INSTITUTO MILITAR DE ENGENHARIA

RECONHECIMENTO AUTOMÁTICO DE COMANDOS CONECTADOS

POR

TEN QEM MARCO ANTÔNIO ROCCA DE ANDRADE

TESE SUBMETIDA COMO REQUISITO PARCIAL PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA

Assinatura do Orientador da Tese

_____

TC QEM Sidney Cerqueira Bispo dos Santos – D.Sc.

Assinatura do Co-orientador da Tese

_____

CEL R/1 Roberto Miscow Filho – M.C.

Rio de Janeiro – RJ

Dezembro de 1999

Em memória de meu pai.

AGRADECIMENTOS

Ao Cel R/1 Roberto Miscow Filho pelos ensinamentos transmitidos e pela valiosa e incansável colaboração nas minuciosas correções do texto desta dissertação.

Ao TC QEM Sidney Cerqueira Bispo dos Santos pela orientação no decorrer do trabalho.

Ao Cap QEM Dirceu Gonzaga de Silva pela implementação das rotinas de treinamento de *HMM* contínuo.

Aos vários locutores que participaram do desenvolvimento do sistema, dentre eles: Douglas Corbari Corrêa, Maurício Capra, Alexandre Guedes de Melo, André Gustavo de Carvalho Albuquerque, Carlos Eduardo Marques Silva, Roberson Fernandes Loriato, Álvaro Marcos Antônio de Araujo Pistono, Erick Simões da Camara e Silva, Jorge Audrim Morgado de Gois, Luiz Augusto Cavalcante Moniz de Aragão Filho, Fernando Monteiro da Silva, Rodrigo dos Santos Marques Porto, Alexandre de Macedo Torturela, Jorge Bonfim, André Luís Miguez Oliveira, Marcos Paulo, Marco Aurélio Rocca de Andrade, Rafael Fernandes da Rocha, Edgar Tavares Crespo Neto.

A Maria José Rocca de Andrade pelos maternos incentivos.

A Lilian Caroline Barella de Oliveira pela compreensão, paciência e incentivo.

Ao Departamento de Engenharia Elétrica do IME (DE/3), agradeço a pronta cooperação oferecida pelo corpo docente e funcionários.

RESUMO

Neste trabalho é proposto um sistema de reconhecimento automático de palavras conectadas, para a língua portuguesa, voltado para controle de servo-mecanismos.

É proposto um método alternativo para a determinação dos pontos terminais (*end-points)* das elocuções. Este método apresentou resultados superiores aos obtidos pelo processo da energia e da taxa de cruzamentos por zero.

Os atributos da voz utilizados são: 5 primeiros coeficientes *PLP-Cepstrum* com suas primeiras e segundas derivadas, e log-energia de tempo curto com sua primeira e segunda derivadas. É proposta uma adaptação do método *PLP* para operar na freqüência de amostragem de 11.025 Hz.

Para modelagem matemática das palavras por processo estocástico é usado um algoritmo de *HMM* contínuo e para decodificação de frases é empregado o algoritmo *Level-Building.*

O vocabulário é composto por 45 palavras, incluindo os dígitos, e a gramática regular proposta permite a composição de 6 tipos diferentes de frases, variando de 1 até 10 palavras em uma sentença. Por meio do número de *bigramas* estabeleceu-se 137 frases para treinamento dos modelos estocásticos. É proposto também um pós-processador sintático para elevar a taxa de acerto baseado nas peculiaridades da tarefa a ser controlada. Foi utilizado um conjunto de 41 frases para verificação dos resultados. Para treinamento do sistema independente do locutor foram utilizados 15 locutores para treinamento e 5 para verificação.

O sistema foi implementado tanto para dependência como para independência de locutor.

Após a realização dos testes verificou-se que a maioria dos erros ocorrem por substituição de palavras, e com menor freqüência ocorrem inclusões de vocábulos. Em geral, a substituição de dígitos ocorre com freqüência menor do que as demais palavras treinadas. A taxa de acerto foi de aproximadamente 99,5% para locutor único e 94,5% para o sistema independente de locutor.

ABSTRACT

In this work, na automatic recognition system of connected words for the Brazilian Portuguese language is proposed so as to control servomechanisms.

An alternative method is proposed in order to determine the end-points of the elocutions. This method presented better results than those obtained by the process of the energy and zero crossings measures.

The features of speech are: first 5 coefficients PLP-Cepstrum with their first and second derivative, and log-energy of short time with its first and second derivative. An adaptation for the PLP method is proposed to operate in the sampling rate of 11.025 Hz.

To mathematical model the words by stocastic process, an algorithm of continuous HMM is used, and, to decode sentences, the Level-Building algorithm is used.

The vocabulary is composed by 45 words, including the digits; and the regular grammar proposed allows the composition of 6 diferent types of sentences, varying from 1 to 10 words in a sentence. By means of the number of bigrams, it was fixed 137 sentences for stocastic model training. It is also proposed a syntactic post-processor to elevate the success rate based on the peculiarities of the task to be controlled. A group of 41 sentences was used for verification of the results. To train the independent speaker system, 15 speakers were used for training and 5 for verification.

The system was implemented for speaker dependence and independence.

After the accomplishment of the tests, it was verified that most of the mistakes is due to substitution of words, and, with smaller frequency, there are inclusions of words. In general, the substitution of digits occurs less often than the other trained words. The success rate was of approximately 99,5% for unique speaker and 94,5% for the independent speaker system.

SUMÁRIO

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SÍMBOLOS

PLP	*Perceptual Linear Predictive*
HMM	*Hidden Markov Model*
LB	*Level Building*
A/D	Analógico-digital
LPC	*Linear Predictive Coefficients*
FFT	*Fast Fourier Transform*
fdp	Função de distribuição de probabilidade
MKM	*Modified K-means*
DTW	*Dinamic Time-Warping*

CAPÍTULO 1

INTRODUÇÃO

1.1- IMPORTÂNCIA DA PESQUISA DE VOZ NO CONTEXTO ATUAL

O grande desenvolvimento da tecnologia atual, gerando sistemas e dispositivos cada vez mais rápidos, exige uma interação mais intensa entre homens e máquinas. A extensão dessa capacidade permitirá a utilização dos computadores por toda uma nova gama de usuários, sem necessidade de treinamento específico ou de equipamentos especiais. Várias formas de comunicação desenvolveram-se pelo uso de teclados e terminais, porém a fala permanece como a forma mais simples, mais natural e eficaz.

Além da facilidade de interação com as máquinas, outros fatores têm estimulado o desenvolvimento de pesquisas nessa linha, entre os quais podem ser citados o crescimento da capacidade computacional aliado ao decrescimento do custo dos microcomputadores e o surgimento de microprocessadores especializados para as diversas formas de tratamento da voz.

1.2- DIMENSÃO DA DIFICULDADE

Um dos objetivos da tecnologia do reconhecimento de voz[1,2], num sentido amplo, é construir sistemas que recebam a informação falada e atuem apropriadamente sobre tal informação. Nestes termos, máquinas que realizem reconhecimento automático de voz são parte de um grupo de máquinas chamadas *artificialmente inteligentes* que podem ouvir, entender, atuar (devido à informação transmitida através da voz), e falar (permitindo troca de informações). Porém, a construção de máquinas robustas, inteligentes e que conversem fluentemente, permanece como distante meta a ser alcançada.

O que se entende por reconhecimento de voz está em contínua evolução em velocidade expressiva. O conhecimento atual ainda é relativamente modesto e de aplicações muito restritas. Atualmente, a maioria dos sistemas práticos reconhecem somente palavras isoladas, com pequeno vocabulário e pouco robustos ao ruído ambiente (local de uso sem tratamento acústico especial). Sistemas que permitam uma comunicação mais natural entre homem e máquina ainda são experimentais. E todos eles apresentam melhor desempenho quando treinados e utilizados por um único locutor. Sistemas para múltiplos locutores ainda possuem desempenho inferior aos que utilizam um único locutor.

Por outro lado, a maioria das aplicações em que se deseje fazer uso de reconhecimento de voz necessita de maior complexidade do que a existente usualmente em sistemas onde palavras isoladas são claramente pronunciadas por um locutor único e sem ruído ambiente. Um sistema para uso mais amigável e robusto deve ser capaz de reconhecer voz contínua de vários locutores, aceitar voz com pronúncias diferentes e com acentos regionais, permitir adaptações de vocabulário, corrigir palavras mal pronunciadas, com diferentes tendências gramaticais e voz em ambientes ruidosos. Tudo isso incorporado em um sistema pequeno, prático, com operação em tempo real e que possa adaptar-se, fazendo aprendizado de novos modelos léxicos, sintáticos, semânticos e pragmáticos. Dentro deste contexto, o reconhecimento de voz encontra-se em um estágio que pode ser chamado de preliminar, mesmo com o avanço dos últimos anos. Apesar disso, o progresso obtido permite a utilização limitada do reconhecimento de voz em tarefas de grande interesse[1,2].

Um dos grandes problemas[3] no reconhecimento de voz é que ainda não se sabe perfeitamente todas as etapas do processo que ocorre no aparelho auditivo e no cérebro humano durante o reconhecimento. Outro problema é a natureza não linear da audição, indicando que modelos matemáticos lineares e simples são inadequados para análise da voz. Atualmente, a forma encontrada para superar a falta de conhecimento sobre os mecanismos biológicos do reconhecimento de voz é o uso de processos computacionais de alto desempenho baseados em modelos estocásticos. Nesses processos, a quantidade de informação utilizada é menor do que em sistemas determinísticos equivalentes, porém há um esforço computacional maior para o treinamento dos modelos matemáticos representantes dos eventos acústicos.

1.3 - OBJETIVO DO TRABALHO

Este trabalho tem por objetivo: propor, implementar e avaliar o desempenho de um sistema para reconhecimento de conjuntos de comandos vocais conectados, em língua portuguesa, voltado para o controle de deslocamentos de objetos genéricos no espaço e acionamentos eletro-mecânicos, fazendo uso de modelos acústicos obtidos por processos estocásticos.

1.4 - COMPOSIÇÃO DO COMPENDIUM

O capítulo 2 apresenta as etapas de pré-processamento da voz para tornar o sinal tratável pela parte de treinamento e reconhecimento do sistema. Sugere uma alternativa para o método de determinação de pontos terminais. Faz também uma explanação sumária do método de análise por coeficiente *PLP* e dos ajustes do

método para o ambiente (no sentido computacional) da dissertação. Apresenta os atributos da voz que serão empregados no restante do trabalho.

O capítulo 3 expõe a teoria básica de modelagem estocástica baseada em modelos de Markov escondidos (*HMM*), as adaptações para a modelagem de palavras e o algoritmo *Level Building (LB)*.

O capítulo 4 apresenta o vocabulário voltado para a tarefa de deslocamento e acionamento de dispositivos eletro-mecânicos, sua gramática e demais aspectos da linguagem sugerida. Apresenta também considerações sobre a montagem das frases usadas para treinamentos e testes.

O capitulo 5 descreve o sistema de reconhecimento proposto, o método usado para modelagem do vocabulário e das frases possíveis e o algoritmo montado para o reconhecimento.

O capítulo 6 apresenta os resultados obtidos, avalia o desempenho do sistema, discute aspectos práticos da tarefa de modelagem e reconhecimento, expõe conclusões atingidas e sugere trabalhos futuros.

CAPÍTULO 2

PRÉ-PROCESSAMENTO DA VOZ

2.1 - INTRODUÇÃO

Em um sistema de reconhecimento automático de comandos a voz, o primeiro passo é a transformação do sinal sonoro em uma seqüência de números que possam ser tratados computacionalmente. Nesta conversão analógico/digital, o mais comum é a obtenção de um arquivo contendo a evolução da amplitude do sinal ao longo do tempo, amostrado em determinada freqüência[4,5,6]. Atualmente estão disponíveis comercialmente várias placas específicas para o processamento de sinais sonoros.

O segundo passo é a determinação dos pontos terminais da elocução sobreposta ao silêncio ou ruído de fundo da gravação ou do ambiente. Esta etapa visa evitar o desperdício de esforço computacional em amostras que não contenham informação relevante para o processo de reconhecimento.

O terceiro passo é a extração de parâmetros ou atributos relevantes do sinal de voz. Essa etapa é uma compressão de dados com o objetivo de reduzir a dimensão do espaço em que são definidas as elocuções, ou seja, consiste em realizar um mapeamento do espaço das elocuções de dimensão *M*, número de amostras do sinal de voz, em um espaço dimensional *N*, número dos parâmetros (atributos ou características) extraídos desse sinal, sendo o número de elementos da dimensão *N* menor que os originais da dimensão *M*. Como exemplo, se um sinal possuir 22.000 amostras, e, no processo de extração de atributos, tiver as amostras agrupadas em quadros[4,5,7] de 110 amostras, e se de cada quadro for gerado um vetor com 5 atributos ou coeficientes, o sinal inicial com M = 22.000 amostras passa a ser representado por uma matriz de 200 quadros x 5 coeficientes, resultando em $N = 1.000 < M$.

Algumas destas técnicas de mapeamento procuram fazer uma representação paramétrica de percepção do sinal de voz de modo que haja uma correlação com o que é percebido pelo sistema de audição humano[3,8].

2.2 - AQUISIÇÃO DE SINAL DE VOZ

No presente trabalho, todos os sinais foram adquiridos com uma placa de som da marca *Sound Blaster*, com taxa de amostragem de 11.025 Hz, 16 *bits* por amostra, em mono-canal, com o controle automático de ganho incorporado ativado, e com microfone acoplado a fones de ouvido da marca *Boeder*.

Um cuidado especial foi tomado visando reduzir o erro de quantização[9] do conversor analógico/digital incorporado à placa *Sound Blaster*. Elocuções com todas as amostras de pouca amplitude foram rejeitadas pois implicavam em uma sub-utilização do conversor. Só foram aceitas gravações de elocuções onde a ocorrência de amostras com amplitude absoluta acima de 25% da amplitude de fundo de escala do conversor A/D fosse superior a 0,5 % do total de amostras. Os valores de 25% e 0,5% foram estipulados após uma seqüência de testes com elocuções de palavras isoladas e frases variadas realizadas por um único locutor. Estes valores podem ser ajustados com um estudo dos fonemas predominantes em um determinado vocabulário, de modo a otimizar a quantização do sinal de voz.

2.3 - DETERMINAÇÃO DE PONTOS TERMINAIS

A diferenciação entre o que é ruído e o que é voz, pelo computador, é uma das tarefas iniciais no reconhecimento automático da fala. Uma diferenciação incorreta pode prejudicar, logo nas primeiras etapas, o processo de reconhecimento. Gravada uma elocução, a determinação do ponto onde ocorre a transição entre sinais pertencentes ao ruído de fundo e os sinais relevantes da voz tem por objetivo orientar a pesquisa para o intervalo em que o sistema de reconhecimento deverá ser aplicado. Conhecidos os pontos terminais, pode-se concentrar o esforço computacional no intervalo de interesse e economizar tempo de processamento nas várias etapas subseqüentes.

Um método simples para a determinação dos pontos terminais da locução é a observação direta da voz digitalizada em um dos diversos programas disponíveis hoje no mercado. Com o auxílio de gráficos e repetição de locução, o operador pode estimar os pontos que limitam o sinal relevante.

Um método mais usado para tornar automático o processo é o da energia e da taxa de cruzamento por zero [3,4,5,6,7,10]. Nesse processo, o sinal é dividido em quadros correspondentes a intervalos de dez milisegundos e as comparações são feitas entre quadros. No início da gravação, considera-se um intervalo em torno de cem milisegundos (ou dez quadros) como sendo constituído por ruído de fundo ambiente. Desse início são extraídas a energia e a taxa de cruzamentos por zeros. A partir desses valores iniciais, são determinados valores que serão usados como limiares de decisão na busca do início e do fim da locução. Após a comparação das características

de um quadro do sinal com os limiares e admitindo que ele contém um sinal relevante, os quadros vizinhos são reexaminados para testar a continuidade da locução e impedir que um pico espúrio seja erroneamente admitido como relevante. Há varias formas de determinação dos limiares de comparação e meios de exclusão de picos espúrios dentro desta linha de determinação de pontos terminais[3,10].

Os dois métodos citados apresentam alguns inconvenientes. O primeiro é demorado, tedioso e não pode ser aplicado em tempo real. O segundo é relativamente vulnerável às condições de ruído ambiente, e os atributos e quadros usados nem sempre são aproveitadas no restante do processo de reconhecimento.

Para o presente trabalho foi proposto um terceiro método regido pela idéia de que o sinal de voz apresenta uma variação entre atributos (coeficientes Mel-Ceps ou Mel-PLP, por exemplo) de janelas adjacentes maior do que as variações do sinal de ruído ambiente. A simples observação de espectrogramas de elocuções realizadas em ambientes com baixo ruído permite encarar a determinação de pontos terminais sobre esta ótica. A Figura 2.1 apresenta o espectrograma da elocução da frase *bom dia*, ladeada por amostras de ruído ambiente. Observa-se a relativa uniformidade do ruído em comparação com o trecho da elocução relevante (amostras de 3.000 a 7.000 aproximadamente na Figura 2.1). Assim, distâncias euclidianas entre janelas que contenham atributos de ruído ambiente estariam abaixo de um limiar, indicando pouca variação e sinal irrelevante, e distâncias entre janelas que contenham atributos de sinal de voz estariam acima de um limiar, indicando uma

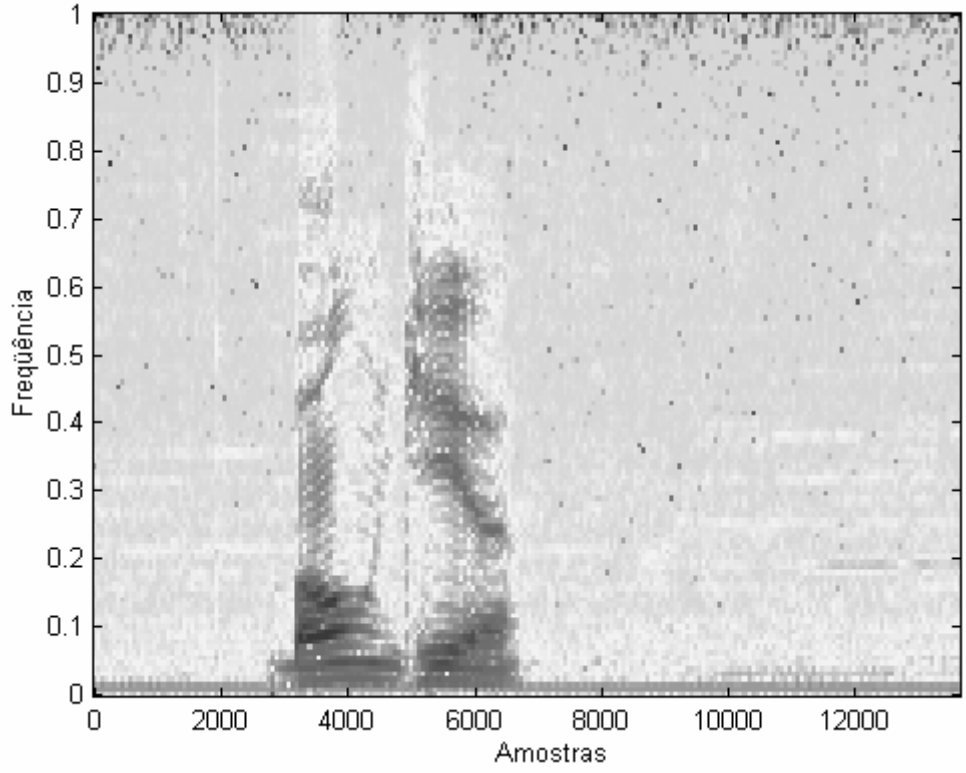

FIGURA 2.1 – Espectrograma da elocução *bom dia,* com eixo vertical normalizado por f_s = 11.025 Hz.

seqüência de transições que provavelmente seria causada pela seqüência de fonemas, indicando um sinal relevante para identificação. Supõe-se, neste ponto, que o ruído ambiente possui a menor variação de atributos entre janelas, ou, de outro modo, é o trecho mais uniforme da gravação, e que um fonema muito prolongado não traz informação importante para a tarefa de reconhecimento de voz, podendo assim ser confundido com ruído sem causar, entretanto, problemas quando apresentar poucas variações entre janelas vizinhas. Outro fator que guiou o método proposto foi tornar a etapa de determinação de pontos terminais parte do processo de extração de atributos (tratada no próximo item), utilizando valores intermediários desse processo para determinar um limiar de decisão para o início e o fim da elocução com maior precisão.

O cálculo do limiar de decisão foi feito com base nas cinqüenta primeiras janelas do intervalo gravado, admitiu-se que neste início havia somente atributos (por exemplo: coeficientes Mel-Ceps ou PLP-Ceps) do ruído ambiente com uma distribuição próxima da gaussiana. Este intervalo possui duração e função similares ao intervalo inicial do método da energia e taxa de cruzamento por zero. Eram calculados as médias e os desvios padrão de cada atributo ao longo das cinqüenta janelas iniciais. Os valores foram consolidados em um único limiar determinado pela distância euclidiana entre um vetor *M+ (M mais),* formado pelas médias mais os desvios, e um vetor *M- (M menos)* formado pelas médias menos os respectivos desvios padrão. Caso a distância entre duas janelas do restante da gravação fosse maior que este limiar, estas janelas serão candidatas a representar sinais relevantes de voz.

A determinação do ponto inicial da locução dentro da gravação é feita varrendo-se as janelas ao longo do tempo, iniciando-se a varredura logo após a última janela usada na caracterização do ruído de fundo. O primeiro vetor da janela de teste que estiver afastado da janela vizinha além do limiar de decisão é considerado como forte candidato a representar uma janela de locução válida. Para se ter a confirmação de que a janela corresponde realmente ao início de uma locução válida, são examinadas as nove janelas seguintes, e se dentre elas mais de seis também forem válidas, a primeira janela deste conjunto é aceita como início de locução relevante. Caso contrário, prossegue-se na busca. Este procedimento visava evitar falso reconhecimento causado por picos espúrios.

A determinação do ponto final da locução segue o mesmo processo da determinação do ponto inicial, porém iniciando a pesquisa pelo final da elocução.

Este método proposto supõe que os intervalos de silêncio ou ruído ambiente são pequenos se comparados com a elocução contida entre eles, e que o locutor tem controle sobre o período de gravação. Este controle (semelhante ao *push-to-talk* em rádio-transmissores) delimita o tempo de gravação útil. Atendidas estas

restrições, obtém-se que o esforço computacional para a extração de atributos de todo o período gravado não é maior que duas vezes o tempo gasto para a extração dos atributos do trecho relevante, para o caso de uma palavra isolada. À medida que o número de palavras de uma elocução aumentava, a parte do sinal correspondente a silêncio tende a ter uma menor parcela da duração da gravação.

Sob o ângulo deste método proposto, a determinação dos pontos terminais passa a ser incorporada na etapa de extração de atributos, podendo ocorrer no final ou em alguma etapa intermediária da extração, fazendo uso, por exemplo, de valores obtidos em bancos de filtros.

A tabela comparativa entre os métodos é apresentada no capítulo de resultados.

2.4 - ATRIBUTOS DA VOZ

Existem vários atributos que podem ser extraídos a partir de uma seqüência digitalizada de voz. Cada atributo apresenta desempenho próprio para cada atividade do reconhecimento. Por exemplo, a *freqüência fundamental, freqüências formantes e co-articulação nasal* tem emprego na identificação de locutores[5,6], *energia e taxa de cruzamentos por zero* no reconhecimento de palavras[4,7], enquanto *coeficientes de predição linear (LPC),coeficientes cepstrum* e *mel-cepstrum* podem ser aproveitados em ambos os casos[4,5,6,7].

Para o presente trabalho optou-se por fazer uso de coeficientes oriundos da análise de *predição linear perceptiva (Perceptual Linear Predictive PLP)*[8], tendo em vista os resultados atingidos, em tarefas de reconhecimento, superiores aos dos demais atributos citados anteriormente[10,11]. Além dos coeficientes PLP, fez-se uso do atributo *energia de tempo curto* e coeficientes *delta* e *delta-delta*.

2.4.1 - Coeficientes PLP[8]

A abordagem mais utilizada para a estimação da envoltória espectral suavizada de um sinal de voz é a que utiliza coeficientes de predição linear (*LPC*). Existem diversos algoritmos eficientes e rápidos para o cálculo e para realizar transformações entre diversos espaços paramétricos. Quando a ordem do modelo só de pólos é bem escolhida, o modelo faz uma boa aproximação das áreas de alta concentração de energia do trato vocal, enquanto despreza harmônicos de baixa energia e outros detalhes espectrais menos relevantes. Entretanto, essa propriedade não é consistente com as peculiaridades do ouvido humano.

Com o objetivo de levar em consideração as características do sistema auditivo, o método *PLP* altera o espectro do sinal de voz antes da análise de predição linear. Faz-se uso de um banco de filtros assimétricos de banda não muito estreita espaçados pela escala Bark[12]. Faz-se uso também de pré-ênfase e compressões com o

objetivo de simular determinadas características da audição humana. Os passos envolvidos na transformação para cada janela do sinal de voz são citados a seguir:

- Análise espectral: realizada com o cálculo da transformada rápida de Fourier e do espectro de potência.

- Análise de banda crítica: pondera-se o espectro de freqüência, de acordo com a escala Bark[11], utilizando a seguinte equação[8]:

$$\Omega(\omega) = 6\ln\left(\frac{\omega}{1200\pi} + \sqrt{\left(\frac{\omega}{1200\pi}\right)^2 + 1}\right) \tag{2.1}$$

onde ω é a freqüência angular em *rad/s*. Em seguida, faz-se a convolução do espectro ponderado com o espectro de potência de uma banda crítica simulada, onde a forma do filtro é dada pela seguinte equação[8]:

$$\Psi(\Omega) = \begin{cases} 0 & \Omega < -1{,}3 \\ 10^{2{,}5(\Omega + 0{,}5)} & -1{,}3 \le \Omega \le -0{,}5 \\ 1 & -0{,}5 \le \Omega \le 0{,}5 \\ 10^{-\Omega + 0{,}5} & 0{,}5 \le \Omega \le 2{,}5 \\ 0 & \Omega \ge 2{,}5 \end{cases} \tag{2.2}$$

e a convolução por[8]:

$$\Theta(\Omega_i) = \sum_{\Omega = -1{,}3}^{2{,}5} S(\Omega - \Omega_i)\Psi(\Omega) \tag{2.3}$$

A função $\Theta(\Omega)$ é amostrada em intervalos de aproximadamente 1 Bark. O valor exato do intervalo de amostragem é escolhido de forma que um número inteiro de amostras espectrais cubra todo o intervalo de análise. Como exemplo, para cobrir uma faixa de 0 a 5 kHz, equivalente a 0 até 16,9 Barks, pode-se utilizar 18 amostras espectrais espaçadas de 0,994 Bark cada uma.

- Pré-ênfase: o espectro amostrado $\Theta(\Omega)$ é pré-enfatizado por uma curva que simula as diferenças de sensibilidade do ouvido humano às diversas freqüências. O sinal resultante é representado por[8]

$$\Xi[\Omega(\omega)] = E(\omega)\Theta[\Omega(\omega)] \tag{2.4}$$

onde $E(\omega)$ é dado por[8]

$$E(\omega) = \frac{(\omega^2 + 56{,}8x10^6)\omega^4}{(\omega^2 + 6{,}3x10^6)^2(\omega^2 + 0{,}38x10^9)(\omega^6 + 9{,}58x10^{26})} \tag{2.5}$$

para sons em níveis moderados e em freqüências de 0 até superiores a 5.000 Hz. Esta curva de simulação de sensibilidade não é muito rígida, podendo sofrer ligeiras alterações sem que ocorram mudanças significativas. O fato de ainda não serem conhecidas todas as características da audição humana média deixa em aberto a determinação de $E(\omega)$.

- Compressão cúbica da amplitude: Para simular a relação não linear entre a intensidade do som real e a percebida, faz-se uma compressão cúbica conforme a equação[8]

$$\Theta(\Omega) = \Xi(\Omega)^{0,33} \tag{2.6}$$

- **Modelo só de pólos**: A partir da função $\Theta(\Omega)$ é calculada a transformada inversa de Fourier e obtém-se um modelo só de pólos utilizando o método da autocorrelação.

- **Transformações adicionais**: na utilização do PLP no reconhecimento de voz é comum a realização de transformações adicionais sendo a mais comum a passagem de PLP para PLP-Cepstro e suas primeiras e segundas derivadas[11].

Finaliza-se assim a extração dos atributos PLP de uma janela do sinal de voz.

O número de coeficientes a serem gerados e utilizados depende da aplicação. Para tarefas de reconhecimento de *locutor* são utilizados mais de 12 coeficientes (modelos de ordens elevadas), e para tarefas de reconhecimento de voz *independente do locutor* são utilizados modelos de ordens inferiores com 5 coeficientes apenas.

2.4.2 – Considerações Práticas Sobre PLP

O esforço computacional para o cálculo do PLP é comparável ao gasto para a análise preditiva linear tradicional[8], sendo que as etapas de análise de banda crítica e pré-ênfase podem ser unidas e seus coeficientes calculados a priori. A operação mais custosa é a transformada rápida de Fourier. A passagem para coeficientes auto regressivos se realiza com poucos pontos oriundos da análise de banda crítica.

Para o presente trabalho, optou-se por realizar a FFT apenas com número inteiros (16 *bits*), em vez de números de ponto flutuante (80 *bits*) para acelerar o cálculo computacional. Esta redução de precisão não causou variações muito acentuadas nos espectogramas gerados a partir dos resultados com os dois níveis de precisão, e evitou a normalização dos valores inteiros obtidos da placa de aquisição de sinal. Os efeitos foram mais atuantes na transformada inversa. Os coeficientes de ordem mais elevada passaram a apresentar valores diferentes dos obtidos com precisão extendida. Porém, como os coeficientes elevados não possuem utilidade para o reconhecimento independente do locutor, esta divergência foi considerada como sem relevância para o trabalho.

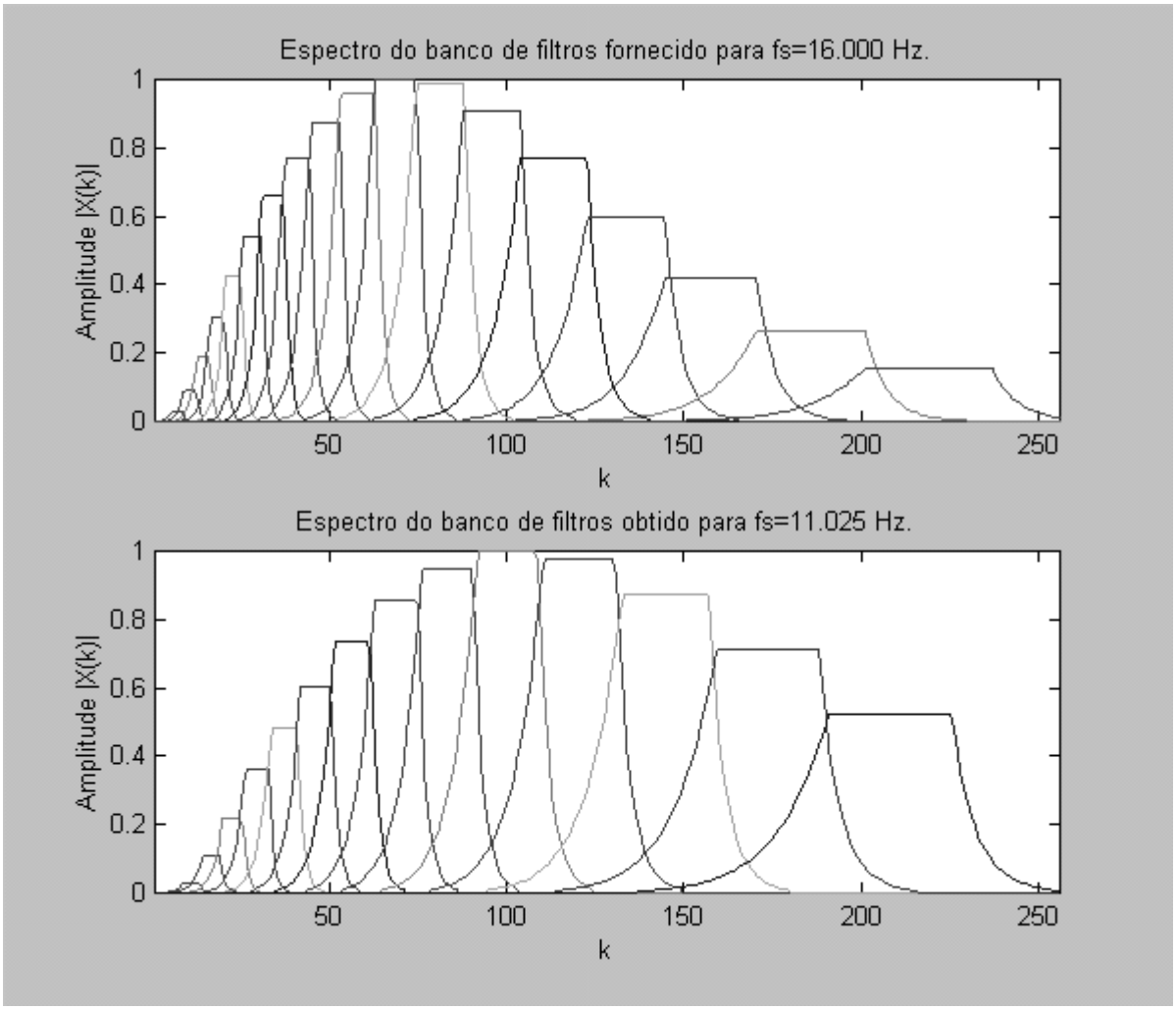

FIGURA 2.2 – Bancos de filtros.

O programa para cálculo de FFT direta e inversa com números inteiros em linguagem *C*, foi adquirido via Internet através do endereço http://www.jjj.de/fxt/, de autoria de Tom Roberts, sofrendo pequenas correções em função de teste realizados antes da inclusão na rotina de extração de coeficientes PLP, utilizada nesta dissertação. O autor do método PLP forneceu, posteriormente à publicação do seu artigo, os valores dos bancos de filtros das bandas críticas já ponderados por uma curva de pré-ênfase. Porém, os dados eram para as freqüências de amostragem de 8.000 Hz ou para 16.000 Hz, com uma curva de pré-ênfase diferente da utilizada no artigo[8]. Para compatibilizar o banco de filtros com a freqüência de amostragem de *fs*=11.025 Hz, de uso comum no Instituto, realizou-se a interpolação da envoltória do banco de filtros correspondente a *fs*=16.000 Hz para se obter a nova curva de pré-ênfase. O número de filtros também foi modificado de 19 para 15, com espaçamento de 1,08 Bark. Manteve-se o número de 512 pontos para a transformada. O gráfico superior da Figura 2.2 contém as curvas dos 19 filtros fornecidos para cálculo de coeficientes PLP para uma f_s = 16 kHz. O gráfico inferior contém as curvas dos 15 filtros propostos para a f_s = 11.025 Hz, seguindo as equações do item 2.3.1., porém equalizado por uma curva interpolada do gráfico superior. Em ambos os gráficos na Figura 2.2, a

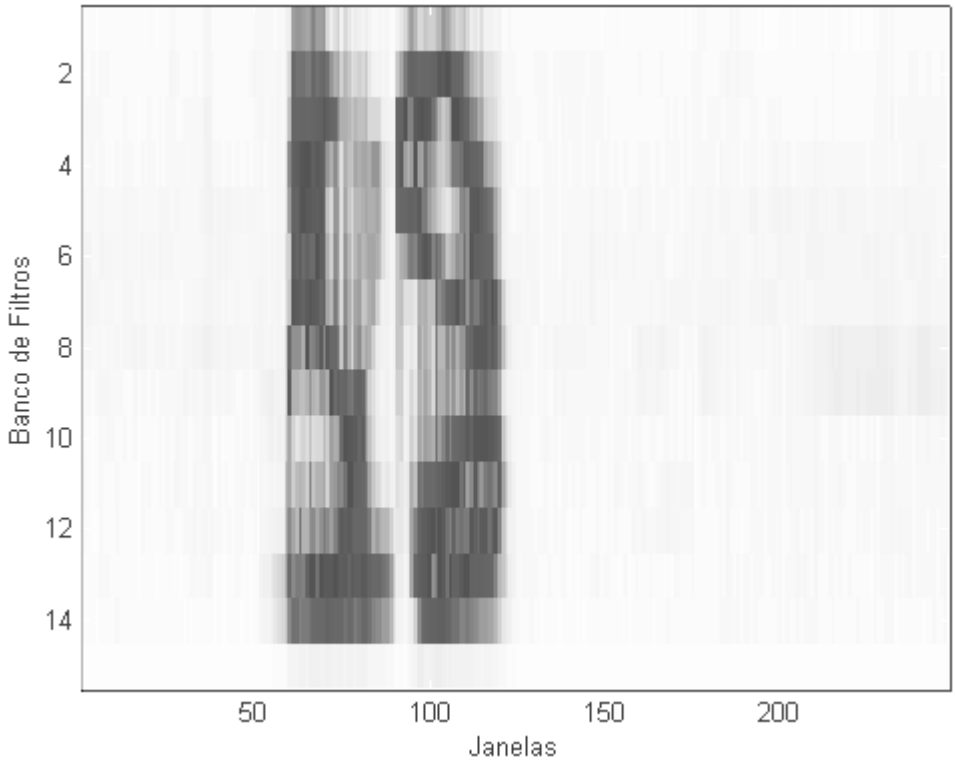

FIGURA 2.3 - Espectrograma obtido após análise de banda crítica da elocução *bom dia*.

primeira curva, correspondente a banda de freqüências mais baixas, possui amplitude e comprimento muito reduzidos, ficando imperceptível neste gráfico.

A Figura 2.3 apresenta o espectrograma obtido após análise de banda crítica da elocução *bom dia*. O banco de filtros está numerado da freqüência mais baixa para a mais alta. Comparando-a com a Figura 2.1, nota-se as *deformações* e a perda de definição do espectro durante a extração de coeficientes PLP causadas pela passagem pelo banco de filtros. Lembrar que a ordenação das freqüências, na vertical, é diferente para cada figura.

2.4.3 – Energia de Tempo Curto

A energia de tempo curto é uma das representações mais simples de um sinal de voz. No caso de um sinal no tempo discreto, *x(n)*, a energia de tempo curto, em geral, é definida como[3]:

$$E_n = \sum_{m=-\infty}^{\infty} x^2(m)h(n - m * N_f), \qquad 0 \le n \le N\text{-}1. \tag{2.7}$$

onde *N* representa o número de janelas, *M* o número de amostras, N_f é a duração de cada janela e *h(.)* é a função de janelamento de Hamming[9].

Para o presente trabalho foi empregado o logaritmo na base 10 da energia de tempo curto[3,7,11,12] obtida da Equação (2.7).

2.4.4 – Coeficientes Delta e Delta-Delta

Estes parâmetros são obtidos através das derivadas de primeira e segunda ordem dos atributos. Podem ser utilizados para representar as mudanças dinâmicas no espectro da voz e, desse modo, detectar variações bruscas dentro do espectro. Uma das aproximações mais populares é[3,11,12]:

$$s(n) \equiv \frac{d}{dt}x(n) \approx x(n+2) - x(n-2). \tag{2.8}$$

Os parâmetros de segunda ordem são obtidos reaplicando a derivada sobre os resultados obtidos na primeira derivação[3,11,12]:

$$\dot{s}(n) = \frac{d}{dt}s(n) \approx s(n+2) - s(n-2). \tag{2.9}$$

2.5 - CONCLUSÃO

Apresentou-se neste capítulo as transformações sofridas pelo sinal de voz até torná-lo tratável pelas rotinas de aprendizagem e reconhecimento. Uma alternativa para o processo de determinação de pontos terminais foi apresentada para aumentar a eficiência e eficácia do mesmo. O método PLP foi ajustado para as condições do sinal usado na dissertação.

CAPÍTULO 3

FUNDAMENTOS DE MODELOS DE MARKOV ESCONDIDOS (HMM) EM RECONHECIMENTO DE VOZ

3.1 – INTRODUÇÃO

Uma cadeia de Markov[13,14,15] é um conjunto finito de elementos, formando uma máquina de estados. Nessa máquina de estados as transições entre os estados não são governadas por regras determinísticas mas por probabilidades de transição entre eles. Porém, se em cada estado, uma determinada saída ou observação puder ser gerada de acordo com uma distribuição de probabilidade (em vez das regras determinísticas normalmente encontradas em máquinas de estados) e se somente a saída ou observação (e não o estado que a gerou) for visível a um observador externo ao processo, então os estados estarão 'escondidos' do exterior. Daí o nome de Modelos de Markov Escondidos[3,5,12,13,16] (Hidden Markov Model – HMM).

Diversos fenômenos podem ser modelados por meio de máquinas de estados finitos. E quando os

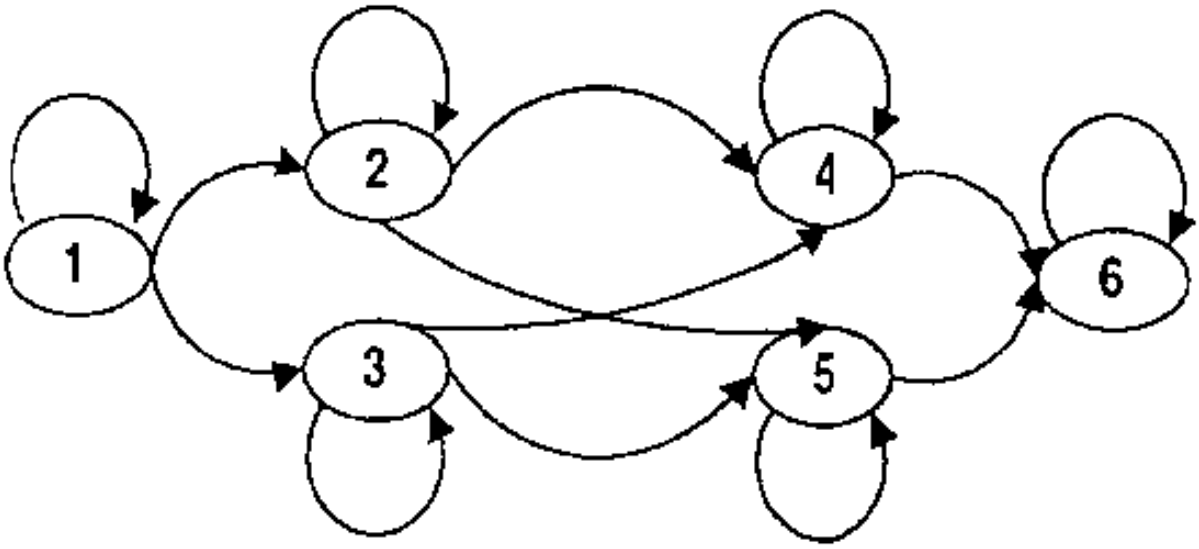

FIGURA 3.1 – Diagrama genérico de uma máquina de estados com 6 elementos.

fenômenos possuem características de processos estocásticos, pode-se pensar em usar HMM's como formas de modelá-los.

Como exemplo de um fenômeno trivial, em que se pode usar HMM para a modelagem, considere-se o caso de um senhor, que recebe da esposa a tarefa de ir à feira e trazer várias frutas relacionadas em uma lista de

compras. Após algum tempo, o senhor retorna e apresenta à esposa as frutas compradas conforme a lista. Neste caso, as saídas ou observações são as frutas obtidas e é somente o que a dona de casa pode avaliar como observadora externa do processo. Não está explicito nas frutas qual foi o caminho percorrido pelo consumidor através das finitas barracas e o que motivou esse caminho. Aspectos como distâncias, preços, quantidades, e qualidades das frutas influenciaram no trajeto, e estes aspectos podem possuir características estocásticas. Ir à feira com uma lista de compras pode ser um problema modelável com HMM.

Uma vez associado o modelo ao fenômeno, pode-se responder a perguntas tais como: qual o melhor trajeto de visita pelas barracas ou se determinado trajeto possibilita a aquisição de toda a lista de compras de forma satisfatória.

3.2 – ELEMENTOS DE UM HMM

Para definir um HMM de forma completa são necessários os seguintes elementos:

- O número de estados do modelo, N;
- O número de símbolos observáveis em um alfabeto, M. Para símbolos discretos, M pode ser inteiro e finito, para símbolos contínuos, M pode ser infinito;
- O conjunto de probabilidades de transição de estados $A = \{a_{ij}\}$:

$$a_{ij} = p\{q_{t+1} = j \mid q_t = i\}, \qquad 1 \le i,j \le N; \tag{3.1}$$

onde q_t denota o estado corrente. As probabilidades de transição devem satisfazer as condições estocásticas:

$$a_{ij} \ge 0, \qquad 1 \le i,j \le N \qquad \text{e} \tag{3.2}$$

$$\sum_{j=1}^{N} a_{ij} = 1 \qquad 1 \le i \le N. \tag{3.3}$$

- A distribuição de probabilidade de cada estado, $B=\{b_j(k)\}$:

Se a distribuição é discreta,

$$b_j(k) = P\{o_t = v_k \mid q_t = j\}, \qquad 1 \le j \le N, \qquad 1 \le k \le M; \tag{3.4}$$

onde v_k denota o k-ésimo símbolo do alfabeto, e o_t a observação corrente. Novamente a seguintes condições estocásticas devem ser satisfeitas:

$$b_j(k) \ge 0, \qquad 1 \le j \le N, \qquad 1 \le k \le N, \qquad \text{e} \tag{3.5}$$

$$\sum_{k=1}^{M} b_j(k) = 1, \qquad 1 \le j \le N. \tag{3.6}$$

Se a distribuição é contínua, normalmente, são especificados os parâmetros de uma função densidade de probabilidade que é representada por um somatório ponderado de M distribuições gaussianas[1,5] ,

$$b_j(o_t) = \sum_{m=1}^{M} c_{jm}\ \mathrm{N}(O_t, \mu_{jm}, U_{jm}) , \qquad (3.7)$$

onde: c_{jm} são os coeficientes de ponderação das gaussianas *M*, μ_{jm} são vetores de médias, e U_{jm} são matrizes covariâncias.

As seguintes condições estocásticas devem ser atendidas por c_{jm} :

$c_{jm} \geq 0$, $\quad i \leq j \leq N$, $\quad 1 \leq m \leq M$, e (3.8)

$$\sum_{m=1}^{M} c_{jm} = 1 , \qquad i \leq j \leq N. \qquad (3.9)$$

- Distribuição do estado inicial, $\pi = \{ \pi_i \}$, onde

$$\pi_j = p\{q_1 = j \}, \qquad i \leq j \leq N. \qquad (3.10)$$

Com os elementos definidos acima, um HMM com distribuição de probabilidade discreta pode ser representado pela notação compacta:

$$\lambda = (A, B, \pi), \qquad (3.11)$$

onde A é a matriz de elementos a_{ij} ,B é o vetor de elementos b_k , e π é o vetor de elementos π_l. E um HMM com distribuição de probabilidade contínua pode ser representado com a notação compacta:

$$\lambda = (A, c_{jm}, \mu_{jm}, U_{jm}, \pi), \qquad (3.12)$$

onde o vetor B é substituído pelos vetores de ponderação c_{jm} , médias μ_{jm} e pela matriz covariância U_{jm}.

Retornando ao exemplo da feira, o número de estados é o número N de barracas existentes. A matriz de probabilidades de transição $A = \{a_{ij}\}$ é uma matriz quadrada de NxN elementos. Os valores destas probabilidades a_{ij} podem ser determinados por um estudo (aprendizado) do comportamento de diferentes fregueses na mesma feira e com a mesma lista de compras. O número de observações ou símbolos do alfabeto é, a princípio, o número de frutas constantes na lista, com M elementos. Porém, se for considerado que cada fruta encontrada possui características únicas e contínuas como tamanho, aparência, preço, qualidade e peso, o número de observações possíveis passa a ser contínuo e incontável. A probabilidade de distribuição $B = \{ b_j(k) \}$ reflete a possibilidade de encontrar na barraca j uma fruta k com boas características de preço ou qualidade ou quantidade, etc. A função de distribuição de probabilidade (*fdp*) associada a B pode não ser uma simples gaussiana para um determinado estado ou barraca, e pode apresentar vários pontos de máximos locais tornando-

se uma função multimodal na prática. Supondo que os picos da fdp da qualidade das frutas fossem para frutas verdes e para frutas 'passadas', pode-se tentar modelar esta fdp por uma combinação de no mínimo duas gaussianas. A combinação de gaussianas é uma forma muito popular de obter aproximações para fdp complexas[3,5,12,13]. Para o modelo em questão, além do atributo qualidade outros atributos também são relevantes e devem participar da composição de B. Assim, para o atributo preço poderá existir também uma combinação de gaussianas que melhor representará o comportamento da fdp associada, o mesmo ocorrendo para os demais atributos pertinentes ao modelo para cada estado. Assim, em vez do vetor B para o caso discreto, ter-se-á os vetores c_{jm} e μ_{jm}, e a matriz U_{jm} englobando todas as gaussianas de todos os atributos de todos os estados para representar o modelo λ. Novamente, o levantamento destas probabilidades exige um estudo (aprendizagem) sobre o que é apresentado pelos feirantes em suas barracas. E, finalmente, as barracas que se localizam nas pontas da feira possuem uma maior probabilidade de serem visitadas inicialmente por um freguês recém chegado. Esta barracas apresentariam valores de π_j maiores que as demais, e um estudo seria necessário para avaliar π para todas as barracas. Ir à feira com uma lista de compras teria um modelo representado pela Equação (3.12).

3.3 – SIMPLIFICAÇÕES NA TEORIA DE HMM

Com o objetivo de facilitar o trato matemático e computacional, algumas suposições são feitas na teoria de HMM:

- **Suposição de Markov**: o estado seguinte na máquina de estados depende somente do estado atual. A aplicação desta suposição gera um modelo chamado *markoviano* de primeira ordem. Pode-se construir modelos de ordem maiores em que o próximo estado dependa do estado atual e de n estados anteriores, porém o trato matemático e computacional cresce em complexidade muito mais do que a qualidade dos resultados.
- **Suposição de estacionaridade**: as probabilidades de transição de um estado para outro não se alteram no tempo.
- **Suposição de independência de saídas**: uma dada observação de saída é estatisticamente independente da observação das saídas anteriores.

Embora estas suposições sejam limitadoras, em geral o desempenho dos HMM não é seriamente prejudicado[5,16].

3.4 – PROBLEMAS BÁSICOS DOS HMM

Após optar por modelar um fenômeno por meio de HMMs, três problemas são de grande interesse:

- **Problema da avaliação**: dado um modelo λ e uma seqüência de observações $O=o_1,o_2,o_3,...,o_T$, qual a probabilidade desta seqüência de observações ter sido gerada pelo modelo λ, $p\{O|\lambda\}$?
- **Problema da decodificação**: dado λ e uma seqüência de observações $O=o_1,o_2,...,o_T$, qual a melhor seqüência dentro do modelo capaz de gerar essas observações ?
- **Problema do aprendizado**: dado λ e uma seqüência de observações $O=o_1,o_2,...,o_T$, como ajustar os parâmetros $\lambda=\{A, B, \pi\}$ de modo a maximizar o valor $p\{O|\lambda\}$?

3.4.1 – Problema da Avaliação

Determinar $p\{O|\lambda\}$ a partir de $O=o_1,o_2,o_3,...,o_T$ e do modelo λ pode ser realizado por meio de processos probabilísticos básicos, mas esse cálculo envolve um número de operações da ordem de N^T, onde T é o número de observações. Mesmo que o número de observações não seja muito grande, o número de operações é elevado. Assim, métodos alternativos foram criados para reduzir a complexidade computacional para encontrar

$$p\{O|\lambda\} = \sum_{i=1}^{N} p\{O, q_t = i \mid \lambda\}. \tag{3.13}$$

Um desses métodos faz uso das variáveis auxiliares $\alpha_t(i)$ (chamada de *forward variable* ou variável progressiva) e $\beta_t(i)$ (chamada de *backward variable* ou variável regressiva). A variável progressiva é definida como a probabilidade da seqüência parcial de observação $o_1, o_2,..., o_T$, quando esta termina no estado i, ou seja,

$$\alpha_T(i) = p\{ o_1, o_2,..., o_T, q_T = i \mid \lambda\}. \tag{3.14}$$

E a variável auxiliar $\beta_t(i)$ é definida como a probabilidade da seqüência parcial de observação $o_{t+1}, o_{t+2}, o_{t+3},..., o_T$ ter ocorrido sendo i o estado atual e λ o modelo,

$$\beta_t(i) = p\{o_{t+1}, o_{t+2}, ..., o_T \mid q_t = i, \lambda\}. \tag{3.15}$$

Desta forma, aplicando recursividade, obtém-se as relações:

$$\alpha_{T+1}(j) = b_j(o_{T+1}) \sum_{i=1}^{N} \alpha_T(i) a_{ij}, \qquad i \le j \le N, \qquad 1 \le t \le T-1, \qquad \text{e} \tag{3.16}$$

$$\beta_t(i) = \sum_{j=1}^{N} \beta_{t+1}(j) a_{ij} b_j(o_{t+1}), \qquad 1 \le i \le N, \qquad 1 \le t \le T-1, \tag{3.17}$$

sendo $\alpha_1(j) = \pi_j b_j(o_1)$, $1 \le j \le N$, e $\beta_T(i) = 1$, $1 \le i \le N$. Disto resulta que a probabilidade procurada pode ser obtida por:

$$p\{O|\lambda\} = \sum_{i=1}^{N} p\{O, q_t = i \mid \lambda\} = \sum_{i=1}^{N} \alpha_T(i) = \sum_{i=1}^{N} \beta_1(i) = \sum_{i=1}^{N} \alpha_t(i)\beta_t(i) \qquad (3.18)$$

A complexidade do cálculo reduz-se de N^T para N^2T operações.

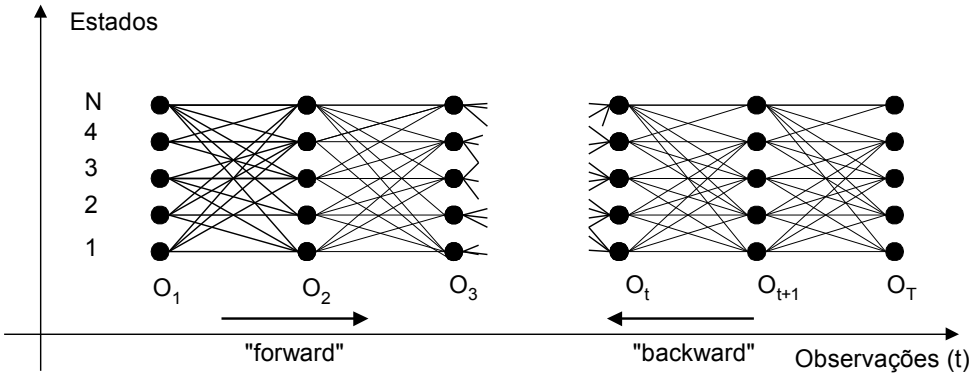

FIGURA 3.2 : Implementação do cálculo das variáveis "Forward" e "Backward".

3.4.2 – Problema da Decodificação

Dado um modelo λ e uma seqüência de observações $O = o_1\ o_2, ..., o_T$, deseja-se saber qual a seqüência de estados com maior probabilidade de ter gerado O.

Uma forma de solução seria procurar, na seqüência, qual o estado mais provável como gerador de determinada observação e, depois, encadear todos os estados encontrados. Um problema nessa abordagem é que a solução pode conter seqüências sem significado para o modelo.

Existem vários critérios de otimização que evitam esse problema. Um dos mais usados é o que encontra a melhor seqüência simples de estados, ou seja, maximiza $P(Q/O,\lambda)$, onde Q representa uma seqüência de estados como $q_1, q_2,..., q_t$. Para encontrar a seqüência Q faz-se uso de um algoritmo de programação dinâmica chamado de algoritmo de Viterbi[1,3,5,12,16]. Nele é definida a variável auxiliar $\delta_t(i)$ como se segue:

$$\delta_t(i) = \max_{q_1, q_2, ..., q_{t-1}} p\{q_1, q_2,..., q_{t-1}, q_t = i, o_1, o_2,..., o_{t-1}| \lambda\}, \qquad (3.19)$$

a qual fornece a maior probabilidade que uma dada seqüência parcial de observação ser gerada por uma seqüência de estados até o estado t, quando o estado atual é i.

Fazendo $\delta_1(j) = \pi_j b_j(o_1)$, $1, \leq j \leq N$ e usando a recursividade, chega-se à relação:

$$\delta_{t+1}(j) = b_j(o_{t+1}) \left[\max_{1 \leq i \leq N} \delta_t(i) a_{ij}\right], \quad 1 \leq i \leq N \qquad 1 \leq t \leq T\text{-}1 \qquad (3.20)$$

É necessário fazer um acompanhamento do argumento que é maximizado para cada instante t para decodificar a seqüência de estados mais provável de gerar O.

O processo possui uma sistemática semelhante ao utilizado em algoritmos de alinhamento temporal dinâmico[1,3,12] (Dinamic Time Warp - DTW).

3.4.3 – Problema do Aprendizado

Geralmente, o problema do aprendizado é como ajustar os parâmetros do HMM do modelo λ para que, dado um grupo de observações chamadas de observações de treinamento, o modelo passe a representar estas observações da melhor forma para uma determinada aplicação. Não existe solução ótima para uma quantidade finita de observações de treinamento. O que se faz é tentar maximizar localmente a função[3] $P(O/\lambda)$ para um dado modelo λ. Como critério de maximização mais usado cita-se o da máxima verossimilhança (Maximum Likelihood – ML). Entre os algoritmos mais utilizados atualmente estão o Algoritmo de Reestimação de Baum-Welch[3,12] (baseado no processo *Forward & Backward*), o Procedimento de Viterbi (baseado no algoritmo de Viterbi) e o *Segmental K-means*[12] .

O algoritmo de Baum-Welch faz uso das variáveis auxiliares α e β (progressiva e regressiva) para compor uma terceira variável $\gamma_t(i)$, chamada de variável de probabilidade a posteriori, correspondente a probabilidade de estar no estado i no tempo t dada a seqüência de observações O, representada pela relação:

$$\gamma_t(i) = P(q_t = i \mid O,\lambda). \tag{3.21}$$

De posse das três variáveis auxiliares e dos elementos que compõem o modelo inicial a ser ajustado, os parâmetros do novo modelo discreto λ^* é dado por:

$$\pi_j^* = \text{nº esperado de vezes do estado } q_i \text{ no tempo t} \tag{3.22}$$

$$a_{ij}^* = \frac{\text{nº esperado de transições do estado i para o estado j}}{\text{nº esperado de transições do estado i}} \tag{3.23}$$

$$b_j(k)^* = \frac{\text{nº esperado de vezes que } o_k \text{ é observado em } q_j}{\text{nº esperado de transições pelo estado j}} \tag{3.24}$$

Para o caso de λ ser um modelo contínuo, o novo modelo terá c_{jm}, μ_{jm}, e U_{jm} ajustados por:

$$c_{jm}^* = \frac{\text{nº esperado de ocorrer a Gaussiana m no estado j}}{\text{nº esperado de ocorrências do estado } q_j} \tag{3.25}$$

$$\mu_{jm}^* = \frac{\text{nº esp. de ocorrer a Gaussiana m em } q_j \text{ ponderada por } o_t}{\text{nº esperado de ocorrer } q_j \text{ e na mistura m}} \tag{3.26}$$

$$U_{jm}^{*} = \frac{\text{nº esperado de ocorrência da Gaussiana m em } q_j \text{ ponderado pela matriz covariância}}{\text{nº esperado de estar no estado } q_j \text{ e na mistura m}} \quad (3.27)$$

Substitui-se $\lambda_{inicial}$ por λ^{*} e repete-se as reestimações até que não ocorra melhorias significativas em $P(O|\lambda^{*})$.

No procedimento de Viterbi, em vez de valores esperados, são usadas as somas das transições ocorridas e observações encontradas ao longo da melhor seqüência de estados obtida para as observações fornecidas.

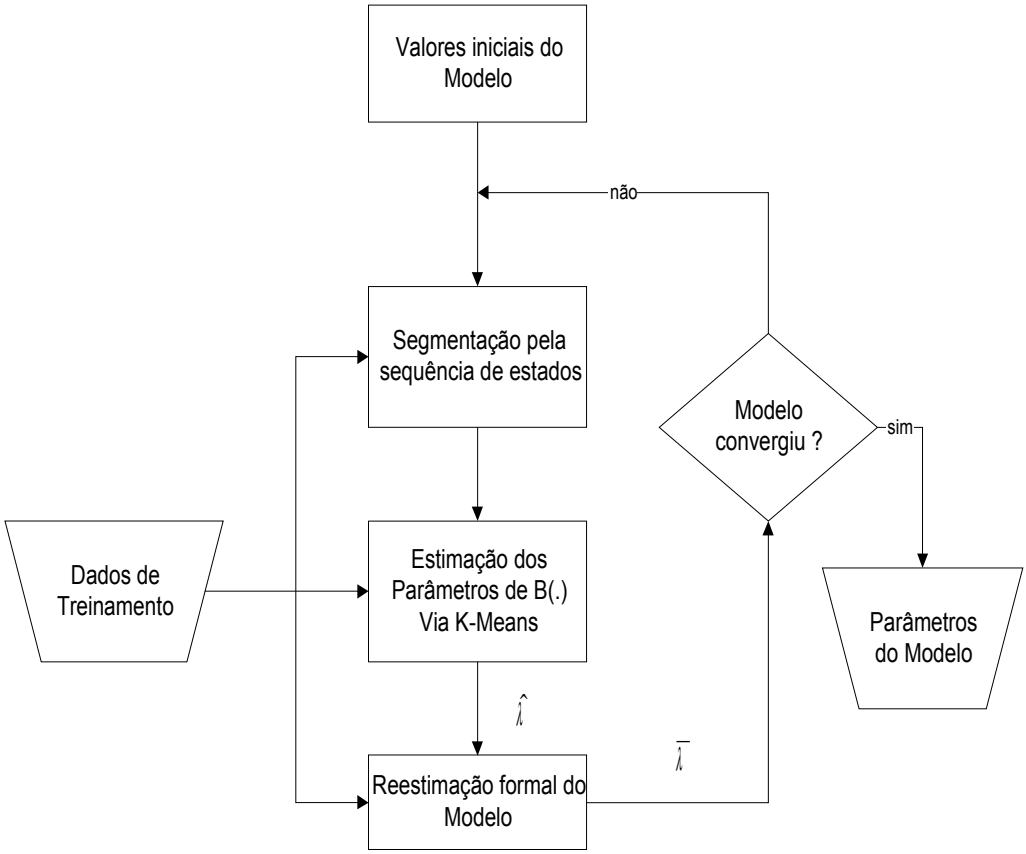

FIGURA 3.3 Algoritmo *Segmental K-means*

Os parâmetros a_{ij}^{*} são obtidos pela contagem do número de transições do estado i para o estado j, dividido pelo número de transições feitas a partir de q_i. As médias, covariâncias e coeficientes de misturas são obtidos para cada estado após o agrupamento dos vetores de observação em M grupos pelo algoritmo *K-means* modificado[5,17]. A média é realmente a média de todas as observações associadas a uma determinada Gaussiana, o mesmo ocorrendo para a covariância. O coeficiente de mistura é dado pelo número de observações classificadas no grupo dividido pelo número total de observações classificadas no estado.

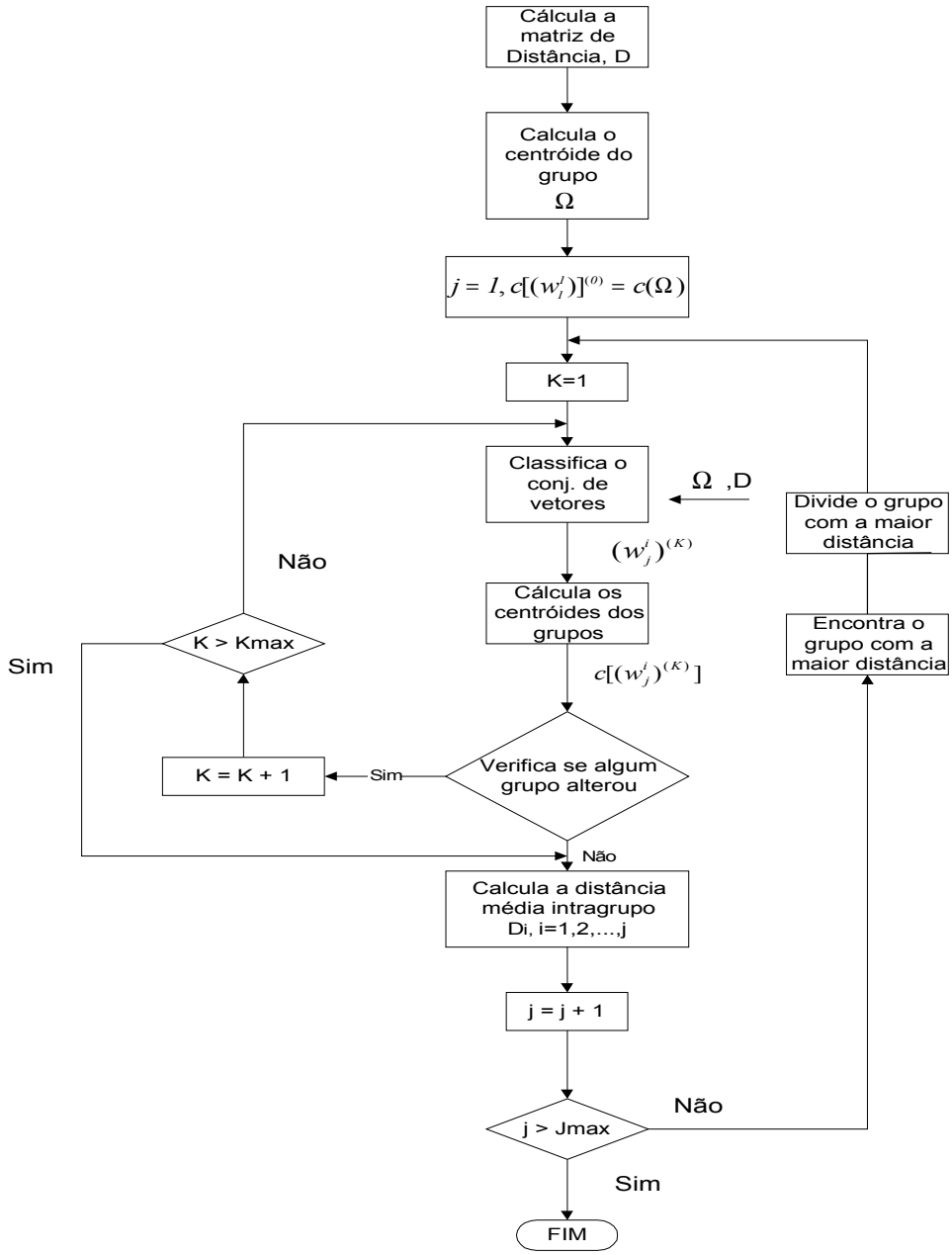

FIGURA 3.4 : Fluxograma do Algoritmo "K-means" Modificado (MKM)

O algoritmo *Segmental K-means* surgiu como tentativa para solucionar um problema de sensibilidade aos valores iniciais do modelo λ observado nos dois algoritmos citados anteriormente. Neste processo, inicialmente divide-se as observações pelos estados (agrupamento) de forma seqüencial, aplica-se o procedimento de Viterbi para a obtenção de um modelo λ^*, o qual é usado no algoritmo de Baum-Welch para uma nova reestimação de todos os parâmetros. A diferença ou a verossimilhança dos modelos iniciais e reestimados são comparadas. Caso as diferenças estejam dentro de uma tolerância limite, então o processo

atingiu uma convergência e finaliza. Caso contrário, o novo modelo é reintroduzido no algoritmo até que ocorra a convergência.

Uma descrição mais completa das fórmulas para treinamento de um HMM contínuo com o método *Segmental K-means* com mais de uma elocução pode ser encontrada na literatura de referência[3,5,12].

3.5 – APLICAÇÃO DE HMM EM RECONHECIMENTO DE VOZ[1,3,5,7,11,12,16]

Os sinais acústicos da fala podem ser convertidos em uma seqüência de números que represente a variação de amplitude no decorrer do tempo. A seqüência pode ser subdividida em quadros ou janelas que se superpõem, abrangendo amostras vizinhas. E destas janelas pode-se extrair atributos da voz que sejam mais representativos do que puramente uma seqüência de amplitudes. O encadeamento de variações de atributos no tempo pode ser modelado por uma máquina de estados finita.

Na modelagem de voz por HMM, os atributos servem para caracterizar um dado fenômeno acústico como, por exemplo, o início de um fonema. Em geral a duração da janela (unidade básica de tempo para fim de processamento de voz) fica em torno de 20 milisegundos, de modo que determinado evento acústico ocorre em um período de algumas janelas. A partir de uma avaliação estatística do comportamento dos atributos pertencentes às janelas de um dado fenômeno acústico (etapa de treinamento) é possível criar um conjunto de valores numéricos - médias, covariâncias, coeficientes de ponderação e de transição para um HMM contínuo[1,11,12] – que passarão a representar aquele fenômeno, caracterizando assim um estado do modelo de Markov. Assim, uma dada elocução pode passar a ser representada por uma seqüência de grupos de valores estatísticos, os chamados estados. É comum atribuir-se um estado a representação do início, um segundo para representar o meio e um terceiro para o fim de cada fonema[18]. Também é comum a adoção de uma máquina de estados onde as transições só possam ocorrer da esquerda para a direita, isto é, de um estado só podem ocorrer transições para o próprio estado, para o seguinte a direita ou para o segundo a direita. Esta configuração de máquina de estados é chamada de modelo de *Bakis*[3] e a Figura 3.5 segue este modelo.

Em uma etapa de decodificação, verifica-se se uma nova elocução possui em seus atributos características estatísticas semelhantes à seqüência de estados treinados. Obtém-se então um valor de verossimilhança entre elocução e modelo. Comparando-se valores de verossimilhança entre diferentes modelos e elocuções é possível determinar o par que melhor se adapta, e realizar o reconhecimento de uma palavra ou locutor.

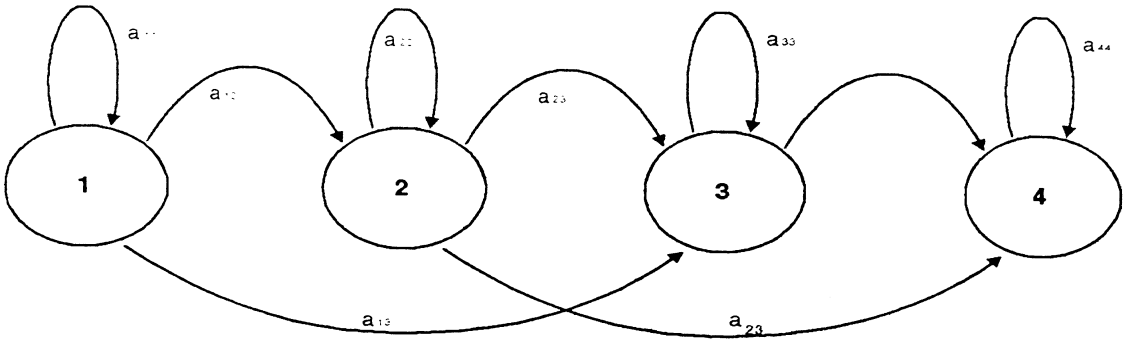

FIGURA 3.5: Máquina de estados segundo o modelo de Bakis

3.6 – RECONHECIMENTO DE PALAVRAS ISOLADAS

Um método simplificado para reconhecer palavras pode consistir nas seguintes etapas: determinar o vocabulário limitado; determinar atributos da voz a serem extraídos; realizar gravações de várias repetições de cada palavra isolada para montar o banco de dados para o levantamento estatístico de modelos; determinar os pontos terminais (*end-points*) de todas as elocuções e extrair seus atributos; determinar apropriadamente, com uso da fonética[19,25], quantos estados serão usados para representar cada palavra; realizar a etapa de aprendizado com *HMM;* obter novas elocuções dos vocábulos e realizar a etapa de avaliação dos modelos do *HMM*, determinado qual modelo fornece a maior verossimilhança e associar o modelo à palavra reconhecida.

Na literatura[3,7,12,17,20] é possível encontrar várias considerações adicionais sobre as etapas citadas.

3.7 – RECONHECIMENTO DE PALAVRAS CONECTADAS

Reconhecer palavras conectadas é mais complexo que reconhecer palavras isoladas.

Em palavras conectadas, sem pausas entre si, existe a co-articulação[11,19] do final da palavra predecessora sobre o início da palavra seguinte e vice-versa. Além disso, modelos de palavras que só considerem o silêncio como limites iniciais e finais não apresentam bons resultados em reconhecimentos de sentenças encadeadas com vários verbetes[21,22,23].

3.7.1. – Modelos Co-articulados

Para criar modelos de HMM de palavras conectadas é necessário extrair de frases de treinamento o segmento da elocução que contém a palavra que se deseje modelar. Com repetições de determinada palavra em vários contextos extraída de diferentes frases de treinamento é possível criar um modelo de HMM que englobe em sua representação as várias influências das co-articulações possíveis, tanto iniciais como finais, para essa palavra.

Existem várias técnicas para realizar a segmentação automática de frases (seqüência de palavras co-articuladas) em palavras isoladas que contenham os efeitos das pronúncias das palavras vizinhas[11,21]. Um método que apresenta bons resultados[11] é baseado no algoritmo de Viterbi. Sabendo-se quais as palavras que compõem uma frase de teste, pode-se montar um modelo de HMM complexo[11] composto, inicialmente, dos modelos de HMM de palavras isoladas. Assim, no caso de uma frase, contendo três palavras representadas por seis estados cada uma, a frase possui um modelo composto por dezoito estados arranjados na ordem dos vocábulos. A matriz *A* correspondente possui 18x18 elementos, por exemplo. Com o modelo complexo realiza-se a etapa de decodificação da frase de teste e, como resultado do algoritmo de Viterbi, obtém-se a relação das janelas da elocução que correspondem a cada estado do modelo complexo. Sabendo-se os estado associados a cada palavra, pode-se relacionar quais as janelas que pertencem a cada palavra, e finalizar a segmentação.

Com as palavras co-articuladas segmentadas, realiza-se novo processo de aprendizagem e obtém-se modelos atualizados. Esses modelos, por sua vez, são usados para re-segmentações. A cada segmentação, verifica-se o valor do somatório da verossimilhança de cada a palavra segmentada avaliada por seu modelo atualizado correspondente (processo de avaliação de HMM). Quando a variação obtida de uma segmentação para outra atingir uma convergência, considera-se o processo de treinamento terminado.

3.7.2. – Reconhecimento de Frases

Com os modelos de palavras influenciadas pelas co-articulações, pode-se iniciar a etapa de reconhecimento de frases com seqüências desconhecidas, formadas com palavras previamente modeladas. Existem várias formas de abordar o reconhecimento de palavras conectadas[3,12,21]. Dentre elas há o algoritmo *Level-Building (LB)*. Nesse algoritmo, o número máximo de vocábulos de uma sentença a ser reconhecida deve ser previamente fixado. Pode-se adotar a existência de tantos *níveis* quanto o número de vocábulos possíveis na frase. Assim, uma etapa de reconhecimento onde se espera encontrar dez palavras terá um algoritmo *LB* com dez níveis de processamento.

Para o primeiro nível do método, o algoritmo *LB* realiza um processo de avaliação de HMM ao longo de todas as janelas correspondentes à elocução. Partindo da primeira janela no tempo, com verossimilhança zero, o algoritmo vai até uma janela *i* e, dentro desse intervalo, avalia qual modelo de palavra permite obter a maior verossimilhança e armazena o código desse modelo, ou palavra, correspondente a essa janela e a sua verossimilhança. Ao chegar na última janela da elocução, obtém-se uma tabela que indica para cada janela, qual a palavra que apresenta a maior verossimilhança (palavra essa que seria a palavra reconhecida, caso a elocução terminasse em dada janela *i* e contivesse apenas uma palavra).

Para o segundo nível, o algoritmo parte da tabela obtida no primeiro nível. Em vez de fixar o início da varredura de janelas na primeira janela temporal, inicia-se a varredura para cada janela *i* até uma janela *j*, sendo $j > i$, e tenta-se *encaixar* ou avaliar os modelos possíveis de ocorrerem como segunda palavra na frases dentro destas janelas do intervalo de *i* a *j*. Nesta etapa obtém-se nova tabela onde associa-se à janela *j* os seguintes valores: a verossimilhança total (obtida com a soma da verossimilhança em *i* oriunda do nível anterior com a verossimilhança obtida com o modelo que melhor se ajustou no trecho de *i* a *j*); o valor de *i* correspondente ao início que possibilitou atingir *j* (através do modelo que forneceu a maior verossiminhança); e o código da segunda palavra identificada da sentença, caso a elocução terminasse em *j*.

Entre a transição dos níveis há uma etapa intermediária que avalia se a verossimilhança total atingida até a janela seguinte a *i* é maior caso $i+1$ seja considerada como ainda pertencente ao final da palavra (ou modelo) anterior ou como início do modelo seguinte, realizando um ajuste fino do ponto de término da palavra anterior e início da seguinte.

Nos níveis seguintes o processo se repete, iniciando o processamento do nível atual com base na tabela do nível predecessor. Em cada nível, o algoritmo procura, seguindo uma gramática, combinar os modelos da forma que melhor se ajustem (baseado na verossimilhança) em um comprimento parcial da elocução, e com o número de palavras igual ao número do nível atual. Ao atingir o nível máximo permitido, consulta-se na última tabela gerada os valores correspondentes à última janela da elocução. Daí obtém-se: a última palavra da frase e em que janela *k* seu modelo começou. Com a janela *k* consulta-se a penúltima tabela (obtida no nível anterior) para saber qual a penúltima palavra reconhecida e onde ocorreu o início de seu modelo. E assim sucessivamente, percorrendo as tabelas de trás para frente até chegar na indicação da primeira palavra da seqüência que gerou a maior verossimilhança para a sentença. Finaliza-se assim o reconhecimento da sentença, em um processo semelhante ao *DTW*[12].

Existem várias considerações que podem ser feitas visando reduzir o esforço computacional do método *LB*[3,11,12,21]. Dentre elas pode-se citar o uso do paralelogramo de Itakura[12] para restrição do caminho global a ser percorrido (de modo análogo ao que ocorre em alinhamento temporal dinâmico – *DTW*[12]). O uso de uma gramática que estabeleça uma ordem na seqüência das palavras pode facilitar a busca do *LB* pela sentença correta[3,11,12,22]. Em dado nível (posição dentro da frase) a gramática restringiria quais as palavras do vocabulário treinado podem ocorrer, evitando de pesquisar todas as palavras modeladas.

Uma apresentação mais formal do algoritmo *Level-Building* e suas equações pode ser encontrada na literatura de referência[3,12,21].

CAPÍTULO 4

GRAMÁTICA REGULAR ADOTADA

4.1. INTRODUÇÃO

O reconhecimento de palavras conectadas, mesmo para dicionários pequenos, exige alta quantidade de memória, para armazenar todos os modelos correspondentes, e uma velocidade elevada de processamento, para realizar a decodificação próxima do tempo real[2]. Uma vez identificada a tarefa específica em que uma interface de reconhecimento de voz será utilizada, pode-se fazer uso das características inerentes à tarefa para otimizar o vocabulário, a memória e a velocidade necessários para o reconhecimento.

4.2. VOCABULÁRIO

A tarefa escolhida foi a de realizar controle de deslocamentos de objetos genéricos no espaço e acionamentos eletro-mecânicos. Para tal tarefa, o vocabulário deve conter os dígitos que permitam a designação de objetos e a determinação de quantidades de deslocamento. O vocabulário deve conter também palavras que ajudem a designação de objetos, mas que estes objetos não sejam específicos, de modo a não haver perda de generalidade. Devem ser incluídas também as direções, unidades de deslocamento e ações possíveis . Sendo assim, adotou-se a lista de palavras constante na Tabela 4.1 como vocabulário de trabalho.

Para efeito de simplificação de modelos, o par ‘para a’ é considerado como palavra única; por isso, nas frases de exemplos, o par ‘para a’ é escrito como ‘para_a’ para lembrar esta simplificação.

Para não ampliar demais o número de unidades do vocabulário, não foram inseridas palavras tais como: dezenas e centenas de números. Embora essa inserção fosse uma facilidade para o locutor, acarretaria um aumento de modelos a serem tratados, modelados e decodificados. Assim, em vez de dizer ‘trezentos e sessenta graus’, para facilidade de implementação, o usuário deverá pronunciar ‘três seis (meia) zero graus’. Esta transcrição é comum na troca de informação de valores, de uma pessoa para outra.

TABELA 4.1: Vocabulário adotado

No. de ordem	Palavra	No. de ordem	Palavra	No. de ordem	Palavra
1	Zero	16	Circuito	31	Passo
2	Um	17	Dispositivo	32	Grau
3	Uno	18	Ande	33	Metros
4	Dois	19	Mova	34	Centímetros
5	Três	20	Gire	35	Passos
6	Quatro	21	Vire	36	Graus
7	Cinco	22	Rode	37	Para a
8	Seis	23	Ligue	38	À
9	Meia	24	Desligue	39	Pra
10	Sete	25	Pare	40	Esquerda
11	Oito	26	Abra	41	Direita
12	Nove	27	Feche	42	Cima
13	Motor	28	Volte	43	Baixo
14	Unidade	29	Metro	44	Frente
15	Sistema	30	Centímetro	45	Trás

4.3. GRAMÁTICA

A gramática é o conjunto de regras que define como as palavras de um dicionário (caso de reconhecimento de palavras conectadas) podem ser encadeadas dentro de uma linguagem, como o português falado no Brasil, por exemplo[24]. Para a tarefa específica de reconhecimento de comandos conectados do presente trabalho, pode-se definir uma *gramática regular*[3] específica para reger como as palavras do vocabulário restrito podem ser encadeadas de modo a gerar sentenças, com significado tanto para a máquina quanto para a pessoa que as gerou. Existem várias formas de abordagem de modelos de linguagem[1,3], porém, a gramática regular tem a vantagem de ser facilmente representada por uma máquina de estados finitos e poder ser modelada por um *HMM*.[13,11]

Para o presente trabalho, fixou-se algumas regras para compor a gramática regular que rege a linguagem do sistema de reconhecimento. Essas regras foram norteadas tanto pela gramática portuguesa quanto pelas limitações e simplificações do algoritmo de reconhecimento.

4.3.1. Sentenças Longas – 10 Palavras

A gramática regular básica estipulada para executar a tarefa de deslocamentos no espaço permite a composição de sentenças com dez palavras divididas em cinco grupos na seguinte seqüência: *sujeito*, *complemento do sujeito*, *verbo*, *complemento de direção* e *complemento de quantidade de deslocamento*.

O sujeito é expresso por uma das palavras de 13 a 17 da Tabela 4.1.

O complemento do sujeito é composto por dois dígitos (palavras de 1 a 12 da Tabela 4.1), cobrindo as combinações de ‘01’ a ‘16’ para permitir distinguir entre 2^4 (4 *bits* em circuito físico) elementos diferentes de uma mesma categoria de sujeito.

O verbo é expresso por uma das palavras de 18 a 28 da Tabela 4.1. Dentre os verbos possíveis, os correspondentes as palavras de 18 a 22 admitem os complementos de direção e de quantidade de deslocamento, sendo esse usados para compor as sentenças longas.

O complemento de direção é composto por duas palavras: uma preposição representada pelas palavras de 37 a 39, e por uma direção dada pelas palavras de 40 a 45 da Tabela 4.1. Todas as três preposições possíveis terminam com a vogal ‘a’, e com a co-articulação com as palavras de direção, permitem ao locutor *compor* as palavras ‘abaixo’ e ‘atrás’ ao longo da sentença, suprindo a deficiência conseqüente do vocabulário adotado.

O complemento da quantidade de deslocamento é composto por três dígitos (palavras de 1 a 12) e uma unidade de deslocamento (palavras de 29 a 36 da Tabela 4.1). Os três dígitos podem formar os números de ‘001’ a ‘399’ e atender a uma rotação de 360 graus. Translações com valores maiores que 399 centímetros podem ser feitas com o uso do múltiplo ‘metro’. A palavra ‘passo’ pode ser atribuída a qualquer quantidade, de rotação ou de translação.

Para facilitar o reconhecimento em etapa posterior, na montagem das frases de treinamento, existe uma pausa (vírgula) entre o complemento do sujeito e o verbo, e outra pausa (vírgula) entre o complemento de direção e o complemento de quantidade de deslocamento. Assim, as seguintes frases são exemplos de sentenças longas geradas pela gramática adotada:

- Motor zero um, gire para_a direita, dois cinco quatro graus.
- Sistema uno dois, mova a cima, zero zero um metro.
- Unidade zero três, ande para_a baixo, sete quatro dois centímetros.

4.3.2. Sentenças de 7 Palavras

As sentenças de sete palavras são montadas na seguinte ordem: verbo, complemento de direção e complemento de quantidade de deslocamento. Elas seguem a mesma estrutura das sentenças longas, porém iniciando na posição da primeira vírgula. Essas sentenças visam permitir instruções mais curtas, em que o sujeito e o complemento do sujeito são subentendidos como sendo os mesmos da última instrução dada ao sistema. Como exemplo de sentenças válidas pode-se usar os mesmos exemplos do item anterior, ignorando as palavras antes da primeira vírgula. Caso fossem dadas as seguintes instruções em duas sentenças como se segue:

- Unidade zero meia, vire para a direita, zero nove zero graus;
- Ande para_a frente, zero zero um metro;

a segunda sentença tem como sujeito e seu complemento a seqüência ‘unidade zero meia’ da primeira sentença.

4.3.3. Sentenças de 6 Palavras

As sentenças de seis palavras são montadas na seguinte ordem: sujeito, complemento do sujeito, verbo e complemento de direção. Elas seguem a mesma estrutura das sentenças longas, porém terminando na posição da segunda vírgula. Estas sentenças visam permitir instruções mais curtas para servo-mecanismos que possuam uma quantidade de deslocamento padronizada. Como exemplo de sentenças válidas, pode-se usar os mesmos exemplos do item 4.3.1., ignorando as palavras após a segunda vírgula. Pode-se fixar um deslocamento de translação padrão de 10 centímetros e um deslocamento de rotação padrão de 45 graus, por exemplo.

4.3.4.Sentenças de 4 Palavras

As sentenças de quatro palavras são montadas na seguinte ordem: sujeito, complemento do sujeito e verbo. Elas seguem a mesma estrutura das sentenças longas porém terminando na posição do verbo. A presença da vírgula entre o complemento do sujeito e o verbo é mantida para facilidade de montagem das frases para treinamento. Estas sentenças visam permitir instruções que não necessitem de complemento para o verbo. As palavras de 23 a 28 da Tabela 4.1 já eliminam naturalmente a necessidade de complemento, e as palavras de 18 a 22 podem ser usadas desde que um deslocamento padrão seja adotado. Como exemplo de sentenças válidas têm-se:

- Sistema zero uno, desligue.
- Unidade uno uno, volte.
- Dispositivo uno meia, pare.

4.3.5. Sentenças de 3 Palavras

As sentenças de três palavras são montadas na seguinte ordem: verbo e complemento de direção. O sujeito e seu complemento são os mesmos de uma instrução anterior. O complemento de quantidade de deslocamento é subentendido como sendo um valor padrão. Como exemplo de sentenças pode-se usar os mesmos do item 4.3.1., porém considerando somente as seqüências de palavras entre as vírgulas.

4.3.6. Sentenças de 1 Palavra – Palavra Isolada

As sentenças de apenas uma palavra contém apenas um verbo (palavras de 18 a 28 da Tabela 4.1). O sujeito é entendido como sendo o mesmo de alguma instrução imediatamente anterior. Caso o verbo necessite de algum complemento, o sistema internamente assume um complemento padrão.

Como exemplo, no caso de uma cadeira de rodas, dada a instrução 'volte', um pós-processador sintático de alto nível deveria buscar o sujeito da instrução anterior (se foi toda a cadeira ou se foi apenas uma das rodas) e fazer executar o retorno do movimento utilizando o complemento de direção contrário e o mesmo complemento de quantidade de deslocamento da instrução anterior, trazendo a cadeira para a mesma posição do início da instrução anterior ao 'volte'.

4.4. MÁQUINA DE ESTADOS

As regras apresentadas visam facilitar o processo de reconhecimento sem que isso prejudique a elaboração de elocuções pelo próprio usuário do sistema proposto.

Fica claro, pelo exposto nos itens anteriores, que para sentenças com menos dez palavras é necessário o uso de um pós-processamento sintático complexo após o reconhecimento da seqüência de palavras contidas em uma elocução. Esse pós-processamento complexo visa completar a frase com complementos padrões e/ou buscar o sujeito e/ou complementos em frases anteriores. A etapa de pós-processamento complexo é muito específica para o tipo de servo-mecanismo a ser controlado (sujeito), não sendo, assim, alvo deste trabalho.

O conjunto de regras da gramática regular adotada permite a sua representação pela seguinte máquina de estados finitos da Figura 4.1, onde podem ocorres as seguintes palavras nos caminhos *C* entre estados:

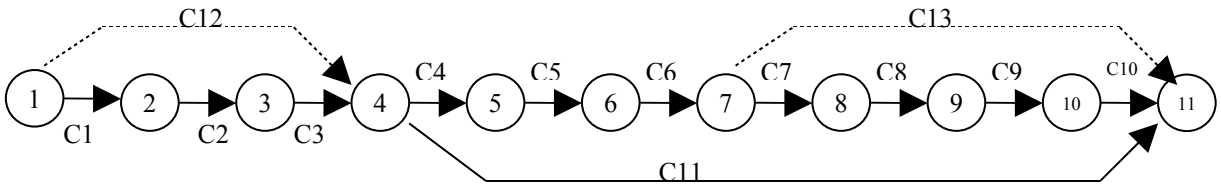

FIGURA 4.1 – Máquina de estados da gramática regular adotada

C1 = {motor, unidade, sistema, circuito, dispositivo};

C2 = {zero, um, uno};

C3, C8 e C9 = {zero, um, uno, dois, três, quatro, cinco, seis, meia, sete, oito, nove};

C4 = {ande, mova, gire, vire, rode};

C5 = {para_a, à, pra};

C6 = {direita, esquerda, cima, baixo, frente, trás};

C7 = {zero, um, uno, dois, três};

C10 = {metro, metros, centímetro, centímetros, passo, passos, grau, graus};

C11 = {ligue, desligue, pare, ande, abra, feche, volte, mova, gire, vire, rode}.

C12 e C13 = transição nula (sem ocorrência de palavras).

No que tange ao fato de um estado não poder retornar para si próprio, a máquina de estado representativa da gramática regular pode ser modelada por um *HMM* com uma configuração mais simples que a usada no modelo de Bakis[3]. Dentro da gramática proposta, as transições entre estados têm probabilidades iguais para todas as transições. E a probabilidade de ocorrência é igualmente distribuída por todas as palavras possíveis em um dado ramo ou caminho.

Um problema do modelo com *HMM* ocorre na determinação da unidade de deslocamento para a sentença longa. Como a ocorrência do estado seguinte só depende do estado atual, ao atingir o estado 10 o processo não traz informação sobre qual verbo foi usado na transição do estado 4 para o estado 5. Por isso, pode ocorrer que, por exemplo, 'gire' e 'metros' ocorram em uma mesma frase decodificada erroneamente, trazendo um conflito entre rotação e translação. De modo semelhante, no estado 10 o sistema não tem a informação de todos os três dígitos pronunciados anteriormente. Então, podem ocorrer erros entre o plural e o singular das unidades de deslocamento. Para minimizar esse problema pode-se fazer uso de um pós-processador sintático[11] simplificado que examine os dígitos anteriores e realize a substituição de singular por plural, ou vice-versa, caso necessário, ou indique a necessidade de repetição da sentença por parte do locutor em caso de conflito entre translação e rotação.

4.5. FRASES PARA O TREINAMENTO

Após a determinação do vocabulário e da gramática regular a ser seguida, fez-se um estudo de quais palavras podem co-articular em uma dada sentença válida. Há palavras que não podem ocorrer em seqüência como: 'motor para_a' e há palavras que só ocorrem com uma palavra predecessora tal como: 'um metro'.

Uma abordagem mais impulsiva poderia levar a realizar o treinamento com todas as frases possíveis de serem geradas pela gramática. Com isso poderia se obter todas as co-articulações possíveis de ocorrência. Entretanto, o número de possibilidades dessa gramática é elevado e praticamente impossível de realizar a gravação de todas as frases por um locutor cooperativo. Utilizando cálculos de matriz de transitividade fechada[23], chega-se à conclusão de que o número de combinações possíveis para as frases com dez palavras (todos os complementos) é algo em torno de 24 milhões de possibilidades.

Uma das soluções propostas[23] para reduzir o número de frases a serem pronunciadas por locutores para montar o banco de dados para treinamento de palavras co-articuladas é fazer uso da ocorrência de *bigramas*. Um bigrama é um par de palavras. A pronúncia corrida faz com que a primeira palavra interfira (co-articule) com a segunda e vice-versa. Assim, o que passa a ser essencial no conjunto de frases de treinamento é a presença mínima de um determinado número de bigramas (ou ocorrência de determinado par de palavras). É possível reduzir significativamente o número de frases de treinamento se elas forem geradas criteriosamente de forma que cada sentença contenha o maior número de novos bigramas.

No presente trabalho, adotou-se as seguintes considerações para a determinação das frases de treinamento:

- Ocorrência de, no mínimo, dois exemplares de cada bigrama em todo o conjunto de frases de treinamento;
- Sendo o locutor cooperativo e observando as pausas (vírgulas) durante a elocução, as palavras separadas pelas vírgulas possuem fraca co-articulação. Forçando essa fraca co-articulação, pôde-se omitir os bigramas correspondentes à contagem obrigatória, reduzindo em muito o número final de frases necessárias;
- Com base nas pausas, os três segmentos básicos da frase, limitados pelas vírgulas, puderam ser gravados também de forma isolada. Isto evitou a repetição seguida do sujeito quando o que se desejava obter eram bigramas do complemento de deslocamento, reduzindo o tempo total de gravação;
- As palavras que possuíam bigramas com co-articulações semelhantes como por exemplo ‘para_a’ em ‘gire-para_a’ e em ‘vire-para_a’, onde os finais das primeiras palavras são similares, tiveram uma redução no número obrigatório de ocorrências;
- Embora na gramática regular adotada não seja possível a ocorrência das seqüências representativas de números maiores que 399, para a geração das frases de treinamento isso não foi respeitado. As frases foram geradas com a possibilidade de ocorrência de seqüências de ‘000’ até ‘999’. Com isso foi possível obter, em um menor número de frases, uma quantidade maior de novos bigramas.
- Os verbos sem complemento foram considerados como palavras isoladas. A justificativa disso está baseada na fraca co-articulação do início da palavra e no silêncio do final da sentença;

As considerações acima possibilitaram que o número de frases de treinamento fosse de 137, entre frases completas e segmentos, e o tempo de gravação médio fosse de cinqüenta minutos por locutor. Antes desse processo, o número de frases obtidas sem a presença das vírgulas ou pausas passava de 500 o que tornaria a tarefa de gravação muito enfadonha, mesmo para um locutor cooperativo.

TABELA 4.2 : Frases de treinamento

Código	Frase
F001	Motor zero dois, mova a esquerda, dois um zero metros.
F002	Unidade zero três, ande a direita, dois uno zero centímetros.
F003	Sistema zero um, mova a cima, seis sete dois passos.
F004	Dispositivo zero dois, gire a baixo, três uno quatro graus.
F005	Motor zero três, vire a frente, zero seis um passos.
F006	Unidade zero quatro, mova para a esquerda, uno quatro três metros.
F007	Sistema zero um, mova pra esquerda, um sete zero centímetros.
F008	Dispositivo zero dois, rode a esquerda, um um nove graus.
F009	Motor um dois, ande para a direita, uno nove zero metros.
F010	Unidade um três, ande a direita, nove seis zero passos.
F011	Sistema um um, mova pra cima, quatro sete quatro centímetros.
F012	Dispositivo um dois, gire para a baixo, sete nove meia graus.
F013	Motor uno dois, ande pra direita, cinco oito quatro centímetros.
F014	Unidade uno três, mova a cima, três quatro cinco metros.
F015	Sistema uno um, gire a baixo, meia dois zero passos.
F016	Dispositivo uno dois, vire pra frente, nove cinco três passos.

Código	Frase
F017	Circuito zero uno, feche.
F018	Circuito zero uno, ligue.
F019	Circuito um uno, desligue.
F020	Circuito uno uno, abra.
F021	zero zero um metro.
F022	zero zero uno metro.
F023	zero zero um centímetro.
F024	zero zero uno centímetro.
F025	zero zero um passo.
F026	zero zero uno passo.
F027	zero zero um grau.
F028	zero zero uno grau.
F029	mova para a cima.
F030	gire pra baixo.
F031	vire para à frente.
F032	vire à frente.
F033	rode para à trás.
F034	rode pra trás.
F035	três dois dois metros.
F036	um quatro zero centímetros.
F037	quatro dois três metros.
F038	cinco dois quatro metros.
F039	quatro quatro seis metros.
F040	cinco quatro meia metros.
F041	seis quatro sete metros.
F042	meia quatro oito metros.
F043	sete quatro nove metros.
F044	cinco seis dois centímetros.
F045	quatro três três centímetros.
F046	seis cinco cinco centímetros.
F047	meia seis seis centímetros.
F048	meia cinco meia centímetros.
F049	sete cinco sete centímetros.
F050	sete seis oito centímetros.
F051	oito cinco nove centímetros.
F052	seis nove quatro passos.

Código	Frase
F053	um quatro cinco passos.
F054	meia oito seis passos.
F055	nove seis meia passos.
F056	meia meia sete passos.
F057	sete sete oito passos.
F058	sete meia nove passos.
F059	seis três zero graus.
F060	oito oito dois graus.
F061	oito meia três graus.
F062	seis cinco cinco graus.
F063	dois quatro seis graus.
F064	nove oito sete graus.
F065	três quatro oito graus.
F066	cinco oito nove graus.
F067	quatro quatro zero metros.
F068	sete três um metros.
F069	cinco quatro uno metros.
F070	nove nove dois metros.
F071	seis oito três metros.
F072	cinco seis quatro metros.
F073	quatro meia cinco metros.
F074	meia seis seis metros.
F075	sete cinco meia metros.
F076	quatro nove sete metros.
F077	seis meia oito metros.
F078	oito cinco nove metros.
F079	nove cinco zero centímetros.
F080	meia quatro um centímetros.
F081	zero cinco uno centímetros.
F082	três dois dois centímetros.
F083	seis nove três centímetros.
F084	cinco um cinco centímetros.

F085	cinco sete seis centímetros.
F086	sete meia meia centímetros.
F087	oito seis sete centímetros.
F088	meia sete oito centímetros.

TABELA 4.2 : Frases de treinamento (continuação)

Código	Frase
F089	oito meia nove centímetros.
F090	um seis uno passos.
F091	oito quatro dois passos.
F092	Meia um três passos.
F093	sete nove quatro passos.
F094	meia uno cinco passos.
F095	sete uno seis passos.
F096	oito nove meia passos.
F097	oito sete sete passos.
F098	nove oito oito passos.
F099	zero nove nove passos.
F100	zero meia zero graus.
F101	nove sete um graus.
F102	zero oito uno graus.
F103	dois cinco dois graus.
F104	nove uno três graus.
F105	sete zero quatro graus.
F106	dois três cinco graus.
F107	seis dois seis graus.
F108	oito um meia graus.
F109	oito zero sete graus.
F110	nove um oito graus.
F111	uno um zero centímetros.
F112	um uno zero passos.
F113	meia dois zero graus.
F114	três três zero metros.
F115	quatro cinco zero passos.
F116	três seis zero graus.
F117	uno meia zero metros.
F118	uno oito zero passos.
F119	dois nove zero graus.
F120	uno sete dois metros.
F121	dois oito dois centímetros.
F122	três nove dois passos.
F123	cinco cinco três centímetros.
F124	quatro seis três passos.
F125	dois meia três graus.
F126	dois sete três metros.
F127	três oito três centímetros.
F128	quatro nove três passos.
F129	dois um zero metros.
F130	uno uno zero centímetros.
F131	um dois zero passos.
F132	zero três zero graus.
F133	dois quatro zero metros.
F134	seis cinco zero centímetros.
F135	cinco seis zero passos.
F136	três meia zero graus.
F137	três sete zero metros.

A Tabela 4.2 apresenta a lista de frases de treinamento. Com esta lista de sentenças, garantiu-se que cada par de palavras com co-articulação entre fonemas semelhantes ocorre pelo menos duas vezes. A palavra

‘zero’ apresentou o maior número de ocorrências. Entre as causas disso estão as várias composições com unidades de deslocamento no singular (palavras de 29 a 32 da Tabela 4.1, frases F021 a F028 da Tabela 4.2).

4.6. FRASES PARA VERIFICAÇÃO

Foi criado um conjunto de frases específico para a verificação do desempenho do sistema. Seguiu-se a gramática regular adotada nos ítens anteriores para criar as novas sentenças de teste com diversos comprimentos, sentenças essas listadas a seguir, precedidas de seus respectivos códigos:

TABELA 4.3 : Frases de teste

F190	Motor zero um, gire a direita, zero quatro cinco graus.
F191	Motor zero dois, gire para a esquerda, uno zero zero passos.
F192	Motor zero três, ande pra frente, zero uno zero centímetros.
F193	Motor zero quatro, mova para a cima, zero zero dois metros.
F194	Sistema zero cinco, ande para a baixo, três zero zero passos.
F195	Sistema zero uno, rode a direita, dois zero quatro graus.
F196	Unidade uno zero, vire para a trás, zero zero cinco passos.
F197	Unidade zero uno, ande para a esquerda, zero zero seis passos.
F198	Unidade zero quatro, ande para a trás.
F199	Sistema uno dois, rode para a direita.
F200	Unidade zero sete, vire para a frente.
F201	Dispositivo zero um, mova pra frente.
F202	Dispositivo zero oito, ande para a esquerda.
F203	Dispositivo uno três, gire pra cima.
F204	Unidade zero nove, mova para a baixo.
F205	Dispositivo zero sete, desligue.
F206	Circuito zero uno, ligue.
F207	Circuito zero dois, desligue.
F208	Dispositivo zero um, abra.
F209	Unidade uno uno, feche.
F210	Sistema zero uno, volte.
F211	Unidade zero meia, pare.
F212	Unidade uno zero, volte.
F213	Vire para a direita.
F214	Gire para a esquerda.
F215	Rode pra cima.
F216	Mova pra baixo.
F217	Ande a direita.
F218	Volte.
F219	Pare.
F220	Abra.
F221	Feche.
F222	Ligue.
F223	Desligue.
F224	Ande.
F225	Ande para a direita, zero zero quatro metros.
F226	Rode pra baixo, zero nove zero passos.
F227	Mova a esquerda, uno zero zero centímetros.
F228	Gire pra cima, zero dois cinco graus.
F229	Vire pra trás, zero zero uno grau.
F230	Mova a frente, zero zero um passo.

4.7 – AVALIAÇÃO DO DESEMPENHO NO RECONHECIMENTO

O desempenho dos reconhecedores de voz contínua normalmente é definido em termos de taxa de acerto de palavras dentro do conjunto de palavras de teste.

Ao ocorrer divergências entre o conteúdo de uma sentença e o resultado de sua identificação, três erros podem ter ocorrido. Caso a sentença original e a reconhecida tenham o mesmo número de palavras, pode ocorrer uma ou mais *substituições* entre palavras. Caso as sentenças sejam de tamanhos diferentes, além da substituição, pode ter ocorrido a *exclusão* ou a *inserção* de uma ou mais palavras.

Assim, uma forma comum de determinar a taxa de acerto é dada por[11]:

$$Ta = 100 * \left(1 - \frac{S + I + E}{Nt} \right) \tag{4.1}$$

onde *Ta* é a taxa de acerto, *S* é o número de substituições ocorrido no conjunto de sentenças de teste, *I* é o número de inserções ocorridas, *E* é o número de exclusões contadas e *Nt* é o número total de palavras contidas nas sentenças de teste.

CAPÍTULO 5

SISTEMA DE RECONHECIMENTO DE PALAVRAS CONECTADAS

5.1 INTRODUÇÃO

Definidos o vocabulário e a gramática regular a ser seguida, passou-se à criação de modelos de palavras isoladas, método de segmentação de frases de treinamento, modelagem de palavras com co-articulação e algoritmo de decodificação das frases de teste.

5.2 MODELOS DE *HMM* PARA PALAVRAS ISOLADAS

5.2.1 Atributos da Voz

Para o presente trabalho, com base na literatura de referência[3,4,5,6,8,11,12], adotou-se o seguinte conjunto de características para representar os sinais de voz: quadros de 10 milisegundos, superposição de 50% entre quadros adjacentes (implicando em janelas de 20 milisegundos), freqüência de amostragem de 11.025 Hz, log-energia de tempo curto com sua primeira e segunda derivadas, 5 primeiros coeficientes PLP-Cepstrum com suas primeiras e segundas derivadas, totalizando 18 atributos para cada janela de sinal de voz .

5.2.2 Número de Estados por Palavra

Uma das primeiras etapas da modelagem de palavras por *HMM* é a determinação de quantos estados serão usados para representar cada palavra. A existência de palavras de vários comprimentos dificulta o uso de um número fixo de estados para todas as palavra. Como exemplo, o número de estados adotados para a palavra ‘a’ (número 38 da Tabela 4.1) foi 3, enquanto para a palavra ‘centímetros’ (número 34 da Tabela 4.1) adotou-se 15 estados.

O número de estados para cada palavra do vocabulário foi fixado com base nos fonemas presentes, usando-se 3 estados por fonema[11,19] da língua portuguesa.

Para algumas palavras foram acrescidos alguns estados para abrangerem diferentes pronúncias da letra ‘r’ e de encontros vocálicos, visando tornar o modelo menos sensível à pronúncias regionalistas dos locutores.

Tomando como base os resultados encontrados na literatura de referência[11], não foram testadas outras distribuições de estados.

Desta forma, o número de estados por palavra do vocabulário foi distribuído conforme apresentado na Tabela 5.1 .

TABELA 5.1: Vocabulário adotado com a distribuiçao de estados

nº	Palavra	estados	nº	Palavra	estados	nº	Palavra	estados
1	Zero	9	16	Circuito	15	31	Passo	6
2	Um	3	17	Dispositivo	15	32	Grau	6
3	Uno	6	18	Ande	6	33	Metros	9
4	Dois	6	19	Mova	6	34	Centímetros	15
5	Três	9	20	Gire	6	35	Passos	6
6	Quatro	9	21	Vire	6	36	Graus	6
7	Cinco	6	22	Rode	6	37	Para a	9
8	Seis	6	23	Ligue	6	38	À	3
9	Meia	9	24	Desligue	9	39	Pra	6
10	Sete	6	25	Pare	9	40	Esquerda	12
11	Oito	9	26	Abra	9	41	Direita	12
12	Nove	6	27	Feche	6	42	Cima	6
13	Motor	9	28	Volte	6	43	Baixo	9
14	Unidade	12	29	Metro	9	44	Frente	9
15	Sistema	12	30	Centímetro	15	45	Trás	6

5.2.3 Parâmetros dos Modelos

Para todos os testes foi adotado o modelo de Bakis[3], 5 gaussianas por estado, 18 atributos da voz (conforme Capítulo 2), 10 interações máximas para o algoritmo *Segmental k-means* e um fator de convergência de 0,1% para o término dos treinamentos dos modelos.

5.2.4 Banco de Elocuções

Foram criados dois bancos de dados contendo gravações para avaliar o desempenho do sistema. Um banco com elocuções de apenas um locutor e outro para vários locutores. A aquisição e pré-processamento do sinal seguem o que foi descrito no Capítulo 2 para todas as gravações.

Para o locutor único, foram realizadas gravações de 30 repetições de cada palavra do vocabulário, de forma isolada. Esse processo demorou 8 horas ao todo.

Para o treinamento e testes no modo independente do locutor, 12 pessoas do sexo masculino pronunciaram 4 vezes cada palavra isolada.

5.2.5 Equações de Treinamento de *HMM* Contínuo

Para a criação dos modelos foi usada a mesma rotina empregada na literatura de referência[1,7,11], a qual faz uso de *HMM* contínuo com as seguintes equações[1,11] de reestimação do modelo $\lambda = (A, c_{jm}, \mu_{jm}, U_{jm}, \pi)$:

- elementos da matriz de transição[1,11]:

$$\bar{a}_{ij} = \frac{\sum_{k=1}^{K}\sum_{t=1}^{T_k-1}\hat{\alpha}_t^{(k)}(i).a_{ij}.b_j(O_{t+1}^{(k)}).\hat{\beta}_{t+1}^{(k)}(j)}{\sum_{k=1}^{K}\sum_{t=1}^{T_k-1}\sum_{j=1}^{N}\hat{\alpha}_t^{(k)}(i).a_{ij}.b_j(O_{t+1}^{(k)}).\hat{\beta}_{t+1}^{(k)}(j)} \tag{5.1}$$

- coeficientes de ponderação das gaussianas[1,11]:

$$\bar{c}_{jm} = \frac{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\sum_{i=1}^{N}\hat{\alpha}_t^{(k)}(j).a_{ji}.b_i(O_{t+1}^{(k)}).\hat{\beta}_{t+1}^{(k)}(i).G_{jm}^{(k)}(t)}{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\sum_{i=1}^{N}\hat{\alpha}_t^{(k)}(j).a_{ji}.b_i(O_{t+1}^{(k)}).\hat{\beta}_{t+1}^{(k)}(i)} \tag{5.2}$$

- médias das gaussianas[1,11]:

$$\bar{\mu}_{jm} = \frac{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\sum_{i=1}^{N}\hat{\alpha}_t^{(k)}(j).a_{ji}.b_i(O_{t+1}^{(k)}).\hat{\beta}_{t+1}^{(k)}(i).G_{jm}^{(k)}(t).O_t^{(k)}}{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\sum_{i=1}^{N}\hat{\alpha}_t^{(k)}(j).a_{ji}.b_i(O_{t+1}^{(k)}).\hat{\beta}_{t+1}^{(k)}(i).G_{jm}^{(k)}(t)} \tag{5.3}$$

- covariâncias das gaussianas[1,11]:

$$\bar{U}_{jm} = \frac{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\sum_{i=1}^{N}\hat{\alpha}_t^{(k)}(j).a_{ji}.b_i(O_{t+1}^{(k)}).\hat{\beta}_{t+1}^{(k)}(i).G_{jm}^{(k)}(t).\left(O_t^{(k)}-\mu_{jm}\right).\left(O_t^{(k)}-\mu_{jm}\right)^T}{\sum_{k=1}^{K}\sum_{t=1}^{T_k}\sum_{i=1}^{N}\hat{\alpha}_t^{(k)}(j).a_{ji}.b_i(O_{t+1}^{(k)})\hat{\beta}_{t+1}^{(k)}(i).G_{jm}^{(k)}(t)} \tag{5.4}$$

Nas equações citadas, α e β são as variáveis progressiva e regressiva, respectivamente; T_k é o número de observações de uma seqüência k ; K é o número de repetições do treinamento; N, o número de estados do modelo; M, o número de gaussianas; O o vetor de observação; e G é dado pela seguinte equação[1,11]:

$$G_{jm}^{(k)}(t) = \frac{c_{jm}.\mathrm{N}\left(O_t^{(k)},\mu_{jm},U_{jm}\right)}{\sum_{s=1}^{M}c_{js}.\mathrm{N}\left(O_t^{(k)},\mu_{js},U_{js}\right)} \tag{5.5}$$

O cálculo de π reestimado não se altera[1,11].

Os detalhes sobre o desenvolvimento destas equações pode ser encontrado na literatura de referência[1,11]. O algoritmo de treinamento opera conforme os princípios apresentados no Capítulo 3 (*Segmental k-means*), porém utilizando as Equações (5.1) a (5.5) para a reestimação dos modelos.

5.3 MODELOS DE *HMM* PARA PALAVRAS CONECTADAS

5.3.1 Banco de Elocuções

Com locutor único, foram realizadas gravações de quatro repetições de cada frase de treinamento. Com 15 locutores, todos do sexo masculino, cada um pronunciou uma vez cada frase de treinamento.

5.3.2 Segmentação de Frases de Treinamento

Após a obtenção dos modelos de palavras isoladas, esses modelos foram utilizados para segmentar as frases de treinamento e assim obter os limites das palavras co-articuladas dentro das elocuções.

Para a segmentação, usou-se o algoritmo de Viterbi com modelos das frases de treinamento montados a partir dos modelos de palavras isoladas. A matriz de transição A do modelo $\lambda = (A, c_{jm}, \mu_{jm}, U_{jm}, \pi)$ da frase foi composta pelo encadeamento, na diagonal principal, das matrizes A dos modelos de palavras isoladas, preenchendo os valores restantes com zeros. Foi realizado um ajuste no último elemento de A de uma palavra isolada. Esse elemento normalmente tem valor 1, mas foi verificado[11] que o uso de 0,8 em seu lugar e 0,2 para a posição imediatamente à direita do elemento 1 original produz melhor resultado na segmentação. Esta modificação leva em consideração a co-articulação dos modelos concatenados[11]. A Figura 5.1 exemplifica o processo, $a_{ij}1$, $a_{ij}2$ e $a_{ij}3$ são matrizes de transição de três modelos de palavras isoladas diferentes a serem concatenados nessa ordem. A matriz $a_{ij}inicial$ representa a concatenação sem ajuste e a matriz $a_{ij}ajustado$ apresenta os ajustes feitos nos elementos 3-3, 3-4, 6-6 e 6-7, visando representar a co-articulação entre modelos.

$$a_{ij}1 = \begin{bmatrix} a1_{11} & a1_{12} & 0 \\ 0 & a1_{22} & a1_{23} \\ 0 & 0 & 1 \end{bmatrix} \quad a_{ij}2 = \begin{bmatrix} a2_{11} & a2_{12} & 0 \\ 0 & a2_{22} & a2_{23} \\ 0 & 0 & 1 \end{bmatrix} \quad a_{ij}3 = \begin{bmatrix} a3_{11} & a3_{12} & 0 \\ 0 & a3_{22} & a3_{23} \\ 0 & 0 & 1 \end{bmatrix}$$

$$a_{ij}inicial = \begin{bmatrix} a1_{11} & a1_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a1_{22} & a1_{23} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a2_{11} & a2_{12} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a2_{22} & a2_{23} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a3_{11} & a3_{12} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a3_{22} & a3_{23} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$a_{ij}ajustado = \begin{bmatrix} a1_{11} & a1_{12} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a1_{22} & a1_{23} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0{,}8 & 0{,}2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a2_{11} & a2_{12} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & a2_{22} & a2_{23} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0{,}8 & 0{,}2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a3_{11} & a3_{12} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a3_{22} & a3_{23} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

FIGURA 5.1 – Concatenação de modelos de palavras isoladas[11].

5.3.3 Refino de Modelos

Com os segmentos de gravações correspondentes a cada palavra co-articulada, fez-se novo treinamento. Os novos modelos de *HMM* obtidos foram então concatenados para fazer nova segmentação das frases de teste. Estes novos segmentos passaram por um processo de verificação com os modelos correspondentes e foi armazenado o somatório das verossimilhanças encontradas. Os novos segmentos também foram utilizados para atualizar os modelos, o que originou nova segmentação, avaliação e verossimilhança total.

Admitiu-se que o processo de refino de modelos atingia uma convergência quando a variação entre as verossimilhanças totais de duas etapas subseqüentes fosse menor ou igual a 0,1% [11].

A convergência foi atingida com 3 etapas de refino, tanto para o sistema dependente do locutor quanto para o sistema independente do locutor.

Os parâmetros do algoritmo de *HMM* são os mesmos apresentados para o caso das palavras isoladas.

5.4 RECONHECIMENTO DAS FRASES

5.4.1 Banco de Elocuções

Para a verificação do sistema no modo independente do locutor, realizou-se uma gravação de cada frase de teste, totalizando 41 frases, com um total de 218 palavras. Para o sistema com vários locutores, cinco pessoas do sexo masculino, diferentes daqueles que geraram as frases de treinamento, gravaram uma repetição de cada frase de teste, totalizando 1.090 palavras.

5.4.2 Implementação do Algoritimo *Level-Building (LB)*[3,11,12]

Formalmente, o algoritmo é o seguinte:

Considera-se os modelos de palavras P_k, $1 \leq k \leq K$, do dicionário e uma seqüência de observações de teste O_t, $t=1, 2, 3,...,T$. A tarefa do reconhecedor é decodificar O em uma seqüência de unidades $\{P_1, P_2,...,P_p\}$, onde p é o número de palavras contidas na sentença decodificada, que melhor descreva a seqüência de observações, de modo que a probabilidade conjunta da seqüência de observações e da seqüência de estados seja maximizada.

No nível $I=1$ (nível inicial), apresenta-se o modelo q de uma palavra à seqüência de observações O (seqüência de janelas da elocução), começando pela janela 1. O cálculo das verossimilhanças é realizado utilizando-se o algoritmo de Viterbi, da seguinte forma:

- Inicialização:

$$\delta_1(1) = b_1^q(O_1) \tag{5.6}$$

onde $\delta_t(j)$ está relacionado à probabilidade conjunta da seqüência parcial de estados e de observações até o instante t e estado j, dadas as probabilidades de transições A e as probabilidades de saídas B

$$\delta_1(j) = 0, \qquad j = 2,3,...,N. \tag{5.7}$$

- Recursão: para $2 \leq t \leq T$ e $1 \leq j \leq N$

$$\delta_t(j) = \max_{1 \leq i \leq N}\left(\delta_{t-1}(i).a_{ij}^q\right).b_j^q(O_t) \tag{5.8}$$

- Término:

$$P(I,t,q) = \delta_t(N) \quad , \qquad 1 \leq t \leq T \tag{5.9}$$

$$G(I,t,q) = 0, \tag{5.10}$$

Quando todas as palavras tiverem sido apresentadas no nível 1, devem ser salvos os seguintes valores:

$$P_{\max}(I,t) = \max_q\left(P(I,t,q)\right) \tag{5.11}$$

$$G_{\max}(I,t) = G\left(I,t,\arg\max_{q} P(I,t,q)\right) \tag{5.12}$$

$$W_{\max}(I,t) = \arg\max_{q}\left(P(I,t,q)\right) \tag{5.13}$$

onde P_{max} é a melhor probabilidade de saída do nível, G_{max} é o ponteiro de retorno do nível (armazena a janela de início da palavra corrente) e W_{max} é o indicador da palavra mais provável de ter produzido a seqüência de vetores de observações.

O cálculo para o segundo nível e os subsequentes diferem somente quanto à avaliação da verossimilhança no primeiro estado da palavra, pois é preciso considerar a co-articulação das palavras para determinar o ponto exato onde a palavra do nível anterior termina e a do nível corrente inicia.

A inicialização desses níveis é realizada da seguinte forma:

$$\delta_1(1) = 0, \tag{5.14}$$

$$\delta_t(1) = \max\left(P_{\max}(I-1,t-1), \delta_{t-1}(1).a_{11}^{q}\right).b_1^{q}(O_t), \tag{5.15}$$

$$\alpha_t(1) = \begin{cases} t-1, se P_{\max}(I-1,t-1) > \delta_{t-1}(1).a_{11}^{q} \\ \alpha_{t-1}(1), de\_outra\_maneira \end{cases} \tag{5.16}$$

A Equação (5.15) escolhe a verossimilhança adequada do nível anterior e a equação (5.16) cria o vetor de retorno inicial que armazena a janela em que a palavra do nível anterior termina ou que a atual começa. Durante a etapa de recursão já apresentada, esse vetor é atualizado como se segue:

$$\alpha_t(j) = \alpha_{t-1}\left(\underset{1\le i\le N}{arq\max}(\delta_{t-1}(i).a_{ij}^{q})\right), \tag{5.17}$$

e, no final do nível, a verossimilhança acumulada da palavra q e o vetor de retorno tornam-se:

$$P(I,t,q) = \delta_t(N), \qquad 1 \le t \le T \tag{5.18}$$

$$G(I,t,q) = \alpha_t(N) \qquad 1 \le t \le T \tag{5.19}$$

Quando todas as palavras do nível tiverem sido apresentadas, as verossimilhanças máximas são atualizadas de acordo com as equações (5.11), (5.12) e (5.13), e prossegue-se para o nível seguinte.

O procedimento termina quando se atinge o número máximo de níveis L e $t=T$. A melhor seqüência de palavras, de comprimento L e probabilidade $P_{max}(L,T)$ é obtida utilizando-se o vetor de retorno $G_{max}(I,t)$, que armazena os instantes em que as palavras com maiores probabilidades, em cada nível, começam.

5.4.3 Reconhecimento de Frases

Para reconhecer cada frase de teste, o algoritmo *LB* foi aplicado seis vezes na elocução, uma vez para cada tamanho de frase possível dentro da gramática regular adotada no Capítulo 4, e os valores das verossimilhança encontrados para cada possibilidade foram comparados entre si. A opção com maior valor de verossimilhança foi considerada como a decodificação correta.

Como para cada passagem do *LB* haveria a necessidade do cálculo dos b_j para cada janela da elocução e para cada estado de cada modelo, a primeira tarefa do decodificador foi obter todos esses b_j. Esta etapa consome mais de 90% do esforço de processamento da busca pela frase pronunciada.

O algoritmo *LB* fora proposto para resolver o problema da decodificação em uma única *passada* pala elocução. As informações referentes à gramática devem ser *embutidas* dentro do algoritmo específico para a linguagem, lembrando que modificações na linguagem implicam modificações significativas dentro do algoritmo. Entretanto, pelas possibilidades da gramática adotada e para facilitar a modificação para inclusão ou remoção de possibilidades de frases, optou-se por um algoritmo *LB* genérico que consulta uma lista de palavras possíveis de ocorrer em cada posição de uma frase de tamanho especificado. Para uma frase de tamanho diferente, outra lista é consultada. Modificações na gramática apenas se refletem no número e no conteúdo da lista de palavras. Comparativamente ao tempo gasto pelo cálculo dos b_j, o acréscimo de tempo das várias aplicações do *LB* ficou irrelevante.

Como exemplo, em uma aplicação do algoritmo *LB* em que se deseje encontrar a melhor seqüência de palavras que componham uma sentença de 6 vocábulos, a lista de palavras conterá também 6 divisões. Na primeira parte estarão as palavras de números 13 a 17 da Tabela 4.1, que são os sujeitos possíveis. O *LB* fará então a inicialização do primeiro nível considerando apenas os modelos dessas 5 palavras. Ao passar para o nível 2, a lista indicará que os modelos a serem considerados são os correspondentes ao de número 1 a 3 da Tabela 4.1 (os dígitos correspondentes às dezenas do complemento do sujeito). O algoritmo fará a melhor concatenação possível entre os modelos de 13 a 17 com os modelos de 1 a 3, dentro da seqüência da elocução fornecida. E assim sucessivamente até atingir o nível 6.

A inclusão da gramática dentro do *LB* leva, geralmente, à inclusão de probabilidades de ocorrência de seqüências de palavras (onde os bigramas podem ser incluídos) no cálculo da verossimilhança total. Porém, da forma como a gramática proposta foi montada, os grupos de palavras correspondentes às possibilidades nos ramos da máquina de estados podem ocorrer com igual probabilidade. Sendo assim, a adição dessas probabilidades no cálculo não traria benefícios ao decodificador e foi omitida.

5.5 PÓS-PROCESSAMENTO SINTÁTICO

Como forma da minimizar os erros de decodificação, alguns testes foram incluídos na rotina de decodificação após cada aplicação do algoritmo *LB*.

Com base no vetor de retorno, é possível determinar quantas janelas da elocução foram designadas para cada modelo de palavra. Como os estados são representações matemáticas dos eventos acústicos, que ocorrem geralmente em intervalos de algumas janelas, espera-se que a cada palavra seja atribuído um número de janelas maior que o número de estados que modela aquela palavra. Tendo isto em mente, criou-se um teste em que caso alguma palavra da sentença decodificada tivesse menos janelas atribuídas do que o seu número de estados mais um, a verossimilhança da sentença é zerada, excluíndo-a do grupo das melhores a serem consideradas ao final de todas as passagem do *LB*. Para o vocabulário utilizado, a exceção no teste ocorre com a palavra 'a', por ter a duração mais curta e ser facilmente envolvida pela coaticulação com as palavras vizinhas na elocução.

Para o caso das unidades de deslocamento no singular, a ocorrência da seqüência predecessora de dígitos '001' no complemento de quantidade de deslocamento implicou na substituição automática pelo singular correspondente. Por exemplo, a elocução '... zero zero um metros.' foi corrigida para '...zero zero um metro.'

Pela forma que o algoritmo de decodificação foi implementado, a seqüência de palavras no princípio da decodificação é composta somente pela palavra 'zero' (código 0 para a rotina), sendo substituída pelas palavras corretas ao final do *LB*. Em testes preliminares, houve casos de erro de decodificação em que a frase de saída apresentava a palavra 'zero' em seu início, o que, segundo a gramática, não é possível. Assim, caso a frase decodificada iniciasse por 'zero', sua verossimilhança é zerada, excluíndo-a também do grupo das melhores a serem consideradas ao final de todas as passagem do *LB*.

5.6 CÁLCULO DA TAXA DE ACERTO

Após a passagem pelo pós-processador sintático, a frase decodificada está pronta para ser comparada com a seqüência de palavras da elocução.

Foi feita uma alteração na forma comum de contagem de palavras erradas. Como o que se deseja decodificar é uma ação a ser executada por servo-mecanismos, a troca de palavras decodificadas por seus sinônimos não foi considerada como erro. Assim as decodificações de 'gire' por 'vire', de 'um' por 'uno' e de 'pra' 'para a' ou por 'a' não foram computadas como erros já que não perturbam o significado da mensagem.

Para o caso das demais substituições, incrementou-se um contador para cada ocorrência ao longo das frases de teste.

Ocorrendo tamanhos diferentes na frase de teste e na a frase decodificada, incrementou-se um contador de inserções ou o contador de exclusões conforme o caso, tantas vezes quanto a diferença de comprimento encontrado. Nessas disparidades de tamanho, observou-se em teste preliminares que, por vezes, o reconhecedor indicava o início da frase correto e errava o final, e em outras vezes errava o início e acertava o final. Assim, para o cômputo das substituições, as frases de tamanhos diferentes foram verificadas do início para o final e vice-versa, e o caminho que apresentasse maior semelhança era usado para avaliar as trocas de palavras.

Como exemplo, considere-se a sentença 'motor zero dois, vire para_a direita, zero dois zero graus' que foi decodificado como 'rode a direita, zero dois zero graus'. Neste caso há 3 exclusões correspondentes ao sujeito e seu complemento. Comparando no sentido direto, há 7 substituições: 'motor zero dois, vire para_a direita, zero' por 'rode a direita, zero dois zero graus'. Já comparando no sentido inverso, há somente 1 substituição: 'vire para_a direita, zero dois zero graus' por 'rode a direita, zero dois zero graus', já que a troca de 'para_a' por 'a' não é contada como erro. Sendo assim, a comparação no sentido inverso foi usada para o cálculo das substituições deste exemplo.

Com base na contagem das substituições, exclusões e inserções, usou-se a Equação (4.1) para obter a taxa de acerto dos sistemas para um locutor e para vários locutores.

CAPÍTULO 6

RESULTADOS, CONCLUSÕES E SUGESTÕES

6.1 DETERMINAÇÃO DE PONTOS TERMINAIS

6.1.1 Resultados

Para testar o método proposto, foram feitas dez gravações, cada uma contendo a elocução de um dos dez dígitos.

O ambiente de gravação usado continha como ruído de fundo sons comuns em um ambiente doméstico. As gravações foram feitas em um cômodo situado no terceiro andar, com janela aberta para uma rua de pouco movimento, próximo a um aeroporto regional, e com um aparelho de televisão ligado em volume moderado no cômodo vizinho.

Para avaliar os três métodos citados anteriormente, cada gravação teve seus pontos iniciais e finais determinados por cada método. A Tabela 6.1 apresenta os resultados. O método A corresponde à edição manual da gravação. O método B é o que emprega somente os valores de energia e taxa de cruzamento por zero. O método C, ora proposto, utiliza desvio padrão de amostras de ruído.

TABELA 6.1: Comparação dos resultados de cada método

	Número de ordem da janela inicial			Número de ordem da janela final		
Método / Dígito	A	B	C	A	B	C
Um	124	122	123	164	167	164
Dois	120	67	122	188	207	189
Três	226	1	226	268	472	267
Quatro	212	23	212	184	301	187
Cinco	178	4	175	264	336	262
Seis	188	159	187	268	322	269
Sete	200	142	198	260	276	260
Oito	224	10	224	284	492	284
Nove	198	198	199	262	276	265
Zero	192	46	191	260	438	263

6.1.2 Conclusões

Pelo estudo dos resultados da Tabela 6.1, comparando-se especificamente as colunas A e C, nota-se que o método proposto aproxima muito dos valores determinados pela edição manual das gravações, em que o ouvido humano é a principal ferramenta.

No ambiente ruidoso das gravações, o método B realizou vários erros de estimação, quanto ao ponto inicial e ao ponto final das locuções. Este método apresentou resultados próximo aos demais nas gravações dos dígitos 'um' e 'nove', em que o ruído de fundo foi casualmente mais baixo.

O método proposto C é mais complexo, mais demorado em termos computacionais porém com melhor proximidade dos valores encontrados pelo método A. Nenhuma correção seria necessária para os pontos terminais determinados dessa forma.

No desempenho dos três métodos, uma melhor caracterização do ruído ambiente permite uma determinação mais eficiente de pontos terminais.

O número de atributos usados para a determinação dos pontos terminais implica maior ou menor esforço computacional. É necessário um estudo das características das locuções que serão processadas para identificar se o aumento de atributos compensará o acréscimo no tempo de processamento. Caso o silêncio represente a maior parte das gravações a serem analisadas, o uso de um menor número de atributos é sugerido para que não se perca a maior parte do tempo analisando o que é irrelevante no processo, sendo necessário avaliar as modificações no desempenho.

A mudança de atributos usados pode modificar em muito o desempenho da tarefa da determinação dos pontos iniciais e finais. Dependendo do sistema de reconhecimento a ser usado, subconjuntos de atributos podem ser calculados previamente para a determinação dos pontos terminais de forma mais eficiente.

O processo automatizado de aquisição de elocuções, pelo qual o próprio locutor supervisionava o trecho editado, permitiu que o número de correções de pontos terminais após a montagem do banco de gravações (mais de 7.000 arquivos) ficasse próximo de zero.

O teste de nível de sinal realizado diretamente na seqüência de números inteiros obtidos da placa de aquisição de som eliminou a necessidade de ajustes posteriores. Em trabalhos anteriores[4,5,6,7,11], esse ajuste para o caso de uma elocução de baixa intensidade implicava multiplicação dos valores por um escalar. Isto não traz a mesma definição de *bits* de amplitude que seria obtida caso o locutor fosse orientado a repetir a elocução em voz mais alta. Este é o primeiro trabalho em reconhecimento de voz no Instituto no qual se tentou obrigar que a

intensidade sonora da voz do locutor estivesse integralmente dentro faixa dinâmica do conversor analógico/digital de 16 *bits* da placa de som, visando uma melhor definição em amplitude dos fonemas gravados.

O uso de microfone mecanicamente acoplado aos fones eliminou a preocupação de manter uma distância padrão do microfone e permitiu maior liberdade de movimentos aos locutores durante as gravações.

6.2 SISTEMA DE RECONHECIMENTO DE PALAVRAS CONECTADAS

6.2.1 Resultados

A convergência dos modelos co-articulados (fator de 0,1%) foi atingida com 3 etapas de refino de modelos, tanto no sistema dependente do locutor quanto no sistema independente do locutor.

A Tabela 6.2 apresenta os valores obtidos em cada um dos sistema de reconhecimento, após a aplicação do pós-processador sintático:

TABELA 6.2 Resultados do reconhecimentos das frases de teste

Sistema / Item	Dependente do locutor	Independente do locutor
Taxa de acerto	98,62%	93,67%
Substituições	1	55
Inclusões	2	14
Exclusões	0	0
Total de palavras	218	1090

6.2.1.1 Observações Sobre o Sistema Dependente do Locutor

A única substituição e as duas inclusões ocorreram na mesma frase de teste. A palavra 'pare' foi trocada por 'gire'. A frase correta continha 4 vocábulos, porém identificou a elocução como uma frase de 6 palavras, realizando duas inclusões.

Para todas as demais frases de teste, as substituições ocorridas foram perfeitamente assimiladas pelo pós-processador sintático e devidamente corrigidas.

6.2.1.2 Observações Sobre o Sistema Independente do Locutor

As inclusões ocorreram nas identificações incorretas de frases de 4 palavras interpretando-as como frases de 6 palavras. Em todas as frases com inclusões, o verbo sem complemento foi substituído por um verbo que admitia complemento, seguido da inclusão de um complemento de direção. A razão entre a verossimilhança da frase reconhecida (6 palavras) e a verossimilhança da frase correta (4 palavras) teve o valor mínimo de 93%. Este fato levou à adição de um teste ao pós-processador sintático: caso a melhor seqüência decodificada contenha 6 palavras, verifica-se qual a razão da verossiminhança da melhor frase decodificada com 4 palavras

sobre a verossimilhança da frase de 6 palavras; se a razão estiver entre 90 e 100%, substitui-se a frase de 6 pela de 4 palavras. Caso esse processo não fosse conveniente para o reconhecededor, deveria ocorrer um aumento no número total de exclusões, porém, comparando a Tabela 6.2 como a Tabela 6.3, nota-se que houve um acréscimo na taxa de acerto e uma redução significativa nas inclusões. Este tipo de teste foi considerado como o mais específico da tarefa dentre todos dos testes que compõem o pós-processador sintático. As substituições apresentaram valores próximos porque os verbos deixaram de ser trocados, porém passou a ocorrer erros nos dígitos identificados no complemento do sujeito.

TABELA 6.3: Resultados com troca de frases reconhecidas

Sistema Item	Locutor único troca de frases: 6/4	Vários locutores troca de frases: 6/4
Taxa de acerto	99,54	94,51%
Substituições	1	58
Inclusões	0	2
Exclusões	0	0
Total de palavras	218	1090

As elocuções das palavras isoladas ‘ande’ e ‘desligue’ causaram erros de substituição muito freqüentes.

A palavra co-articulada com maior índice de substituições foi a palavra ‘ande’ (12 substituições por ‘rode’), seguida da palavra ‘trás’ (10 substituições por ‘cima’), sendo comum a ocorrência das substituições nas mesmas frases ditas por diferentes locutores.

No conjunto dos dígitos, a palavra ‘dois’ foi a que mais substituiu outras palavras (7 das 10 substituições numéricas), com ocorrências de trocas de ‘oito’ por ‘dois’ e ‘uno’ por ‘dois’ freqüentes.

Não ocorreram erros na identificação dos sujeitos das sentenças, porém, algumas das substituições numéricas causaram erros nos complementos dos sujeitos, causando identificação incorreta entre servo-mecanismos de uma mesma categoria.

6.2.2 Conclusões

O número de repetições de vocábulos isolados para um locutor foi de 1.350, e para vários locutores foi de 2.160. Esta diferença e o número de locutores envolvidos não alterou o número de interações necessárias para a convergência dos modelos co-articulados para cada sistema.

Os resultados obtidos para o sistema dependente do locutor indicaram que o conjunto de frases de treinamento, a quantidade e o critério de seleção de bigramas foram satisfatórios.

Os resultados obtidos pelo sistema independente do locutor indicaram uma deficiência nos modelos de palavras co-articuladas.

O modelo co-articulado de 'ande' não serviu para o reconhecimento da palavra como elocução isolada. A solução seria acrescentar repetições desta palavra isolada no conjunto de treinamento. As elocuções isoladas de ‘ande’ foram obtidas de apenas 12 locutores e destinadas somente à etapa de segmentação inicial.

No caso dos verbos 'ande' e 'rode', a semelhança fonética dos finais das duas palavras aliado ao reduzido número de ocorrências dos bigramas (compostos por estes verbos e as preposições possíveis) dificultou a distinção pelo algoritmo de reconhecimento. O uso de uma quantidade maior de bigramas contendo estas palavras e/ou a adoção de um fator de convergência menor que 0,001 devem reduzir esses erros de substituição.

No caso das palavras 'trás' e 'cima', verificou-se que as trocas ocorreram quando a palavra 'trás' foi precedida das preposições 'para_a' e 'pra'. As seqüências de fonemas contendo 'r' devem ter causado uma má distribuição dos estados dos modelos da palavras envolvidas ao longo da sentença decodificada. Os estados correspondentes às preposições devem ter abrangido parte das janelas pertencentes a 'trás'. O restante dessas janelas deve ter encontrado melhor posicionamento no modelo da palavra 'cima', causando as 10 substituições sistemáticas encontradas. Uma redefinição no número de estados empregados nos modelos das preposições e das palavras 'trás' e 'cima' deve reduzir estes erros de substituição.

As substituições do dígito 'oito' pelo dígito 'dois' podem ter ocorrido devido às semelhanças fonéticas entre os encontros vocálicos existentes nas duas palavras. Outros dígitos também foram erroneamente identificados como 'dois'. A solução do problema não deve ser pela adição de mais repetições destes dígitos no conjunto de treinamento porque são eles, os dígitos em geral, que apresentam as maiores ocorrências nas frases de treinamento. Sugere-se, então, uma modificação no número de estados do modelo da palavra 'dois' e/ou a adoção de um fator de convergência menor que 0,1% para o refino de modelos co-articulados.

O modelo da palavra isolada 'desligue' não deve estar com um bom dimensionamento de estados. Todas as substituições ocorridas foram por verbos terminados em 'e', e não somente pelo vocábulo 'ligue'. Uma modificação no número de estados do modelo deve reduzir estes erros de substituição.

De acordo com testes preliminares, a combinação de vários critérios de escolha da melhor frase decodificada possibilita uma elevada taxa de acerto. As taxas de acerto obtidas não seriam possíveis caso não fossem considerados quatro fatores básicos: a verosimilhança das frases de diversos tamanhos, a distribuição de janelas temporais ao longo dos estados, a gramática regular utilizada e a semântica orientada à tarefa de acionamentos eletro-mecânicos.

Inicialmente, pensou-se que o critério da verossimilhança não seria útil para distinguir palavras isoladas grandes de frases de 10 vocábulos, porém, os resultados indicaram que, para uma elocução contendo uma única palavra, a verossimilhança desta elocução ainda pode ser maior do que a verossiminhança de uma seqüência de 10 vocábulos erroneamente identificado para a mesma elocução.

O erro de inclusão de dígitos é comum em outros tipos de sistemas[11] onde, em geral, o dígito 'um' aparece em número maior do que o pronunciado. No sistema proposto, seguindo a gramática regular, a única inclusão de dígitos só poderia ocorrer caso houvesse o acréscimo de todo um complemento de quantidade de deslocamento. Isto só seria possível em frases com menos de 10 palavras e tal caso não foi constatado. O uso, na gramática regular, de seqüências fixas de dígitos pode evitar a inclusão indesejada de algarismos.

O estudo dos erros ocorridos não apontou nenhum problema originado pela adoção das pausas na gramática regular. Pausas criteriosamente introduzidas nas elocuções podem diminuir o conjunto de frases de treinamento, diminuir o tempo de gravação para os locutores, facilitar o trabalho dos algoritmos de segmentação e permitir rearranjos na gramática regular. A gramática adotada inicialmente visava reger apenas frases de 10 palavras. Com o desenvolvimento dos algoritmos de reconhecimento e critérios de avaliação foi possível expandir a gramática regular para abranger 6 tipos de frases. Isso ampliou em muito as possibilidades de expressão do operador do sistema.

Nenhum erro foi atribuído a problemas de ruído na gravação. Os atributos da voz utilizados (coeficientes *PLP-Ceps*) mostraram-se robustos aos ruídos dos ambientes de gravação onde as ventoínhas dos computadores eram as fontes mais notáveis de ruído, seguidas de eventuais ruídos urbanos exteriores às salas de gravações.

6.3 SUGESTÕES PARA TRABALHOS FUTUROS.

A) Notou-se que a faixa de amplitude permitida do sinal de voz adquirido era relativamente estreita. Isto trazia alguma dificuldade para alguns locutores adaptarem a intensidade da voz para a faixa desejada. O desenvolvimento de um *compressor de áudio* analógico para uso em um estágio de pré-amplificação do microfone pode trazer facilidade para os locutores e permitir a acomodação da amplitude do sinal de voz dentro de toda a faixa dinâmica do conversor *A/D* da placa de som. Isto garantiria maior definição de amplitude para os fonemas. Seria necessário um estudo da distorção do sinal gerado pelo *compressor de áudio* e a influência disto nos atributos da voz utilizados pelo sistema de reconhecimento.

B) Será interessante estudar a possibilidade de utilizar dois microfones para realizar a aquisição do sinal de voz, fazendo uso da entrada de linha em estereo da placa de som. Vários artigos podem ser encontrados na literatura que indicam que o uso de uma matriz de microfones para a captação de voz permite obter um acréscimo significativo da relação sinal/ruído.

C) Convém verificar o desempenho do sistema utilizando outros atributos da voz e/ou números variados de coeficientes. Testes preliminares indicaram que a redução criteriosa do número de atributos também permite atingir taxas de acerto elevadas.

D) Será interessante realizar testes utilizando técnicas de adaptação de locutor[12] e, baseando-se nos modelos obtidos de vários locutores, adaptá-los para um locutor novo e tentar atingir os resultados obtidos no sistema de locutor único, porém com um número bem menor de elocuções de treinamento.

E) Será oportuno implementar uma versão do mesmo sistema que utilize unidade fonéticas[1,3,11,12] como modelos fundamentais (em vez de palavras), realizar um estudo mais profundo do idioma nacional, aplicar estes conhecimentos na definição dos modelos matemáticos[25] e comparar os resultados.

F) Convém utilizar outro algoritmo de reconhecimento diferente do *LB*, como o *one-pass* e/ou o *one-pass modificado*[11,12] e/ou um sistema *híbrido* (*HMM* com *redes neurais*)[4,7] e comparar os resultados.

G) Finalmente, sugere-se aplicar o sistema no controle de um equipamento real.

REFERÊNCIAS BIBLIOGRÁFICAS

1. SANTOS, S. C. B. e ALCAIM, A.; Fundamentos de Reconhecimento de Voz, Centro de Estudos em Telecomunicações da Pontifícia Universidade Católica do Rio de Janeiro, CETUC-DID-01/95, Setembro de 1995.
2. ROE, D. e WILPON, J.; *Whither Speech Recognition: The Next 25 Years*, IEEE Communications Magazine, Nov 1993.
3. DELLER Jr., J. R. et alii; *Discrete-Time Processing of Speech Signals*, Macmillian Publishing Company, Nova Iorque, 1993.
4. ANDRADE, M.A.R. e VICENTE, J.V.M.; Reconhecimento de Comandos Isolados à Voz, Projeto de Fim de Curso, IME, 1997.
5. PARANAGUÁ, E. D. S.; Reconhecimento de Locutores Utilizando Modelos de Markov Escondidos Contínuos. Dissertação de Mestrado, IME, 1997.
6. BEZERRA, M. R.; Reconhecimento Automático de Locutor para Fins Forences, Utilizando Técnicas de Redes Neurais, Dissertação de Mestrado, IME, 1994.
7. SILVA, D. G.; Comparação entre os Modelos de Markov Escondidos Contínuos e as Redes Neurais Artificiais no Reconhecimento de Voz, Projeto de Fim de Curso, IME, 1997.
8. HERMANSKY, H.; *Perceptual predictive (PLP) analysis of speech*, J. Acoust. Soc. Am. 87 (4), April 1990.
9. OPPENHEIM, A. V. & SCHAFER, R. W.; *Digital Signal Processing*, Prentice-Hall, 1975.
10. LAMEL, L., RABINER, L. R., ROSEMBERG, A., & WILPON, J.; *Improved end-point detector for isolated word recognition*, IEEE Trans. On ASSP, Vol. 29 pp. 777-785, 1981.
11. SANTOS, S. C. B.; Reconhecimento de Voz Contínua para o Português Utilizando Modelos de Markov Escondidos, Tese de Doutorado, Departamento de Engenharia Elétrica, Pontifícia Universidade do Rio de Janeiro, 1997.
12. RABINER, L. R. e JUANG, B. H.; *Fundamentals of Speech Recognition*. Prentice Hall, USA, 1993.
13. PAPOULIS, A.; *Probability, Random Variables, and Stochastic processes*. McGraw-Hill, 1965.
14. HOEL, P. G.; Port, S. C. e Stone, C. J.; *Introduction to Stochastic Processes*. Houghton Miffin Company, 1972.
15. ROSS, S. M.; *Stochastic Processes*. John Wiley & Sons, 1983.

16. WARAKAGODA, N. D.; *A Hybrid ANN-HMM ASR system with NN based adaptive preprocessing.* M.Sc. thesis, Intitutt for Teleteknikk Transmisjonsteknikk, 1996.
17. WILPON, J.G. e RABINER, L.R.*; A Modified K-Means Clustering Algorithm for Use in Isolated Work Recognition*, *IEEE Transactions on Acoustics, Speech, and Processing*, Vol. ASSP-33, No. 3, *June* 1985.
18. SCHWARTZ, R., CHOW, Y., KIMBALL, O., ROUCOS, S., KRASNER, M. e MARHOUL, J.; *Context-dependent Modeling for Acoustic-phonetic Recognition of Continuos Speech*, IEEE, 1985.
19. SILVEIRA, R.C.P.; Estudo de Fonologia Portuguesa, Cortez Editora, São Paulo, 1986.
20. RABINER, L.R., JUANG, B.H., LEVINSON, S.E. e SONDHI, M.M.; *Recognition of Isolated Digitis Using Hidden Markov Models with Continuos Mixture Densities,* AT&T Technical Journal, Vol 64, No. 6, July-August 1985.
21. RABINER, L.R., WILPON, J.G., SOONG, F.K.; *High Performace Connected Digit Recognition Using Hidden Markov Models*, IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 37, No. 8, August 1989.
22. MYERS, C.S., LEVINSON, S.E.; *Speaker Independent Connected Word Recognition Using a Syntax-Directed Dynamic Programming Procedure*, , IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. ASSP-30, No. 4, August 1982.
23. BROWN, M.K., McGEE, M.ª, RABINER, L.R., WILPON, J.G.; *Training Set Design for Connected Speech Recognition*, , IEEE Transactions on Signal Processing, vol. 39, No. 6, August 1991.
24. NICOLA, J. e INFANTE, U.; Gramática Contemporânea da Língua Portuguesa, Editora Scipione, 14ª edição, 1995.
25. ZUE, V.W.; *The Use of Speech Knowledge in Automatic Speech Recognition*, Proceedings of the IEEE, Vol. 73, No. 11, November 1985.
26. RABINER, L.R., WILPON, J.G. and JUANG, B.H.; *A segmental k-means Training Procedure for Connected Word Recognition,* AT&T Techinical Journal.
27. DAVIS, S.B. e MERMELSTEIN,P.; *Comparation of Parametric Representations of Monosyllabic Word Recognition in Continuously Spoken Sentences,* IEEE Transactions on Acoustics, Speech, and Signal Processing, Vol. ASSP-28, No. 4, August 1980.
28. ITAKURA, F.; *Minimum Prediction Residual Principle Applied to Speech Recognition,* IEEE Transactions on Acoustics, Speech, and Signal Processing, February 1975.

29. MAMMONE, R.J., ZHANG, X., RAMACHANDRAN, R.P.; *Robust Speaker Recognition,* IEEE Signal Processing Magazine, September 1996.
30. MESSINA, R.O. e CABRAL Jr.,E.F.; *A Comparation of Robust Features for Speaker Recognition,* SBT-97, 1997.
31. NIEMÖLLER, M., HAUENSTEIN, A., MARSCHALL, E., WITSCHEL, P., HARKE, U.; *A PC-based Real-Time Large Vocabulary Continuous Speech Recognizer for German,* IEEE, 1997.
32. NUNES, H.F., NAGLE, E.J. e SILVA, C.H.; Segmentação Fonêmica Automática para Frases do Português Brasileiro Utilizando-se HMM, SBT-97, 1997.
33. PICONE, J.W.; *Signal Modeling Techniques in Speech Recognition,* Proceedings of the IEEE, Vol. 81, No.9, September 1993.
34. SANTOS, S.C.B. e ALCAIM, A.; Inventário Reduzido de Unidades Fonéticas do Português Brasileiro para o Reconhecimento de Voz Contínua, SBT-97, 1997.
35. SILVA, F.J.F. e SAOTOME, O. Reconhecimento de Fala com Vocabulário Ilimitado para o Português do Brasil, SBT-97, 1997.
36. STROPE, B. e ALWAN, A.; *A Model of Dinamic Auditory Perception and Its Application to Robust Word Recognition,*IEEE Transactions on Speech and Audio Processing, Vol. 5, No. 5, September 1997.
37. STRUM, R.D. e KIRK, D.E.; *Discrete Systems and Digital Signal Processing,* Addison-Wesley Publishing Company, 1989.
38. TAKARA, T.T., HIGA, K. e NAGAYAMA, I.; *Isolated Word Recognition Using teh HMM Structure Selected by the Genetic Algorithm,* IEEE, p. 967, 1997.
39. VASEGHI, S., HARTE, N. e MILNE, B.; *Mult-resolution Phonetic/segmental Features and Models for HMM-based Speech Recognition,* IEEE, p. 1263, 1997.

Tese apresentada por

_____

Ten QEM Marco Antonio Rocca de Andrade

E aproada pelos Srs.:

_____

TC QEM Sidney Cerqueira Bispo dos Santos – D.Sc.

_____

Cel R/1 Roberto Miscow Filho – M.C.

_____

Cel R/1 Antonio Carlos Gay Thomé – Ph.D.

_____

Ernesto Leite Pinto – D.C.

IME, RIO DE JANEIRO – RJ, 29 de dezembro de 1999.