

UNIVERSIDADE FEDERAL DE SANTA CATARINA
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

CODIFICADORES PARAMÉTRICOS DE SINAIS DE VOZ
COM EXCITAÇÃO CODIFICADA E FILTRAGEM ADAPTÁVEL

DISSERTAÇÃO SUBMETIDA À UNIVERSIDADE FEDERAL DE SANTA CATARINA
PARA OBTENÇÃO DE GRAU DE MESTRE EM ENGENHARIA ELÉTRICA

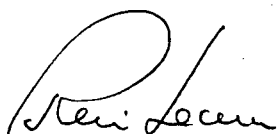
IRVANDO LUIZ SPERANZINI

FLORIANÓPOLIS, OUTUBRO - 1991.

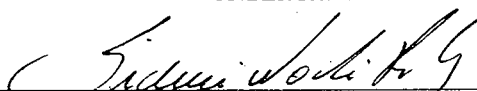
CODIFICADORES PARAMÉTRICOS DE SINAIS DE VOZ
COM EXCITAÇÃO CODIFICADA E FILTRAGEM ADAPTÁVEL

CANDIDATO: IRVANDO LUIZ SPERANZINI

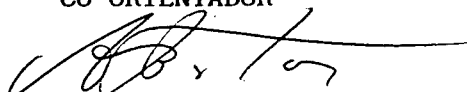
Esta dissertação foi julgada para obtenção do título - **MESTRE EM ENGENHARIA** - Especialidade Engenharia Elétrica aprovada em sua forma final pelo Curso de Pós-Graduação.



Prof. Rui Seara, Dr. Ing.
ORIENTADOR

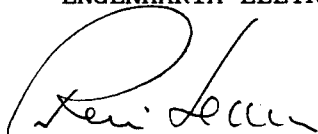


Prof. Sidnei Noceti Filho, D.Sc.
CO-ORIENTADOR

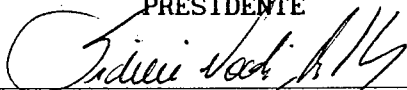


Prof. João Pedro Assumpção Bastos, Dr. D'Etat
COORDENADOR DO CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

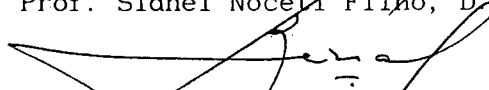
BANCA EXAMINADORA:



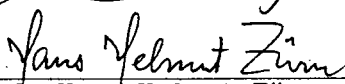
Prof. Rui Seara, Dr. Ing.
PRESIDENTE



Prof. Sidnei Noceti Filho, D. Sc.



Prof. José Carlos Moreira Bermudez, Ph.D.



Prof. Hans Helmut Zürn, Ph.D.



Prof. Giles Lothar Istre, Ph.D.

Aos meus pais.

AGRADECIMENTOS

Aos Professores Rui Seara e Sidnei Noceti Filho pela valiosa ajuda prestada na orientação deste trabalho.

Aos meus colegas, pelo incentivo e sugestões, particularmente a Ricardo Felipe Custódio e Orli Inácio Franzói.

Aos funcionários do Departamento de Engenharia Elétrica.

À FINEP pelo suporte financeiro e material.

Ao LABORATÓRIO DE INSTRUMENTAÇÃO ELETRÔNICA (L I N S E) do Departamento de Engenharia Elétrica da UFSC onde foi desenvolvido este trabalho.

Í N D I C E

LISTA DE FIGURAS.....	viii
LISTA DE TABELAS.....	x
SIMBOLOGIA.....	xi
RESUMO.....	xiii
ABSTRACT.....	xiv
<u>CAPÍTULO 1</u> - INTRODUÇÃO.....	01
<u>CAPÍTULO 2</u> - CODIFICADORES PARAMÉTRICOS - REVISÃO BIBLIOGRÁFICA.....	10
2.1 - Introdução.....	10
2.2 - Codificação Linear Preditiva (LPC).....	10
2.2.1 - Preditor de Curto Atraso (PCA) e Preditor de Longo Atraso (PLA).....	12
2.3 - Codificadores Paramétricos.....	14
2.3.1 - Codificador PELPC.....	15
2.3.2 - Codificador MPELPC.....	16
2.3.3 - Codificador SEV.....	18
2.3.4 - Codificador CELP.....	19
2.4 - Conclusão.....	27

<u>CAPÍTULO 3 - CODIFICAÇÃO DA EXCITAÇÃO COM FILTRAGEM ADAPTÁVEL.....</u>	28
3.1 - Introdução.....	28
3.2 - Requisitos para Avaliação e Comparação do Desempenho da Estrutura Proposta.....	28
3.2.1 - Critério de Medição da Qualidade de Voz Sintetizada.....	29
3.2.2 - Padrão de Comparação dos Resultados Obtidos nas Simulações.....	31
3.3 - CELP com Filtragem Adaptável.....	35
3.3.1 - Resultados das Simulações do CELP Utilizando Filtragem Adaptável.....	37
3.4 - Codificação da Excitação com Filtragem Adaptável.....	39
3.5 - Determinação da Excitação do CEA.....	44
3.5.1 - Desempenho da Estrutura CEA para as Diversas Ordens do Filtro $L(z)$	49
3.6 - Método Alternativo para Determinação dos Coeficientes do Filtro $L(z)$..	56
3.7 - Conclusão.....	61
<u>CAPÍTULO 4 - Conclusões.....</u>	63
BIBLIOGRAFIAS REFERENCIADAS.....	67
BIBLIOGRAFIAS CONSULTADAS.....	75
<u>APÊNDICE A - PROGRAMA PARA DETERMINAÇÃO DOS PARÂMETROS DO CODIFICADOR CELP CONVENCIONAL.....</u>	83

<u>APÊNDICE B</u> - PROGRAMA PARA DETERMINAÇÃO DOS PARÂMETROS DO CODIFICADOR PROPOSTO.....	91
<u>APÊNDICE C</u> - UNIDADE DE AQUISIÇÃO DE DADOS.....	100
C.1 - Introdução.....	100
C.2 - Descrição do Sistema.....	100
C.3 - Especificação do Circuito.....	104
<u>APÊNDICE D</u> - SEGMENTO DE VOZ UTILIZADO NAS SIMULAÇÕES.....	107

LISTA DE FIGURAS

1.1 - A Fonética na comunicação.....	05
1.2 - Representações dos sinais de voz.....	06
1.3 - Qualidade subjetiva de voz para as diversas técnicas de codificação em função da taxa de transmissão em kbits/s.....	07
2.1 - Modelo de produção de voz para codificadores paramétricos.....	11
2.2 - Modelo para a síntese de voz usando preditores de curto e longo atraso.....	13
2.3 - Diagrama em blocos do codificador PELPC.....	15
2.4 - Diagrama em blocos da etapa de análise do codificador multi-pulso.....	17
2.5 - Diagrama em blocos do modelo de síntese do codificador SEV usando 2 preditores de longo atraso (PLA).....	18
2.6 - Distribuição das "codewords" no "codebook".....	19
2.7 - Diagrama em blocos ilustrando os procedimentos para procura da melhor "codeword".....	21
2.8 - Diagrama em blocos ilustrando o procedimento da formulação da análise por síntese para os codificadores CELP.....	22
2.9 - Diagrama em blocos do procedimento de escolha da melhor "codeword".....	26
3.1 - Diagrama em blocos da estrutura CEA para determinação do filtro $L(z)$..	35
3.2 - Diagrama em blocos para determinação da melhor "codeword" da estrutura CEA.....	40
3.3 - Diagrama em blocos para determinação do filtro $L(z)$	41
3.4 - Diagrama em blocos para a síntese de voz utilizando a estrutura CEA...	42

3.5 - Gráfico do comportamento da RSR_S para os diversos valores de f_i e f_f adotados na equação (3.14).....	47
3.6 - Espectro do sinal pseudo-sinusoidal (equação (3.14)).....	49
3.7 - Diagrama em blocos para determinação dos coeficientes do filtro $L(z)$..	56
3.8 - Diagrama da Fig. 3.7 rearranjado.....	57
C.1 - Entradas analógicas.....	101
C.2 - Saídas analógicas.....	102
C.3 - Relógio interno.....	103
C.4 - "Trigger" externo.....	103

LISTA DE TABELAS

3.1 - Resultados das simulações da estrutura CELP convencional.....	33
3.2 - Resultados das simulações da estrutura CELP com filtragem adaptável...	38
3.3 - Resultados das simulações da estrutura CEA.....	43
3.4 - Resultados da estrutura CEA para diversas ordens do filtro $L(z)$, para "codewords" de 10, 20 e 40 amostras.....	50
3.5 - Percentual de quadros sem necessidade do filtro $L(z)$, para "codewords" de 10, 20 e 40 amostras.....	52
3.6 - Resultados das simulações da estrutura CEA dispensando-se o uso do filtro $L(z)$	54
3.7 - Resultados das simulações do codificador CEA quantizado sem o uso do filtro $L(z)$	55

S I M B O L O G I A

- ADPCM - Modulação por Codificação de Pulsos Diferencial Adaptável
- A/D - Conversor Analógico Digital
- AR - "AutoRegressive"
- ARMA - "AutoRegressive Moving Average"
- CCITT - "International Telegraph and Telephone Consultative Committee"
- CEA - Codificação da Excitação com filtragem Adaptável
- CELP - "Code-Excited Linear Predictive"
- DM - Modulação Delta
- DPCM - Modulação por Codificação de Pulsos Diferencial
- DSP - "Digital Signal Processor"
- D/A - Conversor Digital Analógico
- EMO_Q - Erro Médio Quadrático do quadro de voz
- EMO_S - Erro Médio Quadrático do segmento de voz
- f_a - Frequência de Amostragem
- f_i - Frequência inicial
- f_f - Frequência final
- L - Número de "codewords" do "codebook"
- LMS - "Least Mean Square"
- LPC - Codificação Linear Preditiva
- MOS - "Mean Opinion Score"
- MPELPC - "Multi-Pulse Excited Linear Predictive Coder"
- $O(.)$ - Ordem de complexidade
- OD - Ordem do preditor

- PCA - Preditor de Curto Atraso
- PELPC - "Pitch-Excited Linear Predictive Coder"
- PCM - Modulação por Codificação de Pulsos
- PITCH - "Altura" do Sinal de Voz
- PLA - Preditor de Longo Atraso
- RPELPC - "Regular-Pulse Excited Linear Predictive Coder"
- RSR_Q - Relação Sinal-Ruído do quadro de voz
- RSR_S - Relação Sinal-Ruído do segmento de voz
- SEV - "Self Excited Vocoder"
- T_p - Tempo de processamento

RESUMO

Dentre os tipos de codificadores paramétricos aplicáveis a sinais de voz, o CELP ("Code Excited Linear Predictive") é um dos codificadores que possibilitam uma implementação em tempo real a baixas taxas. No entanto, um dos problemas deste codificador é o elevado esforço computacional requerido na etapa de seleção da melhor seqüência de excitação. Muitos pesquisadores propuseram formas de aceleração deste procedimento, alguns impondo restrições à estrutura, outros propondo uma mudança no procedimento de análise, passando do domínio do tempo para outros domínios.

Neste trabalho é proposta uma nova estrutura de codificador de voz baseada na estrutura CELP, a qual apresenta um reduzido esforço computacional para a seleção da melhor seqüência de excitação, quando comparado ao conseguido pelas estruturas CELP convencionais.

São apresentados resultados computacionais que confirmam o bom desempenho da estrutura proposta, tanto em termos de complexidade computacional requerida quanto em termos de qualidade objetiva (Relação Sinal-Ruído) do sinal de voz sintetizado.

A B S T R A C T

Among the several parametric coders applied to voice signals, CELP (Code Excited Linear Predictive) is one that enables a real time implementation at low bit rates. However, the main problem presented by this coder is the large computational effort required to select the best excitation sequence. Many researchers have proposed different means of accelerating this procedure, some by imposing constraints on the structure, others by changing analysis procedures from the time domain to other domains.

In this work, a new voice coder structure based on CELP structure is proposed, which requires a reduced computational effort for the selection of the best excitation sequence, when compared to those realized using conventional CELP structures.

Computational results which demonstrate the performance of the structure proposed are presented. A good performance is obtained both in terms of computational effort and in terms of objective quality (Signal-to-Noise Ratio) of the synthesized voice signal.

CAPÍTULO 1

INTRODUÇÃO

Dentre as primeiras tentativas de sintetizar a voz humana, pode-se citar a máquina falante de Wolfgang von Kempelen. Ele concebeu, por volta de 1791, uma máquina totalmente mecânica que era capaz de gerar todas as vogais e a maior parte das consoantes dos idiomas alemão e latim, como também elaborar e produzir frases curtas nestas línguas [1]. Graham Bell, conhecendo a máquina de von Kempelen, também construiu a sua. Esta, por sua vez, reproduzia frases curtas com alto grau de inteligibilidade, despertando a atenção de vários outros pesquisadores, principalmente aqueles ligados ao campo da fonética experimental [2].

Porém, os estudos da voz humana só tiveram realmente um grande avanço a partir da publicação da teoria de Fourier em 1822. Os primeiros trabalhos científicos em decomposição de sinais de voz foram apresentados por Bertil Malberg antes da Primeira Guerra Mundial [3].

Durante a Segunda Guerra Mundial, os instrumentos de medição de espectro de sinais foram aperfeiçoados, e em 1947, apareceram os primeiros espectrógrafos. Estes instrumentos foram evoluindo e muitos passaram a ser comercializados, tal como o espectrógrafo fabricado pela "Kay Electric". Tais instrumentos utilizavam um papel especial em torno de um tambor, sobre o qual era gravado por meio de uma agulha a energia espectral correspondente aos sinais a serem analisados [3].

Em 1936, foi concebida a primeira máquina destinada a realizar a

codificação da voz com o objetivo de reduzir a taxa para fins de transmissão. Seu inventor, Homer Dudley, também foi o primeiro a utilizar o termo "vocoder", aglutinação do termo "voice coder", do inglês codificador de voz. Contudo, os primeiros "vocoders" comerciais só foram efetivamente utilizados na década de 60 [4].

Com o passar dos anos, diversos foram os sistemas desenvolvidos (eletromecânicos, eletrônicos (analógicos), etc). O advento da microeletrônica aliada ao grande avanço da tecnologia de fabricação de circuitos integrados em larga escala possibilitaram que o processamento e o armazenamento de sinais pudessem ser realizados na forma digital apresentando diversas vantagens em relação aos sistemas anteriores, entre outras, redução do número de componentes mecânicos, redução do tamanho físico, menor custo do "hardware", melhoria da qualidade do sinal de voz e o uso do computador como ferramenta de auxílio à análise, ao projeto e à implementação.

O processamento digital de sinais de voz viabilizou e possibilitou o aprimoramento de diversas aplicações nas áreas industriais, militares e de entretenimento. Como exemplo, pode-se citar:

a) Sistemas de armazenamento digital de voz:

- instrumentos falantes;
- brinquedos falantes;
- dispositivos de alarme;
- dispositivos de avisos em lojas, aeroportos, rodoviárias, etc.;
- dispositivos de mensagens telefônicas, (hora certa, despertador, etc.);
- auxílio a deficientes visuais;
- conversor de texto escrito em falado;
- aplicações automotivas.

b)Sistemas de transmissão digital de voz:

- sistemas de comunicação de segurança;
- estações reforçadoras de sinais;
- telefonia celular digital;
- economia nos canais de comunicação;
- cancelamento de eco.

c)Sistemas de reconhecimento de palavras:

- conversor de texto falado em escrito;
- comunicação homem-máquina.

d)Sistemas de reconhecimento de locutor:

- acesso à informação;
- acesso a áreas restritas;
- transações automáticas de crédito;
- auxílio a deficientes auditivos.

Um dos grandes problemas encontrados na área da codificação de voz é o da medida de sua qualidade [5]. Uma alternativa para resolver este problema foi o estabelecimento de uma escala de medida denominada MOS ("Mean Opinion Score"), por meio da qual pode-se ter uma avaliação subjetiva da qualidade da voz sintetizada. Esta escala varia de 1 a 5, onde:

- nível 5 denota qualidade excelente ou transparente;
- nível 4, boa;
- nível 3, satisfatória;
- nível 2, ruim;
- nível 1, inaceitável.

Nesta escala, níveis iguais ou superiores a 4 caracterizam uma alta qualidade de voz. Para valores em torno de 2 e 3, tem-se uma voz sintética não

adequada para aplicações em sistemas de reconhecimento de locutor.

Como figura de mérito para medir a qualidade do sinal de voz, neste trabalho, foi adotado o critério objetivo de medida da Relação Sinal-Ruído apresentado em detalhes no Capítulo 3. Outros critérios de avaliação podem ser encontrados em [6, 7, 8].

Um dos objetivos de grande interesse das pesquisas no campo do processamento de voz, atualmente, é o de procurar obter uma qualidade de voz sintetizada próxima da transparente, com taxas da ordem de 8 kbits/s e, qualidade com nível em torno de 3,5 (escala MOS) para taxas de 4,8 kbits/s ou menor. A principal motivação das pesquisas em processamento de voz é ir ao encontro de novas técnicas e algoritmos para representação do sinal de voz que levem a uma melhoria na qualidade e/ou uma redução na quantidade de parâmetros a ser transmitidos ou processados.

Os sons de uma língua são estudados no campo da fonética que subdivide-se em três ramos principais: o acústico, o auditivo e o articulatório (Fig. 1.1). Segundo Malberg [2], a quantidade de sons em uma língua é quase ilimitada. Não se pronuncia duas vezes seguidas, da mesma maneira, uma vogal ou uma consoante. O contexto sonoro difere de acordo com a entonação, com a velocidade de pronúncia, etc. Esta variação dificulta o estabelecimento de critérios de classificação dos fonemas, penalizando sobremaneira a obtenção das técnicas para o reconhecimento da fala, como também a redução da taxa de bits para transmissão e/ou armazenamento da voz codificada.

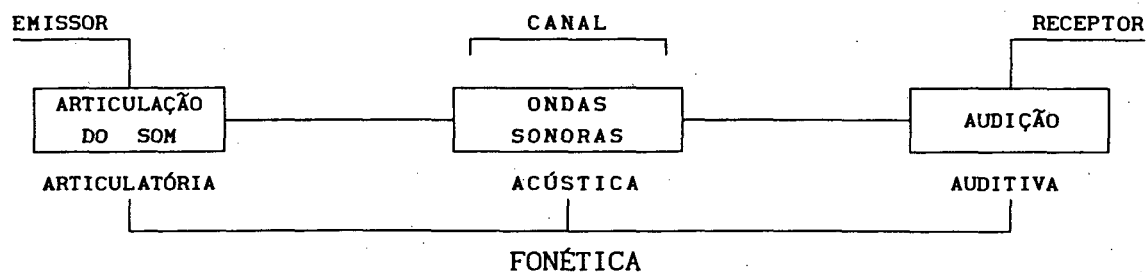


Fig. 1.1 - A Fonética na comunicação

A associação de fonemas para a construção de um segmento de voz qualquer configura uma das três técnicas básicas utilizadas para a representação de sinais de voz. Esta técnica é denominada "representação por regras". Aqui, grupos de fonemas podem ser expressos por pequenas quantidades de dados (aproximadamente a mensagem básica de informação de um texto (60 bits/s)). Todavia, com esta técnica fica bastante difícil se obter uma qualidade de voz "natural", tendo em vista as diferentes características de entonação, acentuação, pontuação, duração dos sons, etc., encontrados nas diversas línguas.

As duas outras formas de representação de sinais de voz são a representação por forma de onda e a representação paramétrica (Fig. 1.2).

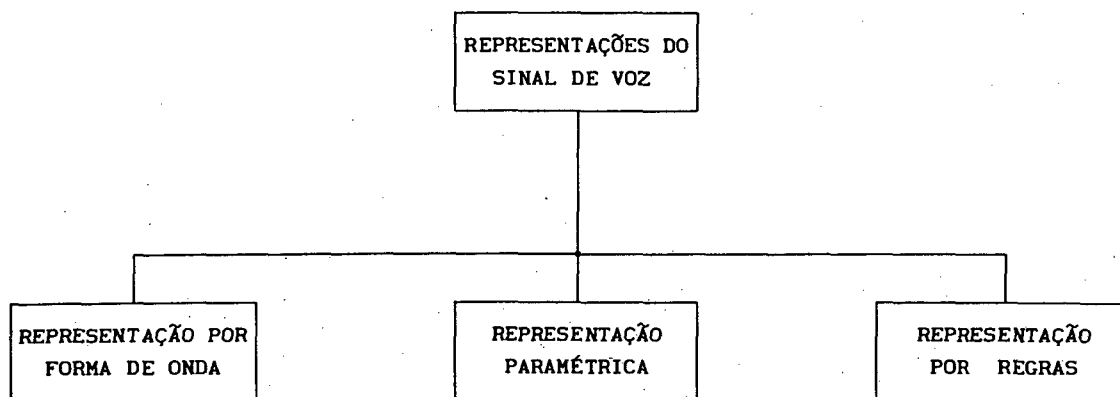


Fig. 1.2 - Representações dos sinais de voz

A representação por forma de onda é a que possui menor complexidade de realização, porém, as taxas obtidas são maiores (normalmente entre 20 kbits/s e 200 kbits/s). A modulação por codificação de pulsos (PCM); a modulação por codificação de pulsos diferencial (DPCM); a modulação por codificação de pulsos diferencial adaptável (ADPCM); a modulação delta (DM), constituem as principais técnicas deste tipo de representação [9, 10, 11, 12]. O fato de permitir obter uma alta qualidade de voz e o de ser relativamente mais simples de implementar fazem com que a representação por forma de onda, ainda hoje, seja a técnica que tem maior apelo industrial/comercial. Para compatibilizações de sistemas, o CCITT ("International Telegraph and Telephone Consultative Committee") criou algumas recomendações, como a G711 (64 kbits/s), a G721 (32 kbits/s), a G722 (24 ou 40 kbits/s), que servem de base para este tipo de codificação [9, 11].

A representação paramétrica constitui-se em um dos principais ramos de

estudos para realizar a compressão de sinais de voz, pois com este tipo de representação, é possível combinar baixas taxas, desde 500 a 20.000 bits/s, com alta qualidade de voz sintetizada. Fazem parte desta técnica os codificadores MPDPC ("Multi-Pulse Excited Linear Predictive Coder"), RPELPC ("Regular-Pulse Excited Linear Predictive Coder"), CELP ("Code-Excited Linear Predictive Coder"), entre outros.

Uma comparação entre as técnicas de codificação paramétrica e as por forma de onda, considerando a escala MOS versus a taxa de transmissão, é apresentada na Fig. 1.3.

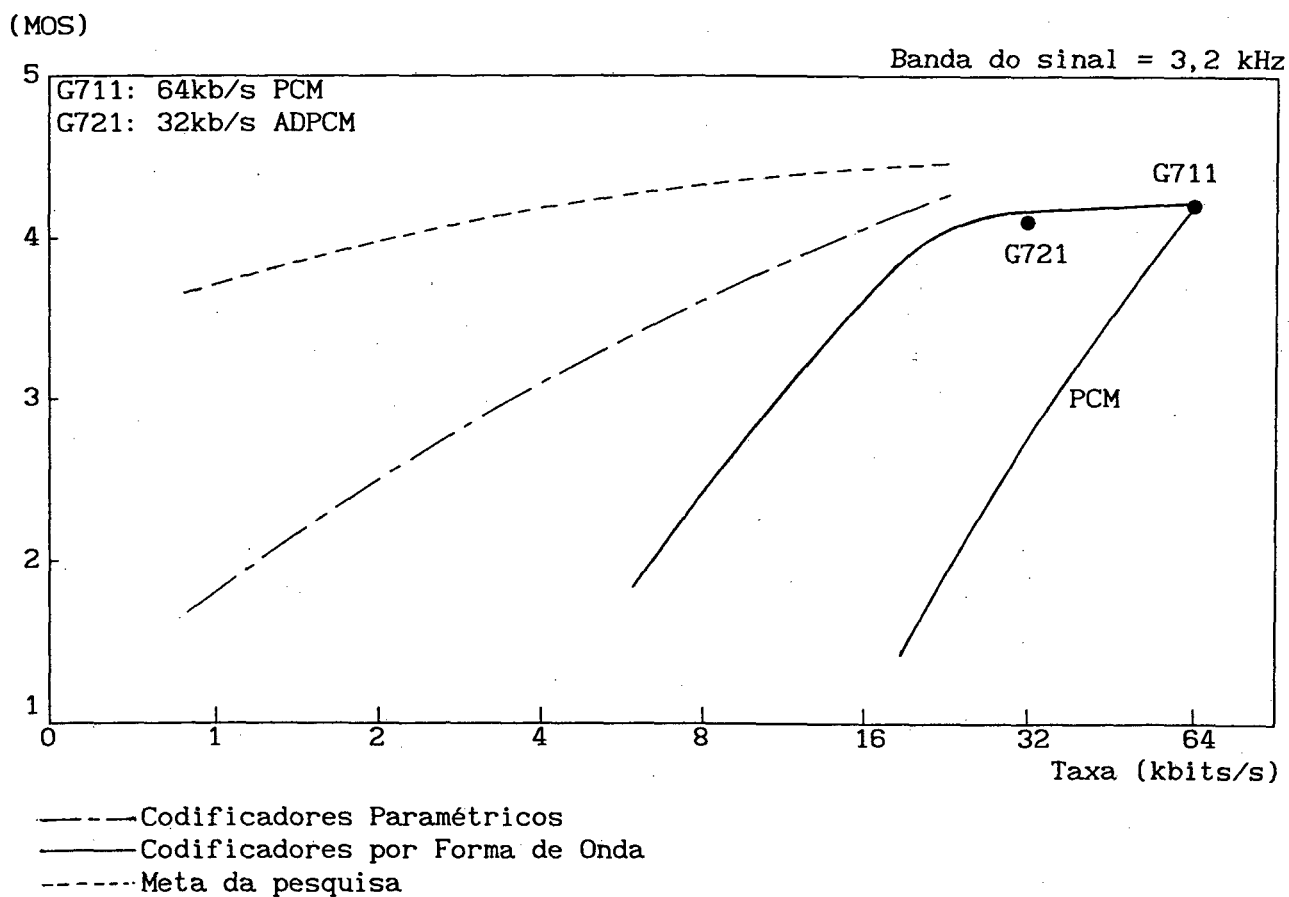


Fig. 1.3 - Qualidade subjetiva de voz para as diversas técnicas de codificação em função da taxa de transmissão em kb/s/s.

Dentre os tipos de codificadores paramétricos aplicáveis a sinais de voz, o CELP é um dos codificadores que podem possibilitar uma implementação a taxas tão baixas quanto 4,8 kbits/s [13]. No entanto, um dos problemas deste codificador é o elevado esforço computacional requerido na etapa de seleção da melhor seqüência de excitação. Muitos pesquisadores propuseram formas de aceleração deste procedimento, alguns impondo restrições à estrutura, outros propondo uma mudança no procedimento de análise do domínio do tempo para outros domínios [14, 15, 16, 17]. Neste trabalho é proposta uma nova estrutura, a qual apresenta um reduzido esforço computacional (análise e síntese) quando comparado com a do CELP convencional, permitindo ainda aplicar diretamente sobre a mesma alguns dos procedimentos de aceleração na fase de análise já consagrados para as estruturas CELP. Neste trabalho, os esforços foram dirigidos no sentido de se reduzir a quantidade de operações aritméticas envolvidas nos algoritmos sem grandes perdas na qualidade da voz sintetizada.

No Capítulo 2 é apresentada uma revisão bibliográfica de algumas das estruturas de codificadores paramétricos mais importantes. É também apresentado um equacionamento completo do codificador CELP convencional [14]. São revistos pontos importantes da predição linear aplicada a sinais de voz.

No Capítulo 3 é obtida, a partir da estrutura do codificador CELP básico, a nova estrutura de codificação. São apresentados os caminhos que conduziram à proposição da nova estrutura. Também são mostrados e discutidos os resultados das diversas simulações que serviram de base para este estudo.

No apêndice "A" apresenta-se o programa que implementa um algoritmo para obtenção dos parâmetros do CELP básico proposto em [14].

No apêndice "B" apresenta-se o programa que implementa um algoritmo para obtenção dos parâmetros da estrutura proposta.

No apêndice "C" apresentam-se os detalhes e as características principais do sistema de aquisição de dados A/D e D/A, desenvolvido para a realização deste trabalho.

No apêndice "D" apresenta-se as características da locução utilizada nas diversas simulações neste trabalho.

CAPÍTULO 2

CODIFICADORES PARAMÉTRICOS - REVISÃO BIBLIOGRÁFICA2.1. INTRODUÇÃO

A representação paramétrica é baseada no modelo do aparelho fonador, na qual os parâmetros de excitação e os parâmetros do trato vocal são codificados com o intuito de diminuir as taxas de transmissão do sinal e/ou armazenagem.

O processo de codificação é obtido através da partição de uma seqüência de amostras em quadros e sub-quadros. Cada quadro com duração entre 10 e 30 ms (período este em que o sinal de voz é aproximadamente estacionário) permite a determinação dos parâmetros relativos ao trato vocal (envelope espectral). Os sub-quadros com duração entre 1 e 5 ms permitem, por sua vez, a obtenção de parâmetros concernentes à excitação (estrutura espectral fina).

2.2. CODIFICAÇÃO LINEAR PREDITIVA (LPC)

A técnica de predição linear possibilita que um sinal (estacionário) possa ser estimado através da combinação linear das entradas (presente e passadas) e saídas (passadas) de um determinado sistema. A predição linear é aplicada eficazmente em vários campos: estatísticos, econômicos, de comunicações, biomédicos, etc. [18].

Como os sinais de voz podem ser considerados aproximadamente estacionários para pequenos segmentos (quadros), é possível, nestes casos,

aplicar proveitosamente tal técnica [19]. Na Fig. 2.1 é apresentado um esquema simplificado que representa os codificadores paramétricos, onde $c(n)$ é o sinal de excitação e $\hat{s}(n)$ o sinal de voz sintetizado, obtido da saída do filtro preditor, cuja função precípua é estimar o trato vocal.

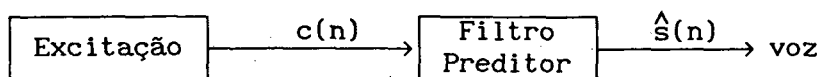


Fig. 2.1 - Modelo de produção de voz para codificadores paramétricos.

No modelo de produção de voz, apresentado anteriormente, a composição dos efeitos do espectro de radiação, das cavidades bucal e nasal e até mesmo da excitação glotal podem ser representados por um filtro pólo-zero (filtro tipo ARMA) invariante no tempo para cada quadro de voz, formando assim, uma estrutura de predição.

O sinal de saída do preditor pode ser representado pela seguinte equação de diferenças:

$$\hat{s}(n) = \sum_{k=1}^{OD} a_k \cdot \hat{s}(n-k) + \sum_{k=0}^{ON} b_k \cdot c(n-k) \quad (2.1)$$

Aplicando-se a Transformação em Z à equação (2.1), obtém-se a seguinte função de transferência para o filtro preditor:

$$H(z) = \frac{\hat{S}(z)}{C(z)} = \frac{\sum_{k=0}^{ON} b_k \cdot z^{-k}}{1 - \sum_{k=1}^{OD} a_k \cdot z^{-k}} \quad (2.2)$$

Devido ao problema de não-linearidade do sistema de equações obtido na fase de análise para a estimação dos parâmetros da estrutura tipo ARMA ("AutoRegressive Moving Average") [20], normalmente utiliza-se, para a modelagem do trato vocal, filtros tipo AR ("AutoRegressive") com 10 a 16 pólos em sua função de transferência. Assim a função de transferência do filtro preditor reduz-se à:

$$H(z) = \frac{b_0}{1 - \sum_{k=1}^{OD} a_k \cdot z^{-k}} \quad (2.3)$$

2.2.1. PREDITOR DE CURTO ATRASO (PCA) E PREDITOR DE LONGO ATRASO (PLA)

Em um sinal de voz as amostras próximas entre si e as amostras afastadas de um período de "pitch" são fortemente correlacionadas. Para aproveitar esta correlação, normalmente, prefere-se dividir o filtro preditor em dois filtros em cascata. Um serve para tirar proveito da correlação existente entre as amostras próximas entre si, denominado Filtro Preditor de Curto Atraso, e o

outro tem o propósito de aproveitar a correlação contida entre as amostras separadas por um período de "pitch", denominado Filtro Preditor de Longo Atraso, (Fig. 2.2).

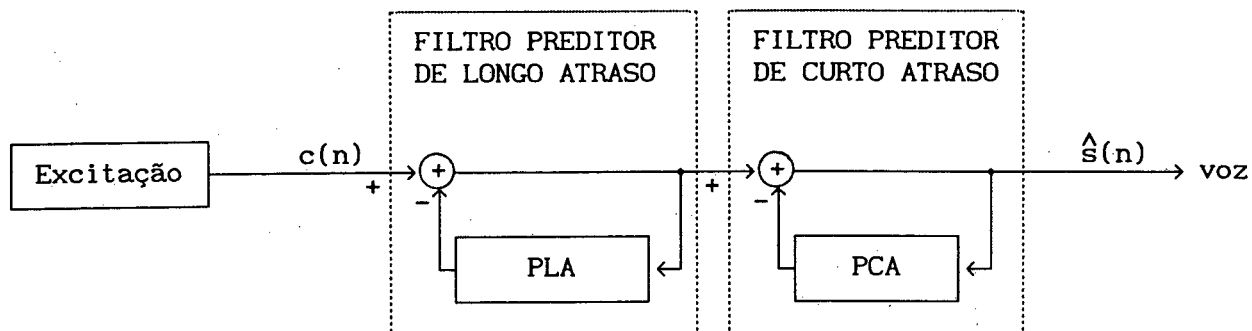


Fig. 2.2 - Modelo para a síntese de voz usando preditores de curto e longo atraso.

O preditor de longo atraso (PLA), representado no domínio da Transformada Z, é dado pela seguinte equação:

$$P_L(z) = \sum_{k=-q}^r d_k \cdot z^{-(D+k)} \quad (2.4)$$

onde D é o número de amostras relativas a um período de "pitch" e d_k são os coeficientes do PLA. Tipicamente o número de coeficientes d_k varia de 1 ($q=r=0$) a 3 ($q=r=1$) [13]. A razão do uso de 3 coeficientes para o PLA, é a garantia de que o tempo correspondente a um período de "pitch" esteja sempre contido entre a primeira e a última dessas 3 amostras.

O preditor de curto atraso (PCA), representado através da Transformada Z, é dado pela seguinte equação:

$$P_C(z) = \sum_{k=1}^{OD} a_k \cdot z^{-k} \quad (2.5)$$

onde a_k são os coeficientes do PCA e OD a sua ordem, que está, tipicamente, entre 10 e 16, para uma frequência de amostragem de 8 kHz [13].

Normalmente os coeficientes dos preditores são determinados com base no sinal de voz original usando técnicas de predição linear [18, 19, 21, 22, 23]. Entre as mais utilizadas destacam-se as seguintes:

- Método da covariância;
- Método da autocorrelação;
- Algoritmo LMS ("Least Mean Square");
- Formulação da "lattice";
- Formulação do filtro inverso;
- Formulação da estimação espectral;
- Formulação da máxima verossimilhança;
- Formulação do produto interno.

A seguir são revistos alguns tipos de codificadores paramétricos mais utilizados para a estimação de sinais de voz.

2.3. CODIFICADORES PARAMÉTRICOS

Os codificadores paramétricos são caracterizados, em sua grande maioria, pelo tipo do gerador de excitação utilizado. Existe uma grande variedade de formas de realização da excitação para estes codificadores. Tipicamente, pode-se classificar a excitação dos codificadores paramétricos em três grupos

principais: a excitação por trem de pulsos regulares, a excitação multi-pulsos e a excitação por dicionário de códigos [13].

Dentre as principais características de desempenho que podem ser aprimoradas nestes codificadores de voz, destacam-se a qualidade da voz sintetizada, a taxa de transmissão (ou, equivalentemente, a quantidade de memória de armazenamento), o atraso de codificação, a complexidade computacional de análise/síntese, a robustez e o custo de implementação.

A seguir serão apresentadas algumas das estruturas mais difundidas destes codificadores.

2.3.1. CODIFICADOR PELPC

O PELPC ("Pitch-Excited Linear Predictive Coder") foi um dos primeiros codificadores propostos com base no mecanismo real de produção da voz humana [19] (Fig. 2.3).

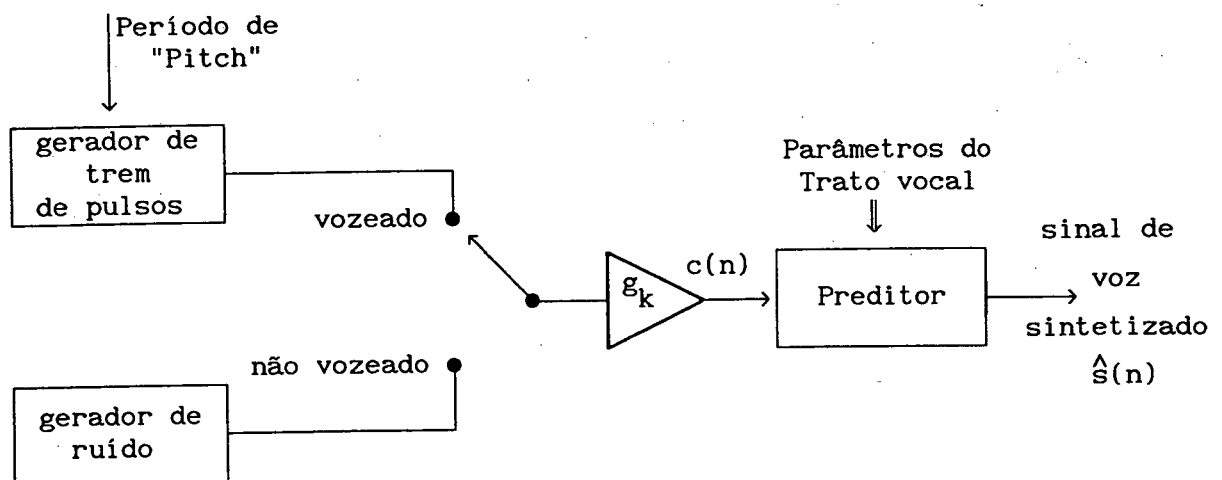


Fig. 2.3 - Diagrama em blocos do codificador PELPC.

Neste sistema são utilizadas duas fontes de excitação. Para os sons vozeados, é utilizado um trem de pulsos periódico com período igual ao "pitch" do sinal de voz. Para os sons não vozeados a excitação consiste em uma seqüência de ruído branco. Os parâmetros relativos a este modelo de codificador são: classificação vozeado/não vozeado, período de "pitch" do sinal de voz, ganho (G) e, os coeficientes do filtro de predição os quais têm a função da modelagem do trato vocal [6, 18, 19, 24].

Testes efetuados com este modelo de excitação [5, 25], mostram que a voz gerada é inteligível, mas não natural, mesmo quando usadas taxas de bits elevadas. A razão exata desta degradação não é bem conhecida. Acredita-se, que o maior problema esteja na modelagem da excitação em termos de um trem de pulsos ou ruído branco [25, 26]. Este tipo de estrutura tem um grande apelo em aplicações com muito baixas taxas (por exemplo: 1 kbits/s).

2.3.2. CODIFICADOR MPCLPC

Inicialmente proposto por Atal & Remde [27, 28], o MPCLPC ("Multi-Pulse Excited Linear Predictive Coder") consiste na utilização de uma seqüência de impulsos com determinada localização e amplitude, tanto para os sons vozeados quanto para os não vozeados, eliminando-se com isto a análise vozeada/não vozeada, uma desvantagem da estrutura do modelo anterior.

Os impulsos são determinados quadro a quadro pelo critério do mínimo erro médio quadrático, calculado entre o sinal de voz original e o sinal de saída do filtro preditor. Infelizmente esta minimização resulta num problema combinatorial cuja solução global exige um grande esforço computacional. Uma forma de resolver este problema é obter a excitação em vários estágios. Em cada estágio a amplitude e a localização de um único impulso é obtida através

de procura exhaustiva. Os demais impulsos são calculados assumindo que os impulsos anteriores são fixos. Isso leva à determinação de um conjunto de impulsos sub-ótimos. Outras técnicas para a determinação da localização dos impulsos são apresentadas em [22]. O número de impulsos necessários para cada quadro de voz depende da qualidade de voz desejada e do período de "pitch", sendo que, a qualidade aumenta de acordo com o aumento do número de impulsos por quadro. Segundo Singhal & Atal [25], 6 a 8 impulsos por quadro de voz de 10 ms de duração são suficientes para sintetizar a voz com alta qualidade. Observa-se que vozes de locutores femininos, por conterem componentes de frequência mais elevada, requerem mais impulsos por quadro do que as de locutores masculinos, para uma qualidade de voz comparável.

A Fig. 2.4 apresenta o diagrama em blocos da etapa de análise do codificador multi-pulso.

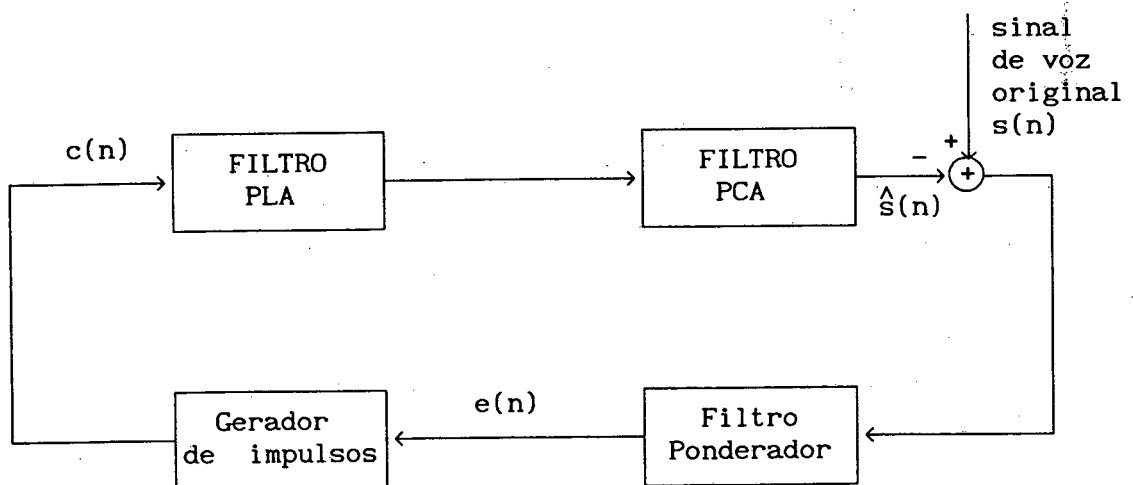


Fig. 2.4 - Diagrama em blocos da etapa de análise do codificador multi-pulso.

Nesta figura, o gerador de impulsos produz uma seqüência $c(n)$ que é aplicada à entrada do filtro preditor de longo atraso. O sinal de erro entre o sinal de voz original $s(n)$ e o sinal de voz sintetizado $\hat{s}(n)$ é aplicado em um filtro ponderador de erro gerando o sinal $e(n)$, que controlará o processo. Maiores detalhes desta estrutura podem ser encontrados em [6, 13, 22, 25, 26].

2.3.3. CODIFICADOR SEV

O SEV ("Self Excited Vocoder") é um codificador constituído de um ou mais preditores de longo atraso (PLA) e um preditor de curto atraso (PCA) onde a excitação não aparece de forma explícita, por isso o nome de auto-excitado [6, 29, 30]. A Fig. 2.5 mostra o diagrama em blocos desta estrutura.

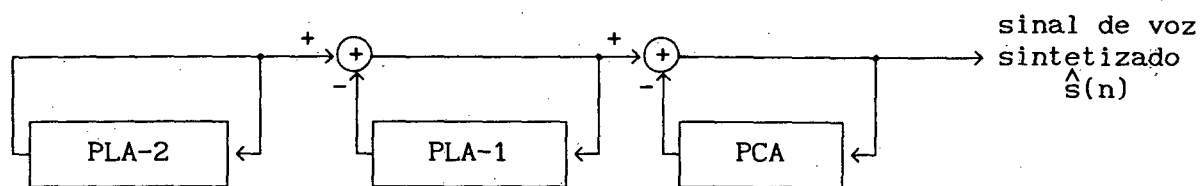


Fig. 2.5 - Diagrama em blocos do modelo de síntese do codificador SEV usando 2 preditores de longo atraso (PLA).

Retirando-se o PLA-1, tem-se o SEV básico. Neste tipo de codificador as amostras atrasadas são utilizadas como excitação. Variando-se o atraso do PLA e seu ganho, pode-se obter a melhor excitação para um determinado quadro. O método de análise empregado para a determinação dos parâmetros neste tipo de codificador é o mesmo utilizado no MPELPC e no CELP, que será discutido a seguir.

2.3.4. CODIFICADOR CELP

Os codificadores CELP (Code-Excited Linear Predictive) foram originalmente propostos por M. R. Schroeder e B. S. Atal [31]. São codificadores que demonstram grande potencial para produzir alta qualidade de voz sintética a taxas tão baixas quanto 4,8 kbits/s [14, 13]. Nesta técnica a excitação consiste na utilização de um dicionário de códigos correntemente denominado por "codebook". A idéia aqui é selecionar neste "codebook" uma excitação ("codeword") que leve à "melhor" qualidade de voz sintetizada, no lugar de, para cada quadro, determinar-se uma nova excitação. É importante notar que neste caso é utilizado o índice da "codeword" para a codificação da excitação, podendo-se diminuir substancialmente a taxa de bits para representá-la.

A Fig 2.6 mostra a estrutura de um "codebook" contendo L "codewords" com N_s amostras. Tipicamente, as "codewords" são obtidas de uma seqüência de ruído branco com distribuição gaussiana [13].

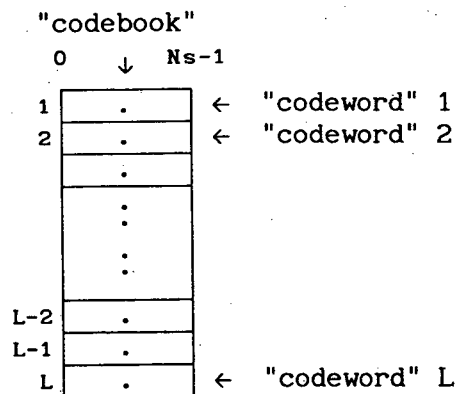


Fig. 2.6 - Distribuição das "codewords" no "codebook".

Assim, a melhor excitação para cada sub-quadro é obtida a partir de uma busca exaustiva no "codebook". A vantagem no que diz respeito à redução da taxa de transmissão é que, por exemplo, para um "codebook" típico de 1024 "codewords" é necessário transmitir apenas 10 bits referentes aos índices. Um dos maiores problemas neste tipo de codificador, contudo, é a elevada complexidade computacional na fase de determinação dos parâmetros, devida, principalmente, ao procedimento de procura das "codewords" no "codebook". Esta limitação tem gerado diversos trabalhos de pesquisa cujo objetivo principal concentra-se na determinação de técnicas para reduzir a complexidade computacional destes codificadores. Dentre estas pode-se citar, por exemplo, a utilização de vetores esparsos [32], a utilização de vetores pseudo-estocásticos [33] e a utilização de vetores com elementos binários [15]. Com o auxílio de outras técnicas consegue-se aumentar ainda mais a eficiência computacional [14], mudando o procedimento de busca da melhor "codeword", do domínio do tempo para o domínio da frequência, onde a operação de convolução da excitação com a resposta ao impulso do filtro preditor é agora reduzida a uma simples operação de multiplicação.

A Fig. 2.7 apresenta o diagrama em blocos para seleção da melhor excitação.

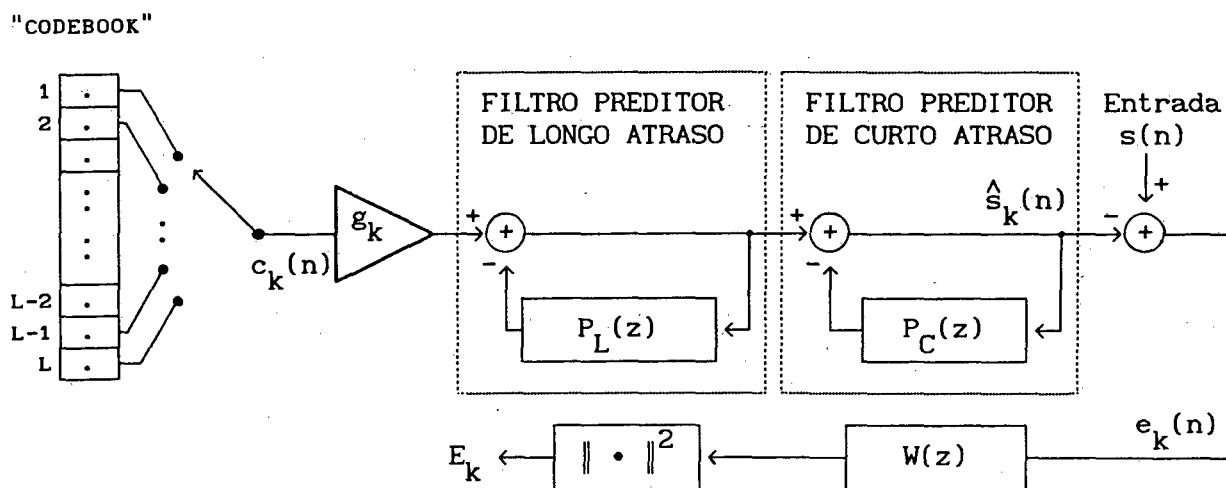
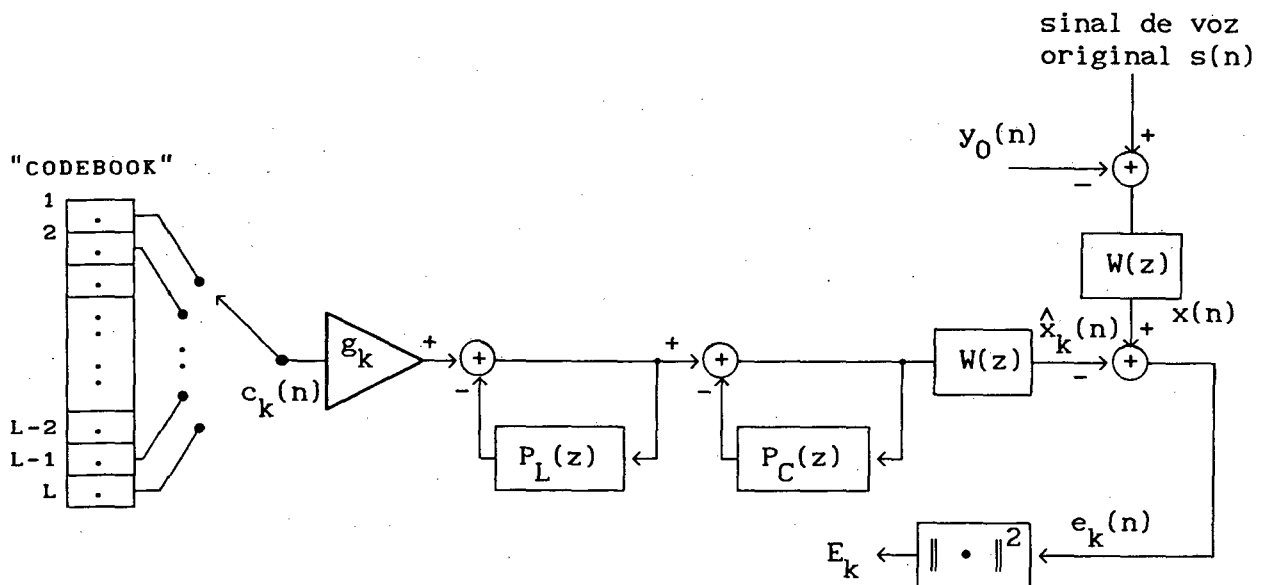


Fig. 2.7 - Diagrama em blocos ilustrando os procedimentos para procura da melhor "codeword".

O procedimento de procura da melhor "codeword" (ilustrado na Fig. 2.7) envolve três passos básicos: a geração de L seqüências $\hat{s}_k(n)$, o cálculo da energia do erro E_k (ponderada) entre cada uma das seqüências dos sinais de voz original e sintetizado e a seleção da seqüência que produz o mínimo erro. A geração das L seqüências $\hat{s}_k(n)$ é a que demanda maior esforço computacional dos três, requerendo aproximadamente $OD \cdot L$ operações aritméticas para cada amostra do sinal de voz, onde OD é a ordem do filtro. Por exemplo, supondo um "codebook" de $L=1024$ "codewords" e um filtro de 20 coeficientes para um segmento adquirido a uma taxa de amostragem de 8 kHz, necessita-se de aproximadamente 160 milhões de operações aritméticas por segundo para a geração das $\hat{s}_k(n)$ seqüências.

A seguir, será apresentado o procedimento da formulação da análise por síntese para os codificadores CELP usando notação matricial. A Fig. 2.8 apresenta o diagrama em blocos que ilustra este procedimento.



$y_0(n)$ é a resposta a entrada nula do filtro composto pelos preditores PLA e PCA.

Fig. 2.8 - Diagrama em blocos ilustrando o procedimento da formulação da análise por síntese para os codificadores CELP.

Seja um "codebook" composto por um sinal de ruído branco com distribuição gaussiana contendo L "codewords" de comprimento N_s . Cada "codeword" é escalada por um fator de ganho g_k , e processada pelos filtros preditores PLA e PCA. O erro entre o sinal de voz original e o sintetizado é processado por um filtro ponderador $W(z)$ e, posteriormente, é calculada a sua energia. A "codeword" que produz a mínima energia é selecionada após uma procura exaustiva no "codebook".

O preditor de longo atraso determinado, por exemplo, para cada sub-quadro de 5 ms (40 amostras para uma frequência de amostragem de 8 kHz), $P_L(z)$, é dado por:

$$P_L(z) = d_1 \cdot z^{-D+1} + d_2 \cdot z^{-D} + d_3 \cdot z^{-D-1} \quad (2.6)$$

onde D representa um atraso na faixa de 2 a 15 ms, que corresponde, tipicamente, à faixa encontrada para o período de "pitch".

O preditor de curto atraso determinado, por exemplo, para cada quadro de 20 ms (160 amostras para uma frequência de amostragem de 8 kHz), $P_C(z)$, é representado por:

$$P_C(z) = \sum_{k=1}^{OD} a_k \cdot z^{-k} \quad (2.7)$$

O propósito do filtro ponderador $W(z)$ é o de proporcionar uma atenuação na distribuição espectral do erro ($e_k(n)$) para as frequências onde o espectro do erro é perceptivelmente mais importante, e amplificá-la nas regiões onde o espectro do erro é menos importante. A motivação para a utilização do filtro ponderador vem do fato de que nas regiões de frequências onde os sinais vozeados têm maior amplitude, nos chamados formantes, as componentes espectrais do erro são menos perceptíveis do que naquelas entre os formantes. O mesmo pode ser constatado para as regiões de frequência de alta e baixa energia dos sons não vozeados. Normalmente o filtro ponderador melhora a qualidade subjetiva e piora a qualidade objetiva do sinal de voz sintetizado.

A função de transferência $W(z)$ pode ser escrita, em termos dos coeficientes do preditor de curto atraso, como:

$$W(z) = \frac{1 - \sum_{k=1}^{OD} a_k \cdot z^{-k}}{1 - \sum_{k=1}^{OD} a_k \cdot \gamma^k \cdot z^{-k}} \quad (2.8)$$

onde γ é um coeficiente que controla a ponderação em função da frequência [1, 14].

O parâmetro γ é dado por:

$$\gamma = e^{-(\pi \cdot \delta) / f_a} \quad (2.9)$$

onde f_a é a frequência de amostragem utilizada e δ caracteriza a redução de magnitude dos pólos de $W(z)$ [14]. Para frequência de amostragem de 8 kHz com γ variando entre 0,73 e 0,92 obtém-se muito bons resultados em termos de tolerância ao ruído [1, 14]. Como o filtro ponderador é linear, é conveniente deslocá-lo para antes da comparação entre o sinal de voz original e o sintetizado com o intuito de simplificar os procedimentos de análise por síntese. Isto equivale a comparar o sinal de voz original ponderado com o sinal sintetizado ponderado.

O resultado na saída do filtro excitado pela k-ésima "codeword" do presente sub-quadro é dada por:

$$\hat{\underline{x}}_k = g_k \cdot H \cdot \underline{c}_k \quad (2.10)$$

onde g_k é o ganho, H é uma matriz triangular inferior de dimensão [Ns x Ns] obtida da resposta ao impulso dos dois filtros preditores PLA e PCA cascadeados com o filtro ponderador W(z), \underline{c}_k é o vetor da k-ésima "codeword" representado por:

$$\underline{c}_k = [c_k(0), c_k(1), \dots, c_k(Ns-1)] \quad (2.11)$$

e $\hat{\underline{x}}_k$ é o vetor do sinal de voz sintetizado subtraído da memória do filtro composto pelos preditores PLA e PCA ($y_0(n)$) cascadeados com o filtro ponderador W(z). Este vetor é representado por:

$$\hat{\underline{x}}_k = [\hat{x}_k(0), \hat{x}_k(1), \dots, \hat{x}_k(Ns-1)] \quad (2.12)$$

O erro quadrático total E_k é definido como:

$$E_k \triangleq \|\underline{x} - \hat{\underline{x}}_k\|^2 \quad (2.13)$$

onde o vetor \underline{x} , representado por

$$\underline{x} = [x(0), x(1), \dots, x(Ns-1)], \quad (2.14)$$

é obtido do sinal de voz original passado pelo filtro ponderador W(z), subtraído da resposta à entrada nula do filtro composto pelos dois filtros

preditores (obtidos pelo $P_L(z)$ e $P_C(z)$) cascadeados com o filtro ponderador $W(z)$. $\| \cdot \|$ representa a norma Euclidiana de um vetor.

O melhor ganho g_k que minimiza E_k , pode ser obtido por $\partial E_k / \partial g_k = 0$.

Assim, obtém-se:

$$g_k = \frac{\underline{x} \cdot H \cdot \underline{c}_k}{\| H \cdot \underline{c}_k \|^2} \tag{2.15}$$

e

$$E_k = \| \underline{x} \|^2 - \frac{(\underline{x} \cdot H \cdot \underline{c}_k)^2}{\| H \cdot \underline{c}_k \|^2} \tag{2.16}$$

A melhor "codeword" pode ser assim obtida pela minimização de E_k ou equivalentemente pelo máximo valor do segundo termo à direita de E_k . Este procedimento pode ser visto no diagrama em blocos da Fig. 2.9 [14].

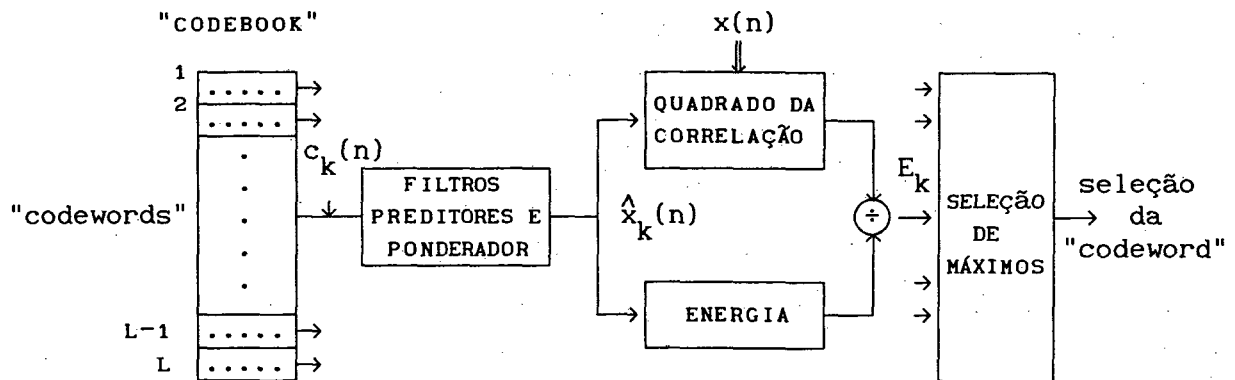


Fig. 2.9 - Diagrama em blocos do procedimento de escolha da melhor "codeword".

2.4. CONCLUSÃO

Neste capítulo foram revistos alguns conceitos sobre predição linear aplicada a sinais de voz e algumas das principais técnicas de codificação paramétricas, sendo analisados com mais detalhes, os codificadores CELP.

Foi evocado que os codificadores CELP demonstram grande potencial para produzir alta qualidade de voz sintética a taxas tão baixas quanto 4,8 kbits/s. Todavia um dos maiores problemas encontrados neste tipo de codificador, é a elevada complexidade computacional na fase de determinação de seus parâmetros.

CAPÍTULO 3

CODIFICAÇÃO DA EXCITAÇÃO COM FILTRAGEM ADAPTÁVEL

3.1. INTRODUÇÃO

Neste capítulo é proposta uma nova estrutura para a codificação de sinais de voz com excitação codificada que apresenta uma menor complexidade computacional na fase de análise do que aquela do codificador CELP convencional.

São apresentados os caminhos que conduziram à proposição desta estrutura, tomando por base o codificador CELP descrito no Capítulo 2.

Esta nova estrutura utiliza um filtro preditor adaptado pelo algoritmo LMS ao invés do tradicional filtro auto-regressivo geralmente determinado pelo método da autocorrelação ou da covariância, entre outros.

Finalmente são apresentados um resumo e alguns comentários dos resultados mais relevantes.

3.2. REQUISITOS PARA AVALIAÇÃO E COMPARAÇÃO DO DESEMPENHO DA ESTRUTURA PROPOSTA

Como requisitos para avaliação do desempenho das diversas estruturas de codificação de sinais de voz destacam-se de modo geral:

- a qualidade do sinal de voz sintetizado;
- a taxa em termos de bits/s envolvidas na transmissão ou armazenamento;

- a complexidade computacional na fase de análise;
- a robustez da estrutura de codificação. Um codificador é dito robusto quando for pequena a sua sensibilidade em relação ao tipo de locutor, ou seja a qualidade da voz sintetizada praticamente não depende do locutor;
- o atraso intrínseco dos sistemas digitais de análise e síntese;
- o custo de implementação em termos de complexidade do "hardware" do codificador/descodificador.

Uma estrutura para codificação de sinais de voz será tanto melhor quanto maior for a qualidade do sinal de voz sintetizado e a robustez da estrutura, e menor forem os demais requisitos apresentados anteriormente.

Todavia, existe um compromisso entre os diversos requisitos, pois na medida que um codificador melhora o desempenho em relação a um, pode piorar em relação a outros. Como exemplo, um codificador que trabalha a baixas taxas, normalmente, requer um elevado esforço computacional acompanhado de uma degradação da qualidade do sinal de voz sintetizado.

Dentre os objetivos que serão perseguidos para o estabelecimento da nova estrutura destaca-se a redução da complexidade computacional, na fase de análise, ao mesmo tempo em que se procura manter a mesma quantidade de parâmetros contidos no codificador CELP convencional adotado como padrão de comparação neste trabalho.

3.2.1. CRITÉRIO DE MEDIÇÃO DA QUALIDADE DE VOZ SINTETIZADA

A figura de mérito utilizada para medir a qualidade objetiva do sinal de voz sintetizado é o erro médio quadrático normalizado, $EMQ_Q(j)$, entre $s(j,n)$ e $\hat{s}(j,n)$, sinais de voz original e sintetizado, respectivamente, onde j é o

índice do respectivo quadro.

Portanto, define-se:

$$EMQ_Q(j) = \frac{\sum_{n=0}^{N-1} [s(j,n) - \hat{s}(j,n)]^2}{\sum_{n=0}^{N-1} s^2(j,n)} \quad (3.1)$$

onde o subscrito Q caracteriza um quadro de voz de comprimento N.

Define-se também, a Relação Sinal-Ruído $RSR_Q(j)$, por:

$$RSR_Q(j) = 10 \log \frac{1}{EMQ_Q(j)} \quad (3.2)$$

Assim, para um segmento de voz composto por diversos, quadros tem-se o erro médio quadrático dado por:

$$EMQ_S = \frac{1}{V} \sum_{j=1}^V EMQ_Q(j) \quad (3.3)$$

e a respectiva Relação Sinal-Ruído do segmento, por:

$$RSR_S = 10 \log \frac{1}{EMQ_S} \quad (3.4)$$

onde o subscrito S caracteriza um segmento de voz composto por V quadros de comprimento N.

3.2.2. PADRÃO DE COMPARAÇÃO DOS RESULTADOS OBTIDOS NAS SIMULAÇÕES

Como padrão de comparação são tomados os resultados de diversas simulações sobre a estrutura do codificador CELP descrito no Capítulo 2.

Para possibilitar e sistematizar as comparações entre os diversos resultados, foram estabelecidos os seguintes parâmetros:

- "codebook" com $L = 1024$ "codewords";
- "codeword" com $N_s = 10, 20$ ou 40 amostras;
- segmento de voz relativo a um locutor masculino com duração aproximada de 14 segundos e foneticamente balanceado (Apêndice D);
- frequência de amostragem de 8 kHz;
- quantização linear de 12 bits (aquisição);
- quadro de análise de voz com $N_a = 80$ ou 160 amostras;
- quadro de medição do $EMQ_0(j)$ com $N = 80$ amostras.

Nestas simulações o "codebook" é constituído de um conjunto de "codewords" obtido de um sinal ruído branco com distribuição gaussiana.

Na Tabela 3.1 são mostrados os resultados das simulações do codificador CELP apresentado no Capítulo 2. As condições nas quais os resultados apresentados foram obtidos são as seguintes:

- Para determinação do PCA é utilizado o método da autocorrelação [19, 18, 22]. O sinal de voz original foi passado por uma janela de Hamming, com recobrimento nas bordas de 5 e 10 ms para os quadros de 10 e 20 ms, respectivamente. Para implementar esse recobrimento, os quadros foram aumentados em 50% em seu tamanho original. Optou-se pela utilização do método da autocorrelação por este ser relativamente simples de implementar, apresentar muito bons resultados e fornecer uma função transferência com fase

mínima para o preditor, o que conduz, contrariamente ao método da covariância, a um filtro PCA sempre estável [19, 22].

- Para a não redução da qualidade objetiva do sinal de voz sintetizado optou-se, neste trabalho, pela não utilização do filtro ponderador $W(z)$, pois este melhora a qualidade subjetiva, porém prejudica a qualidade objetiva utilizada como parâmetro nas avaliações da voz sintética.

- Para compatibilizar estes resultados com os resultados da estrutura proposta, não foi mantida a memória do PCA.

- Não foi considerada, no cômputo total da complexidade computacional, a quantidade de operações requerida na determinação dos coeficientes do PCA por esta ser desprezível frente à complexidade necessária à seleção da melhor "codeword". Em [22] é apresentada uma avaliação mostrando o número de operações aritméticas necessárias à determinação dos coeficientes do PCA quando utilizado os métodos da autocorrelação e da covariância.

- Muitos pesquisadores defendem a importância do PLA nos codificadores CELP [34, 35]. Porém, nas diversas simulações este preditor não foi utilizado em nenhuma das estruturas por aumentar, significativamente, tanto a complexidade computacional na fase de análise como também a taxa de transmissão resultante.

Tabela 3.1 - Resultados das simulações da estrutura CELP convencional.

quadro ms	sub-quadro ms	atraso ms	EMQ_S	RSR_S [dB]	OD	ordem de complexidade
10	1,25	12,5 + T_p	0,064	11,93	6	$O(OD.L+L)$
10	2,5	12,5 + T_p	0,124	9,06	6	$O(OD.L+L)$
10	5	12,5 + T_p	0,206	6,87	6	$O(OD.L+L)$
20	1,25	25 + T_p	0,073	11,37	12	$O(OD.L+L)$
20	2,5	25 + T_p	0,129	8,91	12	$O(OD.L+L)$
20	5	25 + T_p	0,203	6,92	12	$O(OD.L+L)$

OD - Ordem do preditor.

$O(.)$ - Ordem de complexidade de análise por amostra do sinal de voz. A ordem de complexidade é dada em termos de número de operações de multiplicação e adição.

T_p - Tempo de processamento função: da forma de realização, da complexidade de análise/síntese e do processador utilizado no codificador.

L - Número de "codewords" do "codebook".

Como pode ser observado na Tabela 3.1, a complexidade computacional está relacionada diretamente ao número de coeficientes do preditor. Além do mais, também pode ser constatado que, quanto menor for o número de amostras dos sub-quadros, maiores serão os valores de RSR_S . Todavia, a utilização de sub-quadros menores conduz a um acréscimo no número de bits necessário à transmissão, por quadro, dos índices das "codewords" e dos ganhos, penalizando, desta maneira, a taxa de transmissão final.

Os codificadores CELP, apesar de terem demonstrado grande eficiência para codificação de sinais de voz a baixas taxas com alta qualidade [36,37],

apresentam uma elevada complexidade computacional na fase de análise. Somente após o desenvolvimento das novas gerações de DSP's ("Digital Signal Processor") é que foi possível vislumbrar a possibilidade de sua implementação em tempo real.

Na literatura encontram-se várias formulações que permitem acelerar o processo de procura da melhor "codeword". Algumas destas técnicas apresentam restrições ao "codebook", como por exemplo em [15], onde as "codewords" contêm apenas alguns elementos diferentes de zero. Outras propõem mudar o processo de procura da melhor "codeword" do domínio tempo para o domínio frequência, reduzindo, desta forma, a complexidade nesta fase [14].

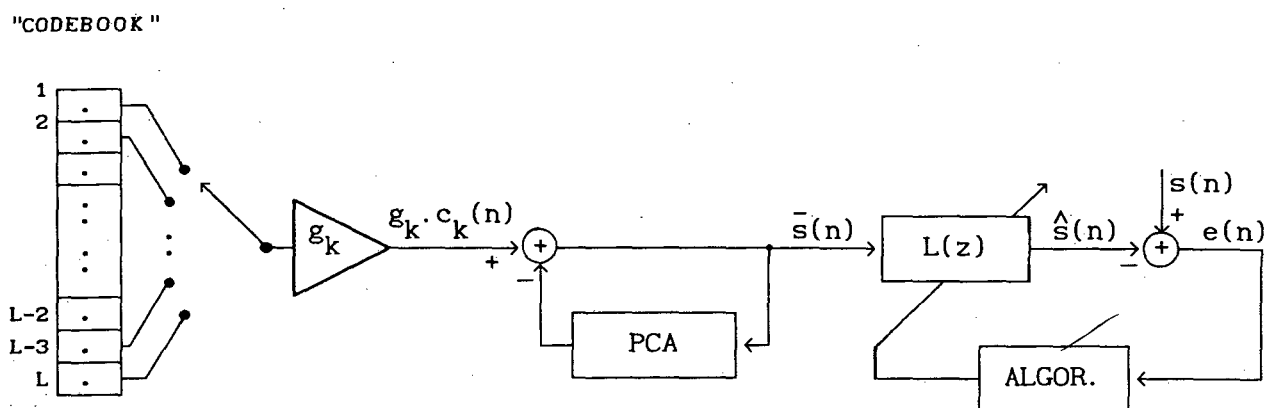
Neste trabalho, pretende-se utilizar uma forma alternativa de procura da melhor "codeword", operando no domínio do tempo e sem qualquer restrição aos elementos do "codebook", de tal forma que ainda exista a possibilidade de aplicação das técnicas de aumento de velocidade propostas para o CELP convencional, acelerando ainda mais o procedimento.

A motivação para se propor alterações na estrutura CELP convencional, surgiu do fato de que o PCA é um filtro sub-ótimo, no sentido de que a determinação de seus coeficientes é realizada sem levar em consideração a forma da excitação. Para contornar este problema, propõe-se inserir um filtro adicional à estrutura CELP, considerando agora a excitação, o qual atuará como um ajuste fino podendo melhorar a qualidade do sinal de voz sintetizado.

A idéia aqui é diminuir o número de coeficientes do filtro PCA de forma que após a adição do novo filtro, o número total de coeficientes seja o mesmo da estrutura original, mantendo-se assim, aproximadamente, a mesma taxa de bits/s. É importante observar que a redução da ordem do filtro PCA implica numa redução da complexidade computacional para a determinação da melhor "codeword".

3.3. CELP COM FILTRAGEM ADAPTÁVEL

Propõe-se neste trabalho a inserção de um filtro adaptável ($L(z)$) com objetivo de realizar um ajuste fino no sinal de saída do codificador CELP. Esta se constitui na primeira tentativa de reduzir a complexidade computacional do CELP convencional, procurando-se manter a mesma qualidade do sinal de voz sintetizado. A Fig. 3.1 mostra o diagrama de análise utilizado para determinação do filtro $L(z)$.



ALGOR. - Algoritmo de determinação dos coeficientes do filtro $L(z)$.

Fig. 3.1 - Diagrama em blocos da estrutura CEA para a determinação do filtro $L(z)$.

A determinação do PCA e a seleção da melhor "codeword" são obtidas da mesma forma que a descrita no Capítulo 2.

Para compensar a falta de zeros do filtro preditor, os quais são importantes, principalmente para sintetizar apropriadamente os sons nasais [19], optou-se por utilizar uma estrutura FIR transversal para o filtro $L(z)$.

Na literatura encontra-se uma série de métodos que poderiam ser

empregados para a determinação do filtro $L(z)$ [38,21,20]. Neste trabalho, optou-se pela utilização do algoritmo LMS ("Least Mean Square"), por esta apresentar uma complexidade computacional relativamente baixa, adequada para aplicações em tempo real e ser facilmente aplicável à estrutura FIR transversal proposta para o filtro $L(z)$.

Assim, temos que $L(z)$ pode ser dado por:

$$L(z) = \sum_{i=0}^{ON} b_i \cdot z^{-i} \quad (3.5)$$

onde ON é a ordem do filtro e b_i seus coeficientes.

Define-se:

$$\underline{\bar{s}}_n^T = [\bar{s}(n), \bar{s}(n-1), \dots, \bar{s}(n-ON)] \quad (3.6)$$

$$\underline{b}^T = [b_0, b_1, \dots, b_{ON}] \quad (3.7)$$

onde $\underline{\bar{s}}_n$ e \underline{b} são os vetores de coeficientes do filtro $L(z)$ e do sinal de voz sintetizado (saída do filtro PCA), respectivamente.

Utilizando-se o algoritmo LMS para atualizar os coeficientes b_i , tem-se:

$$\underline{b}_{n+1} = \underline{b}_n + 2 \cdot \mu \cdot e(n) \cdot \underline{\bar{s}}_n \quad (3.8)$$

Onde \underline{b}_{n+1} e \underline{b}_n são os vetores representando os coeficientes atuais e os coeficientes da iteração anterior, respectivamente.

O erro $e(n)$ na iteração n é dado por:

$$e(n) = s(n) - \hat{s}(n) \quad (3.9)$$

Onde $\hat{s}(n)$ é o sinal de saída do filtro $L(z)$ e $s(n)$ o sinal de voz original.

Na implementação da equação (3.8) foi utilizado $\mu = 0,025 \cdot \mu_c$.

Onde μ_c é o fator de convergência crítico [39], que é dado por:

$$\mu_c = \frac{1}{(ON+1) \cdot E[\bar{s}^2(n)]} \quad (3.10)$$

Este valor $\mu/\mu_c = 0,025$ foi determinado experimentalmente através de diversas simulações e revelou-se bem adequado no que diz respeito à velocidade de convergência do algoritmo para a maioria dos segmentos de voz utilizados.

3.3.1 RESULTADOS DAS SIMULAÇÕES DO CELP UTILIZANDO FILTRAGEM ADAPTÁVEL

Para compatibilizar os resultados destas simulações com os resultados das anteriores é estabelecido que:

- A complexidade computacional para a determinação dos coeficientes do filtro $L(z)$ não será aqui considerada por esta ser desprezível frente às operações necessárias à seleção da melhor "codeword". Utilizando-se apenas 10 a 20 iterações para o algoritmo LMS obtém-se muito bons resultados para a determinação dos coeficientes do filtro $L(z)$.

- A estratégia utilizada será manter o número total de parâmetros, distribuindo 50% dos coeficientes para o PCA e 50% para $L(z)$ ($OD = ON$).

● Para possibilitar uma fácil comparação dos resultados, foram mantidos os mesmos comprimentos dos quadros e sub-quadros.

● Não foi utilizada a memória do filtro $L(z)$.

Os resultados obtidos por simulação, utilizando-se a estrutura apresentada na Fig. 3.1, são mostrados na Tabela 3.2.

Tabela 3.2 - Resultados das simulações da estrutura CELP com filtragem adaptável.

quadro ms	sub-quadro ms	atraso ms	EMQ_S	RSR_S [dB]	OD	ON	ordem de complexidade
10	1,25	12,5+Tp	0,056	12,51	3	3	$O(OD.L+L)$
10	2,5	12,5+Tp	0,108	9,67	3	3	$O(OD.L+L)$
10	5	12,5+Tp	0,208	6,81	3	3	$O(OD.L+L)$
20	1,25	25+Tp	0,054	12,69	6	6	$O(OD.L+L)$
20	2,5	25+Tp	0,117	9,30	6	6	$O(OD.L+L)$
20	5	25+Tp	0,192	7,16	6	6	$O(OD.L+L)$

ON - Ordem de $L(z)$.

OD - Ordem do preditor.

$O(.)$ - Ordem de complexidade de análise por amostra do sinal de voz. A ordem de complexidade é dada em termos de número de operações de multiplicação e adição.

Tp - Tempo de processamento função: da forma de realização, da complexidade de análise/síntese e do processador utilizado no codificador.

L - Número de "codewords" do "codebook".

Como pode ser observado, através da Tabela 3.2, a complexidade computacional, em relação ao CELP convencional foi reduzida. Pode-se notar também que a RSR_S foi mantida. A razão da manutenção da RSR_S é devido à presença do PCA que, mesmo reduzido, ainda consegue modelar as componentes de ruído das "codewords" de tal forma a manter alguma correlação entre o sinal sintetizado pelo PCA e o sinal de voz original.

O atraso resultante da estrutura é o mesmo, já que não foram modificados os tamanhos dos quadros para a determinação do PCA.

3.4. CODIFICAÇÃO DA EXCITAÇÃO COM FILTRAGEM ADAPTÁVEL

A idéia aqui é modificar o sinal de excitação de tal forma que a distribuição espectral deste sinal se aproxime da distribuição espectral do ruído branco passado através do filtro PCA. Assim, poder-se-ia eliminar o PCA e, conseqüentemente, reduzir a complexidade computacional na fase de análise para a referida estrutura. No caso do CELP convencional são necessárias $OD \cdot L + L$ operações aritméticas no procedimento de procura da melhor "codeword" para cada amostra do sinal de voz, onde OD é a ordem do filtro PCA e L é o tamanho do "codebook". Quando $OD = 0$, este número de operações reduz-se para apenas L operações aritméticas.

A princípio é efetuado um estudo desta estrutura mantendo-se o mesmo sinal de excitação utilizado nas simulações anteriores, posteriormente, serão propostas novas formas de sinais de excitação que apresentem um espectro em frequência que se aproxime mais do desejado.

Doravante, a nova estrutura, sem o PCA, será denominada CEA (Codificação da Excitação com filtragem Adaptável). A Fig. 3.2 apresenta o diagrama em blocos do procedimento de seleção da melhor "codeword" para esta estrutura.

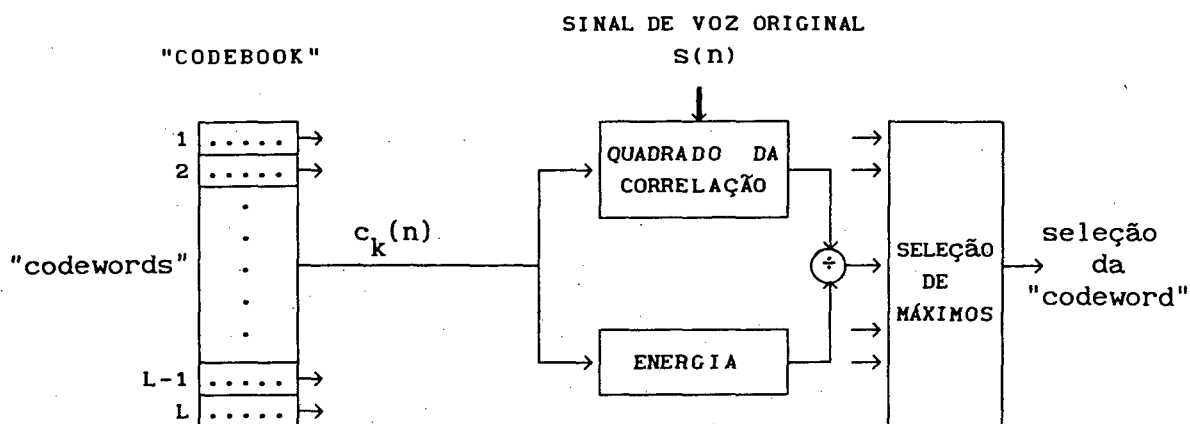


Fig. 3.2 - Diagrama em blocos para determinação da melhor "codeword" da estrutura CEA.

Fazendo-se $H(z) = 1$, o que é equivalente a considerar a matriz H igual à matriz identidade na equação (2.15), determina-se o novo ganho da estrutura que é dado por:

$$g_k = \frac{\sum_{n=0}^{Ns-1} s(n) \cdot c_k(n)}{\sum_{n=0}^{Ns-1} c_k^2(n)} \quad (3.11)$$

Da mesma forma, a equação 2.16, que representa a energia do erro no processo de análise do CELP reduz-se para:

$$E_k = \sum_{n=0}^{Ns-1} s^2(n) - \frac{\left[\sum_{n=0}^{Ns-1} s(n) \cdot c_k(n) \right]^2}{\sum_{n=0}^{Ns-1} c_k^2(n)} \quad (3.12)$$

onde k é o índice da "codeword".

A melhor "codeword" é obtida para o índice k que leve o segundo termo do lado direito da equação (3.12) ao máximo.

Para a determinação do filtro $L(z)$ é utilizada como sinal de excitação a "codeword" selecionada, escalada pelo respectivo ganho conforme o diagrama em blocos mostrado na Fig. 3.3.

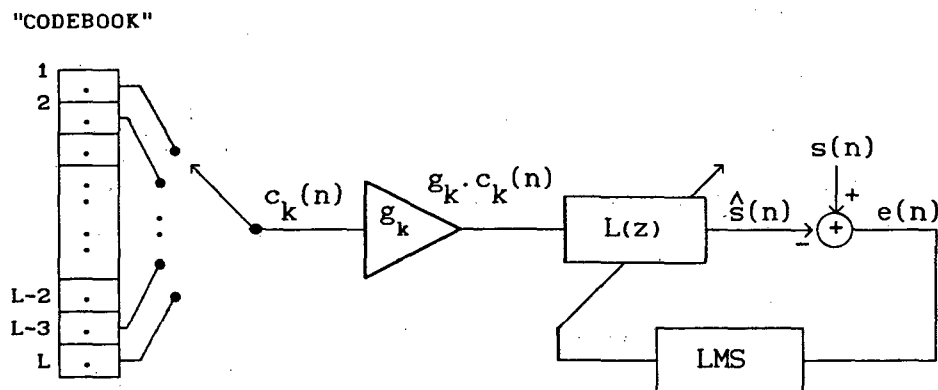


Fig. 3.3 - Diagrama em blocos para determinação do filtro $L(z)$.

Para a determinação dos coeficientes do filtro $L(z)$, seguem-se os mesmos passos apresentados na estrutura anterior, sendo que, na estrutura CEA a excitação do filtro $L(z)$ é dada por $g_k \cdot c_k(n)$.

O diagrama de síntese da estrutura CEA pode ser visto na Fig. 3.4.

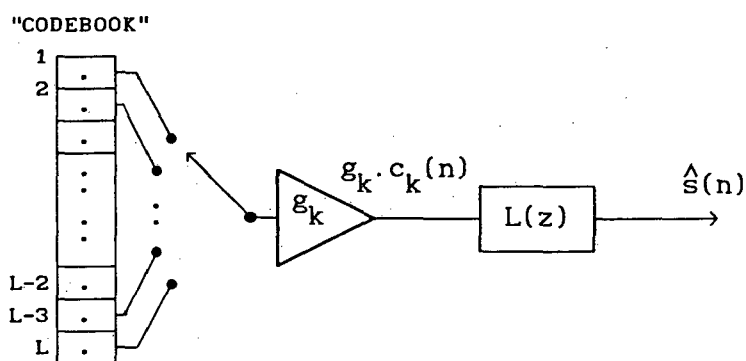


Fig. 3.4 - Diagrama em blocos para a síntese de voz utilizando a estrutura CEA.

O procedimento de síntese nesta estrutura é bastante simplificado, e o sinal de voz sintetizado a cada quadro pode ser obtido por:

$$\hat{s}(n) = g_k \cdot \sum_{i=0}^{ON} b_i \cdot c_k(n-i) \quad (3.13)$$

Para compatibilizar os resultados destas e das próximas simulações com os resultados das simulações do CELP apresentadas na seção 3.2.2, foram estabelecidos os mesmos critérios da seção 3.3.1, sendo que, agora, não será mais utilizado o filtro PCA.

A Tabela 3.3 apresenta os resultados das simulações quando utilizada a estrutura CEA.

Tabela 3.3 - Resultados das simulações da estrutura CEA.

quadro ms	sub-quadro ms	atraso ms	EMQ _S	RSR _S [dB]	OD	ON	ordem de complexidade
10	1,25	1,25+Tp	0,282	5,50	0	0	O(L)
10	2,5	2,5+Tp	0,531	2,75	0	0	O(L)
10	5	5+Tp	0,764	1,17	0	0	O(L)
20	1,25	1,25+Tp	0,282	5,50	0	0	O(L)
20	2,5	2,5+Tp	0,531	2,75	0	0	O(L)
20	5	5+Tp	0,764	1,17	0	0	O(L)

a) Resultados sem o filtro PCA e sem o filtro L(z).

quadro ms	sub-quadro ms	atraso ms	EMQ _S	RSR _S [dB]	OD	ON	ordem de complexidade
10	1,25	10+Tp	0,126	9,00	0	6	O(L)
10	2,5	10+Tp	0,257	5,73	0	6	O(L)
10	5	10+Tp	0,458	3,39	0	6	O(L)
20	1,25	20+Tp	0,116	9,34	0	12	O(L)
20	2,5	20+Tp	0,247	6,07	0	12	O(L)
20	5	20+Tp	0,431	3,66	0	12	O(L)

b) Resultados sem o PCA e com o filtro L(z).

ON - Ordem de L(z).

O(.) - Ordem de complexidade de análise por amostra do sinal de voz. A ordem de complexidade é dada em termos de número de operações de multiplicação e adição.

Tp - Tempo de processamento função: da forma de realização, da complexidade de análise/síntese e do processador utilizado no codificador.

L - Número de "codewords" do "codebook".

Observando a Tabela 3.3, nota-se que a complexidade computacional foi reduzida em relação aos resultados anteriores. Também o atraso pode ser reduzido tendo-se em vista que o tempo de processamento, T_p , está relacionado diretamente com o grau de complexidade da estrutura. A redução da RSR_S é atribuída ao tipo de sinal particular utilizado como excitação. Nas estruturas em que o filtro PCA é mantido, o sinal de excitação, ruído branco com distribuição gaussiana, era passado através do filtro PCA, o qual é responsável pela correlação existente entre o sinal de voz sintetizado e o sinal de voz original. Agora sem o filtro PCA a excitação é utilizada diretamente como "estimativa" do sinal de voz, o que leva a se obter, no caso deste tipo particular de sinal de excitação, uma correlação muito fraca entre estes dois sinais.

Observou-se que, à proporção que se reduz o número de coeficientes do filtro PCA, algumas informações (envoltória espectral do sinal de voz) relativas a ele devem ser repassadas ao "codebook".

Tudo isso conduz à proposição de novos tipos de sinais de excitação, os quais apresentem uma correlação mais forte com o sinal de voz original.

3.5. DETERMINAÇÃO DA EXCITAÇÃO DO CEA

O melhor conjunto de sinais de excitação para o CEA consiste na composição de um sinal genérico que contenha todas as variações possíveis dos sinais de voz. Porém, como a quantidade de sons, mesmo em uma única língua, é quase ilimitada, não é possível a obtenção de um "codebook" composto por este tipo de sinal genérico que seja viável para uma aplicação prática. Para contornar este problema, existem formas alternativas de sinais que podem representar, convenientemente, uma aproximação deste tipo de sinal.

Uma proposta para o sinal de excitação do CEA é a utilização de um sinal sinusoidal com frequência variável. Esta proposição é baseada no fato de que os sinais sinusoidais são elementos fundamentais de uma grande variedade de sinais. A idéia aqui é compor um segmento de um sinal de voz através da justaposição de pequenos segmentos (10 a 40 amostras) de sinais elementares obtidos de um conjunto de sinais pseudo-senoidais que permitam a representação das diversas nuances e variações do sinal de voz.

Doravante, o "codebook" composto por "codewords" independentes será substituído por um vetor de excitação denominado de "codevector". Este "codevector" será composto por "codewords" dependentes umas das outras de tal forma que as "codewords" contidas no "codevector" estejam deslocadas, seqüencialmente, de uma amostra ao longo deste. A motivação que deu origem a esta mudança no "codebook" foi a redução da quantidade de memória necessária para armazenar a excitação (no caso de "codewords" de 40 amostras, o "codevector" é 40 vezes menor do que o "codebook") e a possibilidade de aplicação direta deste tipo de excitação à estrutura CEA.

Como os sinais de voz, em aplicações práticas, são limitados em frequência, de um lado pela frequência máxima dada pelo teorema da amostragem e de outro pela frequência mínima dada pelos formantes, (elementos fundamentais que distinguem os fonemas) é possível determinar um intervalo de variação de frequência do sinal pseudo-sinusoidal que será utilizado na composição do "codevector".

Como a maior concentração de energia do sinal de voz está em torno do primeiro formante, propõe-se utilizar este valor como frequência inicial do sinal pseudo-sinusoidal. Segundo levantamentos através de espectrogramas de sinais de voz, tem-se que o primeiro formante está situado em torno de algumas centenas de Hz [3,2,19], tendo seu valor médio aproximadamente igual a 500Hz.

Sendo assim, será adotada como frequência inicial do sinal pseudo-sinusoidal este valor de frequência.

A frequência final é adotada como sendo a metade da frequência de "Nyquist", ou seja, 4kHz (para $f_a=8\text{kHz}$) pois o sistema de aquisição analógico/digital utilizado praticamente elimina as frequências superiores a este valor.

Adotou-se uma lei de variação linear do período em função do número de ordem de amostra da "codeword" para implementar a variação de frequências do sinal pseudo-periódico. Desta forma, serão enfatizadas as frequências de maior energia do sinal que estão localizadas próximo do primeiro formante.

O sinal pseudo-senoidal pode ser representado pela seguinte expressão:

$$c(\ell) = A \cdot \text{sen} \left[\frac{2 \cdot \pi \cdot (\ell-1)}{T(\ell)} \right] \quad , 1 \leq \ell \leq L + N_s \quad (3.14)$$

Onde A é a amplitude, $c(\ell)$ e $T(\ell)$ são a ℓ -ésima amostra e o ℓ -ésimo pseudo-período do sinal, respectivamente.

Pelo que foi proposto anteriormente, obtém-se, para o período do sinal pseudo-senoidal, a seguinte expressão:

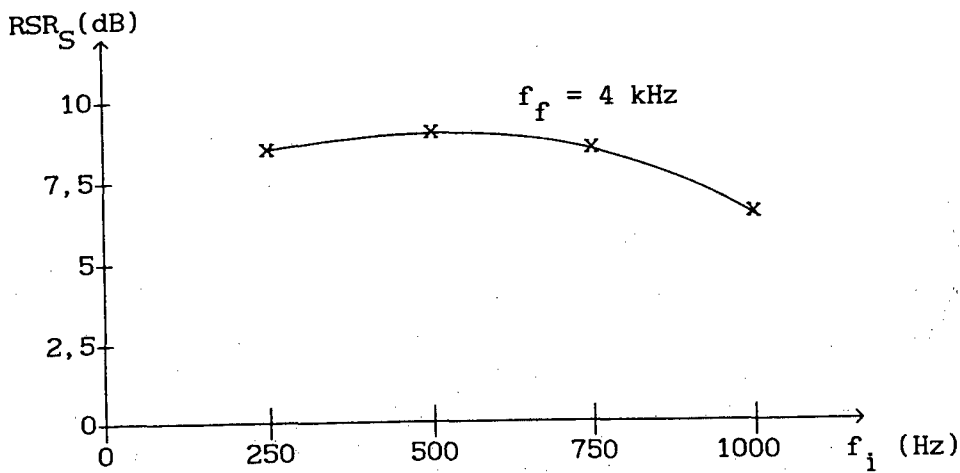
$$T(\ell) = \frac{f_a}{L+N_s-1} \cdot \left[\left[\frac{1}{f_i} - \frac{1}{f_f} \right] \cdot \ell + \left[\frac{L+N_s}{f_f} - \frac{1}{f_i} \right] \right] \quad (3.15)$$

Onde f_i e f_f são as frequências inicial e final do sinal pseudo-sinusoidal, respectivamente; L é o número total de "codewords" do "codevector", e em nosso caso foi utilizado $L=1024$; N_s é o número de amostras da "codeword" e f_a é a frequência de amostragem do sinal de voz original, que é igual a 8 kHz.

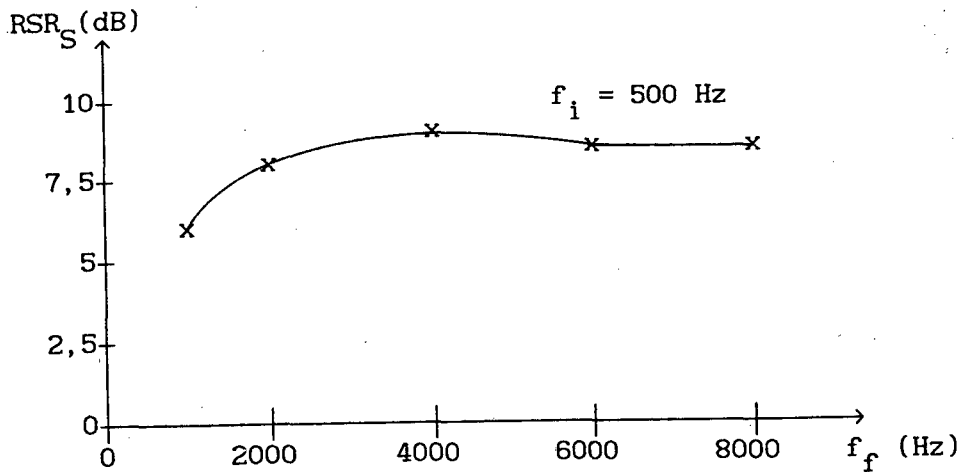
Agora, a n -ésima amostra da k -ésima "codeword", $c_k(n)$, é dada por:

$$c_k(n) = c(k+n) \quad \begin{array}{l} , 1 \leq k \leq L \\ , 0 \leq n \leq Ns-1 \end{array} \quad (3.16)$$

Os gráficos da Fig. 3.5 apresentam resultados de simulações que ilustram o comportamento da RSR_S função dos valores adotados para f_i e f_f .



a) Para $f_f = 4 \text{ kHz}$;



b) Para $f_i = 500 \text{ Hz}$;

Fig. 3.5 - Gráfico do comportamento da RSR_S para os diversos valores de f_i e f_f adotados na equação (3.14).

Para a obtenção dos resultados das simulações apresentados na Fig. 3.5, foram adotados os seguintes parâmetros:

- "codevector" com $L = 1024$ "codewords";
- "codeword" com $N_s = 20$ amostras;
- segmento de voz relativo a um locutor masculino com duração aproximada de 14 segundos e foneticamente balanceado;
- quadro de análise de voz com $N_a = 160$ amostras;
- quadro de medição do $EMQ_Q(j)$ com $N = 80$ amostras;
- ordem do filtro $L(z)$ ON = 12.

Segundo as curvas da Fig. 3.5 constata-se que a RSR_S é máxima quando $f_i = 500$ Hz e $f_f = 4$ kHz. Na medida em que nos afastamos destes limites a RSR_S diminui. Porém, para uma avaliação mais rigorosa, seria necessário efetuar um levantamento exaustivo para as diversas combinações entre f_i e f_f com diversos arquivos de voz. Todavia, isto não foi possível devido à necessidade de elevado número de simulações para a realização deste levantamento.

Por outro lado, cabe aqui mencionar que foi realizada uma grande quantidade de simulações com diversos arquivos de sinais de voz, para diversos valores de f_i e f_f , e com diversas outras formas de variação de $T(\ell)$. Os resultados destas simulações confirmam que a escolha do sinal representado pela equação (3.14) com $T(\ell)$ regido pela equação (3.15) para $f_i = 500$ Hz e $f_f = 4$ kHz é bastante adequada para a aplicação pretendida, ou seja, compor o "codevector" da estrutura CEA.

O gráfico da Fig. 3.6 mostra a distribuição espectral do sinal do "codevector" obtido pela equação (3.14) para $T(\ell)$ determinado pela expressão (3.15) com $f_a = 8$ kHz, $f_i = 500$ Hz, $f_f = f_a/2$, $L = 1024$, $N_s = 20$ e $A = 2048$.

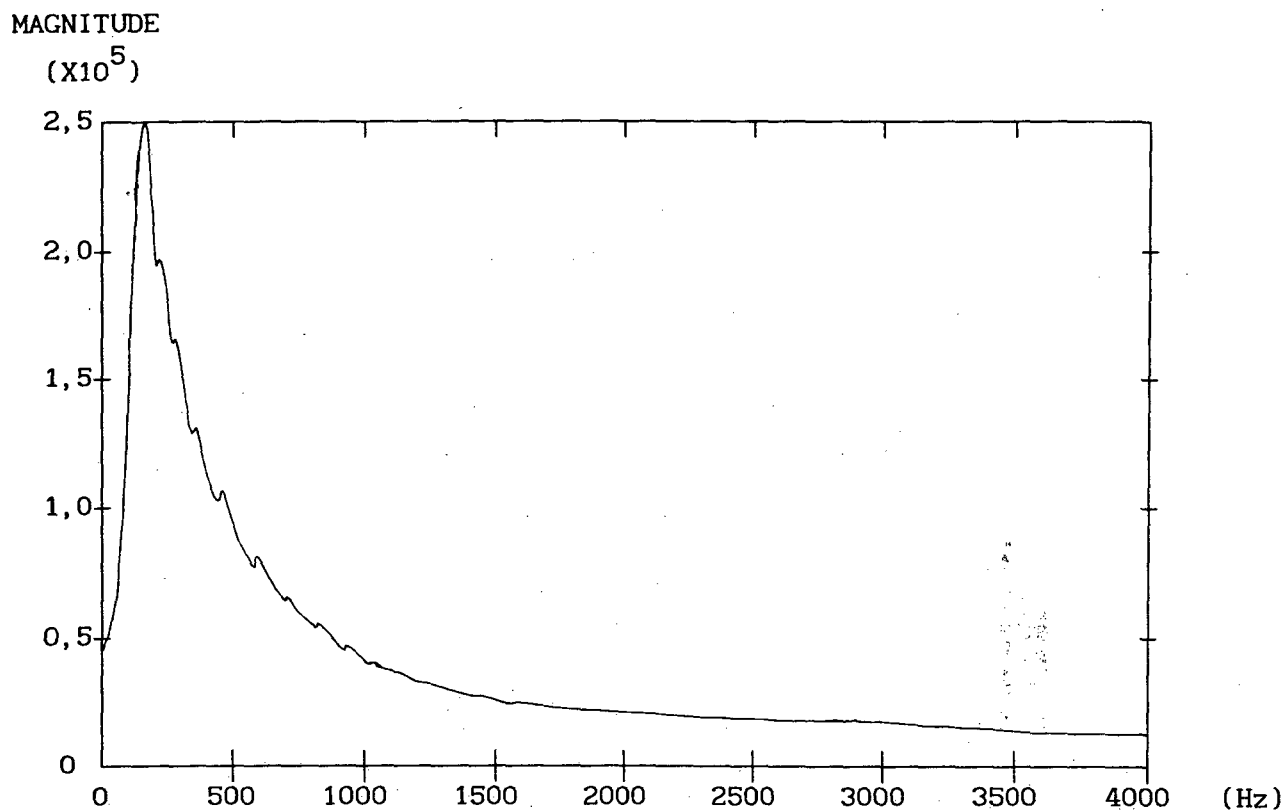


Fig. 3.6 - Espectro do sinal pseudo-sinusoidal (equação (3.14)).

Neste gráfico nota-se, claramente, uma forte presença das baixas frequências, as quais forçosamente acentuam os sons graves conforme observado nitidamente, durante as diversas simulações de síntese de voz.

3.5.1. DESEMPENHO DA ESTRUTURA CEA PARA AS DIVERSAS ORDENS DO FILTRO $L(z)$

Na avaliação do desempenho da estrutura CEA para as diversas ordens do filtro $L(z)$, foi realizado um conjunto de simulações tomando por base o "codevector" composto pelo sinal pseudo-sinusoidal discutido anteriormente.

A Tabela 3.4 apresenta os resultados destas simulações para "codewords" com 10, 20 e 40 amostras.

Tabela 3.4 - Resultados da estrutura CEA para diversas ordens do filtro $L(z)$, para "codewords" de 10, 20 e 40 amostras.

quadro ms	sub-quadro ms	atraso ms	EMQ_S	RSR_S [dB]	ON	ordem de complexidade
10 ou 20	1,25	$1,25 + T_p$	0,069	11,60	0	O(L)
10	1,25	$10 + T_p$	0,068	11,65	1	O(L)
10	1,25	$10 + T_p$	0,067	11,76	3	O(L)
10	1,25	$10 + T_p$	0,064	11,93	6	O(L)
10	1,25	$10 + T_p$	0,060	12,24	12	O(L)
20	1,25	$20 + T_p$	0,067	11,71	3	O(L)
20	1,25	$20 + T_p$	0,066	11,81	6	O(L)
20	1,25	$20 + T_p$	0,063	12,00	12	O(L)
20	1,25	$20 + T_p$	0,060	12,20	24	O(L)

a) "codewords" de 10 amostras.

quadro ms	sub-quadro ms	atraso ms	EMQ_S	RSR_S [dB]	ON	ordem de complexidade
10 ou 20	2,5	$2,5 + T_p$	0,140	8,55	0	O(L)
10	2,5	$10 + T_p$	0,138	8,61	1	O(L)
10	2,5	$10 + T_p$	0,136	8,66	3	O(L)
10	2,5	$10 + T_p$	0,131	8,82	6	O(L)
10	2,5	$10 + T_p$	0,122	9,13	12	O(L)
20	2,5	$20 + T_p$	0,137	8,62	3	O(L)
20	2,5	$20 + T_p$	0,135	8,70	6	O(L)
20	2,5	$20 + T_p$	0,129	8,89	12	O(L)
20	2,5	$20 + T_p$	0,121	9,19	24	O(L)

b) "codewords" de 20 amostras.

quadro ms	sub-quadro ms	atraso ms	EMQ_S	RSR_S [dB]	ON	ordem de complexidade
10 ou 20	5	5 + T_p	0,306	5,14	0	O(L)
10	5	10 + T_p	0,304	5,17	1	O(L)
10	5	10 + T_p	0,301	5,21	3	O(L)
10	5	10 + T_p	0,297	5,27	6	O(L)
10	5	10 + T_p	0,279	5,55	12	O(L)
20	5	20 + T_p	0,303	5,19	3	O(L)
20	5	20 + T_p	0,300	5,23	6	O(L)
20	5	20 + T_p	0,290	5,37	12	O(L)
20	5	20 + T_p	0,277	5,58	24	O(L)

c) "codewords" de 40 amostras.

ON - Ordem do filtro $L(z)$.

O(.) - Ordem de complexidade de análise por amostra do sinal de voz. A ordem de complexidade é dada em termos de número de operações de multiplicação e adição.

T_p - Tempo de processamento função: da forma de realização, da complexidade de análise/síntese e do processador utilizado no codificador.

L - Número de "codewords" do "codebook".

Como pode ser observado nos resultados apresentados na Tabela 3.4, a contribuição do filtro $L(z)$ cresce à medida em que aumentamos a ordem do mesmo. Também foi observado que em alguns quadros a contribuição na RSR_S devido ao filtro $L(z)$ é muito pequena.

Este fato nos induz a pensar em uma estrutura que possibilite incluir ou não o filtro $L(z)$ dependendo de sua contribuição na RSR_0 . Isto certamente levará a uma redução na taxa de transmissão, já que o número de coeficientes a transmitir é menor.

Este estudo levou aos resultados da Tabela 3.5, onde o filtro $L(z)$ foi suprimido dos quadros em que sua contribuição na RSR_Q era inferior a 5%.

Tabela 3.5 - Percentual de quadros sem necessidade do filtro $L(z)$, para "codewords" de 10, 20 e 40 amostras.

quadro ms	sub-quadro ms	$L(z)=1$ (%)	atraso ms	EMQ_S	RSR_S [dB]	ON	ordem de complexidade
10	1,25	99,36	10+Tp	0,069	11,61	1	O(L)
10	1,25	94,36	10+Tp	0,068	11,67	3	O(L)
10	1,25	82,31	10+Tp	0,065	11,84	6	O(L)
10	1,25	71,99	10+Tp	0,061	12,17	12	O(L)
20	1,25	97,13	20+Tp	0,069	11,63	3	O(L)
20	1,25	89,55	20+Tp	0,068	11,69	6	O(L)
20	1,25	77,08	20+Tp	0,065	11,90	12	O(L)
20	1,25	71,63	20+Tp	0,061	12,13	24	O(L)

a) "codewords" de 10 amostras.

quadro ms	sub-quadro ms	$L(z)=1$ (%)	atraso ms	EMQ_S	RSR_S [dB]	ON	ordem de complexidade
10	2,5	97,78	10+Tp	0,139	8,57	1	O(L)
10	2,5	93,48	10+Tp	0,138	8,60	3	O(L)
10	2,5	77,87	10+Tp	0,133	8,74	6	O(L)
10	2,5	59,32	10+Tp	0,124	9,07	12	O(L)
20	2,5	97,42	20+Tp	0,139	8,57	1	O(L)
20	2,5	90,40	20+Tp	0,137	8,62	3	O(L)
20	2,5	70,20	20+Tp	0,132	8,80	6	O(L)
20	2,5	47,85	20+Tp	0,122	9,13	12	O(L)

b) "codewords" de 20 amostras.

quadro ms	sub-quadro ms	$L(z)=1$ (%)	atraso ms	EMQ_S	RSR_S [dB]	ON	ordem de complexidade
10	5	96,63	10+Tp	0,305	5,15	1	O(L)
10	5	90,40	10+Tp	0,303	5,19	3	O(L)
10	5	81,73	10+Tp	0,300	5,24	6	O(L)
10	5	52,15	10+Tp	0,281	5,52	12	O(L)
20	5	93,27	20+Tp	0,304	5,17	1	O(L)
20	5	89,54	20+Tp	0,302	5,20	3	O(L)
20	5	66,77	20+Tp	0,293	5,33	6	O(L)
20	5	46,99	20+Tp	0,278	5,56	12	O(L)

c) "codewords" de 40 amostras.

ON - Ordem do filtro $L(z)$.

O(.) - Ordem de complexidade de análise por amostra do sinal de voz. A ordem de complexidade é dada em termos de número de operações de multiplicação e adição.

Tp - Tempo de processamento função: da forma de realização, da complexidade de análise/síntese e do processador utilizado no codificador.

L - Número de "codewords" do "codebook".

Pode ser observado destes resultados, que na proporção em que é reduzido o número de coeficientes do filtro $L(z)$, aumenta o percentual de quadros em que esta filtragem não apresenta melhora significativa na RSR_Q . O mesmo é observado à medida em que se diminui o número de amostras das "codewords". Estes resultados mostram que é elevado o número de quadros em que não é necessária a filtragem por $L(z)$. Esta conclusão possibilita uma avaliação do codificador sem a utilização do filtro $L(z)$. A Tabela 3.6 apresenta os resultados das simulações para os diversos comprimentos dos sub-quadros nesta condição.

Tabela 3.6 - Resultados das simulações da estrutura CEA dispensando-se o uso do filtro $L(z)$.

sub-quadro ms	atraso ms	EMO_S	RSR_S [dB]	ordem de complexidade
1,25	$1,25 + T_p$	0,069	11,60	$O(L)$
2,5	$2,5 + T_p$	0,140	8,55	$O(L)$
5	$5 + T_p$	0,121	5,14	$O(L)$

$O(.)$ - Ordem de complexidade de análise por amostra do sinal de voz. A ordem de complexidade é dada em termos de número de operações de multiplicação e adição.

T_p - Tempo de processamento função: da forma de realização, da complexidade de análise/síntese e do processador utilizado no codificador.

L - Número de "codewords" do "codebook".

Dispensando-se o filtro $L(z)$ é necessário transmitir somente o índice da "codeword" selecionada e o respectivo ganho, reduzindo-se sobremaneira a taxa de transmissão. Outra grande vantagem na operação desta estrutura é o reduzido atraso requerido na fase de análise, já que não é mais necessário utilizar um quadro do sinal de voz sintetizado para a determinação dos coeficientes do filtro. Desta maneira, tem-se que o atraso pode ser de 10 a 20 vezes menor do que aquele do CELP convencional. Nesta condição, tem-se uma ligeira degradação na RSR_S , porém, por outro lado, têm-se ganhos consideráveis quanto à complexidade computacional e atraso requeridos na fase de análise.

A seguir são apresentados alguns resultados da estrutura CEA, sem o uso do filtro $L(z)$, onde os parâmetros de codificação foram quantizados.

Tabela 3.7 - Resultados das simulações do codificador CEA quantizado sem o uso do filtro $L(z)$.

taxa bits/s	sub-quadro ms	atraso ms	EMQ_S	RSR_S [dB]	ordem de complexidade
12.000	1,25	1,25 + T_p	0,084	10,76	O(L)
6.000	2,5	2,5 + T_p	0,151	8,20	O(L)
3.000	5	5 + T_p	0,316	5,00	O(L)

O(.) - Ordem de complexidade de análise por amostra do sinal de voz. A ordem de complexidade é dada em termos de número de operações de multiplicação e adição.

T_p - Tempo de processamento função: da forma de realização, da complexidade de análise/síntese e do processador utilizado no codificador.

L - Número de "codewords" do "codebook".

Para quantização do ganho foram utilizados 5 bits, um para o sinal e os restantes para a amplitude. No processo de quantização da amplitude do ganho foi utilizada uma escala logarítmica. Para representar o endereço da "codeword" selecionada são necessários 10 bits, tendo-se em vista que o "codebook" utilizado é composto por 1024 "codewords".

Outras formas de quantização do ganho podem ser empregadas, bem como o uso de um número de bits ainda menor para quantização do mesmo, reduzindo, desta forma, ainda mais, a taxa de transmissão. Pode ser observado destes resultados que mesmo após a quantização do ganho a RSR_S não sofreu uma degradação significativa.

3.6. MÉTODO ALTERNATIVO PARA DETERMINAÇÃO DOS COEFICIENTES DO FILTRO $L(z)$

Uma forma alternativa para solucionar o problema da determinação dos coeficientes do filtro $L(z)$, pode ser dada pelo seguinte procedimento.

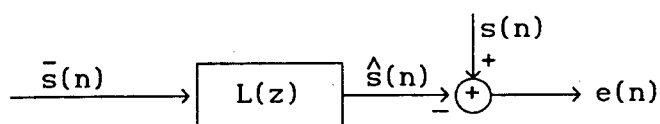


Fig. 3.7 - Diagrama em blocos para determinação dos coeficientes do filtro $L(z)$.

Analisando-se o diagrama em blocos mostrado na Fig. 3.7 constata-se que é possível determinar analiticamente os coeficientes do filtro $L(z)$ com base no conhecimento dos sinais $\bar{s}(n)$ e $s(n)$, sinal de voz sintetizado (saída do filtro PCA) e sinal de voz original, respectivamente.

Considerando a função de transferência do filtro $L(z)$ dada pela equação (3.5), e fazendo-se $b_0=1$, pode-se escrever:

$$L(z) = 1 + P(z) \quad (3.17)$$

onde:

$$P(z) = \sum_{i=1}^{ON} b_i \cdot z^{-i} \quad (3.18)$$

Assim, o diagrama em blocos apresentado na Fig. 3.7 pode ser rearranjado na seguinte forma:

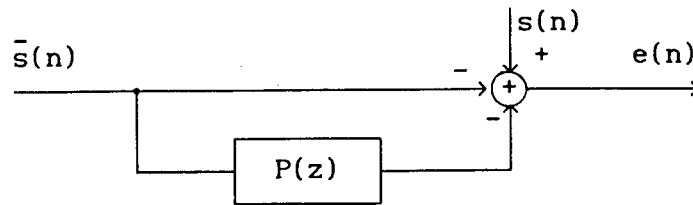


Fig. 3.8 - Diagrama da Fig. 3.7 rearranjado.

O erro $e(n)$ na saída pode ser escrito como:

$$e(n) = s(n) - \bar{s}(n) - \sum_{i=1}^{ON} b_i \cdot \bar{s}(n-i) \quad (3.19)$$

A energia do erro, E , em um quadro de N_a amostras, é dada por:

$$E = \sum_{n=0}^{N_a-1} e^2(n) \quad (3.20)$$

A determinação dos coeficientes ótimos b_i de $P(z)$ é realizada através da minimização da energia do erro dada pela equação (3.20).

Assim,

$$\frac{\partial E}{\partial b_k} = 0 \quad , 1 \leq k \leq ON \quad (3.21)$$

Portanto:

$$\sum_{i=1}^{ON} b_i \cdot \sum_{n=0}^{Na-1} \bar{s}(n-k) \cdot \bar{s}(n-i) = \sum_{n=0}^{Na-1} \bar{s}(n-k) \cdot \Delta(n) \quad (3.22)$$

onde:

$$\Delta(n) = s(n) - \bar{s}(n) \quad (3.23)$$

Fazendo-se:

$$C_{\bar{S}\bar{S}}(k, i) = \sum_{n=0}^{Na-1} \bar{s}(n-k) \cdot \bar{s}(n-i) \quad , 1 \leq k, i \leq ON \quad (3.24)$$

e

$$r_{\bar{S}\Delta}^-(k, 0) = \sum_{n=0}^{Na-1} \bar{s}(n-k) \cdot \Delta(n) \quad , 1 \leq k \leq ON \quad (3.25)$$

tem-se agora a (3.22) dada pela seguinte expressão:

$$\sum_{i=1}^{ON} b_i \cdot C_{\bar{S}\bar{S}}(k, i) = r_{\bar{S}\Delta}^-(k, 0) \quad (3.26)$$

Como pode ser visto da expressão (3.26), a otimização leva a um conjunto de equações lineares ($C \cdot \underline{b} = \underline{r}$) que pode ser facilmente resolvido se a matriz C não for mal condicionada.

Se $N_a \gg ON$ a matriz C é aproximadamente uma matriz Toeplitz o que facilita sobremaneira a solução do sistema de equações. Assim, pode-se então utilizar, para solucionar o sistema de equações dado por (3.26), algoritmos que reduzam substancialmente a complexidade computacional requerida. Desta forma, pode-se utilizar a decomposição de Cholesky [38], ou ainda, os métodos propostos por Levinson [40] e Robinson [41].

É importante observar que a utilização de uma janela de ponderação tanto em $s(n)$ como em $\bar{s}(n)$ podem melhorar substancialmente o condicionamento do sistema linear. Janelas clássicas como a de Hamming, Hanning, etc. podem, então, ser utilizadas.

Objetivando a avaliação deste procedimento foram efetuadas algumas simulações para a determinação dos coeficientes do filtro $L(z)$ utilizando-se o procedimento anteriormente descrito. Os resultados ora apresentados foram obtidos para um segmento de voz estabelecendo-se os mesmos parâmetros utilizados na seção 3.5. Neste caso obteve-se das simulações:

$RSR_S = 8,89$ com a filtragem $L(z)$ utilizando o algoritmo LMS;

$RSR_S = 8,84$ com a filtragem $L(z)$ utilizando o procedimento proposto;

Nota-se que foi obtido uma RSR_S menor para o procedimento proposto em relação ao obtido através do algoritmo LMS. Esta redução é devida aos problemas de instabilidade encontrados na solução do sistema de equações lineares. Caso fossem suprimidos os quadros que apresentaram problemas, certamente, o procedimento proposto levaria a melhores resultados, já que esse conduz à determinação dos coeficientes ótimos.

Como exemplo, são mostrados a seguir os vetores de coeficientes do filtro $L(z)$ (\underline{b}_1^T e \underline{b}_2^T) para um quadro "j" de voz arbitrário, que ilustram os resultados obtidos por simulação utilizando-se os dois procedimentos. Aqui \underline{b}_1^T refere-se ao uso do algoritmo LMS e \underline{b}_2^T ao procedimento proposto.

$$\underline{b}_1^T = [1,000; -0,070; 0,043; -0,064; 0,063; 0,048; 0,029; -0,292; 0,105; 0,151; -0,181; 0,219; -0,137]$$

$$\underline{b}_2^T = [1,000; -0,027; 0,095; -0,094; 0,077; 0,031; 0,062; -0,338; 0,133; 0,181; -0,249; 0,297; -0,169]$$

Neste caso tem-se que:

$$RSR_Q(j) = 9,91 \text{ com a filtragem } L(z) \text{ utilizando o algoritmo LMS;}$$

$$RSR_Q(j) = 9,94 \text{ com a filtragem } L(z) \text{ utilizando o procedimento proposto;}$$

A matriz C para este quadro é:

$$C = \begin{bmatrix} 33523 & 31516 & 28385 & 24765 & 20988 & 17404 & 14359 & 12045 & 10476 & 9589 & 9315 & 8577 \\ 31516 & 33522 & 31511 & 28383 & 24769 & 20992 & 17402 & 14354 & 12045 & 10481 & 9593 & 9324 \\ 28385 & 31511 & 33497 & 31501 & 28403 & 24789 & 20982 & 17377 & 14354 & 12070 & 10501 & 9638 \\ 24765 & 28383 & 31501 & 33493 & 31509 & 28411 & 24785 & 20972 & 17377 & 14364 & 12078 & 10519 \\ 20988 & 24769 & 28403 & 31509 & 33477 & 31493 & 28419 & 24805 & 20972 & 17357 & 14348 & 12042 \\ 17404 & 20992 & 24789 & 28411 & 31493 & 33461 & 31501 & 28439 & 24805 & 20952 & 17341 & 14312 \\ 14359 & 17402 & 20982 & 24785 & 28419 & 31501 & 33457 & 31491 & 28439 & 24815 & 20960 & 17359 \\ 12045 & 14354 & 17377 & 20972 & 24805 & 28439 & 31491 & 33432 & 31491 & 28464 & 24835 & 21005 \\ 10476 & 12045 & 14354 & 17377 & 20972 & 24805 & 28439 & 31491 & 33432 & 31491 & 28464 & 24835 \\ 9589 & 10481 & 12070 & 14364 & 17357 & 20952 & 24815 & 28464 & 31491 & 33407 & 31471 & 28419 \\ 9315 & 9593 & 10501 & 12078 & 14348 & 17341 & 20960 & 24835 & 28464 & 31471 & 33391 & 31435 \\ 8577 & 9324 & 9638 & 10519 & 12042 & 14312 & 17359 & 21005 & 24835 & 28419 & 31435 & 33310 \end{bmatrix}$$

Nota-se, como havíamos afirmado anteriormente, que a matriz é praticamente uma matriz Toeplitz.

Este procedimento apesar de apresentar bons resultados e ter demonstrado eficiência na solução, cerca de 5 vezes mais rápido do que o algoritmo LMS, não foi utilizado nas simulações por apresentar problemas de instabilidade devido ao mal condicionamento da matriz C em alguns quadros.

3.7. CONCLUSÃO

A seguir é apresentado um resumo dos resultados mais relevantes. Pode-se notar claramente as vantagens da utilização da estrutura CEA em relação a do CELP convencional em termos de complexidade computacional. Como exemplo, para o CELP convencional, considerando a ordem do preditor $OD=12$, e o número de "codewords" do "codebook" $L=1024$, tem-se um total de 106 milhões de operações aritméticas para cada segundo de sinal de voz processado. Utilizando-se a estrutura CEA este valor cai para 8 milhões de operações. Nota-se também uma grande vantagem na utilização da estrutura CEA dispensando-se o filtro $L(z)$, principalmente, em relação ao atraso requerido na fase de análise. Esta redução pode ser da ordem de 10 a 20 vezes em relação a requerida pelos codificadores CELP convencionais.

O procedimento de quantização implica apenas em uma ligeira degradação na RSR_S quando são utilizados 5 bits para representar o parâmetro do ganho, tendo em vista que, neste caso, é o único parâmetro a ser quantizado.

Os resultados apresentados abaixo foram obtidos para sub-quadros de 2,5ms e quadros de 20 ms, quando necessário.

- Codificador CELP convencional

$$\begin{aligned} \text{RSR}_S &= 8,91 \text{ dB} \\ \text{EMQ}_S &= 0,129 \\ \text{Atraso} &= 25 + T_p \text{ ms} \\ \text{Ordem de complexidade} &= O(\text{OD.L} + L) \\ \text{Ordem do preditor OD} &= 12 \end{aligned}$$

- Codificador CEA

$$\begin{aligned} \text{RSR}_S &= 8,89 \text{ dB} \\ \text{EMQ}_S &= 0,129 \\ \text{Atraso} &= 20 + T_p \text{ ms} \\ \text{Ordem de complexidade} &= O(L) \\ \text{Ordem do filtro } L(z) \text{ ON} &= 12 \end{aligned}$$

- Codificador CEA dispensando-se o filtro $L(z)$

$$\begin{aligned} \text{RSR}_S &= 8,55 \text{ dB} \\ \text{EMQ}_S &= 0,140 \\ \text{Atraso} &= 2,5 + T_p \text{ ms} \\ \text{Ordem de complexidade} &= O(L) \end{aligned}$$

- Codificador CEA quantizado sem o filtro $L(z)$.

$$\begin{aligned} \text{RSR}_S &= 8,20 \text{ dB} \\ \text{EMQ}_S &= 0,151 \\ \text{Atraso} &= 2,5 + T_p \text{ ms} \\ \text{Ordem de complexidade} &= O(L) \\ \text{Taxa de transmissão} &= 6 \text{ kbits/s} \end{aligned}$$

CAPÍTULO 4

CONCLUSÕES

Foram apresentados, neste trabalho, um estudo e algumas considerações a respeito da implementação de codificadores de sinais de voz com excitação codificada (CELP).

Codificador de sinal de voz com excitação codificada consiste na utilização de um dicionário de códigos, correntemente denominado de "codebook", no qual é criteriosamente escolhida uma excitação ("codeword") que leve à melhor qualidade da voz sintetizada.

Diferentemente dos codificadores paramétricos, que consistem - uma vez determinados os preditores - na estipulação ou determinação específica de uma excitação, nos codificadores CELP o sinal de excitação é determinado através da seleção em um conjunto de sinais contido em um "codebook". Isto implica, naturalmente, em uma diminuição substancial na taxa de bits pois, para representar uma excitação é necessário indicar apenas o índice desta no "codebook".

Foram revistos, no Capítulo 2, alguns conceitos fundamentais à respeito de predição linear aplicada a sinais de voz. Algumas das técnicas principais de codificação paramétrica também foram brevemente discutidas dando-se uma maior ênfase para os codificadores CELP, codificadores estes que estão demonstrando grande potencial para produzir sinais de voz com alta qualidade a taxas tão baixas quanto 4,8 kbits/s. No entanto, tais codificadores apresentam ainda uma elevada complexidade computacional para a determinação de seus parâmetros.

Uma nova estrutura para o CELP consistindo basicamente de um "codebook" excitando um filtro transversal $L(z)$ é proposta e discutida no Capítulo 3. Nessa nova estrutura, primeiramente, é escolhida a seqüência de excitação no "codebook" que mais se assemelha ao sinal de voz original e, posteriormente, esta é passada por um filtro $L(z)$ que ajusta seus coeficientes de modo a obter-se o mínimo erro médio quadrático entre o sinal de voz original e o sintetizado. Para a determinação dos parâmetros do filtro $L(z)$ foi utilizado o algoritmo LMS. Obteve-se desta forma uma redução substancial da complexidade computacional para a procura da melhor "codeword" em relação ao CELP convencional, mantendo-se, aproximadamente, a mesma qualidade objetiva do sinal de voz sintetizado. Todavia foi observado, através de avaliação subjetiva da voz sintetizada por esta nova estrutura (CEA), que os sons graves foram acentuados. Atribui-se este fato ao tipo de sinal utilizado no "codevector".

Para se ter uma idéia, utilizando-se um preditor de ordem igual a 12 com quadros e sub-quadros de 160 e 20 amostras, respectivamente, obteve-se um ganho em termos de operações aritméticas a favor do CEA em relação à estrutura CELP convencional de 12 vezes. Nestas condições a RSR_S manteve-se, sendo para o CELP igual a 8,91 dB e para o CEA 8,89 dB. Nota-se claramente pelos resultados obtidos, que à medida em que é reduzido o número de amostras dos sub-quadros obtém-se um aumento da RSR_S . Como exemplo, se o número de amostras dos sub-quadros forem reduzidos para 10, nas mesmas condições, a RSR_S do CEA passará para 12 dB. Todavia é importante lembrar que quanto menor o tamanho do sub-quadro maior será o número de parâmetros a serem transmitidos e conseqüentemente mais elevada será a taxa de bits.

Esta nova estrutura também possibilita uma redução de até 10 vezes no atraso existente no processo de codificação e decodificação do sinal de voz.

Somente foi possível a viabilização desta nova estrutura devido à mudança do tipo de sinal utilizado na composição da excitação. Este novo sinal consiste de um sinal pseudo-sinusoidal que apresenta, para segmentos consideravelmente curtos, características próximas às dos sinais de voz.

Diversas simulações foram necessárias para obter-se os resultados apresentados neste trabalho. Para se ter uma noção da quantidade de simulações, foram gerados cerca de 500 arquivos de sinais de voz sintetizados entre as várias estruturas estudadas. Considerando-se que foram utilizados apenas microcomputadores tipo IBM-PC, e que o tempo médio de processamento de cada arquivo foi de 10 horas, tem-se um total aproximado de 5.000 horas de CPU ou o equivalente a mais de 200 dias de processamento ininterrupto caso fosse usado apenas um único microcomputador.

Foi proposto também, um método alternativo para a determinação do filtro $L(z)$, que apresenta algumas vantagens em relação à utilização do algoritmo LMS em termos de complexidade computacional. Porém, não foi possível sua utilização nas simulações por este apresentar alguns problemas de instabilidade numérica em certos quadros.

Como sugestões para futuros trabalhos, destacam-se:

- A utilização de um número variável de coeficientes para o filtro $L(z)$, o que poderia levar a uma redução da taxa de bits/s;
- Investigar a respeito da possibilidade da composição dinâmica da excitação. Isso poderia levar à condição de sempre se dispor da melhor excitação para o sinal particular de voz em questão;
- Resolver os problemas de instabilidade numérica encontrada no método alternativo para a determinação do filtro $L(z)$ o que levaria a diminuir em muito a complexidade computacional requerida;

- Efetuar um estudo da quantização dos coeficientes do filtro $L(z)$ da estrutura proposta (CEA). Esta avaliação seria de grande importância na implementação de circuitos digitalizadores de voz, por exemplo.

- Efetuar um estudo da seleção da melhor "codeword" da estrutura CEA no domínio da frequência, o que levaria a uma redução ainda maior da complexidade computacional na fase de análise.

BIBLIOGRAFIAS REFERENCIADAS

- [1] ROCHA, Luis. Procesamiento de Voz. Buenos Aires, Editorial Kapelusz, 1987. 85p.
- [2] MALBERG, Bertil. La Phonétique. Paris, Presses Universitaires de France, 1962. 195p.
- [3] CELDRÁN, E. Martinez. Fonética. Barcelona, Teide, 1984. 406p.
- [4] CHAPMAN, C. Vocoders - o que são, como funcionam, suas possibilidades. Revista Elektor, São Paulo, 41-6, 1989. (1ª parte).
- [5] JAYANT, N. S. High-Quality Coding of Telephone Speech and Wideband Audio. IEEE Communications Magazine, E.U.A., 28(1):10-20, Jan. 1990.
- [6] ROSE, Richard C. & BARNWELL, Thomas P. Design and Performance of an Analysis - by - Synthesis Class of Predictive Speech Coders. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 38(9):1489-503, Set. 1990.
- [7] KITAWAKI, Nobuhiko & NAGABUCHI, Hiromi & ITOH, Kenzo. Objective Quality Evaluation for Low-Bit-Rate Speech Coding Systems. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):242-48, Feb. 1988.

- [8] ATAL, B. S. & SCHROEDER, Manfred R. Predictive Coding of Speech Signals and Subjective Error Criteria. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 27(3):247-54, Jun. 1979.
- [9] JAYANT, N. S. & NOLL, P. Digital Coding of Waveforms Principles and Applications to Speech and Video. New Jersey, Prentice-Hall, 1984. 688p.
- [10] DÈCINA, Maurizio & MODENA, Giulio. CCITT Standards in Digital Speech Processing. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):227-34, Feb. 1988.
- [11] INTERNATIONAL TELEGRAPH AND TELEPHONE CONSULATIVE COMMITTEE. Recommendation G.721 32 Kbit/s Adaptive Differential Pulse Code Modulation (ADPCM). Malaga-Torremolinos, CCITT, 1984. 125-159p.
- [12] OKI Semiconductor. Oki Voice Synthesis LSI. 2. ed., E.U.A., OKI, 1989. 338p.
- [13] KROON, Peter & DEPRETTERE, Ed F. A Class of Analysis-by-Synthesis Predictive Coders for High Quality Speech Coding at Rates Between 4.8 and 16 kbits/s. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):353-63, Feb. 1988.

- [14] TRANCOSO, Isabel M. & ATAL, Bishnu S. Efficient Search Procedures for Selecting the Optimum Innovation in Stochastic Coders. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 38(3):385-96, Mar. 1990.
- [15] LEE, J. I. & UN, C. K. On Reducing Computational Complexity of Codebook Search in CELP Coding. IEEE Transactions on Communications, E.U.A., 38(11):1935-937, Nov. 1990.
- [16] KLEIJN, W. Bastiaan & KRASINSKI, Daniel J. & KETCHUM, Richard H. Fast Methods for the CELP Speech Coding Algorithm. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 38(8):1330-342, Agos. 1990.
- [17] GALAND, C. & MENEZ, J. & ROSSO, M. Complexity Reduction of CELP Coders. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Rio de Janeiro, 1990. Anais...Rio de Janeiro, SBT/IEEE, 1990. p. 181-85, S.9.
- [18] MAKHOUL, John. Linear prediction; a Tutorial Review. Proceedings of the IEEE, 63(4):561-80, Abr. 1975.
- [19] RABINER, L. R. & SCHAEFER, Ronald W. Digital Processing of Speech Signals. New Jersey, Prentice-Hall, 1978. 512p.
- [20] JACKSON, Leland. Digital Filters and Signal Processing. 2. ed., Boston, Kluwer Academic Publishers, 1989. 410p.

- [21] BAÚ, N. & SEARA, R. & BERMUDEZ, J. C. M. Design of IIR Digital Filters from Time-Domain Specifications. In: IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS, Singapore, 1991. Anais... Singapore, IEEE, 1991. p.2443-446.
- [22] CUSTÓDIO, Ricardo Felipe. Codificadores Paramétricos de Sinais de Voz com Excitação Multi-Pulso. Florianópolis, UFSC/LINSE, 1990. 90p. (Dissertação de Mestre em Engenharia Elétrica).
- [23] RAMACHANDRAN, Ravi P. & KABAL, Peter. Pitch Prediction Filters in Speech Coding. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 37(4):467-78, Abr. 1989.
- [24] FLANAGAN J. L. & COKER, C. H. & RABINER, L. R. & SCHAFER, N. Synthetic Voices for Computers. IEEE Spectrum, 22-45, Out. 1970.
- [25] SINGHAL, Sharad & ATAL, Bishnu S. Amplitude Optimization and Pitch Prediction in Multipulse Coders. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 37(3):317-27, Mar. 1989.
- [26] ATAL, B. S. High-Quality Speech at Low Bit Rates: Multi-Pulse and Stochastically Excited Linear Predictive Coders. In: INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING., Tokyo, 1986. Anais... Tokyo, IEEE, 1986. p. 1681-684.

- [27] ATAL, B. S. & REMDE, J. R. A New Model of LPC Excitation for Producing Natural-Sounding Speech at Low Bit Rates. In: IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, Paris, 1980. Anais..., Paris, 1980. p. 614-17, apud SINGHAL, Sharad & ATAL, Bishnu S. Amplitude Optimization and Pitch Prediction in Multipulse Coders. IEEE Transaction on Acoustics, Speech, and Signal Processing, E.U.A., 37(3):317-27, Mar. 1989.
- [28] ATAL, B. S. Linear predictive coding of speech., In Computer Speech Processing, Fallside and W. A. Woods, Eds. Englewood Cliffs, New Jersey, Prentice-Hall, 1985. apud SINGHAL, Sharad & ATAL, Bishnu S. Amplitude optimization and Pitch prediction in multipulse coders. IEEE Transaction on Acoustics, Speech, and Signal Processing, E.U.A., 37(3):317-27, Mar. 1989.
- [29] ROSE, Richard C. & BARWELL III, Thomas P. The Self Excited Vocoder - an Alternate Approach to Toll Quality at 4800 Bps. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Tokyo, 1986. Anais... Tokyo, IEEE, 1986. p. 453-56.
- [30] ROSE, R. C. & BARNWELL, Thomas P. Quality Comparison of Low Complexity 4800 Bps Self Excited and Code Excited Vocoders. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Tokyo, 1987. Anais... Tokyo, IEEE, 1987. p. 1937-940.

- [31] SCHROEDER, M. R. & ATAL, B. S. Code-Excited Linear Prediction (CELP): High-Quality Speech at Very Low Bit Rates. In: IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, Tampa, FL. 1985, V. 1, paper 25.1.1., apud TRANCOSO, Isabel M. & ATAL, Bishnu S. Efficient Search Procedures for Selecting the Optimum Innovation in Stochastic Coders. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 38(3):385-96, Mar. 1990.
- [32] DAVIDSON, Grant & GERSHO, Allen. Complexity Reduction Methods for Vector Excitation Coding. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Tokyo, 1986. Anais...Tokyo, IEEE, 1986. p. 3055-58.
- [33] LIN, D. Speech Coding Using Efficient Pseudo-Stochastic Block Codes. In: IEEE INTERNATIONAL CONFERENCE ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING, 1987. Anais...IEEE, 1987. p. 1354-357.
- [34] KROON, Peter & ATAL, Bishnu S. Quantization Procedures for the Excitation in CELP Coders. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Tokyo, 1987. Anais... Tokyo, IEEE, 1987. p. 1649-652.
- [35] KROON, Peter & ATAL, Bishnu S. Strategies for Improving the Performance of CELP Coders at Low Bit Rates. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Tokyo, 1988. Anais... Tokyo, IEEE, 1988. p. 151-4.

- [36] ATAL, B. S. & SCHROEDER, M. R. Stochastic Coding of Speech Signals at Very Low Bit Rates. In: IEEE INTERNATIONAL CONFERENCE ON COMMUNICATIONS Amsterdam - The Netherlands, 1984. Anais..., Amsterdam - The Netherlands, IEEE, 1984. p. 1610-613. (parte 2), apud TRANCOSO, Isabel M. & ATAL, Bishnu S. Efficient Search Procedures for Selecting the Optimum Innovation in Stochastic Coders. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 38(3):385-96, Mar. 1990.
- [37] ATAL, B. S. & SCHROEDER, M. R. Stochastic Coding of Speech Signals at Very Low Bit Rates: The importance of speech perception. Speech Commun., 4(1-3):155-62, Agos. 1985, apud TRANCOSO, Isabel M. & ATAL, Bishnu S. Efficient Search Procedures for Selecting the Optimum Innovation in Stochastic Coders. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 38(3):385-96, Mar. 1990.
- [38] DUDLEY, H. Remaking Speech. J. Acoust. Soc. Am., 11:169-177, 1939, apud RABINER, Lawrence R. & SCHAEFER, Ronald W. Digital Processing of Speech Signals. New Jersey, Prentice-Hall, 1978. 512p.
- [39] WIDROW, Bernard & STEARNS, Samuel D. Adaptive Signal Processing. New Jersey, Prentice-Hall, 1985. 474p.
- [40] LEVINSON, N. The Wiener RMS error criterion in filter design and prediction, J. Math. Phys., 25(4):261-78, 1947. apud ROCHA, Luis F. Prediccion lineal aplicada a señales de voz. Revista Telegrafica Electronica, Argentina, :962-69, set. 1979.

- [41] ROBINSON, E. A. Estatistical communication and detection. Hainner, N. Y., 1967. apud ROCHA, Luis F. Prediccion lineal aplicada a señaes de voz. Revista Telegrafica Electronica, Argentina, :962-69, set. 1979.
- [42] NOCETI FILHO, Sidnei & BERMUDEZ, José C. M. & SEARA, Rui. Um Novo Método Iterativo para a Compensação do Efeito $\text{sen}(x)/x$ em Conversões D/A In: SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES, 6, Campina Grande, 1988. Anais...Campina Grande, 1988. p. 251-55.

BIBLIOGRAFIAS CONSULTADAS

- [43] CHEN, Juin-Hwey & GERSHO, Allen. Real-Time Vector APC Speech Coding at 4800 Bps with Adaptive Post-filtering. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Tokyo, 1987. Anais...Tokyo, IEEE, 1990. p.2185-188.
- [44] LATHI, B. P. Sistemas de Comunicação. Rio de Janeiro, Guanabara Dois, 1979. 401p.
- [45] GURLEKIAN, Jorge A. & FRANCO, Horacio E. J. & TOLEDO, Guillermo. Procesamiento de señales de habla; el Hombre Dialoga con la Maquina. El QUID De La Ciencia, La Tecnologia Y La Educacion, Argentina, 2(14):87-134, 1983.
- [46] MANSOUR, David & JUANG, Biing Hwang. The Short-Time Modified Coerence Representation and Noisy Speech Recognition. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 37(6):801-2, Jun. 1989.
- [47] MENEZ, J. & GALAND, C. & ROSSO, M. A 2 ms-Delay Adaptive Code Excited Linear Predictive Coder. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Rio de Janeiro, 1990. Anais... Rio de Janeiro, SBT/IEEE, 1990. p. 457-60, S.9.

- [48] MORIYA, Takehiro & HONDA, Masaaki. Transform Coding of Speech Using a Weighted Vector Quantizer. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):425-31, Feb. 1988.
- [49] IRIE, Kazunari & TADA, Yasuhiko & HONMA, Kouichi. APC-AB Codec Modules Operating at 16 and 8 Kbit/s. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):383-90, Feb. 1988.
- [50] RAMAMOORTHY, V. & JAYANT, N. S. & COX, Richard V. & SONDHI, Man Mohan. Enhancement of ADPCM Speech Coding with Backward-Adaptive Algorithms for Postfiltering and Noise Feedback. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):364-82, Feb. 1988.
- [51] SUDA, Hirohito & MIKI, Toshio. An Error Protected 16 Kbit/s Voice Transmission for Land Mobile Radio Channel. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):346-52, Feb. 1988.
- [52] McLAUGHLIN, Michael J. & RASKY, Phillip D. Speech and Channel Coding for Digital Land-Mobile Radio. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):332-45, Feb. 1988.
- [53] NATVIG, Jon E. Evaluation of Six Medium Bit-Rate Coders for the Pan-European Digital Mobile Radio System. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):324-31, Feb. 1988.

- [54] IWADARE, Masahiro & NISHITANI, Takao. 64 Kbit/s Audio Signal Transmission Approaches Using 32 Kbit/s ADPCM Channel Banks. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):307-13, Feb. 1988.
- [55] HOSODA, Kenichiro & NOGUCHI, Osamu & YATSUZUKA, Yohtarō. A 32 Kbit/s ADPCM Algorithm Having High Performance for Both Voice and 9.6 Kbit/s Modem Signals. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):262-73, Feb. 1988.
- [56] YATROU, Paul & MERMELSTEIN, Paul. Ensuring Predictor Tracking in ADPCM Speech Coders Under Noisy Transmission Conditions. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):249-61, Feb. 1988.
- [57] AOYAMA, Tomonori & DAUMER, William R. & MODENA, Giulio Foreword Voice Coding for Communications. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):225-26, Feb. 1988.
- [58] CUPERMAN, Vladimir. On Adaptive Vector Transform Quantization for Speech Coding. IEEE Transactions on Communications, E.U.A., 37(3):261-67, Mar. 1989.
- [59] PARK, Sungkwon & GERHARDT, Lester A. A Robust Spectral Estimation by Modeling an Estimated Autocovariance with an ARMA Model. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 37(2):181-91, Feb. 1989.

- [60] SUKKAR, Rafid A. & LoCICERO, Joseph L. & PICONE, Joseph W. Decomposition of the LPC Excitation Using the Zinc Basis Functions. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 37(9):1329-341, Set. 1989.
- [61] CHEN, Ting-Chung. A Lattice Vector Quantization Using a Geometric Decomposition. IEEE Transactions on Communications, E.U.A., 38(5):704-14, Maio. 1990.
- [62] ADOUL, J-P. & LAMBLIN, C. A Comparison of Some Algebraic Structures For CELP Coding of Speech. In: INTERNATIONAL TELECOMUNICATIONS SYMPOSIUM, Tokyo, 1987. Anais...Tokyo, IEEE, 1987. p. 1953-956.
- [63] SILVEIRA, Regina C. P. Estudos de Fonologia Portuguesa. São Paulo, Cortez Editora, 1986. 254p.
- [64] ATAL, B. S. & HANAUER, L. Suzanne. Speech Analysis by Linear Prediction of the Speech Wave. The Journal of the Acoustical Society of America, New Jersey, 50:637-55, Abr. 1971. (parte 2).
- [65] MENEZ, J. & ROSSO, M. & GALAND, C. & BOTTAU, F. Adaptive Code Excited Linear Predictive Coder (ACELPC). In: INTERNATIONAL TELECOMUNICATIONS SYMPOSIUM, Tokyo, 1989. Anais...Tokyo, IEEE, 1989. p. 132-35.

- [66] JAYANT, N. S. & CHEN, J. H. Speech Coding with Time-Varying Bit Allocations to Excitation and LPC Parameters. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Tokyo, 1989. Anais...Tokyo, IEEE, 1989. p. 65-8.
- [67] SUKKAR, Rafid A. & LoCICERO, Joseph L. & PICONE, Joseph W. Design and Implementation of a Robust Pitch Detector Based on a Parallel Processing Technique. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):441-51, Feb. 1988.
- [68] SUGAMURA, Noboru & FARVARDIN, Nariman. Quantizer Design in LSP Speech Analysis-Synthesis. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):432-40, Feb. 1988.
- [69] CARLSON, A. Bruce. Communication Systems. 2 ed., Kogakusha, McGraw-Hill, 1975. 495.
- [70] TANIGUCHI, Tomohiko & UNAGAMI, Shigeyuki & ISEDA, Kohei & TOMINAGA, Syozi. ADPCM with a Multiquantizer for Speech Coding. IEEE Journal on Selected Areas in Communications, E.U.A., 6(2):410-24, Feb. 1988.
- [71] ADOUL, J-P. & MABILLEAU, P. & DELPRAT, M. & MORISSETE, S. Fast CELP Coding Based on Algebraic Codes. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Tokyo, 1990. Anais...Tokyo, IEEE, 1987. p. 1957-960.

- [72] BELINAZO, C. H. & YAMAMOTO, J. S. Codificação ADPCM a Taxa de Bits Variáveis para Equipamentos Multiplicadores de Circuitos Digitais. In: SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES, 6, Campina Grande, 1988. Anais...Campina Grande, 1988. p. 67-70.
- [73] FERRETI, Marc. & CINARE, François. Synthese Reconnaissance de la Parole. França, Éditests, 1983. 282p.
- [74] BORLAND. Turbo Pascal 4.0. E.U.A., Borland, 1987. 654.
- [75] COX, Richard U. et alii. Real-time Implentation of Time Domain Harmonic Scaling of Speech for Rate Modification and Coding. IEEE Transations on Acoustics, Speech, and Signals Processing, E.U.A., 31(1):258-72, Feb. 1983.
- [76] ATAL, Bishnu S. Stochastic Gaussian Model for Low-Bit Rate Coding of LPC Area Parameters. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Tokyo, 1987. Anais...Tokyo, IEEE, 1987. p. 2404-407.
- [77] BLANKINSHIP, W. A. Note on Computing Autocorrelations. IEEE Transactions on Acoustics, Speech, and Signal Processing, E.U.A., 1976-977, Feb. 1974.
- [78] GREGORIAN, Roubick. & AMIR, Gideon. A Single Chip Speech Synthetizer Using a Switched-Capacitor Multiplier. IEEE Transations on Acoustics, Speech, and Signals Processing, E.U.A., 31(1):313-23, Feb. 1983.
-

- [79] ISTRE, Giles L. Fonologia Transformacional e Natural. Florianópolis, Editora da UFSC, 1980. 266p.
- [80] SILVEIRA, Regina C. P. Estudos de Fonética do Idioma Português. São Paulo, Cortez Editora, 1982. 115p.
- [81] HERNÁNDEZ-GÓMEZ, Luis A. & CASAJÚS-QUIRÓS, F. J. & FIGUEIRAS-VIDAL, A. R. & GARCIA-GOMÉZ, R. On the Behavior of Reduced Complexity Code Excited Linear Prediction (CELP). In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Tokyo, 1986. Anais...Tokyo, IEEE, 1986. p. 469-72.
- [82] MAIA, Eleonora M. No Reino da Fala. São Paulo, Ática, 1985. 128p.
- [83] RABINER, Lawrence R. & GOLD, B. Parallel processing Techniques for Estimating Pitch Periods of Speech in the Time Domain. The Journal of the Acoustical Society of America, E.U.A., 46, 442-48, Agos. 1969.
- [84] STANLEY, Willian D. et alii. Digital Signal Processing. Virginia, Reston Publishing, 1984. 514p.
- [85] OPPENHEIN, Alan V. & SCHAEFER, Ronald W. Digital Signal Processing, New Jersey, Prentice-Hall, 1975. 585p.
- [86] RAO, R. Padmanabha & PEARLMAN, William A. Vector Quantization with Size-Constrained Codebooks. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Rio de Janeiro, 1990. Anais...Rio de Janeiro, SBT/IEEE, 1990. p. 186-90, S.9.

- [87] FLANAGAN, J. L. & SCHROLDI, R. & CROCHIERE, Ronalde & JAYANT, N. & TRIBOLET, José M. Speech Coding. IEEE Transactions on Communications, E.U.A., 27(4):710-37, Abr. 1979.
- [88] WALDMAN, Helio. Processamento de Sinais Digitais. Argentina, La Editorial Kapelusz, 1987. 183p.
- [89] WILLEMS, L. F. & HART, J. & NOOTEBOOM, S. G. & VOGTEN, L. L. M. Manipulação de Sons da Fala. Ciência e Tecnologia, 2(6):179-91, 1983.
- [90] TRANCOSO, Isabel M. & ALMEIDA, Luis B. & TRIBOLET, Jose M. A Study on the Relationships Between Stochastic and Harmonic Coding. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Tokyo, 1986. Anais...Tokyo, IEEE, 1986. p. 1709-12.
- [91] YAMAMOTO, José S. & VIOLARO, Fábio. Investigation on Vector Quantized Multipulse Excitation with fixed Pattern of Positions. In: INTERNATIONAL TELECOMMUNICATIONS SYMPOSIUM, Rio de Janeiro, 1990. Anais...Rio de Janeiro, SBT/IEEE, 1990. p. 191-95, S.9.

A P Ê N D I C E A

PROGRAMA PARA DETERMINAÇÃO DOS PARÂMETROS DO CODIFICADOR CELP CONVENCIONAL

Program CELP;

{Programa de análise e síntese para codificação de um sinal de voz com a estrutura CELP convencional.

Arquivo de entrada : (1) sinal de voz original;
 : (2) "codebook"

Arquivos de saída : (3) endereço da "codeword" e ganho;
 (4) coeficientes do PCA;
 (5) sinal de voz sintetizado.

}

{ Definição das Variáveis }

Const

Nc = 1024;	{Número de "codewords"}
Na = 160;	{Número de amostras do quadro}
Nsq = 20;	{Número de amostras do subquadro}
Ns = 80;	{Número de amostras com recobrimento}
Nq = 10;	{Número de quadros do sinal de voz}
OD = 10;	{Ordem do polinômio do denominador}
ON = 0;	{Ordem do polinômio do numerador}
OM = 10;	{Maior valor entre OD e ON}
voz = 'sidnei.dad';	{Arquivo de voz original}
book = 'codebook.dad';	{Arquivo do codebook}

```

code = 'code.dad';      {Arquivo do endereço da "codeword" e do ganho}
coef = 'coef.dad';     {Arquivo dos coeficientes do preditor}
sint = 'sint.dad';     {Arquivo de voz sintetizada}

```

Type

```

Precisao                = real;
Tipo_subquadro_i        = array[0 .. Nsq-1] of integer;
Tipo_quadro_voz_i       = array[0 .. Na-1] of integer;
Tipo_quadro_voz_r       = array[0 .. Na-1] of precisao;
Tipo_celp1              = array[1 .. 2] of integer;
Tipo_coef_numerador     = array[0 .. ON] of Precisao;
Tipo_coef_denominador   = array[0 .. OD] of Precisao;
Tipo_quadro_sobreposto  = array[0 .. Na+Ns-1] of precisao;

```

var

```

i, j, k, l, q           : integer;      {Variável auxiliar}
indice                  : integer;      {Índice da "codeword" selecionada}
fator                   : integer;      {Número de subquadros}
valor, emq, rsr, max    : precisao;    {Variável auxiliar}
soma, aux1, aux2        : Precisao;     {Variável auxiliar}
ganho                   : Precisao;     {Ganho da "codeword"}
s, ss                   : Tipo_quadro_voz_i;
x, y, x1, y1            : Tipo_quadro_voz_r;
sx                      : Tipo_subquadro_i;
h                       : Tipo_quadro_sobreposto;
codigo                  : Tipo_celp1;
d                       : Tipo_coef_denominador;
b                       : Tipo_coef_numerador;
sintese                 : file of Tipo_quadro_voz_i;
arqvoz                  : file of Tipo_quadro_voz_i;
arqcode                 : file of tipo_subquadro_i;
celp1                   : file of Tipo_celp1;
coefd                   : file of Tipo_coef_denominador;
coefb                   : file of Tipo_coef_numerador;

```

{ Funções }

```
function calcula_erro_saida : precisao;
```

```
var
```

```
  soma : real;
```

```
  aux  : real;
```

```
  i    : integer;
```

```
begin
```

```
  soma := 0;
```

```
  for i := 0 to Na-1 do
```

```
    soma := soma + sqr(x[i]-y[i]);
```

```
  aux := 0;
```

```
  for i := 0 to Na-1 do
```

```
    aux := aux + sqr(x[i]);
```

```
  soma := soma / aux;
```

```
  calcula_erro_saida := soma;
```

```
end;
```

```
function calcula_rsr(emq: precisao) : precisao;
```

```
var
```

```
  aux: real;
```

```
begin
```

```
  calcula_rsr := 10*ln(1/(emq + 1e-30))/ln(10);
```

```
end;
```

```
procedure quantiza_saida;
```

```
var
```

```
  i : integer;
```

```
  aux : real;
```

```
begin
```

```
  for i := 0 to Na-1 do
```

```
    begin
```

```
      aux := y[i] + 2048;
```

```

if aux > 4095 then
  aux := 4095
else
  if aux < 0 then
    aux := 0;
  ss[i] := round(aux);
end;
end;

```

```

procedure janelamento_hamming;

```

```

var
  n      : integer;
  aux    : real;
begin
  aux := (2*pi)/(Na+Ns-1);
  for n := 0 to (Na+Ns-1) do
    h[n] := (0.54 - 0.46*cos(aux*n))*h[n];
  end;
end;

```

```

procedure determina_coeficientes_filtro_durbin(

```

```

  var s : tipo_quadro_sobreposto; { seqüência a ser imitada }
  n      : integer;                { número de amostras }
  on     : byte;                   { ordem do numerador }
  od     : byte;                   { ordem do preditor }
  var bb : tipo_coef_numerador;    { coeficientes do numerador }
  var aa : tipo_coef_denominador); { coeficientes do denominador }

```

```

Var

```

```

R      : array[0..OM] of real ;    { autocorrelação }
E      : array[0..OM] of real ;    { auxiliar }
ap     : tipo_coef_denominador;
parcor : array[1..OM] of real;

```

```

soma      : real;
i, j      : integer;

begin

{ **** autocorrelação **** }
i := 0;
repeat
  soma := 0;
  j := 0;
  repeat
    soma := soma + s[i+j]*s[j];
    inc(j);
  until (i+j) > N-1;
  R[i] := soma;
  inc(i);
until i > od;

{ **** Método de Durbin **** }

E[0] := R[0];
for i:= 1 to od do
  begin
    soma := 0;
    for j := 1 to i-1 do
      soma := soma + ap[j]*R[i-j];
    parcor[i] := (R[i] - soma) / E[i-1];
    aa[i] := parcor[i];
    for j := 1 to i-1 do
      aa[j] := aa[j] - parcor[i]*ap[i-j];
    E[i] := (1-sqr(parcor[i]))*E[i-1];
    ap := aa;
  end;
aa[0] := 1;
for i := 1 to od do
  aa[i] := - aa[i];

```

```

bb[0] := s[0];
end;

```

{Determinação dos coeficientes, seleção da "codeword" e o ganho}

```

begin
  writeln;
  fator := round(Na/Nsq);
  assign(arqvoz, voz);
  reset(arqvoz);
  assign(arqcode, book);
  reset(arqcode);
  assign(celp1, code);
  rewrite(celp1);
  assign(coefd, coef);
  rewrite(coefd);
  assign(sintese, sint);
  rewrite(sintese);

  read(arqvoz, s);
  for i := 0 to Na-1 do
    h[i] := (s[i] - 2048)/2048;

  for q := 1 to Nq do
    begin
      write('quadro ', q, ' indice ');
      read(arqvoz, s);
      for i := 0 to Ns-1 do
        h[i+Na] := (s[i]-2048)/2048;
      for i := 0 to Na-1 do
        x[i] := h[i+round(Ns/2)];
      janelamento_hamming;
      determina_coeficientes_filtro_durbin(h, Na, ON, OD, b, d);
      write(coefd, d);
    end;
  end;

```

```

for i := 0 to fator - 1 do
  begin
    max := -1;
    for j := 0 to Nc-1 do
      begin
        read(arqcode, sx);
        aux1 := 0;
        aux2 := 0;
        for k := (i*Nsq) to (i*Nsq+Nsq-1) do
          begin
            soma := 0;
            for l := 1 to OD do
              if (k-l) >= 0 then
                soma := soma - d[l]*y1[k-l];
            y1[k] := soma + sx[(k-i*Nsq)];
            aux1 := aux1 + y1[k]*x[k];
            aux2 := aux2 + sqr(y1[k]);
          end;
        valor := sqr(aux1)/aux2;
        if valor > max then
          begin
            max := valor;
            indice := j+1;
            ganho := aux1/aux2;
            x1 := y1;
          end;
        end;
        codigo[1] := indice;
        if ganho > 32 then
          ganho := 32;
        codigo[2] := round(ganho * 1000);
        write(ce1p1, codigo);
        for l := (i*Nsq) to (i*Nsq + Nsq - 1) do
          begin
            y[l] := x1[l]*codigo[2]/1000;
            y1[l] := y[l];
          end;
        end;
      end;
    end;
  end;
end;

```

```
        y1[l] := x1[l];
    end;
    write(indice, ' ');
end;
emq := calcula_erro_saida;
rsr := calcula_rsr(emq);
for l := 0 to Na-1 do
    y[l] := y[l]*2048;
    quantiza_saida;
    write(sintese,ss);
    writeln;
    writeln('rsr = ',rsr);
    for i := 0 to Na-1 do
        h[i] := (s[i] - 2048)/2048;
    end;
close(arqvoz);
close(arqcode);
close(celp1);
close(coefd);
close(sintese);
end.
```


A P Ê N D I C E B

PROGRAMA PARA A DETERMINAÇÃO DOS PARÂMETROS DO CODIFICADOR PROPOSTO

Program CEA;

{Programa de análise e síntese para codificação de um sinal de voz usando a estrutura CEA.

Arquivo de entrada: (1) sinal de voz original;

: (2) "codevector"

Arquivos de saída : (3) endereço da "codeword" e do ganho;

(4) coeficientes do filtro L(z);

(5) sinal de voz sintetizado sem o filtro L(z);

(6) sinal de voz sintetizado com o filtro L(z).}

{ Definição das variáveis }

Const

NI	= 20;	{Número de iterações do algoritmo LMS}
Nc	= 1043;	{Número de amostras do "codebook"}
Nasq	= 20;	{Número de amostras do subquadro}
Naq	= 160;	{Número de amostras do quadro}
Nq	= 2792;	{Número de subquadros do sinal de voz}
ONL	= 12;	{Número de coeficientes do filtro L(z)}
fu	= 40;	{Fator de redução do μ crítico}
voz	= 'sidnei.dad';	{Arquivo de voz original}
vetor	= 'codevector.dad';	{Arquivo do codevector}
cod	= 'codigo.dad';	{Arquivo do endereço da "codeword" e do ganho}
coef	= 'coef.dad';	{Arquivo dos coeficientes do filtro L(z)}
sint	= 'síntese.dad';	{Arquivo de voz sintetizada sem o filtro L(z)}
resint	= 'resíntese.dad';	{Arquivo de voz sintetizada com o filtro L(z)}

Type

```

Precisao           = real;
Tipo_s_quadro_voz_i = array[0 .. Nasq-1] of integer;
Tipo_s_quadro_voz_r = array[0 .. Nasq-1] of precisao;
Tipo_quadro_voz_i   = array[0 .. Naq-1] of integer;
Tipo_quadro_voz_r   = array[0 .. Naq-1] of precisao;
Tipo_coef_num_lms   = array[0 .. ONL] of precisao;
Tipo_codigo_celp    = array[0 .. 1] of precisao;
Tipo_energia_codebook = array[0 .. Nc] of precisao;
Tipo_codebook_i     = array[1 .. Nc] of integer;
Tipo_quadro_pleno   = array[-ONL .. Naq-1] of precisao;

```

var

```

indice           : integer;      {Índice da "codeword" selecionada}
i, j, k, l, N, q : integer;      {Variável auxiliar}
soma, somax, melhor : Precisao;    {Variável auxiliar}
rsr, ga, emq     : Precisao;    {Variável auxiliar}
ganho           : Precisao;    {Ganho da "codeword"}
c               : Tipo_codebook_i;
ssq, ssq1      : Tipo_s_quadro_voz_i;
sq, sq1        : Tipo_quadro_voz_i;
w              : Tipo_quadro_pleno;
xsq, ysq       : Tipo_s_quadro_voz_r;
xq, yq, yq1    : Tipo_quadro_voz_r;
bl             : Tipo_coef_num_lms;
codigo         : Tipo_codigo_celp;
cr             : Tipo_energia_Codebook;
celp1         : file of Tipo_codigo_celp;
codebook      : file of Tipo_codebook_i;
sintese       : file of Tipo_quadro_voz_i;
meda          : file of Tipo_s_quadro_voz_i;
arqvoz       : file of Tipo_s_quadro_voz_i;
medaq        : file of Tipo_quadro_voz_i;
arqvozq      : file of Tipo_quadro_voz_i;
coefbl       : file of Tipo_coef_num_lms;

```

{ Funções }

```

function calcula_erro_saida : Preciso;
var
  soma : real;      {Variável auxiliar}
  aux  : real;      {Variável auxiliar}
  i    : integer;   {Variável inteira}
begin
  soma := 0;
  for i := 0 to Nasq-1 do
    soma := soma + sqr(xsq[i]-ysq[i]);
  aux := 0;
  for i := 0 to Nasq-1 do
    aux := aux + sqr(xsq[i]);
  soma := soma / aux;
  calcula_erro_saida := soma;
end;

function calcula_rsr(emq:precisao): precisao;
begin
  calcula_rsr := 10*ln(1/(emq + 1e-30))/ln(10);
end;

procedure quantiza_saida;
var
  i : integer;      {Variável inteira}
  aux : real;       {Variável auxiliar}
begin
  for i := 0 to Nasq-1 do
    begin
      aux := ysq[i] + 2048;
      if aux > 4095 then
        aux := 4095
      else

```

```

        if aux < 0 then
            aux := 0;
        ssq[i] := round(aux);
    end;
end;

```

```

Procedure calcula_coeficientes_lms(

```

```

    r   :tipo_quadro_voz_r;
    rl  :tipo_quadro_pleno;
    bl  :tipo_coef_num_lms);

```

```

var

```

```

    LOOP           : integer;    {Variável de iteração}
    fc             : precisao;   {Variável auxiliar}
    rk             : precisao;   {Variável auxiliar}
    ek             : precisao;   {Variável auxiliar}

```

```

begin

```

```

    { **** zerar os valores neg. de rl **** }

```

```

    for i := -ONL to -1 do

```

```

        rl[i] := 0;

```

```

    { **** inicialização dos coeficientes **** }

```

```

    for i := 0 to ONL do

```

```

        bl[i] := 0;

```

```

    { **** cálculo do fator de convergência **** }

```

```

    soma := 0;

```

```

    for i:= 0 to Naq-1 do

```

```

        soma:=soma+sqr(rl[i]);

```

```

    soma:=soma/Naq;

```

```

    fc:=1/((ONL+1)*soma*fu);

```

```

    { **** cálculo do filtro **** }

```

```

    LOOP := 0;

```

```

    for l := 1 to NI do

```

```

        begin

```

```

for k := LOOP to Naq-1 do
  begin
    rk := 0;
    i := 0;
    for j := k downto k-ONL do
      begin
        rk := rk + bl[i]*rl[j];
        i := i + 1;
      end;
    ek := r[k] - rk;
    i := 0;
    for j := k downto k-ONL do
      begin
        bl[i] := bl[i] + 2*fc*ek*rl[j];
        i := i + 1;
      end;
    LOOP := 0;
  end;
end;
end;

```

```

procedure calcula_saida_filtro_fir(
  xq : tipo_s_quadro_voz_r;
  bl : tipo_coef_num_lms;
  yq : tipo_s_quadro_voz_r);
begin
  for i := 0 to Nasq-1 do
    begin
      soma := 0;
      for l:= 0 to ONL do
        if (i - l) >= 0 then
          soma := soma + bl[l]*xq[i-l];
        yq[i] := soma;
      end;
    end;
  end;
end;

```

```
{ Seleção da melhor "codeword" e do ganho }
```

```
begin
```

```
  assign(codebook, vetor);
```

```
  reset(codebook);
```

```
  read(codebook, c);
```

```
  close(codebook);
```

```
  assign(arqvoz, voz);
```

```
  reset(arqvoz);
```

```
  assign(meda, sint);
```

```
  rewrite(meda);
```

```
  assign(celp1, cod);
```

```
  rewrite(celp1);
```

```
for j := 1 to Nc do
```

```
  begin
```

```
    c[j] := c[j] - 2048;
```

```
    cr[j] := c[j];
```

```
  end;
```

```
for j := 1 to Nc-Nasq+1 do
```

```
  begin
```

```
    soma := 0;
```

```
    for i := 0 to Nasq - 1 do
```

```
      soma := soma + sqr(cr[j+i]);
```

```
    cr[j] := soma;
```

```
  end;
```

```
N := round(Naq/Nasq);
```

```
for k := 1 to Nq*N do
```

```
  begin
```

```
    read(arqvoz, ssq);
```

```
    soma := 0;
```

```
    for i := 0 to Nasq-1 do
```

```
      begin
```

```

        ssq[i] := ssq[i]-2048;
        xsq[i] := ssq[i];
    end;
    melhor := 0;
    for j := 1 to Nc-Nasq+1 do
    begin
        soma := 0;
        for i := 0 to Nasq - 1 do
            soma := soma + xsq[i]*c[j+i];
            somax := sqr(soma)/cr[j];
            if abs(melhor) < abs(somax) then
            begin
                ga      := soma;
                melhor := somax;
                indice := j;
            end;
        end;
        ganho := ga/cr[indice];
        codigo[0] := indice;
        codigo[1] := ganho;
        writeln('quadro ',k,' amplitude ',abs(melhor),' indice ',indice);
        write(celp1,codigo);
        for i := 0 to Nasq-1 do
            ysq[i] := c[indice+i]*ganho;
            quantiza_saida;
            write(meda,ssq1);
            emq := calcula_erro_saida;
            rsr := calcula_rsr(emq);
            writeln('quadro ',k,' rsr ',rsr,' indice ',indice);
        end;
        close(celp1);
        close(arqvoz);
        close(meda);
    end.

```

```

{      Determinação dos coeficientes do filtro L(z)      }

begin
  assign(arqvozq, voz);
  reset(arqvozq);
  assign(medaq, sint);
  reset(medaq);
  assign(coefbl, coef);
  rewrite(coefbl);
  assign(meda, resint);
  rewrite(meda);
  for k := 1 to Nq do
    begin
      writeln('quadro ', i);
      read(arqvoz, sq);
      read(meda, sq1);
      for i:=0 to Naq-1 do
        begin
          xq[i] := sq[i] - 2048;
          w[i] := sq1[i] - 2048;
          yq[i] := w[i];
        end;
      calcula_coeficientes_lms(xq, w, bl);
      write(coefbl, bl);
      for j := 0 to N-1 do
        begin
          calcula_saida_filtro_fir(y1, bl, y);
          for i := 0 to Naq-1 do
            begin
              ysq1[i] := yq[i+j];
              xsq1[i] := xq[i+j];
            end;
          calcula_saida_filtro_fir(ysq1, bl, ysq);
          emq := calcula_erro_saida;
          rsr := calcula_rsr(emq);
          writeln('RSR = ', rsr);
        end;
      end;
    end;
  end;

```



```
        quantiza_saida;  
        write(meda,ssq);  
    end;  
end;  
close(arqvoz);  
close(meda);  
close(coefbl);  
close(meda);  
end.
```

A P Ê N D I C E C

UNIDADE DE AQUISIÇÃO DE DADOS

C.1. INTRODUÇÃO

Neste apêndice serão apresentadas as características gerais e o diagrama em blocos de um sistema completo desenvolvido para a aquisição e a reprodução dos sinais de voz. O sistema, baseado em microcomputadores tipo IBM-PC, compreende a etapa para o tratamento dos sinais analógicos/digitais e os programas básicos de seu gerenciamento.

C.2. DESCRIÇÃO DO SISTEMA

A Fig. C.1 mostra o diagrama em blocos da etapa de entrada. O sistema possui 8 entradas analógicas diferenciais multiplexadas, ou 16 entradas unipolares, selecionáveis por "software". Apenas uma das entradas possui um filtro passa-baixa chebyshev de 8^a ordem, 0,25 dB de "ripple" na banda passante, e banda passante de 3,4 kHz. O amplificador de instrumentação de entrada tem ganho programável de 1, 10, 100 e 1000. Em cada canal foi incorporado um circuito com diodos rápidos para atuar como proteção quando a tensão de entrada exceder a tensão de alimentação.

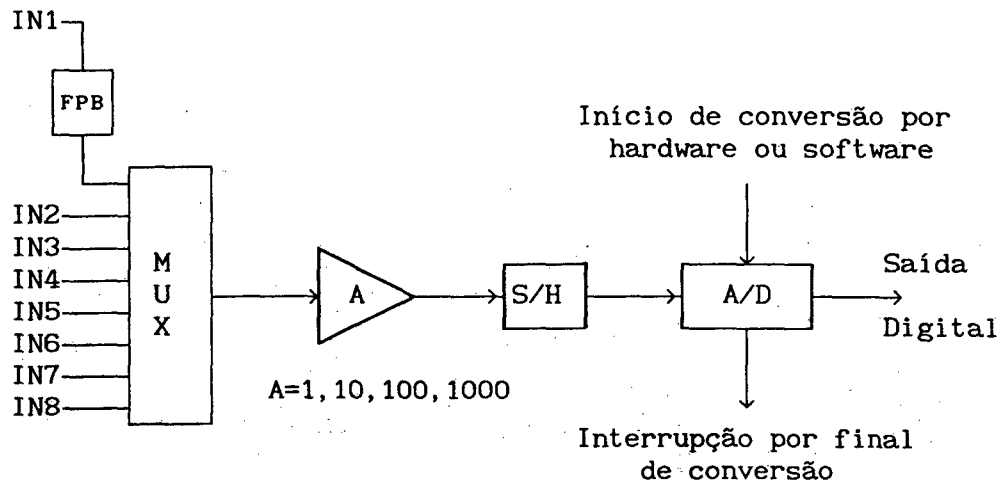


Fig. C.1 - Entradas analógicas

O início da conversão pode ser feito por "hardware" ou por "software". Por "hardware", o contador programável fornece o pulso de início de conversão. Por "software", dois endereços são usados para o início de conversão de 8 e 12 bits.

O final da conversão é realizado por "hardware", sendo programado por dois "jumpers", para gerar uma interrupção na CPU do microcomputador.

A seleção do canal e do ganho é efetuada através da escrita de uma palavra de controle num registrador.

A seleção do endereço base do sistema de aquisição, o modo de entrada (bipolar, unipolar, diferencial), a entrada de $\pm 5V$ e $\pm 10V$, e a escolha do início de conversão por "software" ou por "hardware", é feita por um conjunto de "jumpers" inseridos na placa.

A Fig. C.2 mostra o diagrama em blocos das saídas analógicas. A conversão D/A é feita com um conversor de 12 bits, com tempo de conversão de $3\mu s$. Porém,

a utilização da máxima taxa dependerá da velocidade do microcomputador. São previstas 8 saídas analógicas, em cujos filtros de reconstrução as localizações dos pólos foram convenientemente alteradas de modo a compensar a distorsão $(\text{sen}x)/x$ [42]. Um dos canais possui um amplificador de áudio com capacidade de acionar um alto falante, podendo-se, assim, utilizar o próprio alto falante do microcomputador. A seleção do canal de saída, é feita através da escrita de uma palavra num registrador, sendo que, é importante existir um tempo morto entre a comutação de um canal e outro, para evitar interferências.

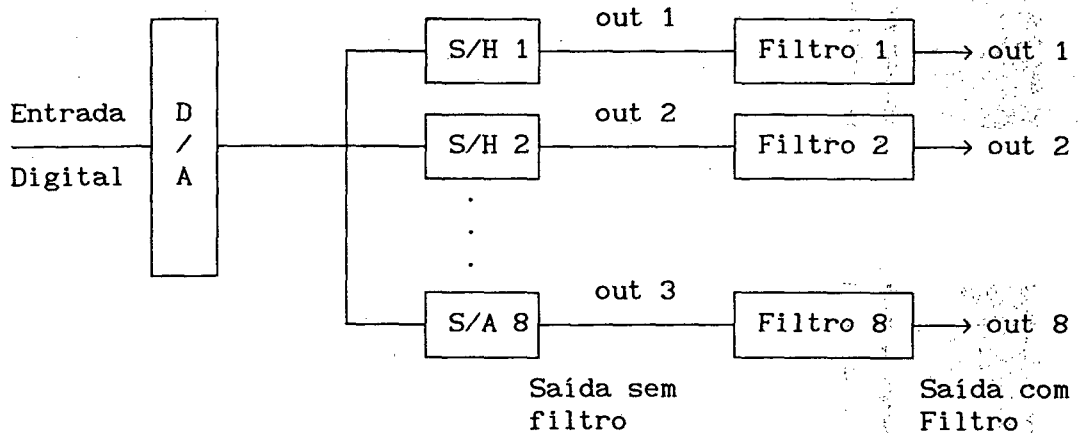


Fig. C.2 - Saídas analógicas

O gerador de pulsos interno pode ser usado para gerar uma interrupção interna, iniciar a conversão do A/D por "hardware" ou gerar um sinal de relógio externo. A Fig. C.3 apresenta o diagrama em blocos de conexão do relógio programável.

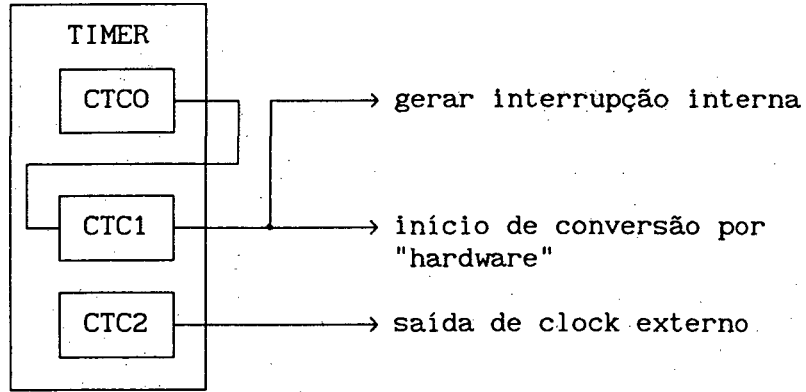


Fig. C.3 - Relógio interno

Para possibilitar um mecanismo de sincronismo das entradas e saídas, foi previsto um dispositivo de "trigger" externo como mostrado na Fig. C.4. O potenciômetro é usado para ajustar o nível de disparo do sinal. Este "trigger" pode ser programado para gerar uma interrupção para o início de uma aquisição ou para a transferência de dados digitais entre o sistema e o mundo exterior.

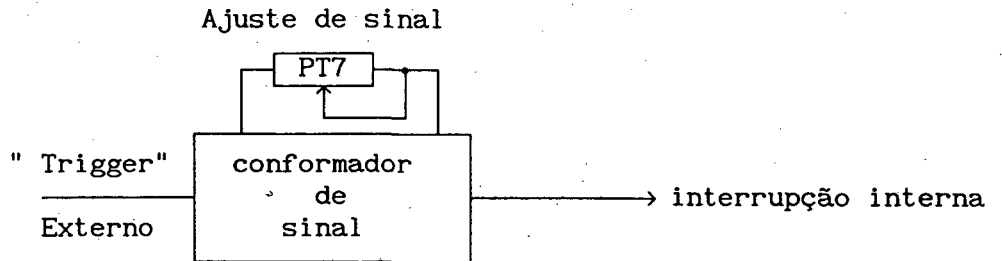


Fig. C.4 - "Trigger" externo

C.3. ESPECIFICAÇÃO DO CIRCUITO

As principais características da unidade de aquisição são:

1. Faixas de frequências

12 bits: CC até 18 kHz

8 bits: CC até 27 kHz

2. Sensibilidades

a) ± 10 V

b) ± 5 V

c) ± 1 V

d) $\pm 0,5$ V

e) ± 100 mV

f) ± 50 mV

g) ± 10 mV

h) ± 5 mV

3. Filtros

a) CC a 3,4 kHz (Chebyshev 8^a ordem, 0,25 dB de "ripple" na banda passante)

4. Frequências de amostragem variável

a) até 37 kHz para 12 bits

b) até 54 kHz para 8 bits

5. Resolução

12 bits: quantização binária uniforme

8 bits: quantização binária uniforme

6. Compensação do efeito $(\text{sen } x)/x$ no D/A

7. Impedância de entrada

1 M Ω

8. Modos de operação das entradas

Modo unipolar: proteção até 100V

Modo bipolar: proteção até 100V

Modo diferencial: proteção até 100V

9. Frequências de corte f_c , (-3 dB) do amplificador diferencial

a) $A = 1 \dots \dots f_c = 1 \text{ MHz}$

b) $A = 10 \dots \dots f_c = 400 \text{ kHz}$

c) $A = 100 \dots \dots f_c = 150 \text{ kHz}$

d) $A = 1000 \dots \dots f_c = 25 \text{ kHz}$

10. Resposta dinâmica do amplificador de áudio

a) frequência de corte inferior (-3dB) = 20 Hz

b) frequência de corte superior (-3dB) = 465 kHz

11. Slew rate do amplificador diferencial

5 V/ μ s

12. Potência do amplificador de áudio

200mW

13. Entradas e saídas analógicas

8 canais de entrada

8 canais de saída

A P Ê N D I C E D

SEGMENTO DE VOZ UTILIZADO NAS SIMULAÇÕES

Nas simulações realizadas, neste trabalho, foi utilizado um segmento de voz foneticamente balanceado com duração aproximada de 14 segundos.

Considera-se um segmento de voz foneticamente balanceado uma locução que contenha todos os fonemas de uma determinada língua natural exibindo a mesma frequência de ocorrência de fonemas que apareceria em uma conversação informal.

A seguir é apresentada a locução que foi utilizada nas diversas simulações:

"ONTEM MEU FILHO FUGIU DE CASA LEVANDO O SEU CACHORRO, UM BONÉ PRETO E AINDA O DINHEIRO QUE GANHOU DA AVÓ. O MEU GOSTOSO, QUENTE E RELAXANTE BANHO, MUDOU MINHA VONTADE DE TRABALHAR, PORQUE GERALMENTE É UM ÓTIMO REMÉDIO NO CASO DE FADIGA".