implicando uma pequena perda na qualidade da voz reconstruída. Nas etapas de cálculo do *pitch* e na modelagem de Fourier, o estudo da influência da utilização dos coeficientes LP obtidos diretamente da análise de predição linear, ou reconstruídos após a quantização vetorial, sugere que ambos os casos são equivalentes em termos de qualidade da voz reconstruída. Por fim, o capítulo apresenta algumas considerações finais acerca da ferramenta PESQ, decorrentes dos estudos dos limiares associados aos parâmetros vs_1 e *peakness*. Dos resultados decorre que é imprescindível que uma avaliação subjetiva informal seja sempre empregada em conjunto com os processos de automação proporcionados pelo PESQ, de forma a garantir a correta aplicação da ferramenta.

5.2 Proposta para trabalhos futuros

A seguir estão algumas possíveis continuações deste trabalho:

- realizar uma avaliação subjetiva formal dos resultados do Capítulo 4;
- avaliar o resultado conjunto das alterações proposta neste trabalho a partir de uma nova versão do codificador;
- avaliar a influência de outros procedimentos descritos no algoritmo do MELP, como, por exemplo, o mecanismo de cálculo do ganho;
- avaliar a influência do tamanho das janelas de amostragem no desempenho do codificador;
- avaliar o desempenho do codificador com a substituição de etapas originais por procedimentos oriundos de outros padrões. Por exemplo, avaliar o desempenho do MELP com a substituição do quantizador vetorial original por de outros padrões; e
- estudar a implementação do MELP nas versões em 1200 bps e 600 bps, por meio do desenvolvimento de um quantizador vetorial que opere em blocos de quadros.

Referências Bibliográficas

[1] ISODE LTD. "STANAG 5066: The Standard for Data Application over HF Radio." Feb. 2008. Disponível em: <http://www.isode.com/whitepapers/stanag-5066.html>.

[2] COUTOLLEAU, M., VILA, P., MEREL, D., et al. "New Studies about a high data rete HF parallel modem." In: *Military Communication Conference, 1998. MILCOM 98. Proceedings, IEEE*, v. 2, pp. 381 – 385. IEEE, 1998.

[3] NATO. "STANAG 4198: "Parameter and Coding Characteristics that must be commin to assure interoperability of 2400 bps linear predictive encoded digital speech"." Feb. 1984.

[4] U.S. DOD. "Federal Standard 1015: "Analog to Digital Conversion of Voice by 2400 bit/second Linear Predictive Coding"." Nov. 1984.

[5] HARRIS CORP PRESS RELEASE. "Harris Corporation Receives $5.3 Million Order from Brazilian Ministry of Defence for Falcon III RF-7800V VHF Tactical Radios." July 2011. Disponível em: <http://www.harris.com/view_pressrelease.asp?act=lookup&pr_id=3272>.

[6] ONU. "MINUSTAH: United Nations Stabilization Mission in Haiti." 2011. Disponível em: <https://www.un.org/en/peacekeeping/missions/minustah/>.

[7] ONU. "Brazilian Flagship for UNIFIL Maritime Task Force." Nov. 2011. Disponível em: <"http://unifil.unmissions.org/Default.aspx?tabid=1499&ctl=Details&mid=3103&ItemID=15777>.

[8] BENTKOVSKI KOBI, HALALY ISRAEL, D. N. A. "MELP - Mixed Excitation Linear Prediction Speech Coder." Jan. 2011. Disponível em: <http://health.tau.ac.il/Communication%20Disorders/noam/speech/melp/index.htm>.

[9] U.S. DOD. "MIL-STD 3005: Analog-to-Digital Conversion of Voice by 2,400 Bit/Second Mixed Excitation Linear Prediction (MELP)." Dec. 1999.

[10] MCCREE, A., BARNWELL, T.P., I. "A new mixed excitation LPC vocoder." In: *Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991 International Conference on*, pp. 593 –596 vol.1, Washington, DC, USA, Apr . IEEE Computer Society. ISBN: 0-7803-0003-3.

[11] MCCREE, A. V. *A new LPC vocoder model for low bit rate speech coding.* Tese de Doutorado, Atlanta, GA, USA, 1992. Director-Barnwell,III, Thomas P.

[12] HAAGEN, J., NIELSEN, H., HANSEN, S. D. "A 2.4 kbps high-quality speech coder." In: *Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991 International Conference on*, pp. 589 –592 vol.1, Apr . ISBN: 0-7803-0003-3.

[13] KEMP, D. P., COLLURA, J. S., TREMAIN, T. E. "Multi-frame coding of LPC parameters at 600-800 bps." In: *Acoustics, Speech, and Signal Processing, 1991. ICASSP-91., 1991 International Conference on*, pp. 609 –612 vol. 1, Washington, DC, USA, Apr . IEEE Computer Society. ISBN: 0-7803-0003-3.

[14] MCCREE, A., BARNWELL, T.P., I. "Military Communications Conference, 1992. MILCOM '92, Conference Record. 'Communications - Fusing Command, Control and Intelligence'., IEEE." In: *A 2400 bps mixed excitation LPC vocoder*, pp. 381 –384 vol.1, Oct. 1992.

[15] TREMAIN, T., KOHLER, M., CHAMPION, T. "Philosophy and goals of the DoD 2400 bps vocoder selection process." In: *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings, 1996 IEEE International Conference on*, v. 2, pp. 1137 –1140 vol. 2, May 1996.

[16] MCCREE, A., TRUONG, K., GEORGE, E. B., et al. "A 2.4 kbit/s MELP coder candidate for the new U.S. Federal Standard." In: *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings, 1996 IEEE International Conference on*, pp. 200 –203 vol. 1, May . ISBN: 0-7803-3192-3. doi: http://dx.doi.org/10.1109/ICASSP.1996.540325.

[17] KOHLER, M. "A Comparison of the New 2400 Bps MELP Federal Standard with Other Standard Coders." In: *ICASSP '97: Proceedings of the 1997 IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP '97 -vol. 2*, p. 1587, Washington, DC, USA. IEEE Computer Society. ISBN: 0-8186-7919-0.

[18] SUPPLEE, L., COHN, R., COLLURA, J., et al. "MELP: The New Federal Standard at 2400 bps." In: *Acoustics, Speech, and Signal Processing, 1997. ICASSP-97., 1997 IEEE International Conference on*, pp. 1591 – 1594 vol.2, Apr . ISBN: 0-8186-7919-0.

[19] M. D. STREET, J. S. C. *Interoperable Voice Communications: Test and Selection of STANAG 4591.* Relatório técnico, NATO NC3A, Oct. 2001.

[20] WANG, T., KOISHIDA, K., CUPERMAN, V., et al. "A 1200 bps speech coder based on MELP." In: *Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE International Conference*, pp. 1375 –1378 vol.3.

[21] CHAMBERLAIN, M. "A 600 bps MELP vocoder for use on HF channels." In: *Military Communications Conference, 2001. MILCOM 2001. Communications for Network-Centric Operations: Creating the Information Force. IEEE*, pp. 447 – 453 vol.1.

[22] CHU, W. C. *Speech Coding Algorithms: Foundation and Evolution of Standardized Coders.* New York, NY, USA, John Wiley & Sons, Inc., 2003. ISBN: 0471373125.

[23] KINDOZ, A., KONDOZ, A. M. *Digital Speech; Coding for Low Bit Rate Communication Systems.* New York, NY, USA, John Wiley & Sons, Inc., 1994. ISBN: 0471950645.

[24] ITU-T. "Rec. P.862: Perceptual evaluation of speech quality (PESQ): An objective method for end-to-end speech quality assessment of narrow-band telephone networks and speech codecs." Feb 2001.

[25] ITU-T. "Rec. P.800 : Methods for subjective determination of transmission quality." Aug 1996.

[26] DE M. PREGO, T. *Aceleração dos Codificadores de Fala G.729 e G.729A.* Tese de Mestrado, COPPE/UFRJ, 2009.

[27] DE M. PREGO, T., NETTO, S. L. "Algoritmo de Busca Eficiente no Dicionário Adaptativo para o Codec ITU-T G.729." In: *XXVI Simpósio Brasileiro de Telecomunicações - SBrT'08*, Set 2008.

[28] TAN, E. C., TEO, T. T. "Real-Time Implementation of MELP Vocoder." *Jounal of The Institution of Engineers, Singapore*, v. 44, n. 3, pp. 38 – 58, 2004.

[29] VENTURA, M. M., NETTO, S. L. "Avaliação Subjetiva de Versão Acelerada do Codificador de Voz MELP." In: Sociedade Brasileira de Telecomunicações (Ed.), *Anais do Simpósio Brasileiro de Telecomunicações*, pp. 1–5, Out. 2011.

[30] OSR. "Open Speech Repository." Disponível em: <http://www.voiptroubleshooter.com/open_speech/>.

[31] SUN MICROSYSTEMS, INC. "Solaris Dynamic Tracing Guide." 2008. Disponível em: <http://docs.oracle.com/cd/E19253-01/817-6223/book-info/index.html>.

[32] APPLE INC. "Instruments User Guide." Disponível em: <https://developer.apple.com/library/mac/documentation/developertools/conceptual/InstrumentsUserGuide/Introduction/Introduction.html>.

[33] SOX. "Sound eXchange." Disponível em: <http://sox.sourceforge.net/>.

[34] PESQ. "Implementação em linguagem C do PESQ." Disponível em: <http://www.itu.int/rec/T-REC-P.862/>.

Apêndice A

Banco de Vozes

Para a realização dos testes objetivos efetuados por meio PESQ, foram utilizados sinais de voz no formato WAV que foram obtidos do *Open Speech Repository* [30]. O projeto OSR disponibiliza arquivos de voz em diversas línguas (inglês americano e britânico, francês, hindi e chinês mandarim) para uso em testes com voz sobre IP ou outras aplicações envolvendo fala. Esses arquivos estão disponíveis sem restrição para uso, publicação, cópia ou distribuição, desde que seja dado crédito ao projeto. Dessa base, foram utilizadas as amostras de voz recortadas manualmente para realização dos estudos em [27].

A intenção de utilizar uma base de vozes na língua inglesa americana teve como propósito principal poder fazer comparações com outros estudos presentes na literatura, evitando-se eventuais discrepâncias decorrentes de alguma especificidade da língua portuguesa. Em todos os testes foram utilizadas amostras de voz masculinas e femininas em igual proporção, de forma a obter uma representação mais ampla de falantes, como, por exemplo, uma maior variação de valores de *pitch*.

Os arquivos WAV correspondem a sinais codificados com PCM 16-bit, com freqüência de amostragem de 8 kHz.

Apêndice B

Ferramentas Computacionais

B.1 DTrace - *Dynamic Tracing*

A ferramenta computacional *DTrace* foi desenvolvida pela *Sun Microsystems* para o sistema operacional *Unix Solaris*, visando a análise e a resolução de problemas de desempenho em tempo real do sistema operacional e aplicativos. De forma a obter as informações de interesse, a ferramenta *DTrace* altera dinamicamente o *Kernel* do sistema e processos do usuário [31].

Posteriormente, a ferramenta foi disponibilizada por meio de uma licença CDDL (*Common Development and Distribuition License*) e agora está implementada em outros sistemas operacionais baseados em *Unix* (*Mac OS X* e *FreeBSD*).

A utilização do *DTrace* se dá por meio da linguagem *script* D, similar em termos de sintaxe à linguagem C. No caso do sistema operacional *Mac OS X*, o uso do *DTrace* pode ser feito por meio do aplicativo *Instruments*, que é integrado ao ambiente de programação *XCode* da *Apple*. Nesse caso, a interação com a ferramenta é facilitada por meio de uma GUI (*Graphical User Interface*) intuitiva, descartando a necessidade de elaboração dos *scripts* em linguagem D [32].

Neste trabalho a ferramenta *DTrace* foi utilizada para realizar o "*Time Profiling*", ou seja, a medida do tempo de execução da operação de codificação e decodificação, que serviu como medida de redução da complexidade computacional.

B.2 SoX - *Sound eXchange*

O SoX é um programa que roda em linha de comando e permite converter arquivos de áudio em diversos formatos, ou ainda aplicar diversos efeitos a esses arquivos. Ele está disponível em diversos sistemas operacionais (Mac OS X, Linux, Windows, etc), e pode ser encontrado em [33].

Essa ferramenta foi utilizada em todos os *scripts* em *Bourne Shell* desenvolvidos

nesta dissertação para transformar os arquivos WAV da base da dados em arquivos RAW utilizados pelo codificador MELP. Após decodificação, a ferramenta era novamente utilizada para converter os arquivos RAW em WAV para uso na avaliação objetiva por meio do PESQ.

B.3 PESQ - *Perceptual Evaluation of Speech Quality*

O algoritmo PESQ se propõe a estimar a nota *Mean Opinion Score* (MOS) de um sinal de voz com banda estreita, conforme descrito na Seção 2.6 do Capítulo 2. O PESQ realiza uma avaliação objetiva da qualidade da voz e está definido na recomendação P.862 da União Internacional de Telecomunicações (*International Telecommunication Union* - ITU) [24].

O programa PESQ, implementado em linguagem C, pode ser obtido no sítio da ITU em [34] e compilado para a plataforma (sistema operacional) de interesse por meio, por exemplo, do compilador GCC (*GNU Compiler Collection*).