

JOSE DO NASCIMENTO PORTELA

TUTSI - SISTEMA TUTORIAL PARA ANÁLISE DE SINAIS

Dissertação apresentada à Coordenação dos cursos de Pós-graduação em ENGENHARIA ELÉTRICA da Universidade Federal da Paraíba, em cumprimento às exigências para obtenção do grau de Mestre.

ÁREA DE CONCENTRAÇÃO: PROCESSAMENTO DA INFORMAÇÃO

BENEDITO GUIMARÃES AGUIAR NETO

Orientador

MARIA DE FÁTIMA CAMÊLO

Co-orientadora

Campina Grande
Maio 1992

DIS
OP. 688 (1992)
PSTSI

Agradecimentos

A minha esposa, pelo apoio e compreensão.

Aos colegas que compartilharam conhecimento e auxiliaram na busca de soluções: Adriano, Cortez, Eduardo, Giovani, Gustavo, Joseana, Lígia, Luis Reyes, Marlon, Mendes, Niomar, Rinaldo, Rosângela Vilar, Suely, Silvana, Washington e Wallington.



P843t Portela, Jose do Nascimento
Tutsi : sistema tutorial para analise de sinais / Jose do Nascimento Portela. - Campina Grande, 1992.
91 f. : il.

Dissertacao (Mestrado em Engenharia Eletrica) - Universidade Federal da Paraiba, Centro de Ciencias e Tecnologia.


1. Programas Tutoriais 2. Processamento da Informacao 3. Computacao - 4. Engenharia Eletrica 5. Dissertacao I. Aguiar Neto, Benedito Guimaraes, Dr. II. Camelo, Maria de Fatima, M.Sc. III. Universidade Federal da Paraiba - Campina Grande (PB) IV. Título

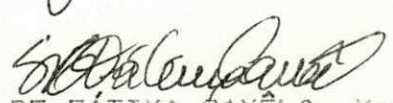
CDU 004.588(043)

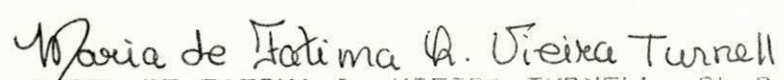
TUTSI: SISTEMA TUTORIAL PARA ANÁLISE DE SINAIS


JOSE DO NASCIMENTO PORTELA

DISSERTAÇÃO APROVADA EM 22.05.92


BENEDITO GUIMARÃES AGUIAR NETO, Dr.-Ing., UFPB
Orientador


MARIA DE FÁTIMA CAMÊLO, Mestre, UFPB
Co-orientadora


MARIA DE FÁTIMA G. VIEIRA TURNELL, Ph.D., UFPB
Componente da Banca


WILSON GUERREIRO PINHEIRO, Ph.D., UFPB
Componente da Banca

CAMPINA GRANDE - PB
MAIO - 1992

RESUMO

O objetivo do presente trabalho foi a concepção e a implementação de um Sistema Tutorial, denominado Tutsi, para comunicação de conhecimento na área de Análise de Sinais.

Tutsi explora os recursos de um sistema de computação para facilitar a aprendizagem de tópicos que envolvem dificuldades especiais no processo ensino-aprendizagem, como convolução, análise espectral, síntese de sinais, e levantamento das características estatísticas de sinais.

Sua interface com o usuário utiliza saídas gráficas na emissão dos resultados, árvore de menus para seleção de operações, possibilita a execução de funções do DOS diretamente ligadas ao seu uso, oferece Help sensível ao contexto e material instrucional para consulta.

ABSTRACT

The objective of this work is the design and implementation of a Tutoring System, called Tutsi, for knowledge communication in signal analysis.

Tutsi realizes basic operations that present special difficulties in the teaching-learning process, such as convolution, spectral analysis, signal synthesis, and statistical characteristics gathering.

Tutsi's user interface explores graphics resources in emitting results, offers a context sensitive Help, selects operations by menus, executes DOS functions closely related to its utilization, and provides a context reading on theory, exercises, examples and references.

ÍNDICE

Capítulo I. Introdução	1
1.1 Proposta de trabalho	2
1.2 Estrutura da dissertação	3
Capítulo II. Sistemas tutoriais	5
2.1 Os sistemas tutoriais	5
2.2 Teorias de aprendizagem	8
2.2.1 Teoria da aprendizagem significativa (Teoria de Ausubel)	9
2.2.2 Teoria de hierarquia de aprendizagem (Teoria de Gagné)	9
2.3 Escolha do domínio	11
Capítulo III. Domínio: Análise de sinais	14
3.1 Características estatísticas	15
3.1.1 Média	15
3.1.2 Variância	16
3.1.3 Desvio padrão	16
3.1.4 Função distribuição	16
3.1.5 Função densidade de probabilidade	16
3.1.6 Função autocorrelação	20
3.1.7 Características de implementação	20
3.2 Gerador de seqüências aleatórias	22
3.2.1 Geração de números pseudo-aleatórios	22
3.2.2 Geração de variáveis aleatórias	25
3.3 Síntese de sinais	28
3.4 Sinais determinísticos	29
3.4.1 Onda quadrada	29

3.4.1.1 Características de implementação	30
3.4.2 Pulso retangular	32
3.4.2.1 Características de implementação	32
3.5 Análise espectral	33
3.5.1 FFT	34
3.5.1.1 Características de implementação	35
3.6 Limitação de banda	36
3.7 Amostragem sample-and-hold	36
3.7.1 Amostragem instantânea	40
3.7.2 Amostragem natural	41
3.7.3 Características de implementação	41
3.7.4 Aliasing	42
3.8 Recuperação do sinal original por interpolação	42
3.9 Convolução	42
3.9.1 Convolução linear	43
3.9.2 Convolução circular	44
3.10 Sistemas de 1ª ordem	49
Capítulo IV. Tutsi - Interface	50
4.1 Interface texto	51
4.2 Menus de operações	53
4.3 Menu rotativo	56
4.4 Crítica de dados	57

4.5	Entrada paralela de dados	58
4.6	Filtragem de teclas	59
4.7	Manipulação de arquivos	60
4.8	Prevenção contra erros de execução	61
4.9	Erros fatais	62
4.10	Help	62
4.11	Interface gráfica	66
4.12	Conversão de escala horizontal	68
4.13	Conversão de escala vertical	69
4.14	Multigráficos	71
4.15	Deteção da placa gráfica	72
4.16	Superposição de curvas	77
4.17	Traçados	79
4.17.1	Características de implementação	79
4.18	Zoom	80
Capítulo V. Conclusão.		83
5.1	Sugestões para continuação do trabalho	84
Referências bibliográficas.		88
Apêndice: Instalação.		90

CAPÍTULO I

INTRODUÇÃO

A comunicação de conhecimento é a ação de causar e/ou dar suporte à aquisição de conhecimento, utilizando-se de um conjunto restrito de operações de comunicação [1].

A comunicação de conhecimento em qualquer área enfrenta dificuldades específicas que demandam recursos adequados para facilitar essa comunicação. A eficiência dos meios de comunicação de conhecimento varia de acordo com a natureza do conteúdo a ser comunicado.

Em Engenharia Elétrica, muitas são as situações em que se requer um meio de comunicação de conhecimento específico que facilite o aprendizado. Isto ocorre, por exemplo, na abordagem dos seguintes tópicos:

- Levantamento de características estatísticas de sinais;
- Convolução;
- Análise espectral; e
- Síntese de Sinais.

Cada uma das operações acima citadas apresenta graus diferentes de dificuldade quando estudadas, quer seja na assimilação de seus conceitos, quer seja na própria realização da

operação. O levantamento de características estatísticas, como a função densidade de probabilidade, por exemplo, requer o cálculo repetitivo de probabilidades de intervalos de amplitude, e o traçado do gráfico da função, tarefas que são extenuantes se realizadas com outro recurso que não o computador. Enquadra-se nesse mesmo caso a Síntese de Sinais a partir da Análise de Fourier. Suponha-se que os termos senoidais e/ou cossenoidais de um sinal periódico tenham sido calculados, e, a partir destes, o aluno deva traçar a curva, utilizando os termos na equação temporal. Seja uma curva com 1000 pontos, tendo sido obtidos 5 termos senoidais, com valor médio A_0 , segundo a equação abaixo:

$$x(t) = A_0 + A_1 \text{ Sen}(wt) + A_2 \text{ Sen}(2wt) + A_3 \text{ Sen}(3wt) + A_4 \text{ Sen}(4wt) + A_5 \text{ Sen}(5wt)$$

Para a obtenção de cada ponto da curva, são necessárias 11 operações. Portanto, para se obterem os 1000 pontos da curva, são necessárias 11000 operações. Se a resposta estiver errada, novo cálculo de termos senoidais deve ser feito e as 11000 operações repetidas. Os dois exemplos supracitados constituem amostras de um conjunto de situações em que o uso do computador em atividades instrucionais se revela imprescindível.

1.1 PROPOSTA DE TRABALHO

O presente trabalho objetivou a concepção e a implementação de um Sistema Tutorial, denominado Tutsi, para auxiliar o estudante na aprendizagem das técnicas e conceitos da 'Análise de Sinais'.

Tutsi realiza operações básicas como Convolução, Análise Espectral, Levantamento de Características Estatísticas, Síntese de Sinais, e outras. São usados recursos gráficos na apresentação dos resultados, Help sensível ao contexto, seleção de operações via árvore de menus, e execução de funções do DOS. É permitido ao usuário consultar um banco de textos que contém um resumo da teoria, exercícios, exemplos e referências bibliográficas, relacionados com a operação em estudo.

1.2 ESTRUTURA DA DISSERTAÇÃO

No Capítulo II, procura-se posicionar o trabalho realizado no conjunto atual de sistemas relacionados. Apresenta-se um breve histórico dos sistemas tutoriais, o conceito de sistemas tutoriais inteligentes e de sistemas hipertexto, e são abordadas algumas teorias de aprendizagem.

No capítulo III, considera-se todo o conteúdo teórico abrangido, adicionando-se ao final de cada tópico descrição de aspectos relevantes da implementação da função que realiza cada operação: algoritmo utilizado, particularidades da implementação, resultados obtidos, etc.

No Capítulo IV trata-se da interface com o usuário. Descrevem-se os vários menus, o Help sensível ao contexto, as opções de consulta ao material instrucional disponível no banco de textos, e o uso das teclas disponíveis para controle da execução pelo usuário. Apresentam-se também os recursos gráficos disponíveis: telas multigráficos, zoom, superposição de curvas,

saídas para a tela e para a impressora, tipos de traçado das curvas, e algoritmos utilizados nas rotinas de saída gráfica.

No capítulo V, apresentam-se um sumário do trabalho realizado, algumas conclusões e sugestões para trabalhos futuros.

Seguem-se após o Capítulo V, as Referências Bibliográficas, e o Apêndice que trata da instalação do Tutsi.

CAPÍTULO II

SISTEMAS TUTORIAIS

INTRODUÇÃO

Um sistema de computação é considerado um meio poderoso de comunicação de conhecimento dada a alta velocidade com que armazena, processa, e recupera grandes quantidades de informação. Comunica-se com o homem e com máquinas, e possibilita o diálogo com o alvo da comunicação, característica peculiar de seres inteligentes. Manipula diversos tipos de informação tais como texto, gráficos, imagem e áudio; torna possível a simulação de experimentos com condições controladas. Com o avanço da tecnologia de hardware, software e dos dispositivos de armazenamento, vislumbram-se sistemas cada vez mais diversificados quanto ao tipo e formato da informação tratada. Não obstante, o uso desses sistemas, para fins instrucionais requer a adequação da informação ao sistema, levando em conta as suas propriedades, assim como o conteúdo de um livro precisa ser adequado para ser transmitido por vídeo, cinema, etc.

2.1 OS SISTEMAS TUTORIAIS

Um Sistema Tutorial é uma ferramenta auxiliar do processo ensino-aprendizagem, que se destina não só às escolas, mas também a empresas e ao apoio à pesquisa. Visa basicamente comunicar conhecimento, e pode fazer simulações, avaliar o aprendizado, e até modificar seu próprio conhecimento no caso dos tutoriais inteligentes. Sua aplicação é eficiente em situações especiais de instrução, como, por exemplo, quando os treinandos precisam manusear substâncias químicas explosivas. Devido ao alto risco envolvido, torna-se conveniente que as operações sejam simuladas antes de realizadas. Os recursos do computador podem auxiliar também no traçado de gráficos tridimensionais, com opção de rotação, mudança de escala, variação de parâmetros, etc, facilitando o estudo de Cálculo e o entendimento de conceitos ou operações matemáticas.

Já na década de 60, surgiram os primeiros sistemas de ensino auxiliados por computador, os sistemas CAI (Computer-Aided Instruction) [2], semelhantes a um livro, com estrutura estática. Em sua maioria, eram desprovidos de qualquer flexibilidade na forma de interação com o usuário, e expunham o conteúdo em forma de instrução programada.

No início dos anos 70, surgiu o SCHOLAR, que foi considerado o primeiro Sistema Tutorial Inteligente [2]. Dispunha de um vocabulário limitado, e era capaz de analisar perguntas e fazer inferências, como, por exemplo, que Brasília está

localizada na América do Sul, uma vez que está no Brasil, que, por sua vez, está na América do Sul [2].

Com o advento das técnicas de inteligência artificial, o nível de interação com o usuário evoluiu até a "linguagem natural", permitindo também diferentes formas de expressão do conhecimento armazenado. Um exemplo desta nova geração, o SOPHIE (Sophisticated Instructional Environment) [3], funciona como um laboratório simulado de eletrônica, onde o estudante pode identificar e eliminar defeitos colocados aleatoriamente pelo sistema em um circuito. O objetivo é que o aprendiz desenvolva estratégias para a localização e eliminação de defeitos, de forma perita. O sistema fornece medições, responde a perguntas do estudante, troca dispositivos supostamente defeituosos, sugere estratégias e alterações na solução, de modo a conduzir a uma solução com noção de causalidade.

O SOPHIE dialoga com o usuário em linguagem natural, utilizando uma gramática que associa a cada categoria de termos da sintaxe um conceito do sistema, como medidas, elementos do circuito, etc.

Um Sistema Tutorial difere de um Sistema Especialista quanto às respostas fornecidas. Enquanto um Sistema Especialista procura a melhor solução para um problema, os Tutoriais auxiliam o usuário, orientando na busca de soluções.

Hartley e Sleeman propuseram uma arquitetura [3] para os Sistemas Tutoriais inteligentes composta dos módulos:

- Especialista do domínio;

- Modelo do estudante;
- Conhecimento pedagógico; e
- Interface com o usuário.

Anderson propôs uma abordagem que não inclui o "Modelo do estudante" [3].

O surgimento da tecnologia de hipertexto trouxe novas perspectivas para os sistemas de comunicação de conhecimento, possibilitando um alto grau de individualização no acesso à informação armazenada.

Um Sistema Hipertexto é um gerenciador de banco de dados [4] que permite conectar telas de informação. Cada tela constitui um "Nó". Um "Nó" pode ser um texto, um gráfico, dados, etc. Estes 'Nós' podem ser conectados uns aos outros, formando uma rede. Em um nível mais sofisticado, o Sistema Hipertexto é um ambiente de software para trabalho de colaboração, comunicação e aquisição de conhecimento.

Determinado domínio pode ser discretizado em nós e posto à disposição do usuário, que está livre para navegar pela rede de "Nós" e concatenar aqueles cuja associação lhe ocorrer. O Hipertexto imita a habilidade do cérebro de acessar informação por referência. Este tipo de atitude é bastante adequado para o uso com a teoria de aprendizagem de Ausubel descrita na seção seguinte.

2.2 TEORIAS DE APRENDIZAGEM

Algumas teorias tentam descrever como ocorre o

aprendizado nos seres humanos. Dentre elas, as teorias de Piaget, Bruner, Skinner, Ausubel, e Gagné [5].

Embora não digam exatamente o que deve ser feito para que a aprendizagem ocorra, duas dessas teorias permitem um planejamento aplicável ao processo ensino-aprendizado auxiliado por computador: as teorias de Ausubel e Gagné [1].

II.2.1 TEORIA DA APRENDIZAGEM SIGNIFICATIVA (Teoria de Ausubel)

A teoria de Ausubel [5] é também conhecida como teoria de subsunção. Ausubel considera que o acúmulo gradativo de informações no cérebro compõe, ao longo do tempo, uma estrutura hierárquica de conceitos, a ESTRUTURA COGNITIVA. Nesta estrutura, conceitos mais específicos são relacionados a conceitos mais gerais, de forma que novos conceitos são "ancorados" à medida que se relacionam com aspectos da estrutura cognitiva. Quando um conceito novo não é associado à estrutura cognitiva, pode ocorrer um tipo de aprendizado que é chamado de "aprendizagem mecânica". Por exemplo, um estudante de primeiro grau pode memorizar e aplicar a fórmula da equação do segundo grau sem, no entanto, fazer associação com lançamento de projétil ou com antenas parabólicas. Se mais tarde ocorrer a associação, terá havido uma "aprendizagem significativa". Portanto, segundo essa teoria, é muito importante detectar o que o aprendiz já sabe, para o planejamento da instrução. Por isso é proveitoso fornecer ao aprendiz, material introdutório com alto nível de generalidade e inclusividade. Tal material, que pavimenta a base para a

aprendizagem significativa é chamado de "organizador prévio".

II.2.2 TEORIA DE HIERARQUIA DE APRENDIZAGEM (Teoria de Gagné)

Para Gagné [4], um conceito a ser aprendido pode ser decomposto em conceitos mais simples, que, dispostos em determinada hierarquia, levam à assimilação do conceito a ser aprendido. Este fato evidencia a importância de se determinarem habilidades já dominadas pelo aprendiz e o ritmo de aprendizado a se executar. Um exemplo de hierarquia de aprendizagem para o conceito de aceleração é mostrado na figura 1.

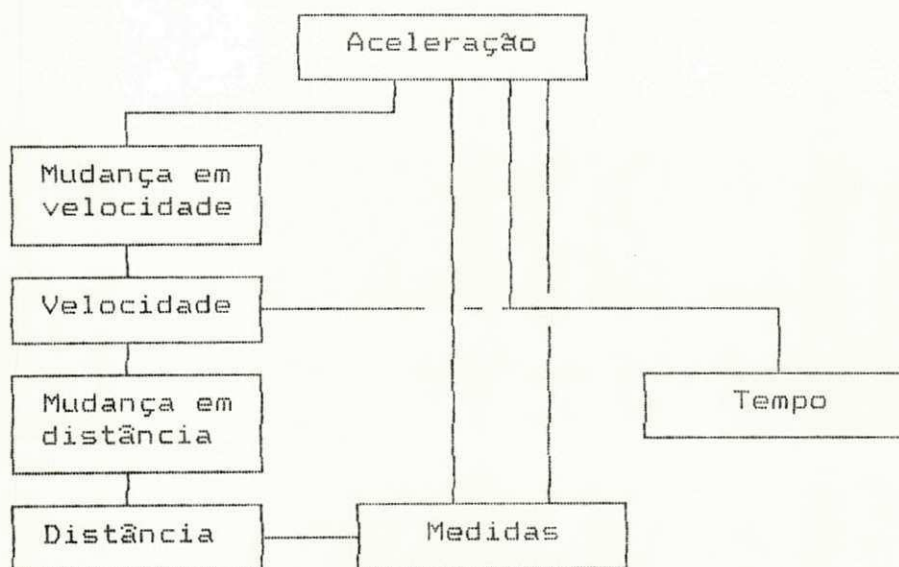


Fig 2.1 Hierarquia de aprendizagem para o conceito de aceleração

Gagné também propõe [1] um modelo de aprendizagem e memorização derivado da teoria de processamento de informação, mostrado na figura 2.2.

No seu modelo, a informação é captada pelo registrador

sensorial e codificada, passando pelo sistema nervoso até a memória de curta duração. Se a informação deve perdurar, é enviada à memória de longa duração. O gerador de respostas é ativado pelas memórias do modelo e faz a interface com o exterior do indivíduo de forma que se pode avaliar como a informação foi processada por ele.

Devido à correspondência entre o processamento da informação e a resposta expressa pelo gerador de respostas (de fato, uma ação do aprendiz) a aprendizagem para Gagné é uma mudança de estado interior observável na mudança de comportamento e na persistência desta mudança. No entanto, a teoria de Gagné não está interessada nos aspectos sociais do comportamento humano, mas nos aspectos cognitivos.

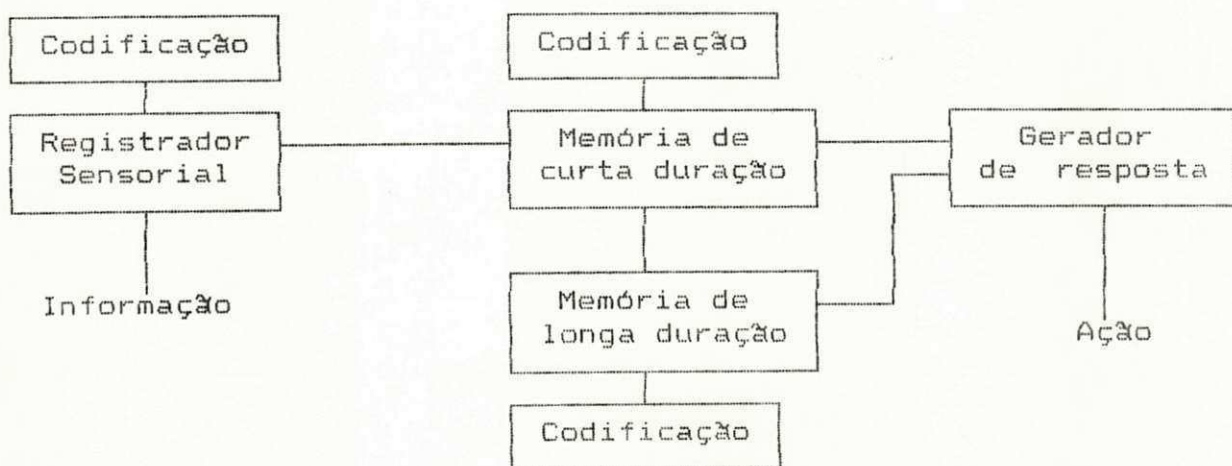


Fig 2.2 Fluxo de informação do modelo de aprendizagem de Gagné.

2.3 ESCOLHA DO DOMÍNIO

O Domínio de um sistema tutorial deve oferecer

oportunidades para exploração das características do computador como meio de comunicação, descritas na introdução deste Capítulo. A Análise de sinais foi escolhida como domínio de Tutsi, devido às dificuldades pedagógicas encontradas no ensino de alguns tópicos, dentre eles:

- Obtenção das características estatísticas de sinais;
- Síntese de sinais;
- Análise espectral; e
- Convolução.

As vantagens do uso do computador no ensino do primeiro e tópicos, já foram destacadas no Capítulo I. Apresentam-se, a seguir, as vantagens do uso de tutoriais relacionados a Análise espectral e Convolução.

ANÁLISE ESPECTRAL

Com um Sistema Tutorial pode-se calcular e esboçar o espectro de um pulso retangular seguidas vezes, variando-se, a cada vez, a duração do pulso. Esta cadeia de operações similares, em que se altera um parâmetro, determina uma relação entre o parâmetro variante e o comportamento do espectro, como se o estudante criasse uma função cujo domínio é o parâmetro variante e cujo contra-domínio é algum aspecto do espectro, por exemplo a largura do espectro.

Esta prática estabelece a relação de causa e efeito, que, se fundamentada teoricamente, conduz à compreensão de fenômenos e mesmo a inferência de fatos novos pelo estudante.

CONVOLUÇÃO

A convolução é reconhecidamente uma operação de difícil assimilação por parte do estudante. O aspecto matemático, integral de um produto, não traz dificuldades, mas a compreensão do fenômeno correspondente à operação matemática é difícil. Além desse fato, é impossível realizar-se uma Convolução sem computador quando um ou os dois sinais envolvidos são de natureza probabilística, não sendo, portanto, definidos por expressões analíticas.

A relação de causa e efeito, que pode ser explorada amplamente no uso de Sistemas Tutoriais, não resulta em compreensão se o estudante não tiver base teórica sobre a operação realizada. Pode ocorrer que o estudante "veja" o que acontece, mas não perceba "por que" acontece. É relevante, portanto, a leitura da teoria envolvida, e desejável que os elementos dessa teoria estejam disponíveis no próprio Tutorial, acessados facilmente para consulta, sem perda de continuidade no estudo.

O uso de Sistemas Tutoriais não visa substituir nenhum outro recurso, mas complementar o conjunto existente. O Tutsi, por exemplo, pode ser de grande utilidade se usado como preparação do estudante para as experiências em laboratório.

CAPÍTULO III

DOMÍNIO : ANÁLISE DE SINAIS

Tutsi realiza operações básicas da análise de sinais e serve de tutor na transmissão e sedimentação dos conceitos e técnicas envolvidos nessas operações.

As operações que podem ser efetuadas são:

- Geração de sinais aleatórios com distribuição de probabilidade escolhida pelo usuário;
- Geração de sinais determinísticos:
 - Trem de pulsos retangulares, com frequência, duração, e amplitude definidos pelo usuário;
 - Onda quadrada com número de harmônicas fixado pelo usuário;
- Levantamento das características estatísticas de sinais: média, variância, desvio padrão, função densidade de probabilidade, e autocorrelação;
- Obtenção da densidade espectral de potência de um sinal;
- Síntese de sinais, a partir das componentes espectrais obtidas da Análise de Fourier;
- Amostragem-e-retenção com taxa constante;
- Recuperação de um sinal, por interpolação, a partir de sua versão amostrada;
- Análise espectral;

- Convolução de seqüências finitas em sistema causal. Convolução linear e convolução circular; e
- Aplicação da convolução a sistemas de primeira ordem.

Evidentemente, outras operações da análise e processamento digital de sinais podem ser adicionadas em continuação a este trabalho.

Descrevem-se, a seguir, uma sucinta revisão das operações enfocadas, os algoritmos utilizados, e tecem-se considerações relevantes sobre a implementação.

3.1 CARACTERÍSTICAS ESTATÍSTICAS

Na análise de sinais, alguns sinais podem ser definidos por expressões matemáticas, como a senóide, a onda triangular e a onda quadrada. Estes são chamados de sinais "determinísticos". Contudo, outros sinais não podem ser definidos por expressões matemáticas, como o sinal de voz e o sinal de vídeo. Estes sinais são chamados de sinais "probabilísticos" e são estudados através de suas características estatísticas.

Consideram-se algumas dessas características, nas subseções seguintes.

3.1.1 MÉDIA

Seja $x(n)$ uma seqüência que representa um sinal elétrico dentro de um intervalo finito de N amostras. A média, μ , de $x(n)$ é calculada da seguinte forma:

$$\mu = \frac{\sum_{i=1}^N X_i}{N} \quad (3.1)$$

3.1.2 VARIÂNCIA

A variância, σ^2 , da seqüência $x(n)$ é calculada da seguinte forma:

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2 \quad (3.2)$$

3.1.3 DESVIO PADRÃO

O desvio padrão, σ , é a raiz quadrada da variância:

$$\sigma = \left[\sigma^2 \right]^{\frac{1}{2}} \quad (3.3)$$

3.1.4 FUNÇÃO DISTRIBUIÇÃO DE PROBABILIDADES

Sejam X uma variável aleatória, e x um número. Define-se, então, o evento $[X \leq x]$. Assim, a probabilidade desse evento, $P[X \leq x]$, depende do valor de x , ou seja, é uma função de x , e é representada por $F(x)$:

$$F(x) = P[X \leq x] \quad (3.4)$$

O domínio de $F(x)$ é o conjunto dos números reais, e seu contradomínio é o conjunto dos números reais no intervalo $[0,1]$. $F(x)$ é uma função não-decrescente em x , e tem as seguintes características:

$$\lim_{x \rightarrow \infty} F(x) = 1$$

$$\lim_{x \rightarrow -\infty} F(x) = 0$$

3.1.5 FUNÇÃO DENSIDADE DE PROBABILIDADE

A função densidade de probabilidade, fdp, é definida como a derivada da função distribuição, F(x):

$$fdp(x) = \frac{d F(x)}{dx} \quad (3.5)$$

Esta definição é utilizada no levantamento de fdp, a partir das amostras de um espaço amostral.

Para se levantar a fdp de um sinal contínuo é necessário amostrá-lo e quantizá-lo em amplitude, de forma a definir um número finito de amplitudes. Portanto, o que se faz é detetar os valores máximo e mínimo assumidos pela forma de onda e discretizar em amplitude essa faixa de valores. Desta forma, cada valor fica dentro de um sub-intervalo, e o número de amplitudes possíveis torna-se finito.

Uma vez limitado o número de valores de amplitude que o sinal pode assumir, trata-se cada ponto da curva como sendo amostras de um experimento, faz-se a contagem da ocorrência de cada amplitude e calcula-se sua probabilidade:

$$P_i = N_i / N \quad (3.6)$$

onde N_i é o número de vezes que a amplitude i ocorre;

N é o número total de amplitudes possíveis ou número de sub-intervalos em que foi dividido o sinal; e

P_i é a probabilidade de ocorrência da amplitude i .

O processo de discretização em amplitude é descrito a seguir:

- Determinam-se os valores máximo e mínimo que o sinal assume em sua forma de onda.
- Define-se o número de intervalos de amplitude em que será discretizado o sinal.
- Procede-se à contagem de ocorrências:

O intervalo a que pertence cada amostra do sinal é obtido da fórmula :

$$i = \frac{(X_i - V_{\min})(I_a - 1)}{V_{\max} - V_{\min}} + 1 \quad (3.7)$$

onde

i é o índice de cada intervalo de amplitude em que foi dividido o sinal;

X_i é o valor da amostra do sinal;

V_{\max} é o valor máximo de amplitude do sinal;

V_{\min} é o valor mínimo de amplitude do sinal; e

I_a é o número de intervalos de amplitude em que foi dividido o sinal.

Calcula-se a probabilidade de ocorrência de cada intervalo, dividindo o número de ocorrências pelo número total de pontos, segundo a equação 3.7.

De posse destas probabilidades individuais, obtém-se a função distribuição (Cumulativa), $F(x) = P(X \leq x)$ com a seguinte fórmula:

$$F(X_i) = P(X_i) + P(X_{i-1}) + \dots + P(1) \quad (3.8)$$

A função densidade de probabilidade, que é a derivada da função distribuição, é obtida derivando-se $F(x)$ ponto a ponto. O exemplo, a seguir, mostra a seqüência de passos para obtenção da contagem de ocorrências, função distribuição e função densidade de probabilidade.

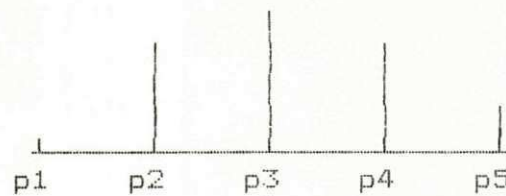


Fig. 3.1 Contagem de ocorrências

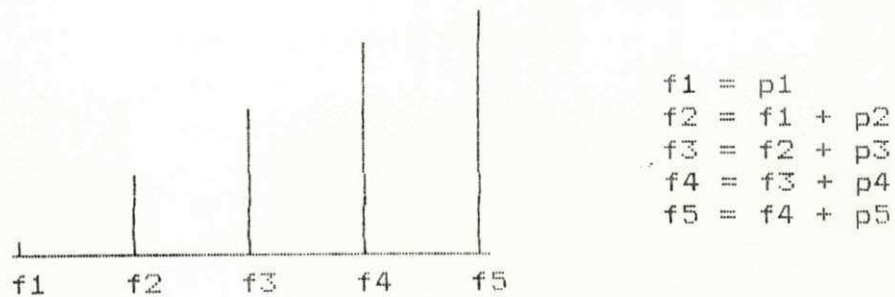


Fig. 3.2 Função distribuição de probabilidade (Função cumulativa)



Fig. 3.3 Função densidade de probabilidade

$$\begin{aligned} d_1 &= (f_2 - f_1) / \alpha \\ d_2 &= (f_3 - f_2) / \alpha \\ d_3 &= (f_4 - f_3) / \alpha \\ d_4 &= (f_5 - f_4) / \alpha \\ d_5 &= (f_6 - f_5) / \alpha \end{aligned}$$

onde o fator α é o intervalo de quantização;

$$f_6 = F(\infty) = 1; e$$

$$f = d(F(x)) / dx .$$

Quando se levanta a fdp de uma seqüência, com N amostras, a curva obtida apenas se aproxima da curva ideal, pois o número de amostras é limitado. Tal aproximação torna-se mais e mais precisa à medida que N tende para infinito.

3.1.6 FUNÇÃO AUTOCORRELAÇÃO

A função autocorrelação fornece uma indicação sobre as dependências estatísticas entre valores de um sinal. Expressa em gráfico, é útil para a detecção de periodicidades não percebidas na forma de onda do sinal. Se o sinal é estacionário podemos obter a densidade espectral de potência, a partir da transformada de Fourier da função autocorrelação (Teorema de Wiener-Kinchinne) [6].

Seja uma seqüência $x(n)$ como definido na seção 3.1.1.

A função autocorrelação[6] é definida pela expressão:

$$R_{xx}(k) = \frac{1}{N} \left[\sum_{n=0}^{N-1} x(n) \cdot x(n+k) \right] \quad (3.9)$$

onde N é o comprimento da seqüência $x(n)$;

n é o índice dos elementos da seqüência $x(n)$; e

k é o índice dos coeficientes de correlação;

k também identifica o número de deslocamentos da seqüência sobre si mesma, portanto k pertence ao intervalo $[0, N-1]$.

CARACTERÍSTICAS DE IMPLEMENTAÇÃO

Tutsi obtém características estatísticas de um

conjunto de amostras em dois módulos:

1. Calcula a Média, a Variância e o Desvio Padrão; faz o levantamento da fdp e traça seu gráfico;
2. Calcula a função Autocorrelação e traça seu gráfico.

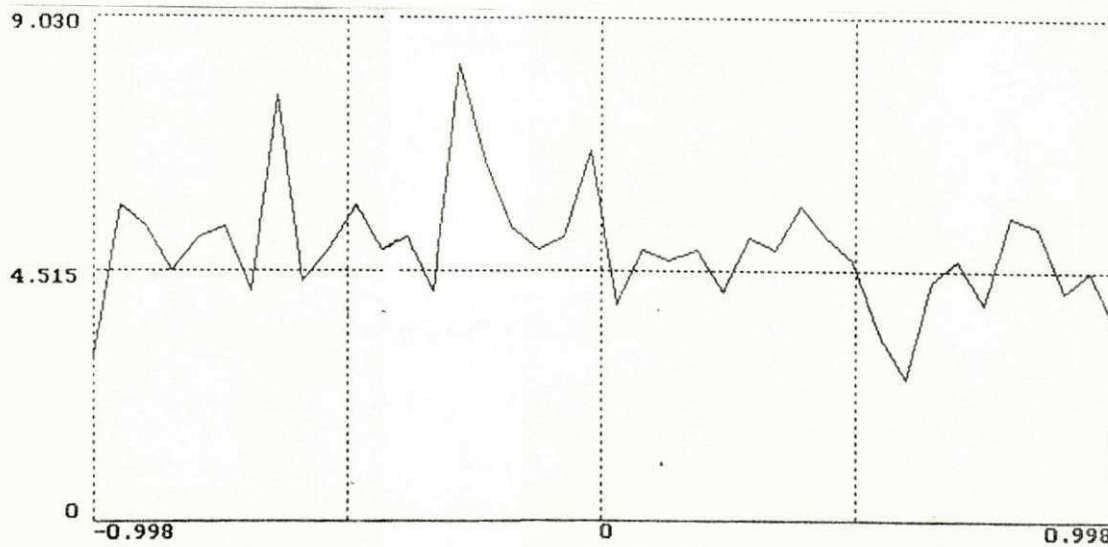


Fig. 3.4 Levantamento de fdp de uma seqüência com distribuição Uniforme

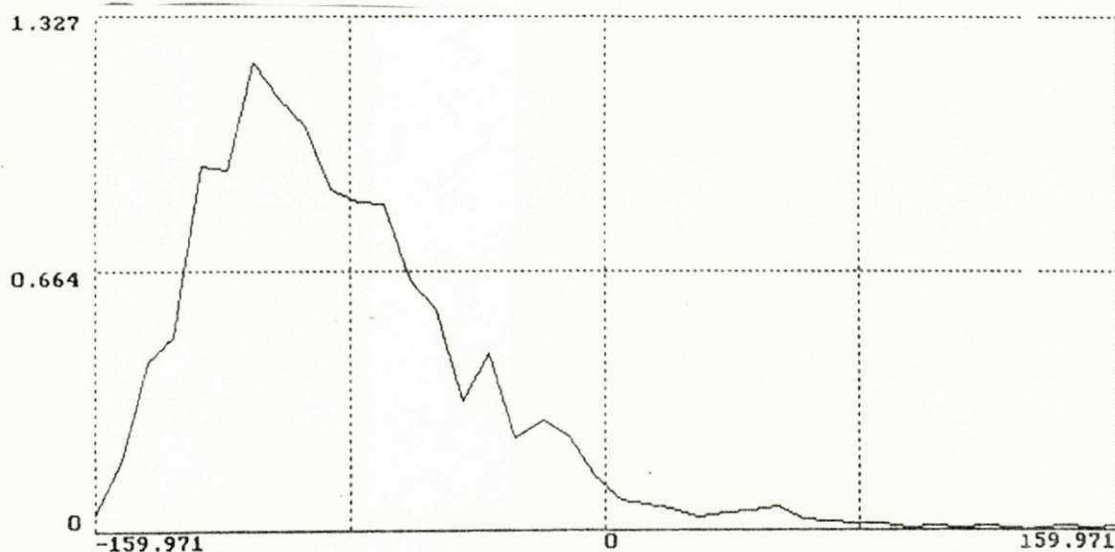


Fig. 3.5 Levantamento de fdp de uma seqüência com distribuição Gamma

A função autocorrelação pode ser normalizada em função da variância do sinal

$$R_{xx}(k) = R_{xx}(k)/R_{xx}(0) \quad (3.10)$$

assumindo valor máximo igual a 1.

PROPRIEDADES DA FUNÇÃO AUTOCORRELAÇÃO

I. $R_{xx}(k) = R_{xx}(-k)$

II. $\lim_{k \rightarrow \infty} R_{xx}(k) = 0$

3.2 GERADOR DE SEQUÊNCIAS ALEATÓRIAS

As seqüências geradas podem ter uma das seguintes distribuições de probabilidade:

- Uniforme;
- Exponencial (Laplace);
- Gaussiana (Normal);
- Gamma;
- Beta;
- Weibull;
- Qui-quadrado;
- Log-normal;
- Cauchy; e
- Rayleigh.

3.2.1 GERAÇÃO DE NÚMEROS PSEUDO-ALEATÓRIOS

Muitos tipos de geradores foram idealizados no

passado. Atualmente, porém, a geração de números aleatórios tem sido efetuada com eficiência pelo método congruencial linear [6].

A partir desse método, 3 tipos de geradores podem ser classificados: o multiplicativo, o aditivo, e o misto. Tais geradores utilizam expressões matemáticas que, aplicadas iterativamente, produzem uma seqüência de números, chamados de pseudo-aleatórios, pois são gerados de forma determinística.

Um gerador congruencial linear aplica a seguinte fórmula:

$$X_n = (C_1 \cdot X_{n-1} + \dots + C_j \cdot X_{n-j} + C_0) \text{ módulo } M \quad (3.11)$$

O método mais amplamente usado é o multiplicativo em que $j=1$, $C_0=0$, e X_n é o resto da divisão inteira de $C \cdot X_{n-1}$ por

M , ou

$$X_n = (C \cdot X_{n-1}) \text{ módulo } M$$

A escolha da constante C e do valor inicial X_0 deve garantir que a seqüência tenha um "período" máximo. "período" neste caso, refere-se ao maior comprimento de seqüência gerada sem repetição [7].

Para implementação em um computador cujo tamanho da palavra é igual a N , considera-se o valor de M igual a 2^N . A operação módulo pode ser feita simplesmente ignorando-se o "overflow" na palavra do computador.

Como o número resultante X_n é um inteiro, deve-se dividi-lo por M e obter-se um número real entre zero e 1.

O valor inicial é também chamado de "semente". O turbo Pascal fornece um gerador de números aleatórios cuja semente é obtida do clock do sistema provendo uma maior diversificação na geração das seqüências.

Tem-se, a seguir, um exemplo de uma sub-rotina em Fortran para o gerador multiplicativo:

```
FUNCTION RCM(IST)
RCM=IST*1220703125*0.291038305E-10
RETURN
END
```

onde IST é a semente;

1220703125 é o multiplicador ; e

o outro número é igual a $1 / M$.

Dada a simplicidade da sub-rotina, pode-se utilizar mais de um gerador em conjunto, como na figura 3.6, para melhorar a aleatoriedade das seqüências produzidas:

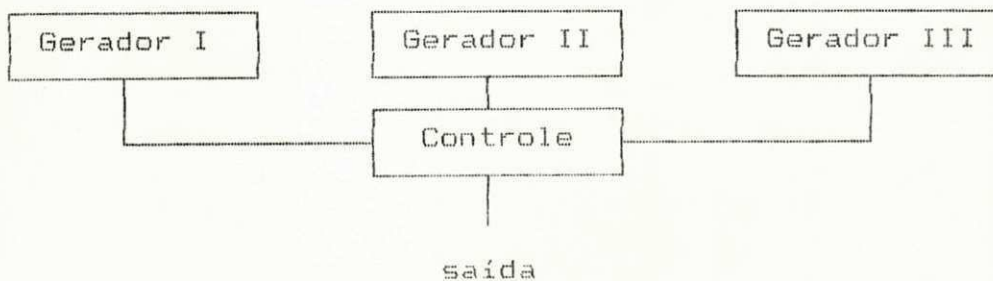


Fig 3.6 Diagrama de blocos de um gerador múltiplo de números pseudo-aleatórios

A rotina de controle seleciona um dos três geradores e envia o número gerado por ele para a saída. A quantidade de geradores pode ser qualquer; no entanto, com dois ou três, já se conseguem períodos bem mais longos e diminui-se o efeito das

tendências de cada gerador.

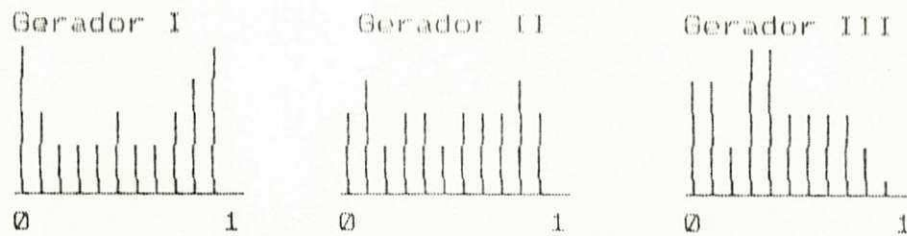


Fig 3.7 Distribuição de 3 geradores diferentes para 1000 números gerados.

A figura 3.7 mostra o histograma de geração de 3 geradores (Valor das amostras versus número de amostras geradas). Observa-se que o gerador I, tem maior probabilidade de gerar números próximos de zero e de 1. O gerador II é mais uniforme que os outros dois, e o gerador III tem baixa probabilidade de produzir números próximos de 1.

3.2.2 GERAÇÃO DE VARIÁVEIS ALEATÓRIAS

O gerador de números aleatórios produz uma seqüência de números no intervalo $[0,1)$ com distribuição uniforme. A partir destes, pode-se gerar seqüências com outras distribuições. Três métodos são utilizados [7] para esse fim:

- método da transformação inversa;
- método da rejeição; e
- método de composição ou mistura (dos dois métodos anteriores).

Dos métodos citados, o mais simples e que consome menos operações para ser computado é o método da transformação inversa. Além de não oferecer restrições ao seu uso, como o faz o

método da rejeição, seus resultados são tão precisos quanto os dos outros. Por estas razões, esse foi o método adotado neste trabalho.

MÉTODO DA TRANSFORMAÇÃO INVERSA

A função distribuição de probabilidades é definida no intervalo $[0,1]$. Pode-se, portanto, gerar números aleatórios distribuídos uniformemente no intervalo $[0,1)$ e considerar que cada um representa um valor de $F(x)$ (Fig 3.8).

Seja R o valor de cada amostra da seqüência do gerador, e x o valor da amostra cuja imagem pertence à função distribuição desejada $F(x)$.

Procede-se à igualdade

$$R = F(x) \quad (3.12)$$

O cálculo de x determina um valor aleatório que pertence a função distribuição desejada, e é feito aplicando-se a função inversa em ambos os lados da equação 3.12:

$$x = F^{-1}(R)$$

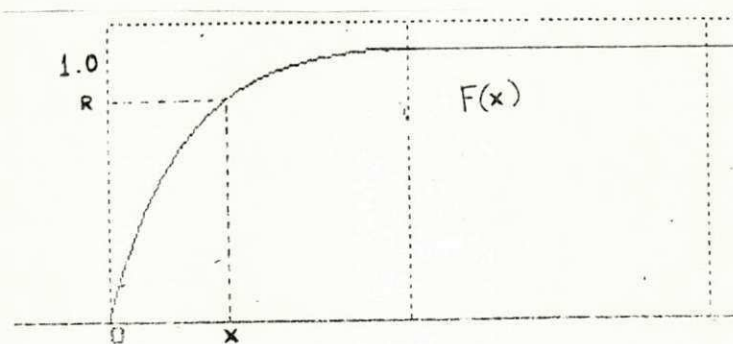


Fig 3.8 Função distribuição de probabilidades

é necessário que a expressão analítica de $F(x)$ seja conhecida, e que a mesma admita a existência de uma função inversa. Nem sempre isto acontece. Às vezes, a própria expressão de $F(x)$ não pode ser obtida a partir da integral da função densidade de probabilidade (fdp). Nestes casos, faz-se uma estimação de $F(x)$ através de valores tabelados da fdp.

Em [7], tem-se a obtenção das seguintes distribuições: Uniforme, Exponencial, Gamma, Qui-quadrado, Beta, Normal, Weibull e Lognormal. As distribuições de Rayleigh e Cauchy são definidas através de expressões analíticas, e, aplicando-se a técnica da transformação inversa, obtêm-se seqüências com estas distribuições. Descreve-se, a seguir, a obtenção da distribuição de Cauchy e da distribuição de Rayleigh, que não constam em [7]. A função distribuição de Cauchy é definida pela expressão:

$$F(x) = \frac{1}{2} + \frac{\text{arctg}(x-\mu)}{\pi} \quad (3.13)$$

Iguala-se $F(x)$ a R , onde R representa uma seqüência com distribuição uniforme, e obtém-se o valor de x .

$$x = \text{tg} \left(\pi \left(R - \frac{1}{2} \right) \right) + \mu$$

A distribuição de Rayleigh é obtida da seguinte expressão [8]:

$$x = \left[2a^2 \cdot \text{Ln} \left[\frac{1}{R} \right] \right]^{\frac{1}{2}} \quad (3.14)$$

onde R é um número pseudo-aleatório integrante de uma seqüência com distribuição uniforme [8], e x é um valor da seqüência de Rayleigh obtida.

3.3 SÍNTESE DE SINAIS

A análise de Fourier [9] demonstra que todo sinal periódico pode ser representado por uma soma de funções senoidais.

Seja $y(t)$ uma função periódica. Da análise de Fourier, pode-se escrever a função $y(t)$ da seguinte forma:

$$y(t) = A_0 + \sum_{n=1}^{\infty} A(n) \cdot \text{Sen}(n\omega t) + B(n) \cdot \text{Cos}(n\omega t) \quad (3.15)$$

Os termos $A(n)$ e $B(n)$ podem ser calculados segundo as equações abaixo:

$$A(n) = \frac{2}{T} \int_{-T/2}^{T/2} y(t) \cdot \text{Cos}(n\omega t) dt \quad (3.16)$$

$$B(n) = \frac{2}{T} \int_{-T/2}^{T/2} y(t) \cdot \text{Sen}(n\omega t) dt \quad (3.17)$$

Na equação 3.15, o termo A_0 é o valor médio de $y(t)$:

$$A_0 = \frac{1}{T} \int_{-T/2}^{T/2} y(t) dt \quad (3.18)$$

onde T é o Período de $y(t)$;

ω é a frequência fundamental de $y(t) = 2\pi f$; e

n é o índice do harmônico.

Para realização da síntese do sinal, é necessário que sejam calculados os termos senoidais, $A(n)$ e $B(n)$, e o valor médio, A_0 .

A síntese do sinal é feita efetuando-se o somatório da equação 3.15. Por se tratar de soma de senóides com frequências diferentes, esta tarefa torna-se longa e cansativa. Para evitar esforço humano desnecessário, este cálculo deve ser realizado por computador, o qual dispõe ainda de recursos para traçar o gráfico da curva sintetizada.

3.4 SINAIS DETERMINÍSTICOS

Os sinais determinísticos desempenham um papel fundamental na análise de sinais. São definidos por expressões analíticas e seu processamento pode tornar-se mais simples. Servem para exemplificar situações e resultados em sessões de estudo, podendo ser manipulados analiticamente, sem necessariamente precisar-se de um método numérico para obtenção dos resultados.

São exemplos de sinais determinísticos: onda quadrada, senóide, dente de serra, onda triangular, pulso retangular, etc. Todos estes sinais podem ser obtidos utilizando-se as operações disponíveis em Tutsi. Descreve-se, a seguir, os sinais que Tutsi gera diretamente sem operações intermediárias.

3.4.1 ONDA QUADRADA

A onda quadrada é um sinal periódico de grande importância na análise de sinais. Sua representação por série de Fourier torna possível a observação do Fenômeno de Gibbs [10]. Gibbs constatou que a série truncada de Fourier não converge na

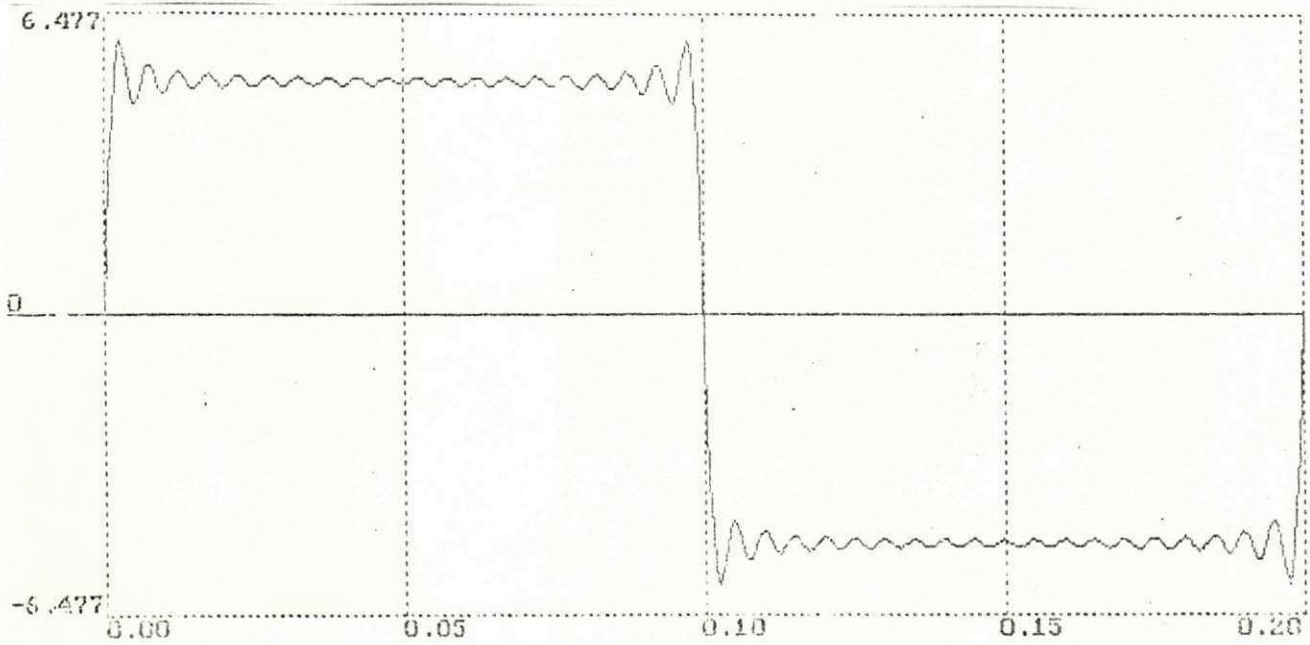
vizinhança da descontinuidade de um sinal, e que o pico do "ripple" apresentado não parece diminuir com o acréscimo de termos na série.

3.4.1.1 CARACTERÍSTICAS DE IMPLEMENTAÇÃO

Com base na Análise de Fourier, Tutsi realiza a síntese de uma onda quadrada a partir de componentes senoidais. Neste caso, Tutsi não requer os coeficientes, mas apenas o número de harmônicas. O tempo de processamento depende do número de harmônicas desejado, podendo ser exaustivo se o usuário escolher uma quantidade muito grande destas, por volta de 100. No entanto, o sistema alerta o usuário para tal fato. Há também um limite imposto pela variável que armazena o número de harmônicas, se esse limite for ultrapassado o dado é rejeitado, para evitar um erro de execução, e solicitado novo dado.

A síntese da onda quadrada com apenas 1 harmônico resulta numa Senóide.

A figura 3.9 mostra uma onda quadrada com 20 harmônicos, na qual pode-se observar o fenômeno de Gibbs.



Síntese da onda quadrada. Número de harmônicas: 20

Nome do arquivo: oq.tut

Número de pontos: 500

Tempo (eixo horizontal) expresso em MILISSEGUNDOS

Valores de amplitude multiplicados por 10^0

Frequência fundamental (kHz): 5.000

BIBLIOTECA DE PESQUISA DE INFORMÁTICA - UFRJ - PÓS-GRADUADO EM ENGENHARIA DE SISTEMAS DE COMPUTAÇÃO

Fig 3.9 Onda quadrada com 20 harmônicas, apresentando o fenômeno de Gibbs

3.4.2 PULSO RETANGULAR

O pulso retangular é um sinal determinístico de grande importância para a análise de sinais, pois contém em sua forma de onda e em seu espectro, características da informação que transita nos sistemas de comunicação digital.

Abaixo, vê-se um trem de pulsos retangulares e a descrição de seus parâmetros.

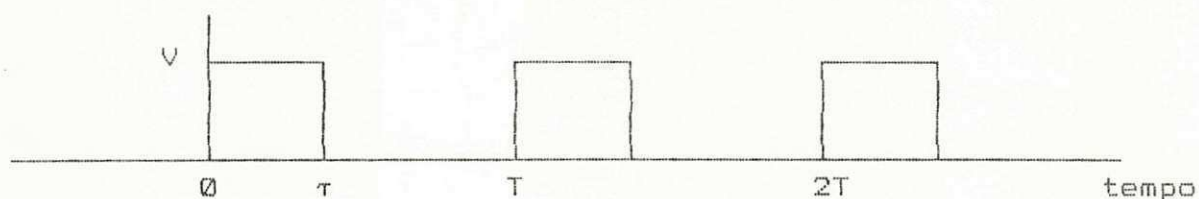


Fig 3.10 trem de pulsos.

onde τ é a duração do pulso;

T é o período do pulso;

V é a amplitude do pulso; e

a relação τ/T é chamada duty cycle.

3.4.2.1 CARACTERÍSTICAS DE IMPLEMENTAÇÃO

Tutsi gera um período de um trem de pulsos retangulares. Na implementação da rotina que gera esse sinal, há um fator limitante que deve ser considerado. Trata-se do valor mínimo do duty cycle. Esse valor relaciona-se com o número de pontos do sinal, do seguinte modo:

Seja um sinal com N pontos. O valor mínimo do duty cycle é $1/N$. Isto corresponde a um pulso com uma (1) amostra

não-nula, e N-1 amostras nulas. Se o duty cycle ultrapassar esse limite, não há como representar o pulso com esse número de pontos.

Tutsi critica os dados fornecidos pelo usuário de forma a obedecer a esta imposição. No Help há orientações sobre a entrada de dados desta operação.

3.5 ANÁLISE ESPECTRAL

A análise espectral pode ser realizada com eficácia através da Transformada de Fourier, que é uma representação, no domínio da frequência, de um sinal temporal. Para o caso de sinais contínuos, esta Transformada é definida pelas expressões abaixo[9].

$$\text{Transformada Direta: } X(\omega) = \int_{-\infty}^{\infty} x(t) \cdot e^{-j\omega t} dt \quad (3.19)$$

$$\text{Transformada Inversa: } x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(\omega) \cdot e^{j\omega t} d\omega \quad (3.20)$$

A Transformada de Fourier pode ser entendida como a superposição de sinais exponenciais, como mostra a própria expressão que a define. Neste caso, porém, diferente da Série de Fourier, $X(\omega)$ é um sinal contínuo na frequência, que não requer que o sinal $x(t)$ seja periódico, como acontece na Série de Fourier. No entanto, faz-se necessário que o sinal tenha energia finita para que a integral convirja.

3.5.1 FFT

A obtenção da transformada de Fourier de um sinal a partir da expressão de sua definição é um processo lento. Porém, existem diversos algoritmos para o cálculo da transformada discreta de Fourier (DFT) que podem ser utilizados, uma vez que se processam sinais contínuos a partir de suas amostras armazenadas em arquivo. Cada ponto do arquivo corresponde a uma amostra, e o arquivo todo a uma seqüência finita.

Os muitos algoritmos existentes para obtenção da DFT têm seu desempenho otimizado em aplicações específicas. O sistema especialista FFTEX [11] seleciona um, entre vários algoritmos, de acordo com a precisão requerida pelo usuário, o tempo de processamento, e a quantidade de memória ocupada.

Para o Tutsi foi selecionado o algoritmo Fourea, de Cooley-Tukey [12], que tem as seguintes características:

- Opera com dados complexos, em uma (1) dimensão, cujo comprimento (N) deve ser uma potência de base 2;
- Transforma direta e inversamente;
- O tempo de execução é proporcional ao produto $N \cdot \log_2 N$;
- Constitui um dos mais simples algoritmos conhecidos; e
- Sua precisão é da mesma ordem de grandeza de outros algoritmos mais complexos, na maioria dos casos [11].

O Fourea originalmente escrito em Fortran padrão, foi traduzido para a linguagem Pascal, para integrar o ambiente do Tutsi.

Embora esse algoritmo se restrinja a dados de entrada cujo comprimento é uma potência de base 2, Tutsi contorna esta restrição por alongar o comprimento de um vetor de dados que não atenda inicialmente a essa restrição, acrescentando ZEROS no final do arquivo, até que o número de pontos seja compatível com Fourea. Tal procedimento não altera a análise espectral, pois as componentes espectrais existentes não sofrem alteração pelo acréscimo de um trecho nulo [5]. O espectro de um sinal de duração T e amplitude zero é nulo ao longo do eixo das frequências.

3.5.1.1 CARACTERÍSTICAS DE IMPLEMENTAÇÃO

Para que Tutsi possa traçar o espectro do sinal, com a unidade de frequência em Hz, é necessário que o usuário informe o intervalo de tempo ocupado pelo seu sinal. Assim, é possível calcular a frequência fundamental e esboçar os valores corretos no eixo horizontal do espectro. Considere o exemplo abaixo: Seja uma seqüência Senoidal com 8 pontos, que ocupa 10 ms no tempo (Dado informado pelo usuário).



Fig 3.11 seqüência senoidal com 8 pontos, 10 ms.

Calcula-se sua frequência fundamental, $f_0 = 1/10 = 0.1$ kHz. A FFT gerará um vetor de saída com 8 pontos, porém apenas 4 são necessários para se expressar seu módulo, porque a FFT produz um

gráfico do módulo que é simétrico em relação ao eixo vertical. Os valores no eixo horizontal do espectro tornam-se então:

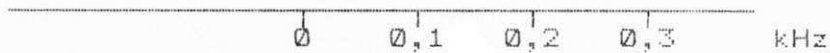


Fig 3.12 Valores da freqüência no eixo horizontal do espectro.

Uma mensagem na tela informa sobre a unidade de freqüência utilizada. As figuras 3.14 e 3.15 apresentam exemplos de obtenção de Espectros, efetuadas por Tutsi.

3.6 LIMITAÇÃO DE BANDA

Esta operação é indispensável à realização de amostragem sem distorção. Todo processo de amostragem tem de obedecer ao teorema de Nyquist, para que o sinal original possa ser recuperado sem distorção, a partir de suas amostras. O teorema de Nyquist requer que o espectro do sinal seja limitado. Isto é conseguido através de filtragem passa-baixas ou passa-faixa, efetuada no sinal que será amostrado.

3.7 AMOSTRAGEM-E-RETENÇÃO

Um sinal contínuo no tempo pode ser representado por amostras igualmente espaçadas. Se se obedece a certas condições, a versão discreta no tempo obtida guarda em si todas as características do sinal contínuo, de modo que é possível se recuperar o sinal contínuo a partir das amostras da versão discreta. As condições que devem ser satisfeitas para que isto

ocorra é o que diz o teorema de Nyquist:

Seja $x(t)$ um sinal cujo espectro é limitado.

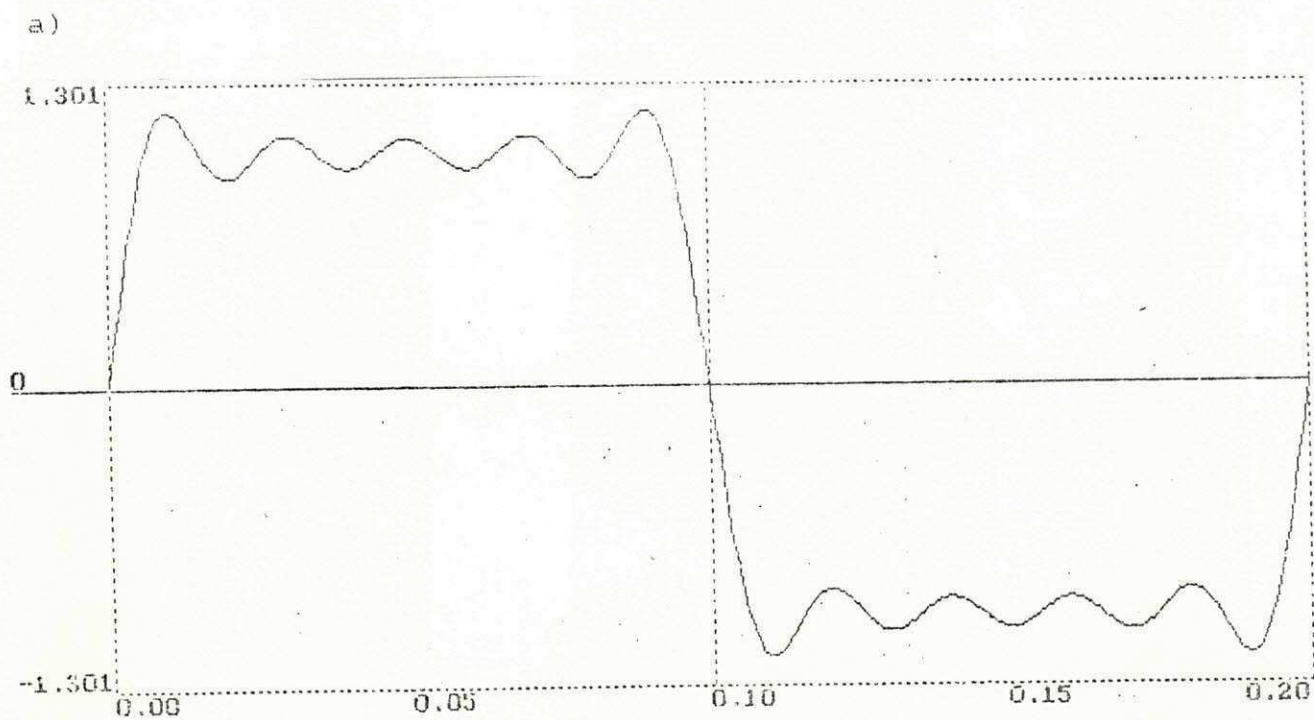
$X(\omega) = 0$, para $|\omega| > \omega_m$.



Então $x(t)$ é unicamente representado por suas amostras $x(nT)$, $n = 0, \pm 1, \pm 2, \dots$, se $\omega_s \geq 2 \omega_m$, onde

$$\omega_m = \frac{2\pi}{T} \quad (3.21)$$

ω_s é chamado "frequência de amostragem".



b)

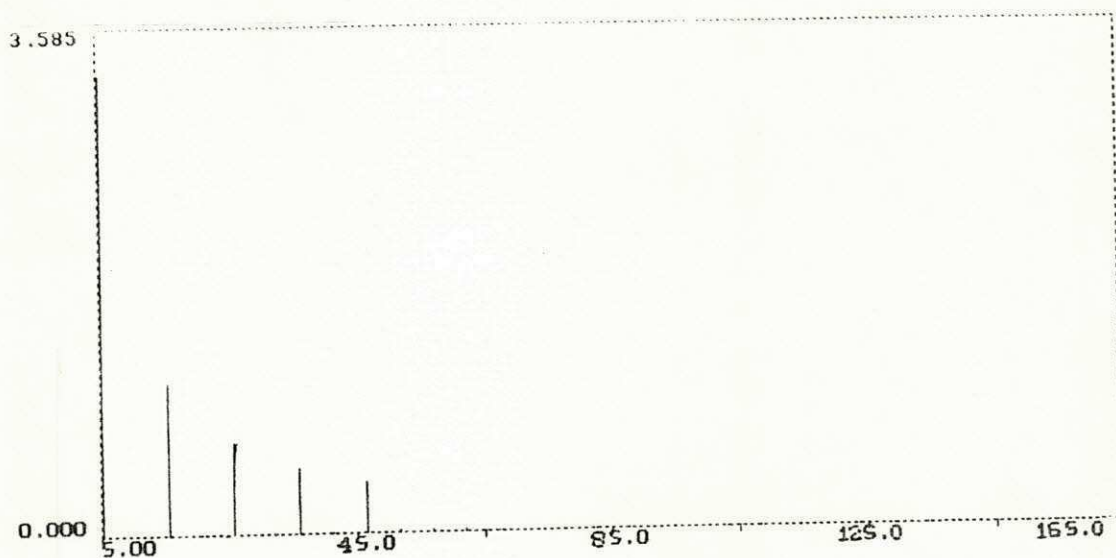


Fig 3.14 a) Onda quadrada com 5 harmônicos. b) Espectro da onda quadrada

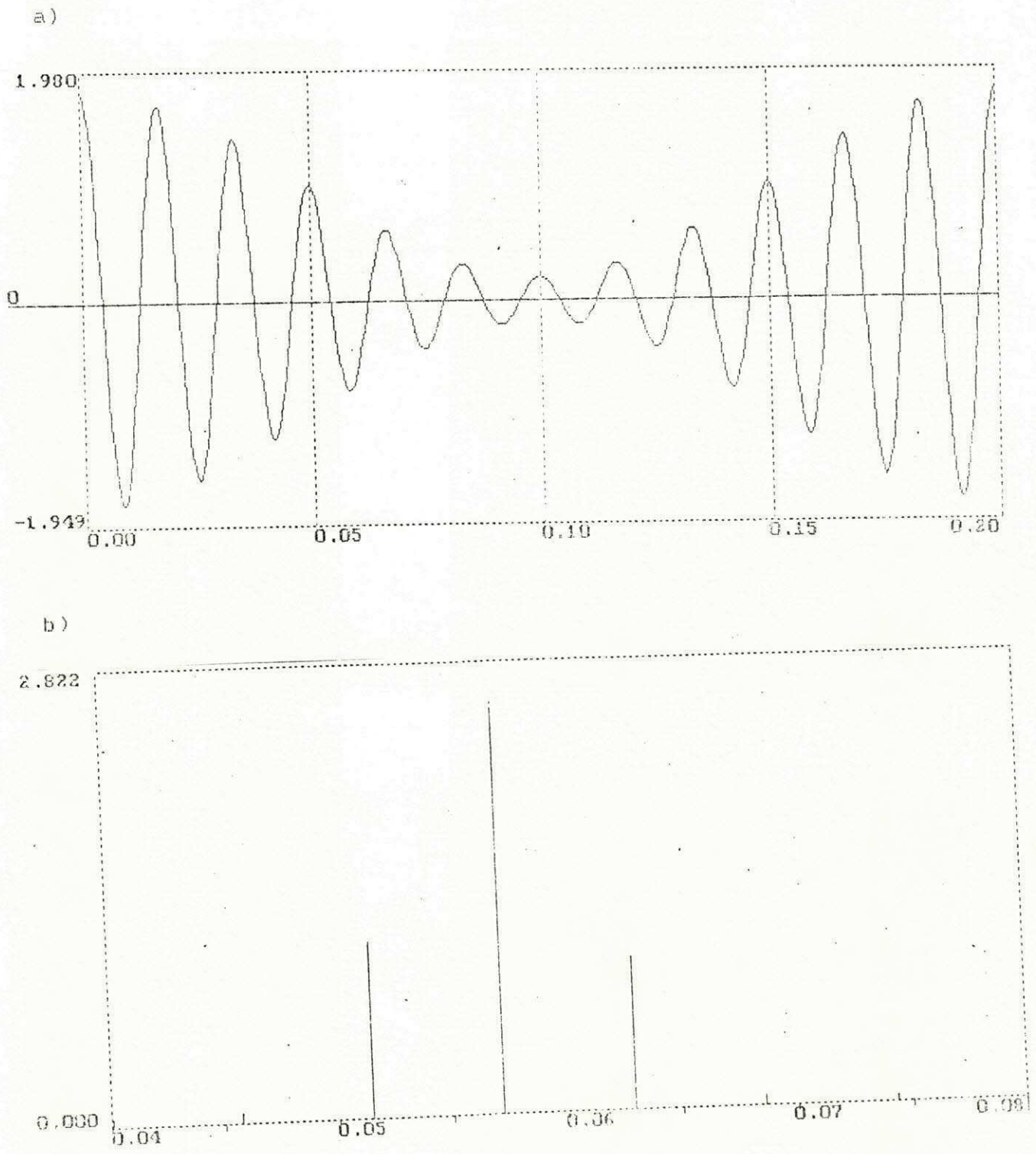


Fig 3.15 a) Sinal AM-DSB. b) Espectro do sinal AM-DSB

3.7.1 AMOSTRAGEM INSTANTÂNEA

A amostragem instantânea é semelhante ao produto de um sinal qualquer $x(t)$ com um trem de impulsos unitários [13], como está mostrado na figura 3.16.

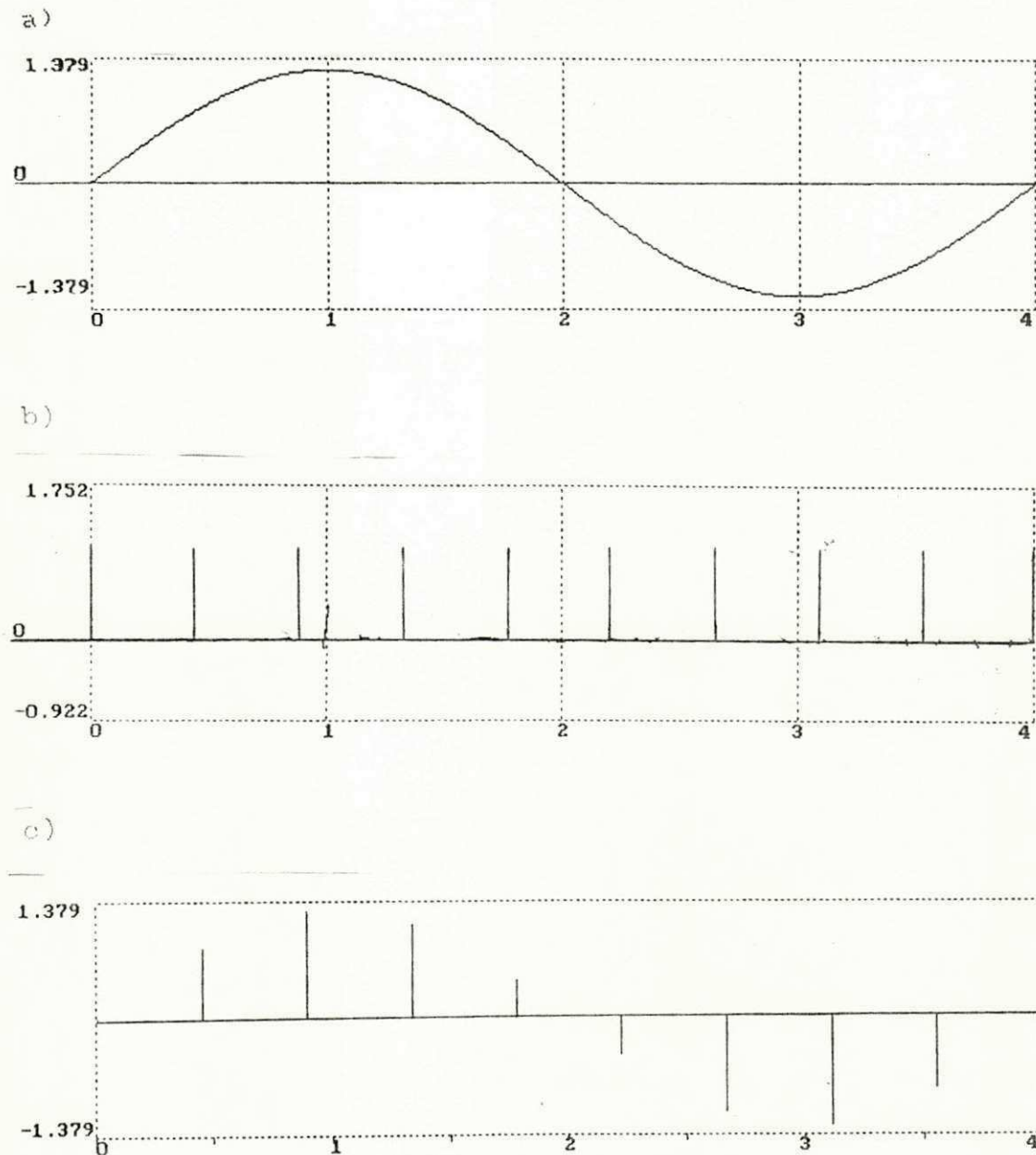


Fig 3.16 a) Sinal $x(t)$ b) trem de impulsos $p(t)$ c) Sinal amostrado $p(t).x(t)$

3.7.2 AMOSTRAGEM NATURAL

A amostragem natural corresponde a se manter constante a amplitude do sinal durante certo tempo, chamado período de amostragem [13]. Este tipo de amostragem é o que é feito na prática, uma vez que o trem de impulsos necessário à amostragem instantânea é muito difícil de ser gerado.

A amostragem-e-retenção, figura 3.16, classifica-se como uma amostragem natural e foi escolhida para ser realizada neste trabalho por ser a mais amplamente utilizada.

3.7.3 CARACTERÍSTICAS DE IMPLEMENTAÇÃO

Para evitar "overflow" nas variáveis do software, limita-se a frequência de amostragem ao intervalo que vai de 1 Hz a 250 MHz, acreditando-se que essa gama de valores atenda ao usuário. Qualquer valor fora desse intervalo é rejeitado como dado inválido e o usuário, avisado por uma mensagem.

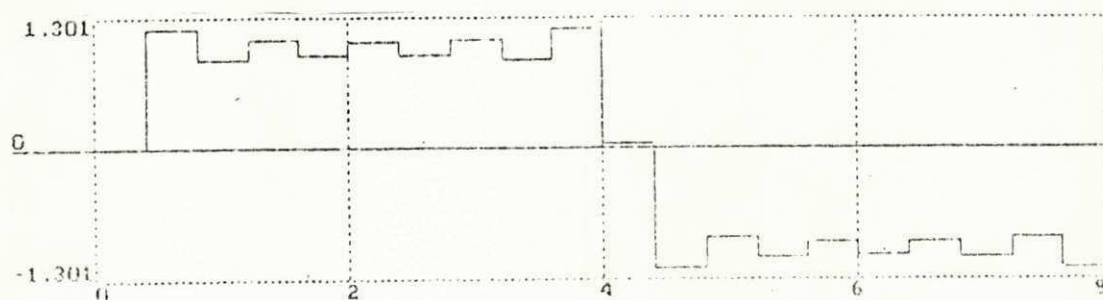


Fig.10 Amostragem-e-retenção

3.7.4 ALIASING

Se um sinal é amostrado sem obedecer ao teorema de Nyquist, ocorre no espectro do sinal uma superposição de raias espectrais, de forma a modificar o espectro original, impedindo a recuperação do sinal no tempo, sem distorção, a partir de sua versão amostrada.

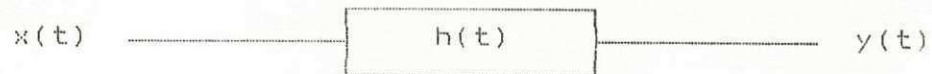
Um exemplo da importância desse fenômeno se vê em cenas de televisão, em que se apresentam ao olho humano 30 quadros por segundo. Esta taxa obedece ao teorema de Nyquist; por isso o espectador tem a noção exata do movimento contínuo da imagem. Caso se diminua tal taxa, chega-se ao ponto de o olho humano, que funciona como filtro passa-baixas, não conseguir recuperar o sinal original sem distorção.

3.8 RECUPERAÇÃO DO SINAL ORIGINAL POR INTERPOLAÇÃO

A interpolação corresponde à filtragem passa-baixas, operação realizada para a recuperação do sinal original a partir da versão amostrada. Isto pode ser conseguido da mesma maneira que foi explanada na operação LIMITAÇÃO DE BANDA.

3.9 CONVOLUÇÃO

Considere um sistema linear invariante no tempo, cuja resposta ao impulso é representada pela função $h(t)$, sendo $x(t)$ o sinal de entrada e $y(t)$, o sinal de saída, como mostra o diagrama de blocos, a seguir.



A relação entre os sinais de entrada, $x(t)$, e de saída, $y(t)$ é definida pela integral abaixo:

$$y(t) = \int_{-\infty}^{\infty} x(u) \cdot h(t-u) du \quad (3.22)$$

A expressão acima pode ser também representada da seguinte forma $y(t) = x(t) * h(t)$, onde a operação entre x e h é chamada de CONVOLUÇÃO.

A Convolução é uma operação fundamental em processamento digital de sinais. A implementação de um filtro digital é comumente feita através da convolução entre o sinal de entrada do filtro e sua resposta ao impulso. Em estimação espectral, a operação básica da Autocorrelação é obtida por convolução entre o sinal e uma versão simétrica no tempo do próprio sinal.

Quando se trata de sinais discretos a operação de convolução assume a seguinte expressão:

$$y(n) = \sum_{-\infty}^{\infty} x(m) \cdot h(n-m) \quad (3.23)$$

Observe-se, na expressão acima, o somatório de infinito a infinito, condição que não pode ser satisfeita em métodos computacionais onde só se trabalha com seqüências finitas ou com sinais limitados no tempo (Sinais com energia finita). Estes limites, portanto, são alterados para valores finitos.

3.9.1 CONVOLUÇÃO LINEAR

A Convolução linear corresponde à operação em que os sinais têm energia finita. Portanto embora se opere no intervalo $(-\infty, \infty)$, de fato o que se faz é assumir que os sinais são NAO NULOS apenas dentro de um intervalo $[t_1, t_2]$.

III.9.2 CONVOLUÇÃO CIRCULAR

A Convolução Circular considera que os sinais são periódicos com períodos iguais [14], de forma que o deslocamento de um sinal sobre o outro, que ocorre na Convolução, leva em conta que o sinal não é nulo após o período considerado. Mostra-se a seguir um exemplo em que um trem de pulsos, sinal X, de período igual a 2 ms, convolui consigo mesmo.

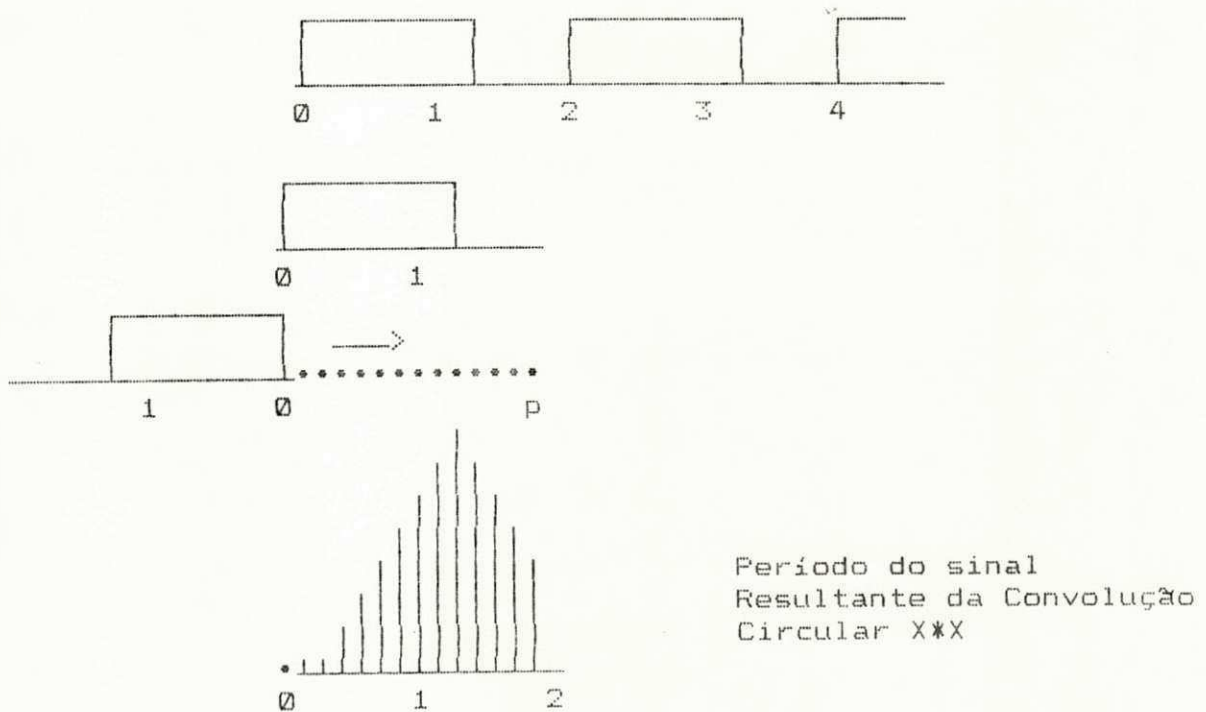


Fig 3.18 Exemplo de convolução circular.

Observe-se que o deslocamento ocorre até o ponto p , pois, daí em diante, o sinal X inicia novo período. Observe-se também que, ao se atingir o ponto p , o valor do produto dos sinais superpostos não é zero; portanto, a Convolução não converge para zero, assim como sempre ocorre na Convolução Linear de sinais limitados no tempo.

A convolução é uma operação de difícil assimilação. Demonstrada em aula convencional, não é compreendida com profundidade. Os livros apresentam exemplos envolvendo sinais que podem ser expressos por funções matemáticas. No entanto, como realizar convolução envolvendo processos estocásticos, os quais não podem ser representados por uma expressão analítica? Uma solução viável é armazenar os dados em um arquivo de dados e fazer a convolução utilizando um algoritmo apropriado. Não é possível efetuar este tipo de convolução usando outro meio que não o computador.

Embora a convolução possa ser definida, no tempo, de infinito a infinito, este não pode ser o caso no computador. Portanto, considera-se que os sinais envolvidos sejam limitados no tempo e em sistema causal (nulos para $t < 0$).

O algoritmo aqui utilizado não se aplica ao processamento em tempo real, pois o processo só inicia depois de os dois sinais estarem gravados em arquivo.

Um fator muito importante a ser considerado é a frequência com que foram amostrados os sinais convolventes. No caso de estas serem diferentes, o sistema detecta isto e promove

uma nova amostragem nos sinais, de maneira a tornar coincidentes, no tempo, as amostras de ambos, pois o deslocamento que um sinal efetua sobre o outro corresponde a se superpor, a cada deslocamento os dois sinais (figura 3.19).

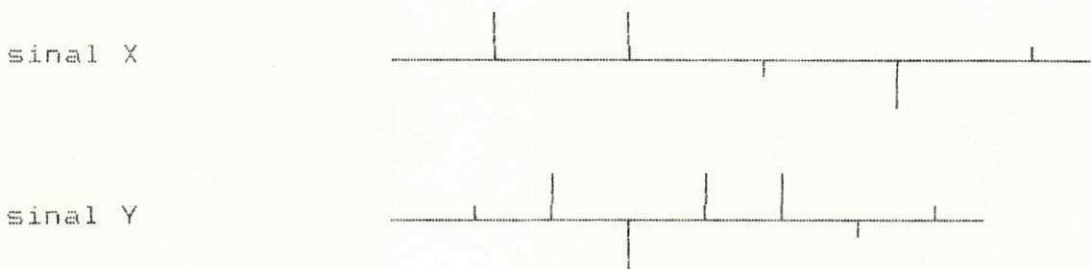


Fig 3.19 Sequências com frequências de amostragem diferentes.

A amostragem aludida é feita da seguinte forma: primeiro, determina-se que sinal deve ser amostrado. Tomando como exemplo os sinais X e Y acima, o sinal a ser amostrado é o sinal X, pois $T_x > T_y$. O período de amostragem é calculado por:

$$T = T_y / (N_y - 1) \quad (3.24)$$

O novo número de pontos que o sinal X assumirá é calculado por:

$$N_x' = (T_x / T) + 1 \quad (3.25)$$

onde T_x é o intervalo de tempo ocupado pelo sinal X;

T_y é o intervalo de tempo ocupado pelo sinal Y;

T é o intervalo de tempo entre duas amostras de Y;

N_y é o número de pontos de Y; e

N_x' é o novo número de pontos de X com o acréscimo das amostras.

Em seguida, interpolam-se amostras ao sinal X pela média de duas amostras vizinhas. Após realizada a interpolação, o

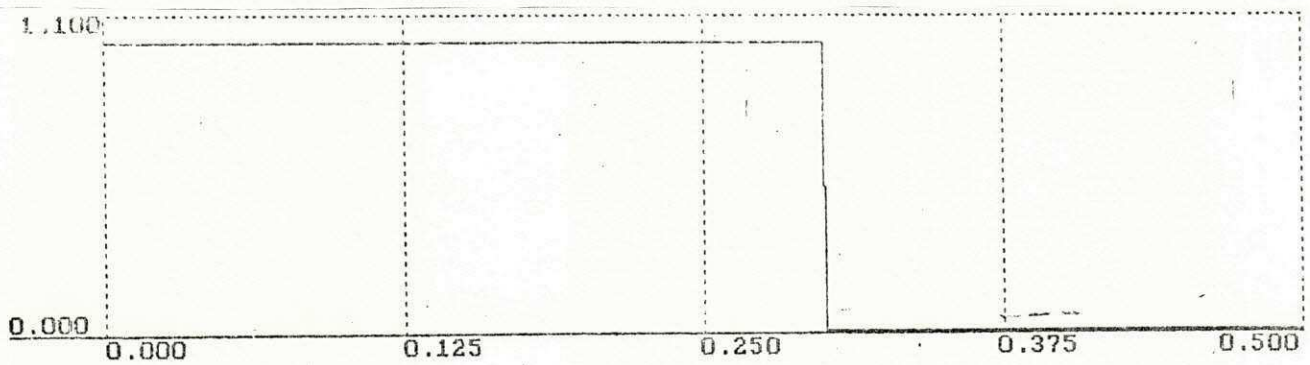
novos número de pontos de X é $N = 2.N_x - 1$. Quando N ultrapassa o número de pontos (N_x') calculados para o sinal, então procede-se à quantização deste de forma que o intervalo de tempo entre duas amostras no sinal X resulta igual ao do sinal Y .

Na figura 3.20 vê-se um exemplo de Convolução Linear realizada por Tutsi, apresentando os dois sinais convoluentes e a convolução resultante.

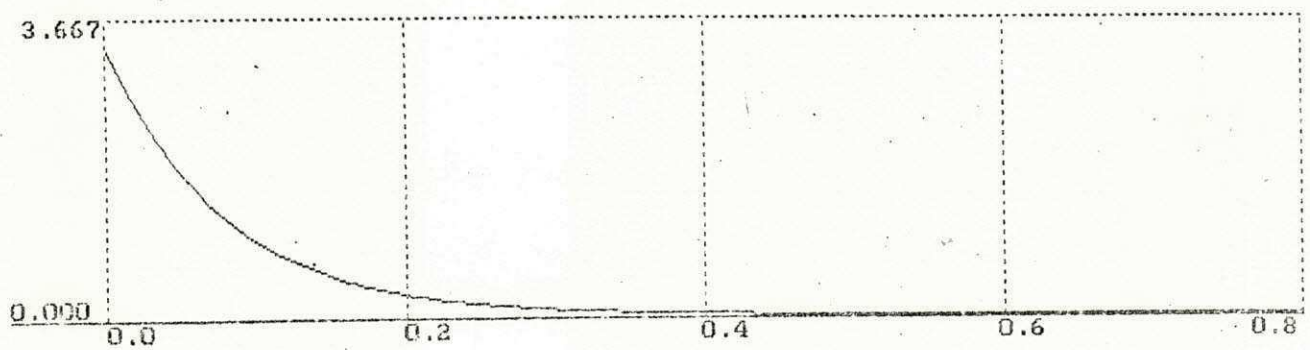
Outro fato a se ressaltar é que há uma sutil diferença entre convolução de sinais contínuos e de sinais discretos, nas expressões que as definem (equações 3.22 e 3.23).

é evidente que a integral não é a mesma coisa que o somatório, pois o somatório não leva em conta os valores do eixo horizontal que por sua vez é imprescindível no cálculo da área na integração. A implementação de um somatório é bem mais simples que a de uma integral, além de consumir muito menos tempo de processamento. Se se usar, para sinais contínuos, o somatório em vez de integral, ter-se-á como resultado uma forma de onda correta com valores de amplitude errados. Portanto, quanto à forma de onda, os dois casos são semelhantes. Por isso Tutsi chama sua operação de "convolução de seqüências finitas" e não convolução de sinais contínuos; embora o usuário possa realizar a convolução de sinais contínuos, limitados no tempo, em sistema causal, obtendo a forma de onda correta, desconsiderando porém os valores verticais.

a)



b)



c)

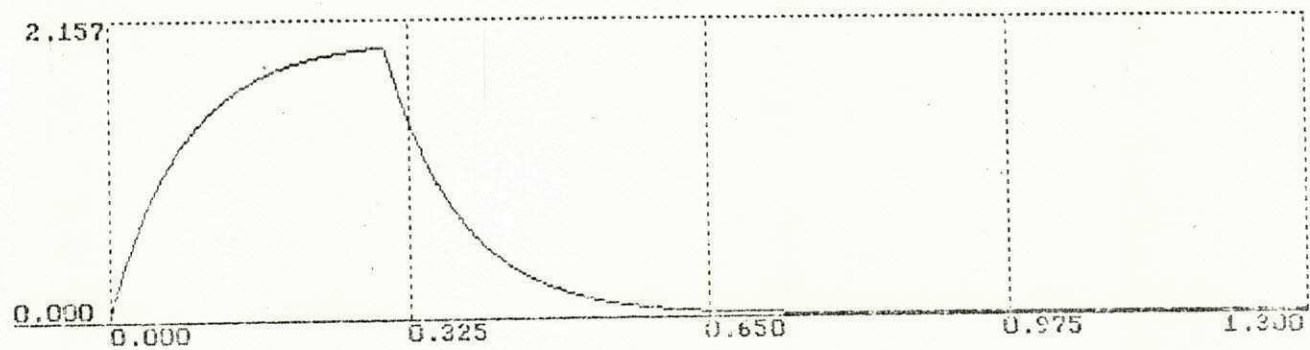


Fig 3.20 a) Sinal X b) Sinal Y c) Convolução linear de X e Y

3.10 SISTEMAS DE 1ª ORDEM

Considere-se um sistema linear invariante no tempo, cujo sinal de entrada é $x(t)$ e cujo sinal de saída é $y(t)$. Trata-se de um sistema de 1ª ordem, quando o mesmo é descrito pela seguinte equação:

$$\tau \frac{dy(t)}{dt} + y(t) = x(t) \quad (3.26)$$

Sua resposta ao impulso é dada pela expressão:

$$h(t) = \frac{1}{\tau} e^{-t/\tau} U(t) \quad (3.27)$$

A Constante τ é chamada de "Constante de tempo", porque seu valor influi na rapidez de decaimento da exponencial na equação 3.27. Sua resposta em frequência é dada por:

$$H(w) = \frac{1}{1 + jw\tau} \quad (3.28)$$

Os filtros Butterworth de 1ª ordem são exemplos de Sistemas de 1ª ordem. São utilizados na operação Limitação de Banda, necessária a amostragem segundo Nyquist.

Tutsi gera $h(t)$, sendo a constante de tempo, τ , expressa em ms, fornecida pelo usuário. Este sinal é uma função exponencial nula para $t < 0$, que tende para zero quando t tende para infinito. Como o número de pontos do arquivo que guardará este sinal é finito, considera-se que $h(t) = 0$ para $t = 10\tau$.

Na figura 3.20, o sinal Y é um exemplo da resposta ao impulso de um sistema de 1ª ordem.

CAPÍTULO IV

TUTSI - INTERFACE

INTRODUÇÃO

A forma de apresentação de um software e a maneira como ele interage com o usuário é muito importante para sua aceitação e disseminação de seu uso. Uma interface ruim pode causar a rejeição de um software, às vezes até sem uma avaliação mais profunda de sua capacidade. Portanto, alguns princípios básicos devem ser considerados ao se projetar a interface com o usuário.

No diálogo com o usuário, deve-se evitar o uso de jargões profissionais da área de computação, termos desconhecidos, neologismos e ambigüidades [15].

Fornecer mensagens de erros claras e objetivas, bem como opções de menu que bem representem o recurso disponível.

CONTROLE DA EXECUÇÃO

Na utilização do software, o usuário deve ter o controle da execução. O processamento de alguma função só deve ocorrer quando requerido por ele. Muito útil também é prover uma

hierarquia de funções, de modo que o usuário ainda não adaptado ao software possa fazer um reconhecimento, percorrendo livremente todos os blocos funcionais do sistema, podendo retornar a qualquer momento ao ponto de onde partiu, e podendo cancelar a qualquer momento uma operação iniciada.

A cada ação do usuário deve corresponder uma resposta do sistema. Se o tempo de resposta for muito longo, sem que o usuário saiba o que está acontecendo, é comum o usuário duvidar da máquina ou do programa.

ROBUSTEZ

Deve-se dar atenção também a ações "perigosas" do usuário, como encerrar o programa indevidamente ao apertar a tecla de saída, apagar um arquivo ou desenho da tela, passar dados inválidos, tentar usar diretório inexistente, etc.

Por fim, deve-se evitar o congestionamento da tela, com quantidade excessiva de mensagens, avisos ou gráficos.

4.1 INTERFACE TEXTO

Chamamos de interface texto a parte da interface do Tutsi que opera no modo texto. Compõem esta interface os menus, o help, as janelas de entradas de dados, as janelas de consulta ao Material Instrucional (Teoria, Exercícios, Exemplos, Referências) e as janelas de execução de funções do DOS.

As operações do Tutsi são oferecidas através de menus, onde a navegação pode ser feita livremente, inclusive para

reconhecimento dos recursos oferecidos, sem necessariamente ser feita a ativa ao de alguma fun ao. O usu rio pode-se aprofundar na  rvore de menus at  as janelas de entrada de dados, que est o no  ltimo n vel antes da execu ao, com a facilidade de retornar a qualquer instante na sub- rvore percorrida, concluindo ou cancelando a tarefa corrente.

Os sinais envolvidos e o resultado das opera es s o gravados em arquivos.

Todo resultado pode ser visto em forma gr fica, pois Tutsi conta com sa da gr fica para tela e para impressora.

Tutsi prev  alguns erros que o usu rio pode cometer, que s o tratados com subsequente mensagem ao usu rio para informa ao e poss vel recupera ao do erro, de forma a evitar fim anormal da execu ao sem conhecimento das causas por parte do usu rio.

Tutsi realiza algumas tarefas do DOS pertinentes  s sess es de estudo, tais como listagem de diret rio, mudan a de diret rio, apagamento de arquivo e renomea ao de arquivo. Esta facilidade evita que o usu rio precise sair do ambiente do Tutsi para manipular os arquivos em que est o seus sinais e resultados de opera es.

Tutsi oferece Help e Material Instrucional sens veis ao contexto, dispon veis a qualquer instante. O Material Instrucional traz um resumo da teoria envolvida na opera ao corrente, exemplos, exerc cios e refer ncias bibliogr ficas para consulta posterior. O Help traz informa es sobre o uso do

software, para auxiliar o usu rio e esclarecer suas d vidas.

Descreve-se a seguir, caracter sticas, funcionalidade e modos de utiliza o do Help e do Material Instrucional.

4.2 MENUS DE OPERA OES

A sele o do item "opera oes" no menu principal apresenta ao usu rio todas as opera oes realiz veis, como na figura 4.1. Os sub-menus s o dispostos de forma hier rquica para permitir o reconhecimento e a busca de forma facilitada. A seguir, s o mostradas as telas com menus e submenus de opera oes do Tutsi.

Opera oes Teoria Exercicios Exemplos Refer ncias Arquivos

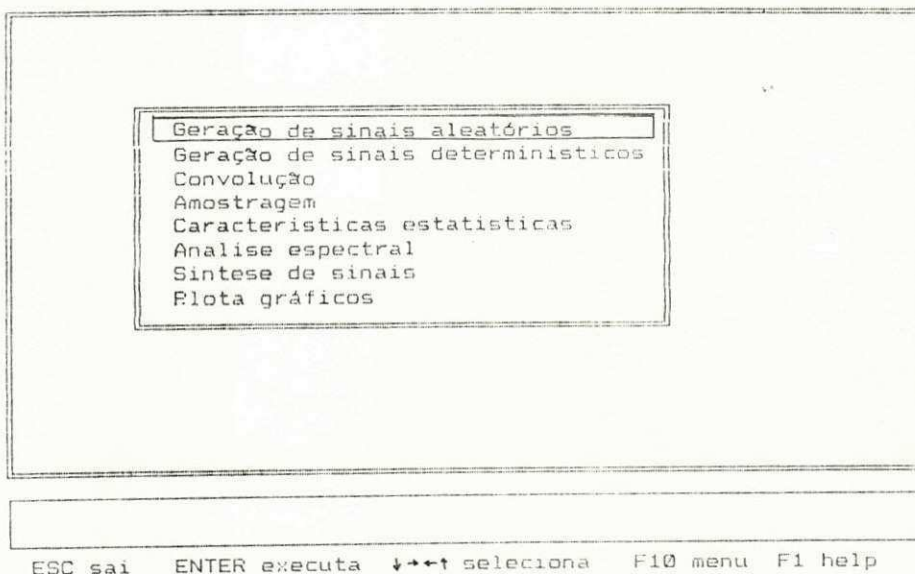


Fig 4.1 Tela do menu principal. Em destaque, a op o "Gera o de sinais aleat rios"

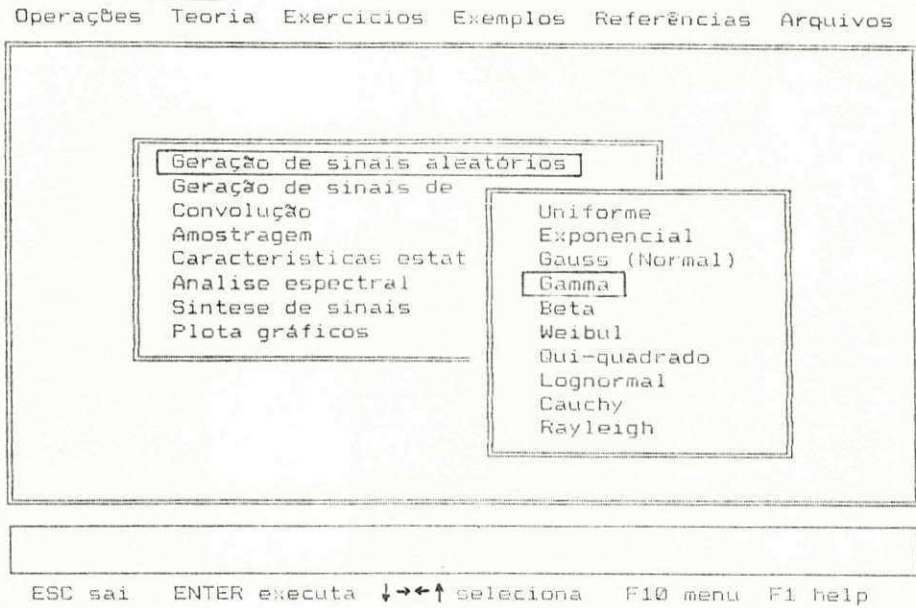


Fig 4.2 Submenu correspondente   op o "Gera o de sinais aleat rios"

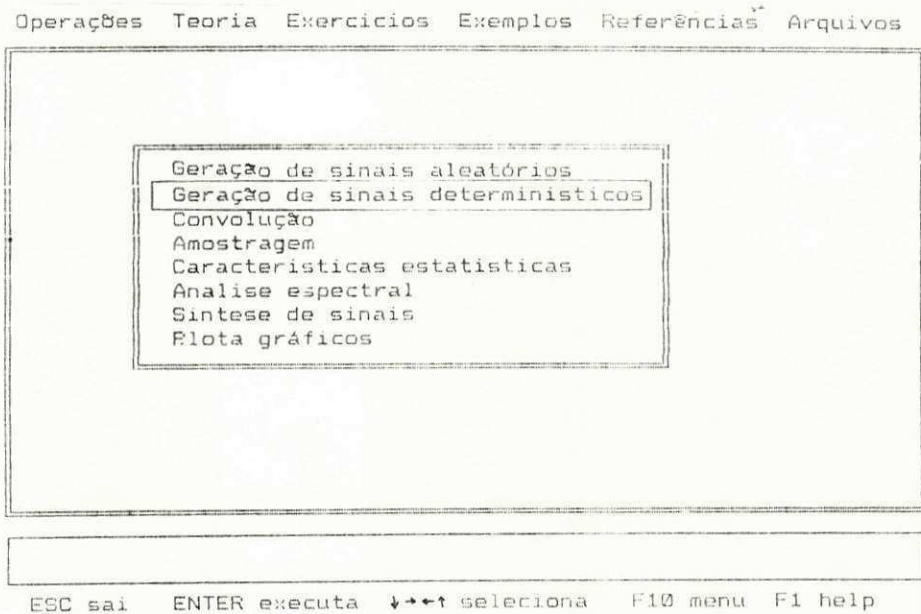


Fig 4.4 Tela do menu principal. Em destaque, a op o "Gera o de Sinais Deterministicos"

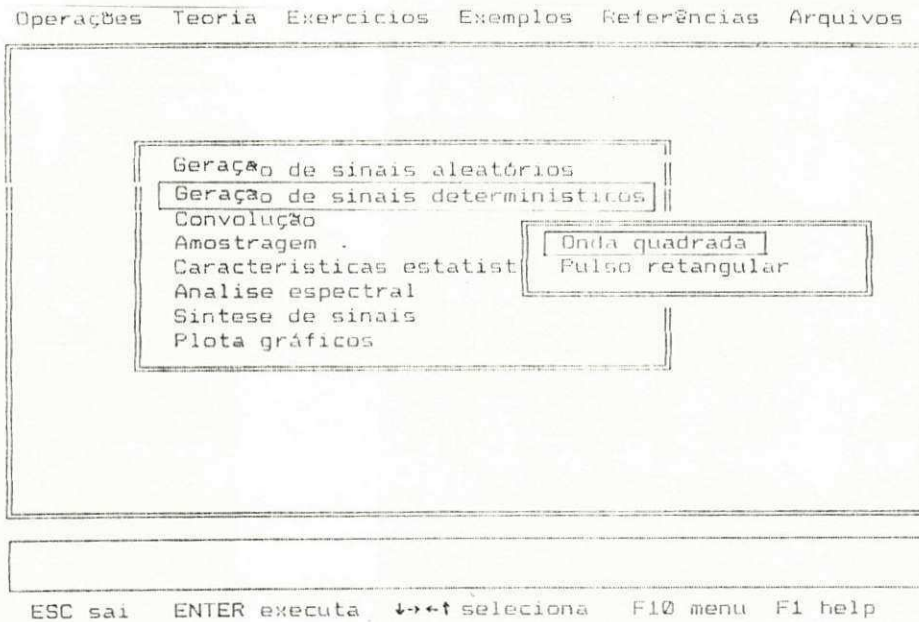


Fig 4.4 Submenu correspondente à opção "Geração de sinais determinísticos"

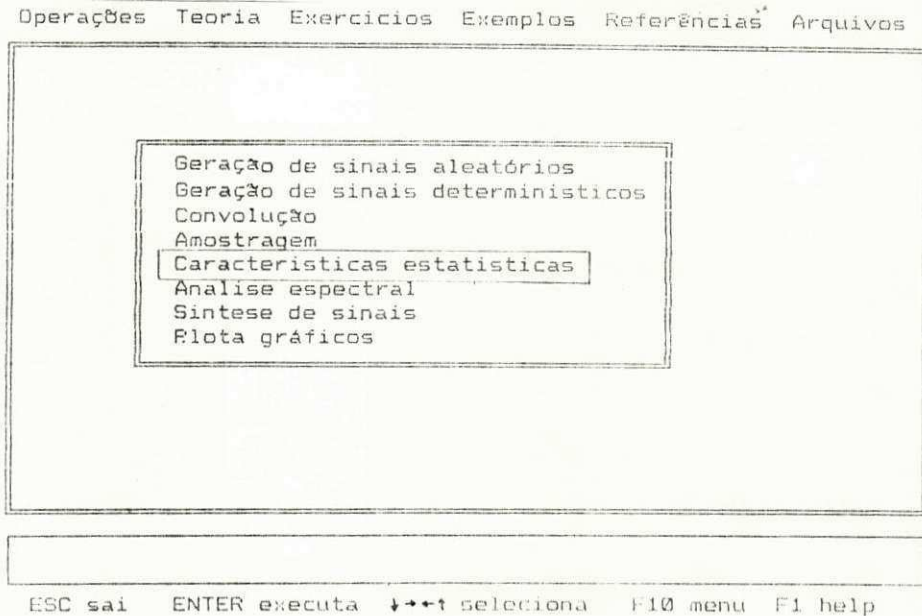


Fig 4.5 Menu de operações. Em destaque, a opção "Características Estatísticas"

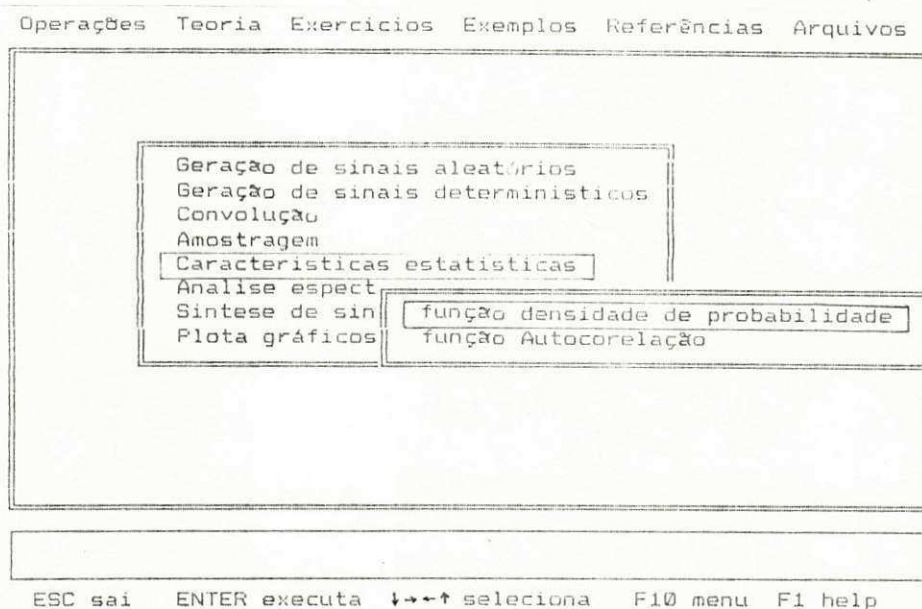


Fig 4.6 Submenu correspondente à opção "Características Estatísticas"

4.3 MENU ROTATIVO

O menu rotativo contém as opções que o usuário pode ativar à medida que sentir necessidade de rever conceitos pertinentes às operações em uso, bem como qual o proceder adotado para a resolução de determinada tarefa. As opções oferecidas são: Operações, Teoria, Exercícios, Exemplos e Referências.

Para fornecer uma janela para consulta com o conteúdo relativo às opções citadas, Tutsi faz uso de um arquivo de acesso aleatório, cujas posições são especificadas pelo programa, de acordo com a requisição do usuário. Um ponteiro de arquivo aponta para posições definidas do texto, à medida que o programa

é executado, de forma que uma requisição a uma dessas opções coincide com o texto relacionado à operação em andamento.

O conhecimento abrangido nas operações do TUTSI foi estruturado de forma que a cada consulta corresponde um texto básico, que aborda os fundamentos teóricos de cada operação estudada.

Para a explanação, é aberta uma janela temporária, sobreposta à atual, possibilitando o retorno do usuário à operação em andamento, ao findar a consulta, sem perda de continuidade.

Reservam-se às opções supracitadas instruções a respeito da teoria de sinais envolvida. Instruções sobre o uso do software são dadas através dos menus e do Help sensível ao contexto. Tenciona-se que as telas oferecidas sejam auto-explicativas, de forma que o usuário tenha um alto grau de certeza sobre o resultado de suas ações. Ao mesmo tempo, qualquer ação tomada pelo usuário deve ser avaliada pelo programa para evitar erros de execução com conseqüente parada do programa, prejudicando o estímulo ao uso do software. Algumas situações, em que o usuário pode causar erros de execução, são previstas e consideradas a seguir.

4.4 CRÍTICA DE DADOS

O módulo de entrada de dados cuida para que toda entrada de dados seja protegida, de forma que um erro de digitação não provoque um erro de execução do programa. Se, por

exemplo, se requer do usuário o valor da frequência de amostragem, dados como os abaixo não são aceitos:

-25	Valor negativo para frequência;
0.0	Valor nulo para frequência;
100000000000	Valor acima do máximo aceito pelo programa;
2,54	Virgula em vez de ponto decimal;
2.4.	Presença de dois pontos;
2.q2	Presença do caractere não numérico 'q';
''	Entrada nula. O usuário teclou ENTER antes de digitar o dado.

4.5 ENTRADA PARALELA DE DADOS

Integra a interface com o usuário um editor de texto simplificado para entrada paralela de dados. A maioria das vezes que requer a entrada de dados, o programa o faz de forma serial, um dado por vez. Uma vez que o usuário tenha passado seu dado para o programa não há mais condição de corrigi-lo. Exemplificando: O programa pede a Média para a geração de uma VA Normal (Gaussiana) e o usuário entra com o valor 0.8 . O programa passa à etapa seguinte solicitando o valor da Variância. Nesse instante, o usuário percebe que, por engano, digitou para Média o valor da Variância. Não é mais possível retroagir e alterar o valor da Média. A entrada paralela de dados se faz imprescindível quando o usuário precisa entrar com grande quantidade de valores, uma vez que aumenta o risco de erros de digitação. Tutsi utiliza a entrada paralela de dados na entrada dos coeficientes de

Fourier para a Síntese da forma de onda. Neste caso, os dados são digitados seguindo-se uma ordem, porém pode-se retornar a qualquer momento aos dados digitados e fazer-se correções. Fornece-se ao usuário um editor de texto simplificado, com recursos para navegação na tela, rolamento de tela, e correção de caracteres erroneamente digitados. Assim, o usuário pode conferir todos os valores na tela, e efetuar rolamento ascendente e descendente da tela, caso a quantidade de dados, um por linha, exceda o número de linhas da tela.

Exemplo: $x(1) = 0.0152$
 $x(2) = -0.21$
 $x(3) = 0.0084$

O usuário pode corrigir qualquer caractere dos dados acima, antes de encerrar sua entrada de dados.

4.6 FILTRAGEM DE TECLAS

Ao requerer a entrada de um dado, o sistema lê caractere por caractere digitado pelo usuário, em vez de ler o dado como um todo. Desta forma, rastreia toda ação do usuário e garante que, a qualquer instante, possa ser cancelada a operação em andamento (tecla ESC), solicitado Help (tecla F1), ou menu rotativo (tecla F10).

O mesmo ocorre quando o sistema lê teclas de navegação de menus (UpArrow, DownArrow, ENTER) e teclas de rolamento de tela (PgUp, PgDn) utilizadas no Help.

Quando o usuário se encontra selecionando opções de

menu, o sistema está pronto para atendê-lo caso queira sair para a janela de consulta. No entanto, a chamada ao Help bloqueia a chamada às opções Teoria, Exercícios, Exemplo, etc., constantes do menu rotativo. O usuário é alertado para este fato na própria tela do Help, que expõe as teclas que podem ser usadas naquele instante. Por outro lado, ocorre também bloqueio do Help, quando o usuário abre uma janela de consulta de uma das opções já citadas.

4.7 MANIPULAÇÃO DE ARQUIVOS

Tutsi opera grande parte do tempo com arquivos, uma vez que os sinais envolvidos nas operações, bem como resultados de operações são guardados em arquivos. Por isso Tutsi oferece ao usuário algumas funções do DOS que operam com arquivos:

- Lista diretório;
- Muda diretório;
- Renomeia arquivo; e
- Apaga arquivo.

Os arquivos GERADOS por Tutsi são sempre do tipo binário com dados reais. Este é o tipo default de arquivos. Contudo Tutsi está apto a trabalhar com os seguintes tipos de arquivos:

ARQUIVOS FORTRAN

Arquivo tipo texto, cujos dados são números reais.

ARQUIVOS DE PLACA DE AQUISIÇÃO DE DADOS

Arquivo binário cujos dados ocupam dois bytes cada um.

4.8 PREVENÇÃO CONTRA ERROS DE EXECUÇÃO

ABERTURA DE ARQUIVOS

Quando o usuário solicita a abertura de um arquivo, o programa verifica se o nome do arquivo é válido. Se o usuário erroneamente tecla ENTER, em vez de uma cadeia de caracteres, o compilador usado não interpreta esta ação como erro e, sim, abre um arquivo cujo nome é ''. Este fato desconfortável para o usuário é levado em conta pelo programa que emite a mensagem abaixo ao usuário:

```
' Arquivo inválido. Repita, por favor : '
```

LEITURA DE ARQUIVOS

Quando não há mais espaço em disco para a escrita de um arquivo, ou quando há defeito físico que causa erro de leitura ou de escrita no disco, a seguinte mensagem é enviada ao usuário:

```
' Erro de leitura/escrita. Tecla ENTER para continuar... '
```

Quando o usuário passa para o programa o nome de um arquivo inexistente, talvez por um erro na digitação, ou por não lembrar do nome exato do arquivo, o programa o informa com a seguinte mensagem:

```
' Arquivo inexistente. Repita por favor: '
```

Se um arquivo estiver sendo lido e os dados armazenados em uma variável do tipo "array", pode ocorrer que a

a quantidade de dados do arquivo ultrapasse o tamanho do array, provocando um "overflow". Neste caso a leitura é interrompida e o usuário, alertado pela mensagem:

```
' Erro de leitura/escrita. Tecla ENTER para continuar... '
```

A crítica de dados mencionada na secção 4.4 também previne contra erros de execução quando da escrita de arquivos. Por exemplo, Tutsi não aceita que um usuário abra um arquivo com apenas 1 Dado (Sinal com 1 ponto apenas), pois isto levaria a variável que armazena o número de pontos a causar uma divisão por zero dentro da rotina de saída gráfica.

4.9 ERROS FATAIS

Ocorre erro fatal, quando os dados de um arquivo não são compatíveis com o arquivo corrente. Exemplo: Se Tutsi estiver configurado para trabalhar com arquivos de números reais e o usuário lhe passar um arquivo de caracteres para ler. Ou ainda: Se Tutsi estiver configurado para trabalhar com arquivos oriundos de placa de aquisição de dados, e o usuário tentar operar com arquivo de números reais, ou de caracteres, ou Arquivo-texto.

4.10 HELP

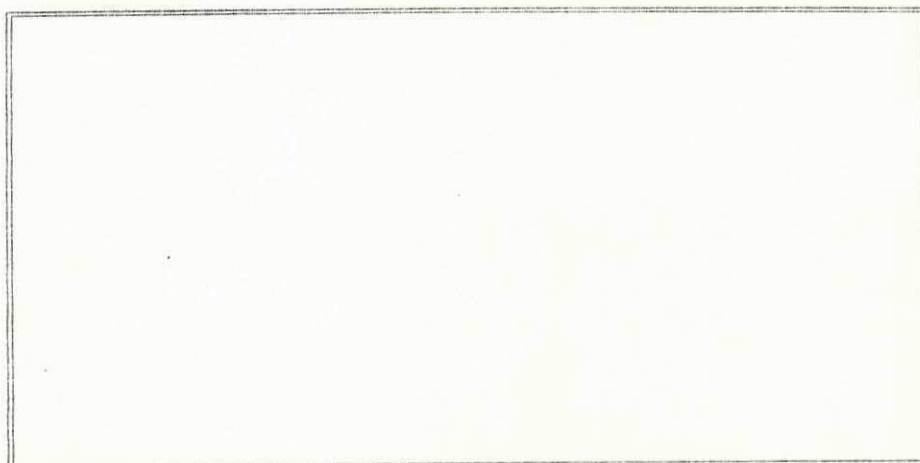
Tutsi provê um Help sensível ao contexto acionado pela tecla F1. O Help só está disponível quando a execução ocorre em modo texto, e contém informações sobre o uso do software, orientações sobre os tipos de dados requeridos do usuário, particularidades do software sobre suas capacidades e restrições

à ação do usuário. Uma seqüência de utilização do Help é descrita a seguir:

Na fig. 4.7, vê-se a primeira tela apresentada ao usuário. Neste ponto da árvore de menus, o contexto é vazio, pois nenhuma matéria do tutorial foi ainda selecionada. O cursor está posicionado no menu rotativo e pode navegar por suas opções: Operações, Teoria, Exercícios, Exemplos, Referências, Arquivos. O Help correspondente, fig. 4.8, é uma descrição dos recursos de Tutsi, independente de qual seja a opção do menu. Caso o usuário acione alguma função de consulta ao Material instrucional, opções "Teoria, Exercícios, Exemplos, e Referências", uma mensagem o orienta:

'Não há contexto. Selecione uma operação.'

Operações Teoria Exercicios Exemplos Referências Arquivos



ESC sai ENTER executa ↵ seleciona F10 menu F1 help

Fig 4.7 Primeira tela apresentada ao usuário

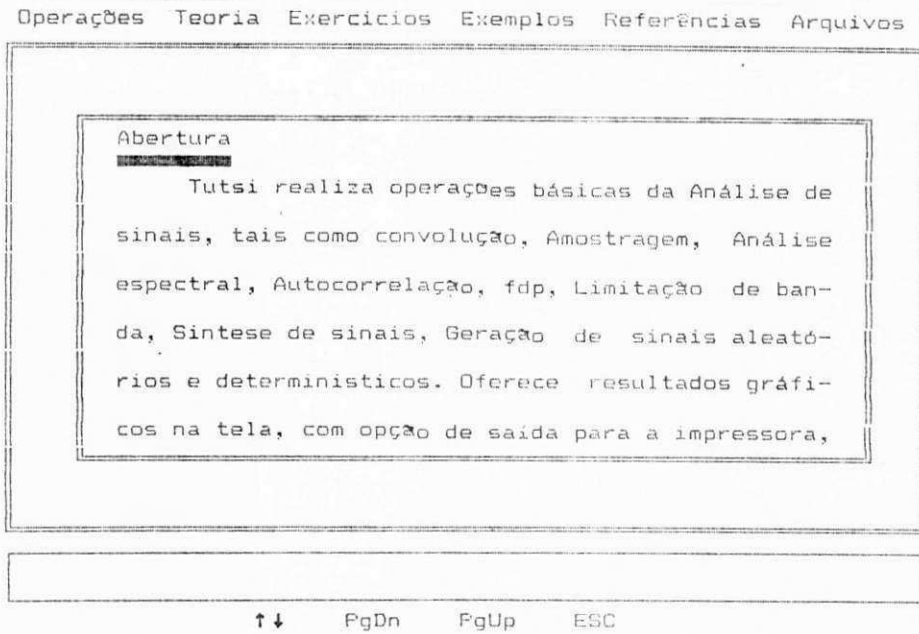


Fig 4.8 Help correspondente à primeira tela

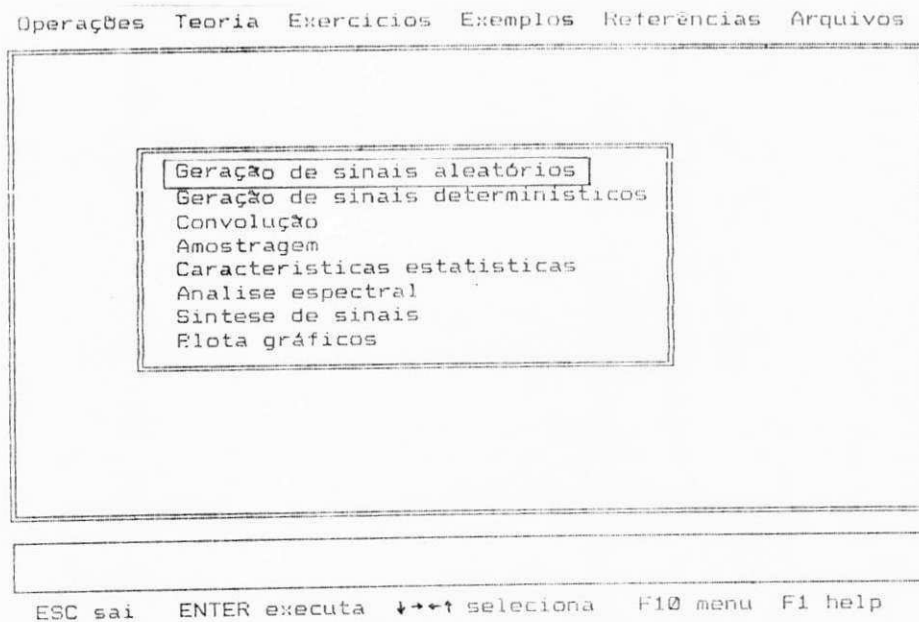


Fig 4.9 Menu de operações. Em destaque a opção "Geração de sinais aleatórios"

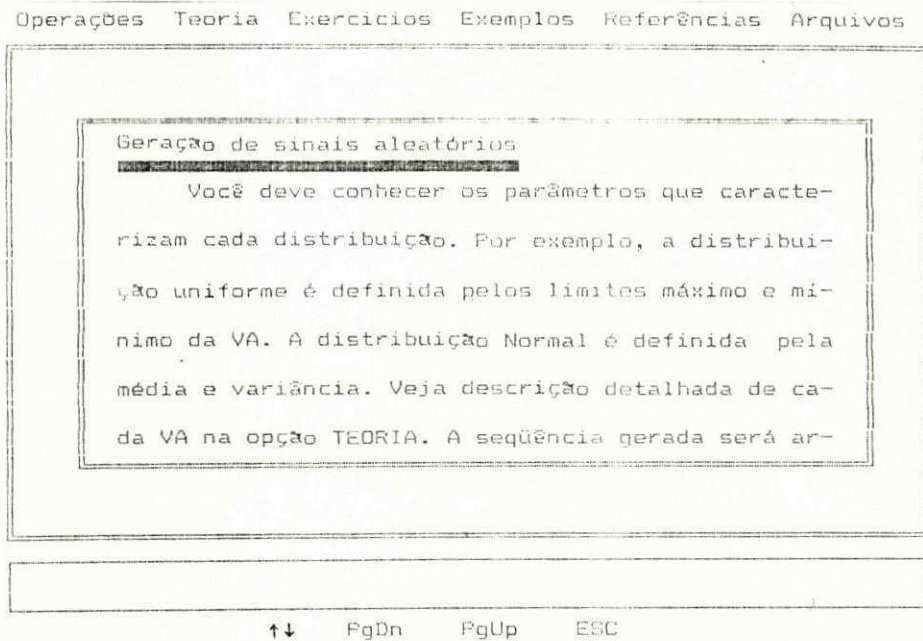


Fig 4.10 Help correspondente à opção "Geração de sinais aleatórios"

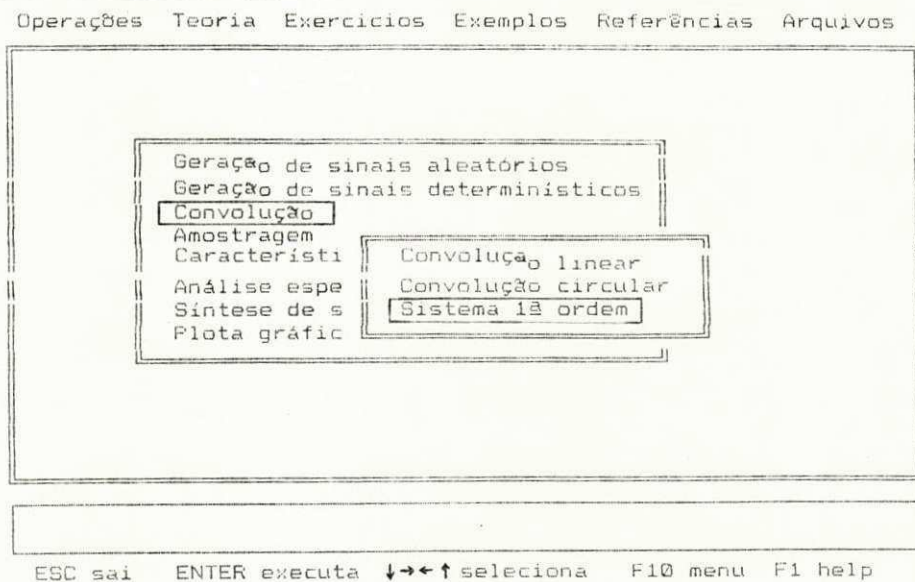


Fig 4.11 Menu de operações com submenu de "Convolução"

Neste caso o contexto abrange a operação de Convolução e a opção do submenu "Aplicação a Sistemas de 1ª Ordem".

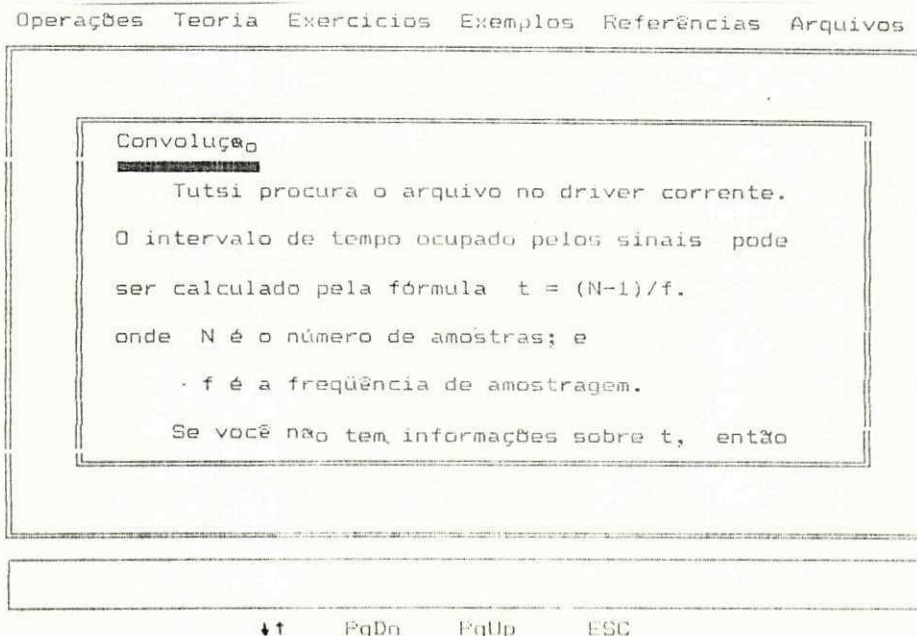


Fig 4.12 Help correspondente à opção "Aplicação a sistema de 1ª ordem"

4.11 INTERFACE GRÁFICA

Todo resultado das operações efetuadas é expresso em gráfico. Para esse fim o sistema conta com as seguintes rotinas:

- Saída gráfica de funções contínuas;
- Saída gráfica de funções descontínuas (Sinal PAM); e
- Saída gráfica tipo trem de impulsos.

As telas gráficas oferecem o recurso de ZOOM caso o usuário deseje detetar detalhes, modificando a escala do gráfico. Contam também com opção de saída para impressora, para fins de documentação.

Esta interface gráfica torna didaticamente ricas as operações, uma vez que, na maioria dos casos se está lidando com formas de ondas, e, noutros casos, com levantamento estatístico, no qual a expressão gráfica é imprescindível.

Uma saída gráfica pode ser acionada pelo sistema de duas formas:

- Requerida pelo usuário; e
- Saída automática ao final de uma operação, como saída de resultados.

Quando o usuário requer uma saída gráfica, deve passar para o sistema o nome do arquivo em que está a curva, os valores máximo e mínimo do eixo horizontal, e qual o tipo de gráfico desejado (Contínuo, Degrau, Trem de Impulsos). Neste caso, pode optar por:

- Traçar 1,2,3 ou 4 gráficos numa mesma tela, em posições distintas; e
- Traçar até 3 (três) curvas superpostas.

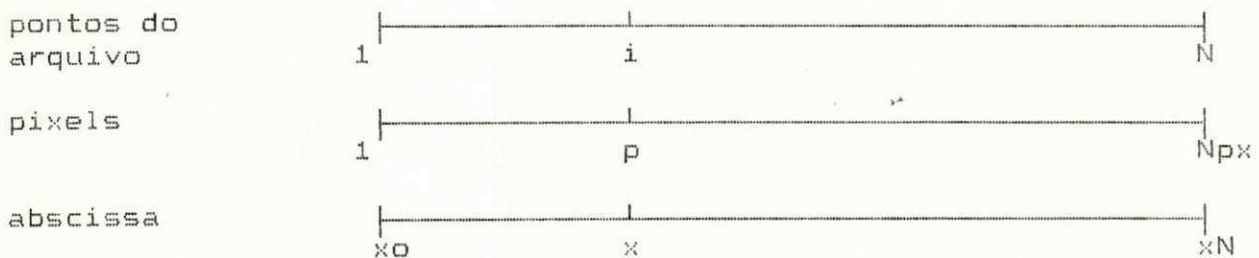
As limitações quanto aos números máximos supracitados são unicamente para evitar um congestionamento da tela. Suponha-se que se desejassem 6 gráficos na mesma tela; embora isto seja possível, os resultados seriam pobres em resolução e em percepção visual. Cada gráfico ocuparia $1/6$ da tela e, conseqüentemente, $1/6$ dos pixels, o que provoca perda na resolução, porque a tela gráfica é tratada como uma matriz.

Cada pixel da tela de vídeo é associado a um par de números inteiros que são suas coordenadas. Em um gráfico

bidimensional, noutro ambiente, cada ponto é associado a um par de números reais. Fica evidente que a transformação de real para inteiro deve ser efetuada sobre as coordenadas da curva em arquivo para representação na tela de vídeo.

4.12 CONVERSÃO DE ESCALA HORIZONTAL

Num arquivo, cujos dados armazenam os pontos de uma curva, o primeiro ponto, de índice 1, refere-se ao valor mínimo de abscissa, e o último dado, de índice N, refere-se ao valor máximo de abscissa. A associação entre índices dos dados, abscissas da curva e pixels é mostrada a seguir:



Os valores de abscissa da curva não são necessários para se efetuar a conversão, porém os valores máximo e mínimo destas são requeridos do usuário para que sejam impressos na tela. O procedimento matemático adotado considera a relação seguinte:

$$\frac{p - 1}{Npx - 1} = \frac{i - 1}{N - 1} \quad (4.1)$$

onde i é o índice do dado no arquivo;

p é a coordenada horizontal do pixel;

Npx é o número de pixels horizontais disponíveis para o gráfico; e

N é o número de pontos do arquivo.

Quando o número de pontos é maior que o de pixels, alguns pontos não são plotados. Quando o número de pixels é maior que o de pontos, alguns pixels não são utilizados.

Quando $npt > npx$, cria-se um loop que incrementa p, a partir de 1 até npx , e calcula-se i :

$$i = \frac{(p-1)(N-1)}{Npx-1} + 1 \quad (4.2)$$

Quando $Npx > N$, o "loop" incrementa i, de 1 a N, e calcula p :

$$p = \frac{(j-1)(npx-1)}{npt-1} + 1 \quad (4.3)$$

Na figura 4.13 mostram-se exemplos dos dois casos acima citados.

4.13 CONVERSÃO DE ESCALA VERTICAL

A tela de vídeo para o desenho do gráfico é tratada como uma matriz de pixels. As coordenadas de cada pixel são números inteiros positivos. Um adaptador gráfico que forneça uma tela de 720x348 pixels, tem para coordenadas do pixel do canto superior esquerdo (0,0) e do pixel do canto inferior direito (719,347). Assim, os valores reais de uma curva devem ser convertidos em coordenadas inteiras, pertencentes à matriz de pixels fornecida pela placa. As coordenadas verticais são obtidas procedendo-se da seguinte forma:

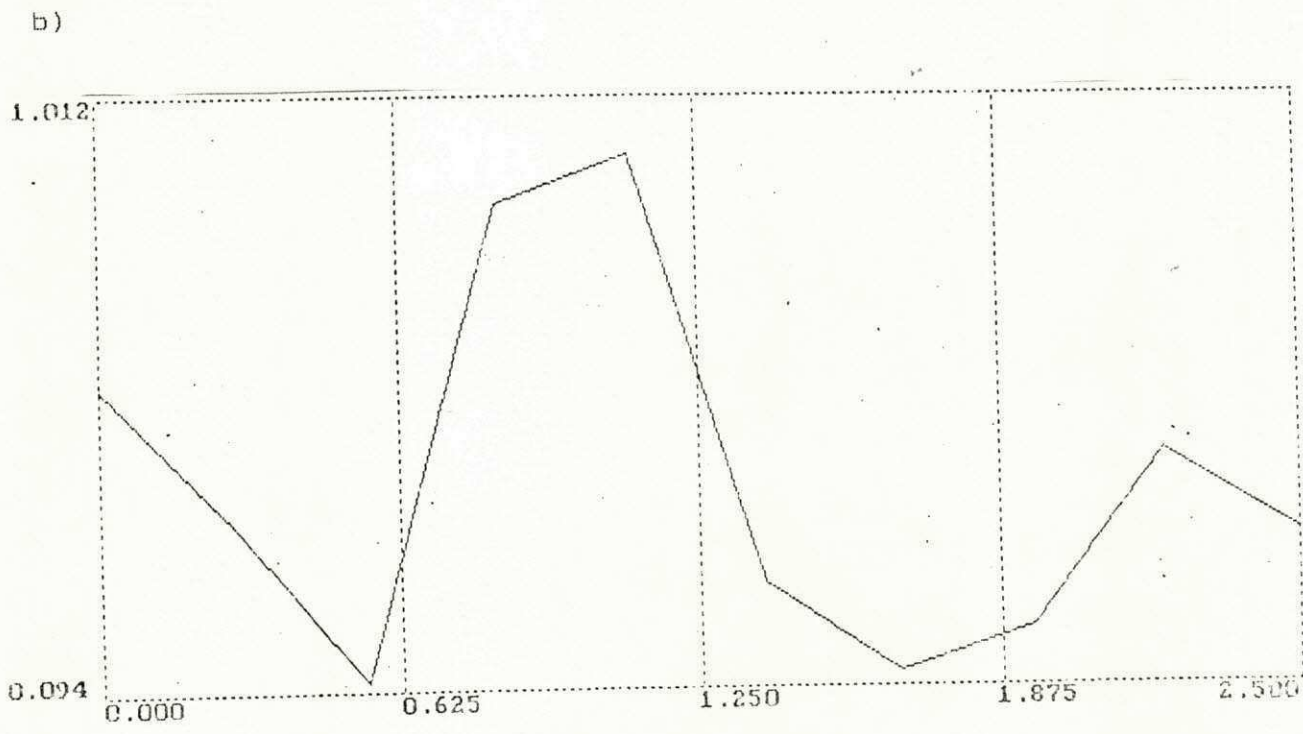
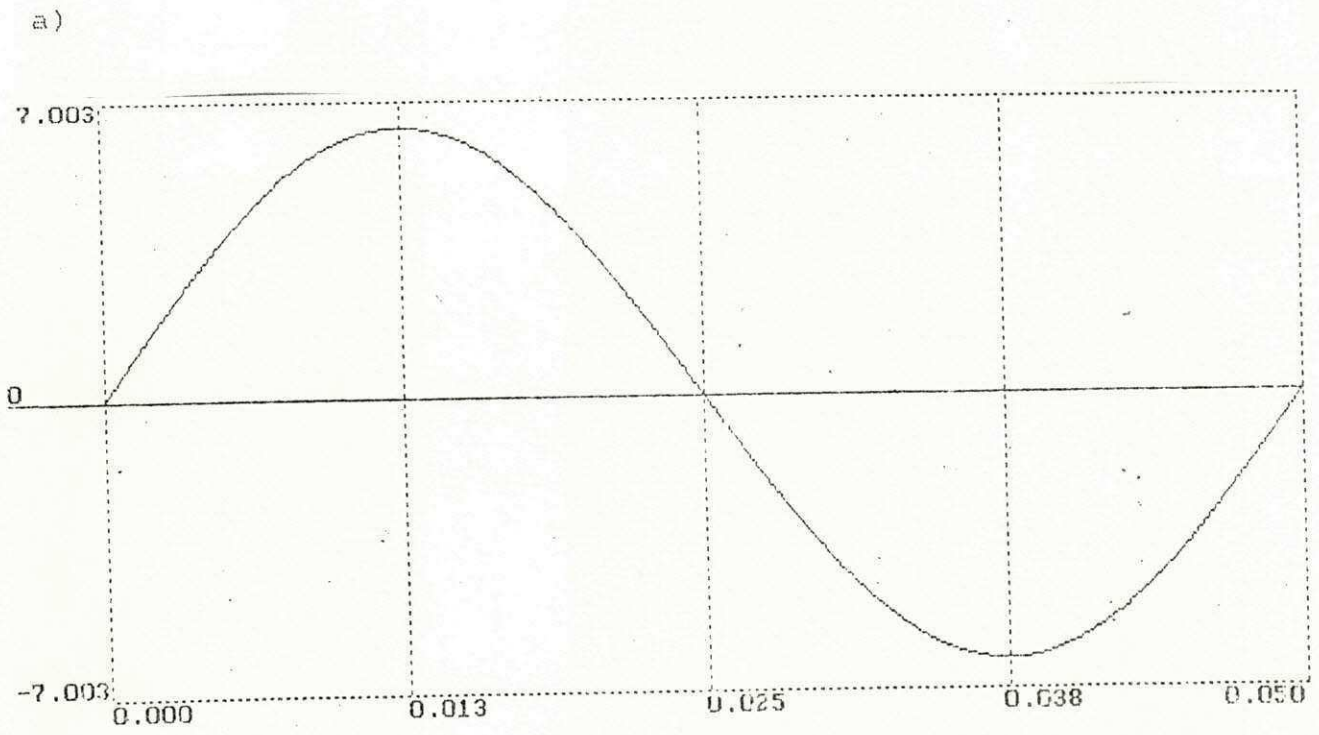


Fig 4.13 a) Número de pontos do arquivo menor que o número de pixels b) número de pixels maior que o número de pontos



Os valores de y , na coluna esquerda, representam as coordenadas verticais dos pixels da tela. Os valores de v , na coluna direita, representam os valores reais da curva. A partir da relação entre v e y , pode-se calcular o valor da coordenada correspondente a qualquer valor da curva, aplicando a relação abaixo:

$$\frac{y}{Y_{\max}} = \frac{V_{\max} - v}{V_{\max} - V_{\min}} \quad (4.4)$$

e o valor de y é obtido:

$$y = \frac{(V_{\max} - v) Y_{\max}}{V_{\max} - V_{\min}} \quad (4.5)$$

onde V_{\max} e V_{\min} são os Valores máximo e mínimo da curva;

v é um valor qualquer da curva;

Y_{\max} é o valor máximo vertical da matriz de pixels; e

y é a coordenada vertical dentro da matriz de pixels.

Tutsi deteta automaticamente os valores máximo e mínimo (V_{\max} e V_{\min}) de um arquivo ao lê-lo.

4.14 MULTIGRÁFICOS

Tutsi diversifica a forma de apresentação dos

resultados, possibilitando a exposição de mais de um gráfico por tela. Um, dois, três e quatro gráficos por tela são as opções disponíveis. Este modo de saída gráfica não é automático, mas controlado pelo usuário, significando que o usuário é quem solicita este modo, acionando a opção 'Flota gráficos' do menu de operações.

Uma forma de saída gráfica automática é quando o usuário realiza uma operação e o sistema devolve o resultado em forma de gráfico.

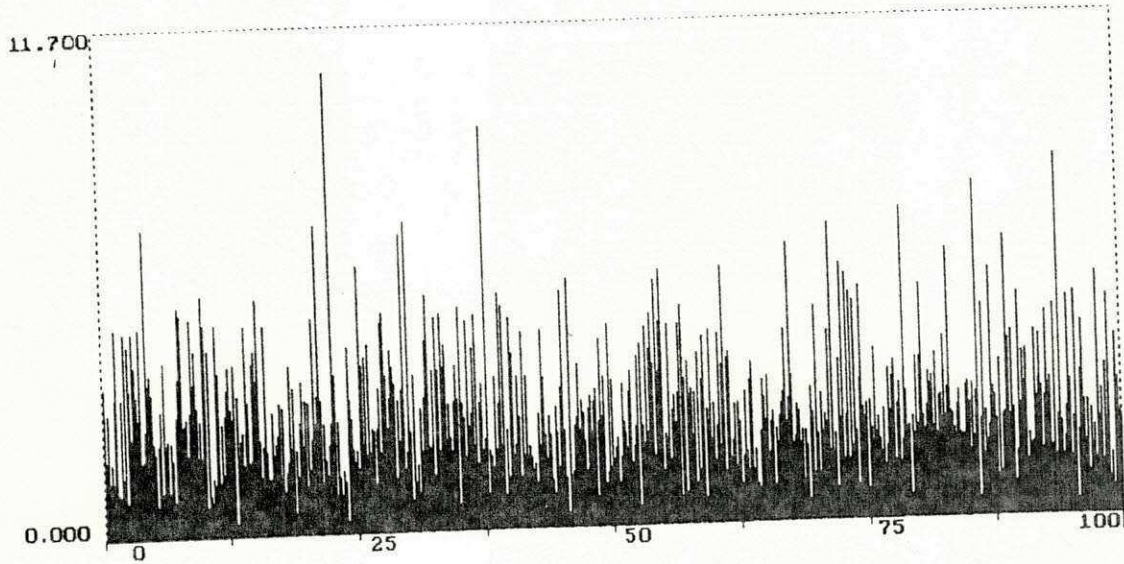
Ao solicitar a saída gráfica, o usuário tem duas opções:

1. Multigráficos; e
2. Superposição de curvas.

A opção 1 requer do usuário o número de gráficos desejados. Para cada gráfico, solicita-se o nome do arquivo, os valores máximo e mínimo de abscissa, e o tipo de traçado desejado. Se o tipo de traçado escolhido for o tipo "Degraus", o sistema ainda requer o número de degraus. Nas figuras 4.14, 4.15, 4.16 e 4.17 são apresentadas as formas de saídas gráficas.

4.15 DETECÇÃO DA PLACA GRÁFICA

Tutsi realiza a detecção automática da placa gráfica em uso, o que contribui para sua portabilidade. A utilização de placas gráficas diferentes é possível, ao se fazer uso dos "drivers" gráficos mencionados no apêndice, que trata da instalação do Tutsi.

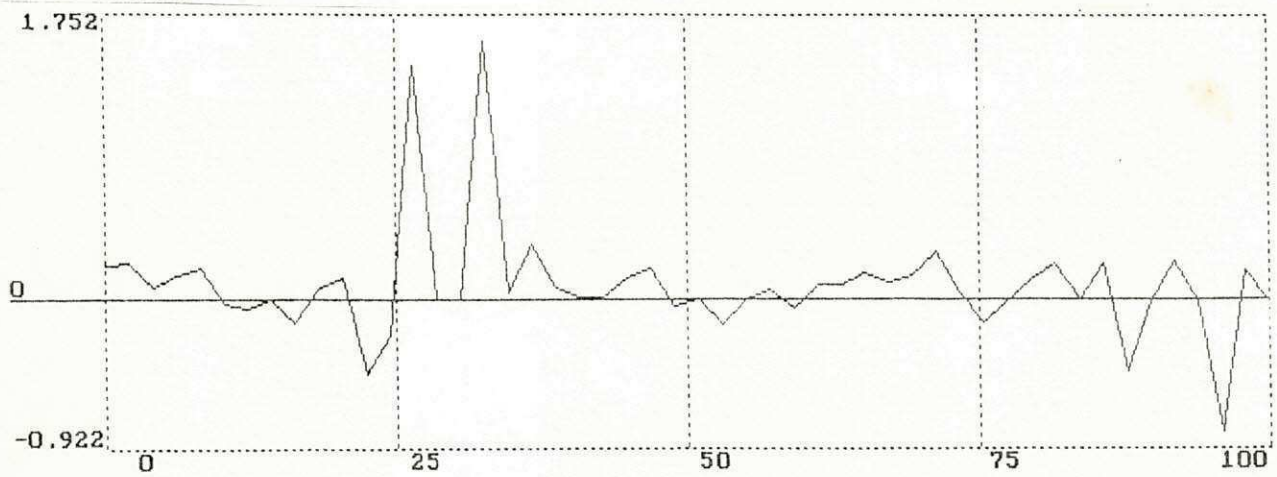


Trasado tren de impulsos

F2 grafico sem zoom F5 zoom ESC sai PRINT SCREEN saida impressora

Fig 4.14 Opção 1 gráfico por tela; tipo de traçado: trem de impulsos

a)



b)

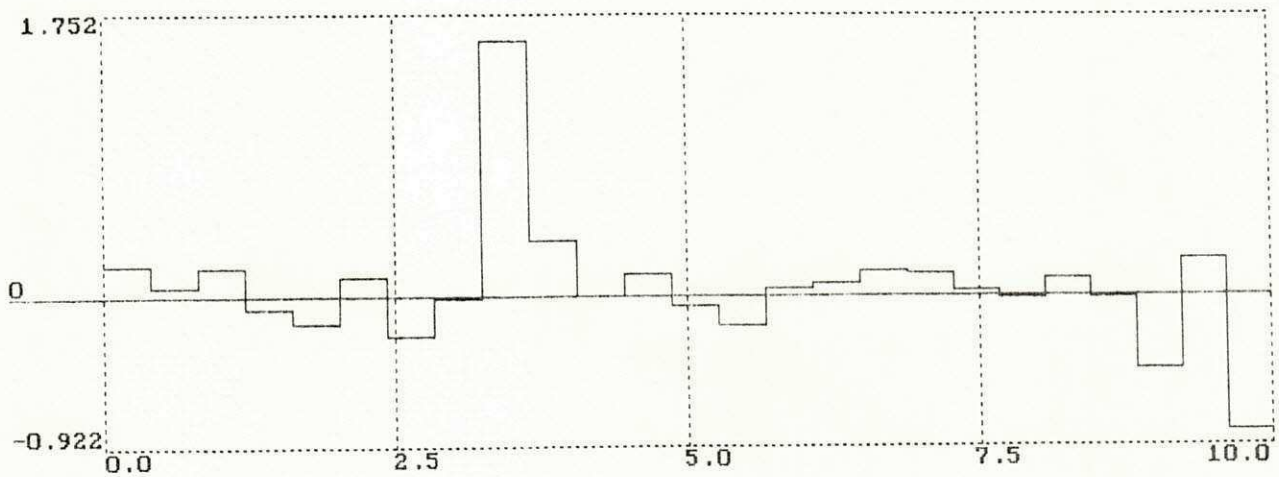
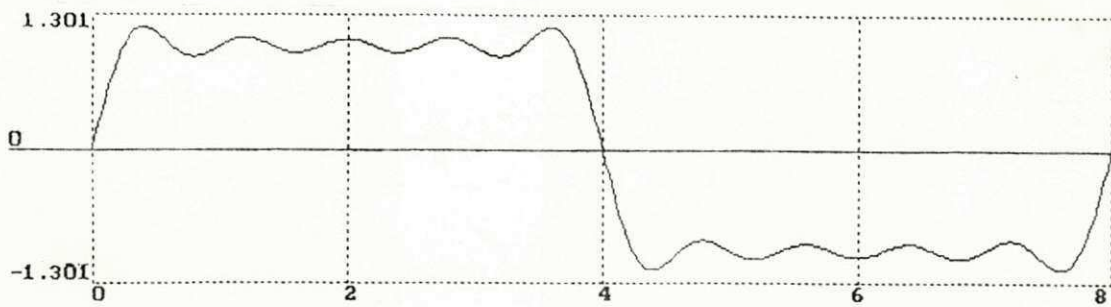
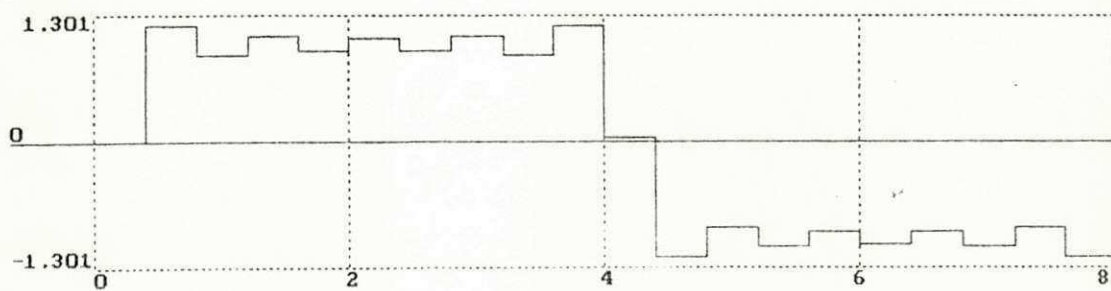


Fig 4.15 Opção 2 gráficos por tela. a) traçado Contínuo
b) traçado degraus

a)



b)



c)

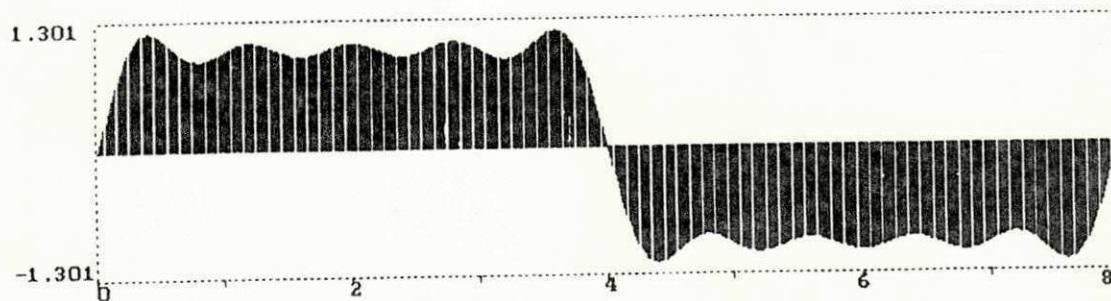


Fig 4.16 Opção 3 gráficos por tela. a) traçado Contínuo
 b) traçado degraus c) traçado trem de impulsos

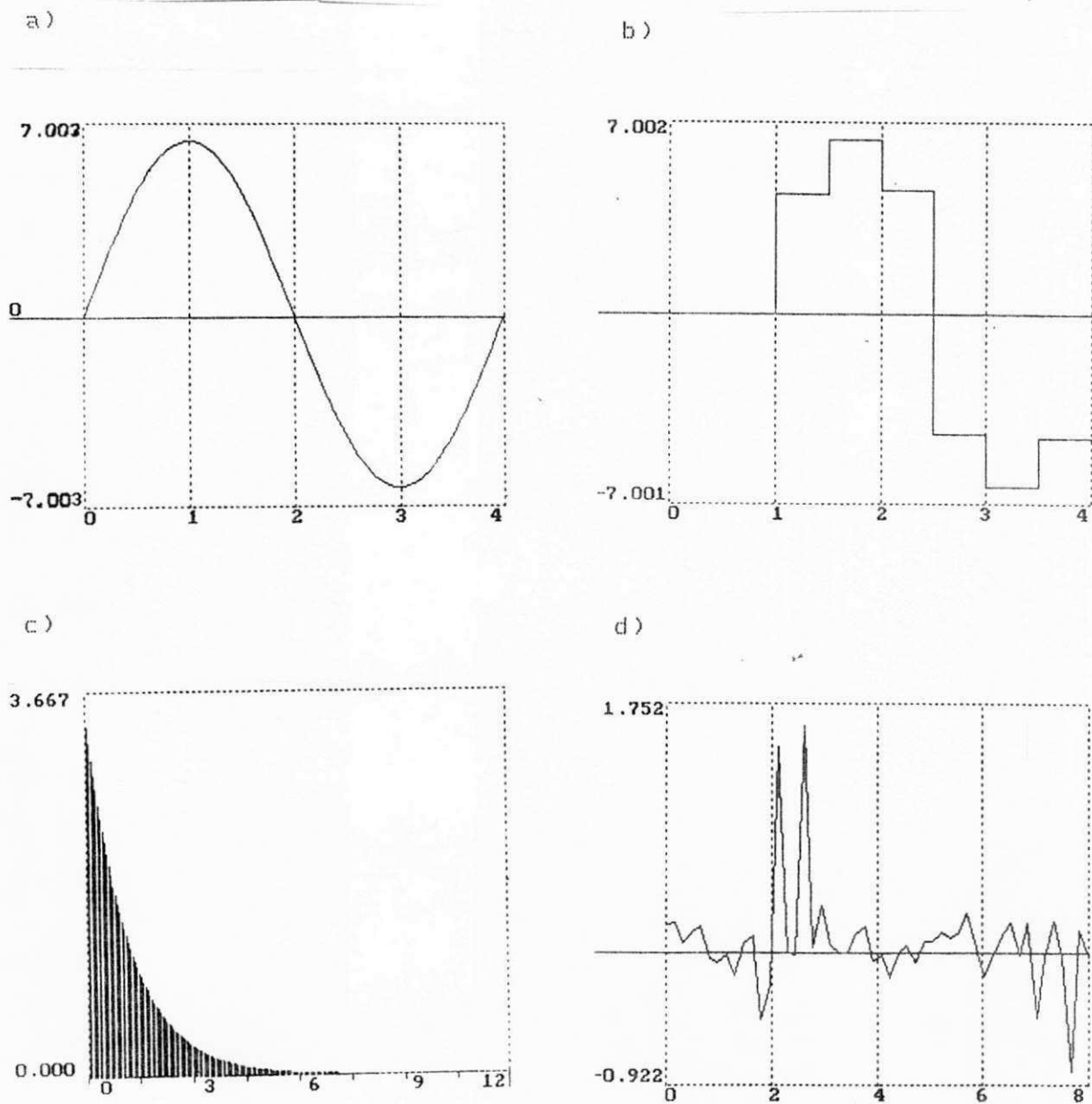


Fig 4.17 Opção 4 gráficos por tela. a) e d) traçado Contínuo
 b) traçado Degraus c) traçado Trem de Impulsos

4.16 SUPERPOSIÇÃO DE CURVAS

A principal dificuldade enfrentada na concepção da rotina que superpõe curvas é a compatibilização das escalas envolvidas, de forma que a escala final englobe todas as participantes. Determinam-se o maior e o menor valores de ordenada das curvas, o maior e o menor valores de abscissa das curvas, e estabelece-se o "Viewport" adequado.

A fig. 4.18 apresenta o caso de 3 curvas que ocupam cada uma o seu respectivo "Viewport". O "Viewport" para superposição é o "Viewport" 4.

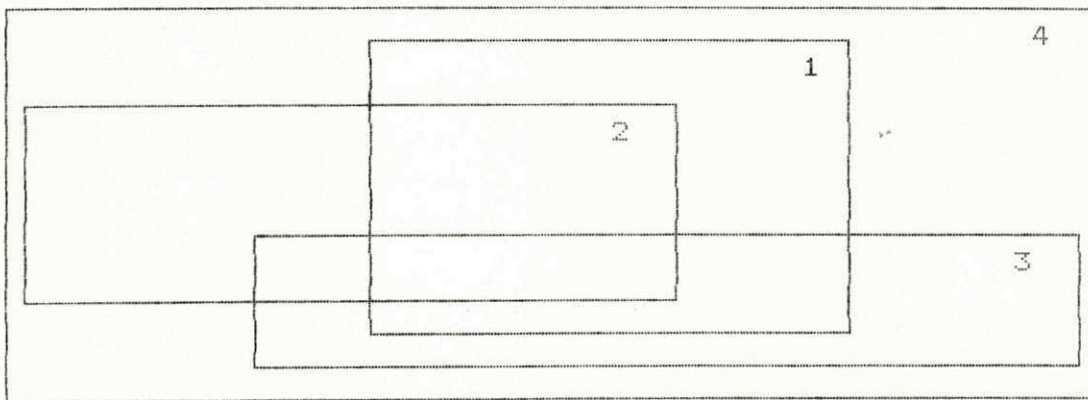


Fig 4.18 Definição de Viewport para superposição de curvas

Tutsi superpõe no máximo três curvas. A figura 4.19 apresenta um exemplo com três curvas superpostas.

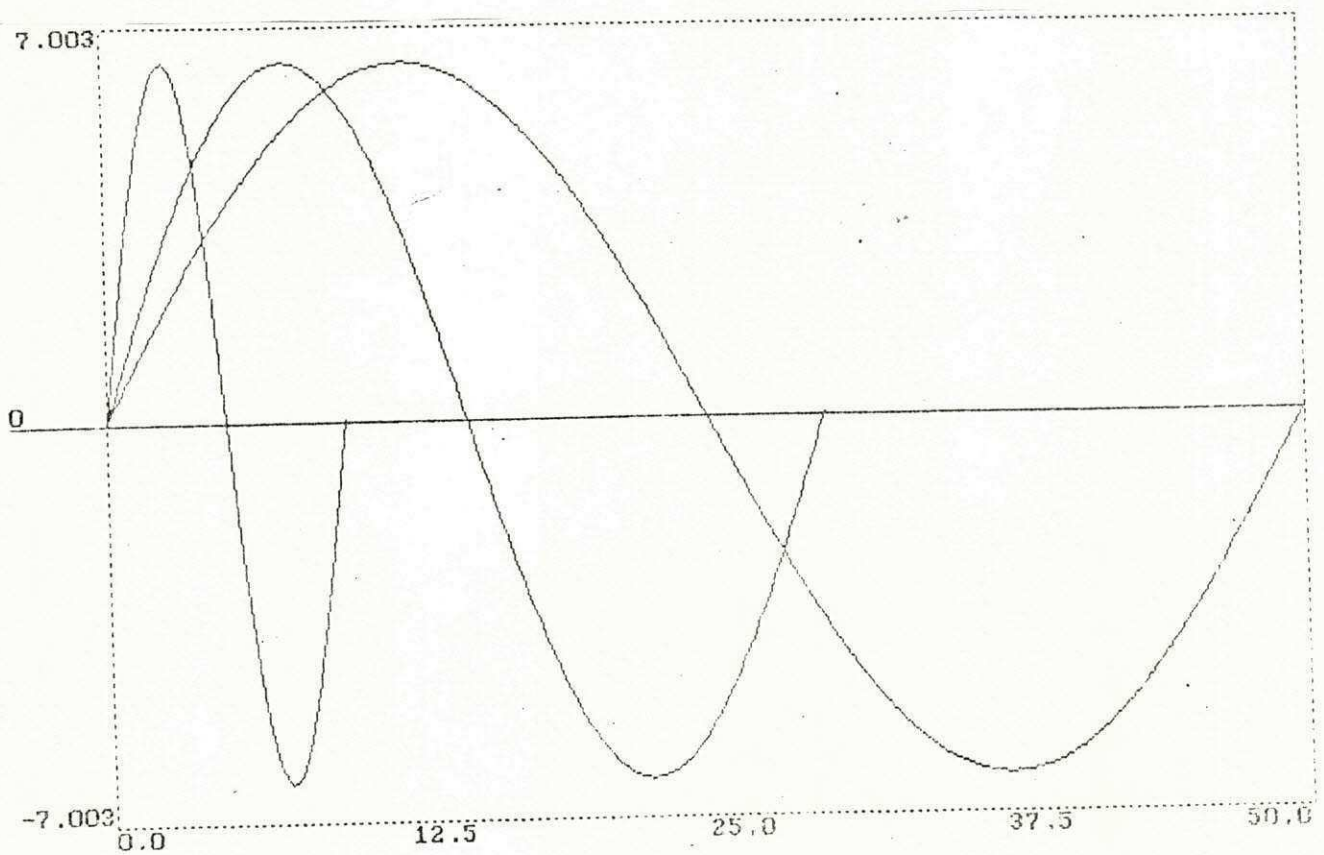


Fig 4.19 Exemplo de tela com três curvas superpostas

4.17 TRAÇADOS

A diversidade da forma de traçado visa atender necessidades relacionadas aos padrões de traçado de determinados sinais. O espectro de um sinal tem sido, na maioria dos casos, traçado em forma de trem de impulsos para esboçar a idéia de componentes espectrais, ou raias. O sinal resultante de uma amostragem tipo "sample-and-hold" é composto de degraus de igual duração. Por isso, Tutsi põe a disposição três formas de traçado, visando a melhoria da qualidade de apresentação de seus resultados.

4.17.1 CARACTERÍSTICAS DE IMPLEMENTAÇÃO

As rotinas dos três tipos de traçado disponíveis, têm em comum as conversões de escala, horizontal e vertical, e a detecção automática do adaptador gráfico em uso. A implementação da rotina que traça sinal em forma de degrau apresenta uma dificuldade particular, que requer consideração:

Se uma curva tem 32 degraus, e a placa em uso fornece 720x348 pixels (Hércules), é fácil calcular quantos pixels correspondem a cada degrau: $720/32$, que resulta 22,5. Convém lembrar, porém, que a quantidade de pixels é um número inteiro positivo. Assim, 22,5 seria aproximado para 23 ou 22. Se se optasse por 22, a curva ocuparia 704 pixels (32x22); mas a tela dispõe de 720, e a curva não ocuparia toda a tela. Se se optasse por 23, a curva ocuparia 736 pixels (32x23), e parte da curva não

seria mostrada. Isto acarreta também uma distorção visual, pois o centro da curva se desloca para a esquerda ou para a direita. Para evitar os inconvenientes citados, a rotina utilizada varia o tamanho do degrau de forma a centralizar a curva e traçá-la integralmente. A tela obtida é, portanto, semelhante a tela do osciloscópio. Observe, na figura 4.20, os resultados obtidos.

4.18 ZOOM

O recurso de Zoom tem uso relevante na análise de gráficos de sinais ruidosos, na verificação dos valores de raias espectrais e na documentação de trechos de sinais.

A rotina que efetua o Zoom divide a tela gráfica em quatro regiões, que podem ter suas imagens expandidas. Ao inicializar, a rotina salva todas as características da imagem inicial, visto que o usuário pode querer retornar ao primeiro gráfico. O número de Zooms sucessivos possíveis depende do número de pontos do arquivo. Cada região corresponde a $1/4$ do gráfico, portanto, a cada Zoom, o número de pontos considerados é igual a $1/4$ do número atual mais um. Como exemplo considere-se a seqüência abaixo:

	N
gráfico inicial	500
1º Zoom	126
2º Zoom	33
3º Zoom	9
4º Zoom	3
5º Zoom	-

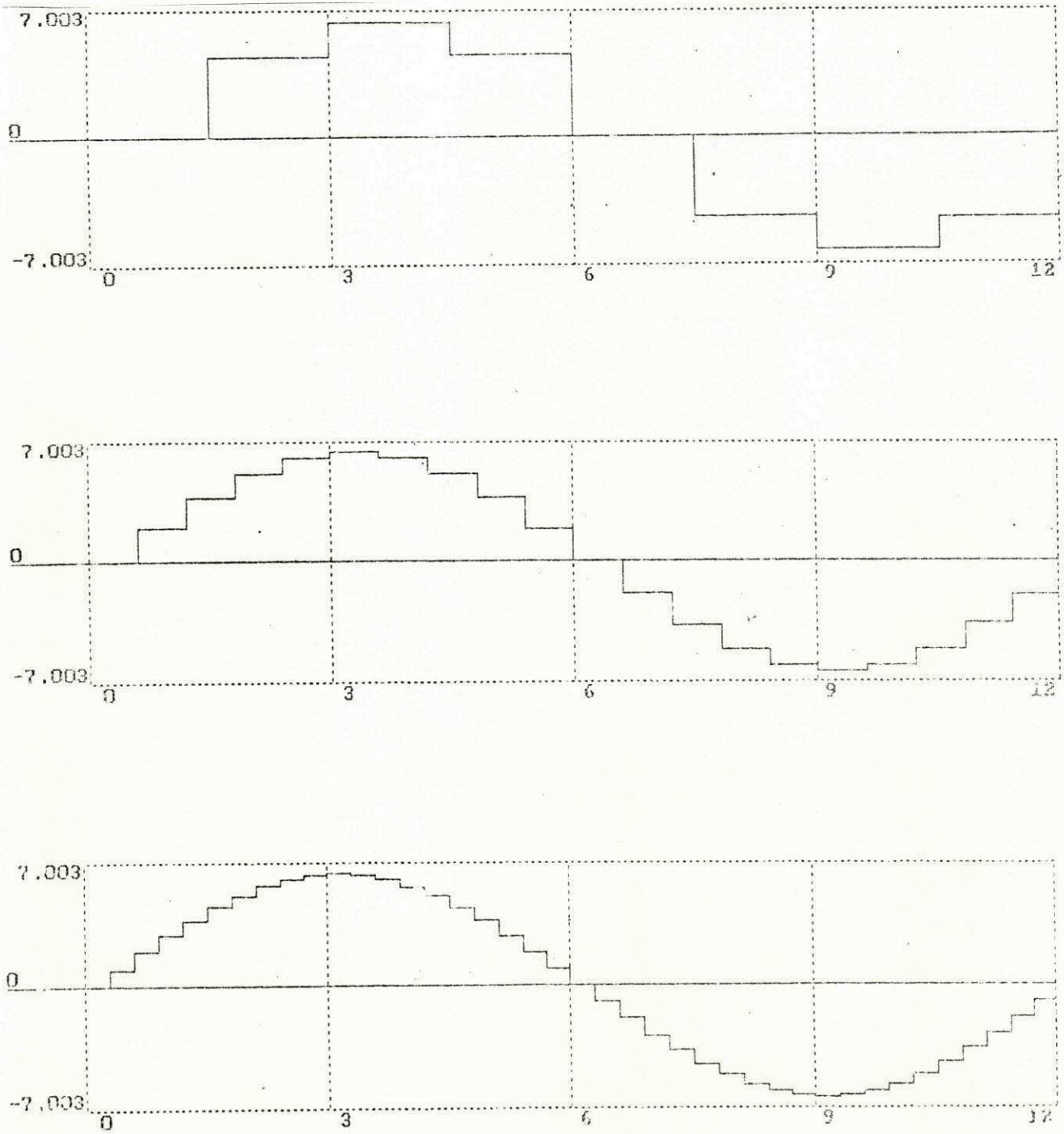


Fig 4.20 Gráficos com 8, 20 e 40 degraus

O 5º Zoom não é possível, pois N seria igual a 1, e não se pode traçar uma curva com apenas 1 ponto. Ademais, a variável que armazena o número de pontos participa em expressões matemáticas que levariam a uma divisão por zero, que causaria um erro fatal. Nesse instante, a tecla F5 que aciona a operação de Zoom é travada.

A ampliação da região selecionada é assimétrica. O eixo horizontal é ampliado quatro vezes, ao passo que o eixo vertical não tem fator de ampliação determinado previamente. A rotina de saída gráfica deteta os valores máximo e mínimo de amplitude na região e os aloca dentro de um "Viewport", cujo topo se baseia no valor máximo, e cuja base se baseia no valor mínimo. Assim, a ampliação no eixo vertical é sempre maximizada.

CAPÍTULO V

CONCLUSÃO

A implementação do Tutsi buscou atender a necessidades específicas de cursos de Engenharia Elétrica, na área de Análise de Sinais. A versão atual do Tutsi concede ao estudante oportunidade de realizar rápida, eficiente e confortavelmente operações básicas como Convolução, Análise Espectral, e Síntese de Sinais, operações estas ímpares no grau de dificuldade que apresentam. Possibilita a leitura da teoria abrangida, na tela, e utiliza recursos gráficos que facilitam a compreensão dos resultados.

O Help sensível ao contexto quase que dispensa a leitura de um manual de utilização. Os menus oferecidos, em conexão com as mensagens que orientam a execução e previnem erros na entrada de dados, a ausência de uma sintaxe de comandos e o reduzido número de teclas de controle, tornam a interface amigável e as telas auto-explicativas.

Tutsi pode também ser utilizado simplesmente como traçador de gráficos, uma vez que suas rotinas de saída gráfica formam um bloco independente, beneficiando especialmente o aluno que não esteja no momento lidando com Análise de Sinais, mas queira obter resultados gráficos de seus trabalhos e/ou

imprimí-los.

O campo de aplicação do Tutsi abrange aulas de laboratório, aulas demonstrativas, sessões individuais de estudo, apoio a pesquisa, e utilização como traçador de gráficos.

A implementação de Tutsi enseja a adição de outras operações da Análise de Sinais ao conjunto implementado, o melhoramento ou ampliação do Material Instrucional.

A seguir, são dadas algumas sugestões para versões futuras de Tutsi.

5.1 SUGESTÕES PARA CONTINUAÇÃO DO TRABALHO

Outras operações podem ser acrescentadas para enriquecer o conjunto já implementado. Por exemplo:

- Degradação de sinais com relação sinal-ruído especificada pelo usuário. Uma vez que Tutsi gera sinais estatísticos com distribuição escolhida, podem-se adicionar dois sinais com relação sinal-ruído dada pela expressão:

$$\text{SNR} = 10 \log \frac{\sigma_x^2}{\sigma_n^2}$$

onde σ_x^2 é a Variância do sinal a ser degradado; e

σ_n^2 é a Variância do ruído.

- Obtenção de outras características estatísticas de sinais, tais como Covariância, Correlação cruzada, Estimação de fdp, etc.

TUTORIAIS INTELIGENTES

Implementação do módulo "Modelo do Estudante", para compor com o domínio e a interface existente, um Sistema Tutorial Inteligente, segundo a arquitetura proposta por Hartley e Sleeman [3].

INTERFACE

- Aplicação de técnicas de hipertexto ao Help e aos textos para consulta.

Os textos do Help e do Material Instrucional, conteriam palavras-chaves, que poderiam referenciar outros textos. Uma palavra-chave seria uma palavra ou expressão que encerrasse um conceito do domínio. Por exemplo, em um texto que resumisse a teoria de Convolução, algumas palavras-chaves seriam: Sistema Linear Invariante no Tempo, Resposta ao Impulso, Sinal de Energia Finita, etc. Cada palavra-chave funcionaria como opção de menu que se ligaria a um texto explicativo do conceito (Figura 5.1).

- Implementação de Help no modo gráfico.

Na versão atual o Help só está disponível em modo texto, mas seria uma facilidade a mais se o usuário pudesse acioná-lo também a partir do modo gráfico.

- Implementação de Zoom em telas multigráficas.

O Zoom ocuparia uma janela temporária no centro da tela, que seria eliminada após a observância dos detalhes por

parte do usuário, figura 5.2.

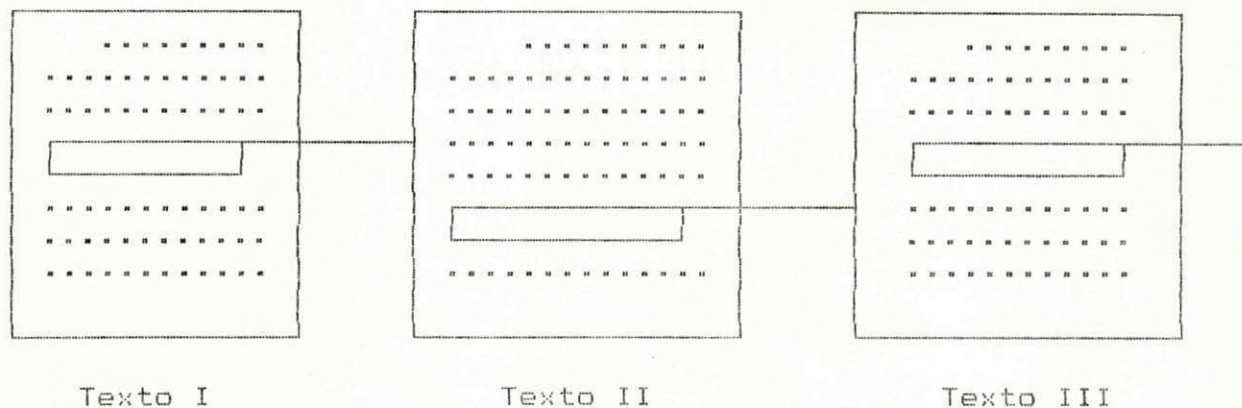


Figura 5.1 Textos do Material Instrucional que formam uma rede de 'nós'

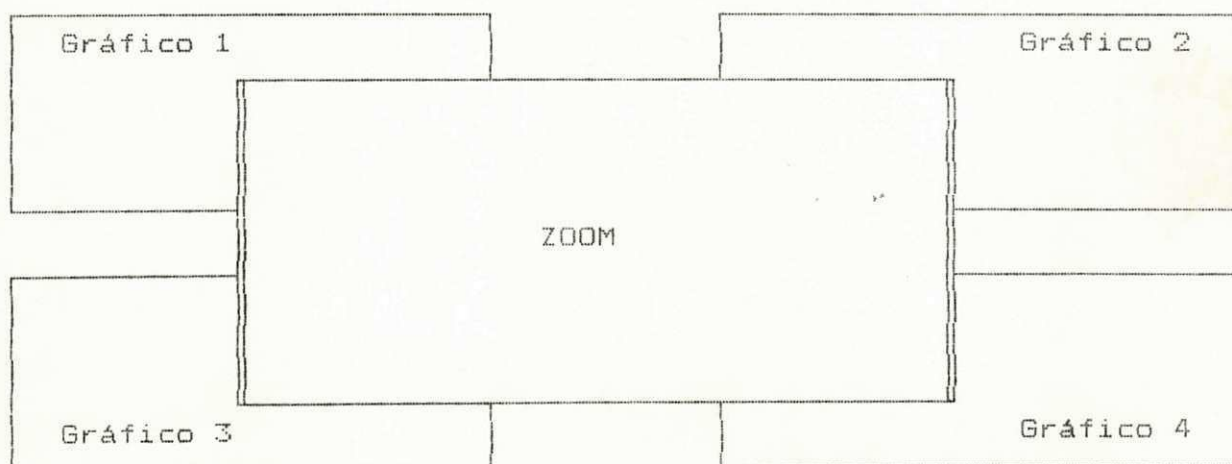


Fig 5.2 "Viewport" de Zoom em telas multigráficas

- Execução automática dos exemplos constantes do Material Instrucional.

Os exemplos dados poderiam ser executados pelo simples acionar de uma tecla de confirmação. Na versão atual, os exemplos não são executados automaticamente, mas o usuário tem que entrar na árvore de menus e efetuar a entrada de dados da forma como é

exemplificado.

- Impressão dos textos do material instrucional.
- Drive para "Mouse".

A utilização de "Mouse" na interface proporcionaria maior conforto ao usuário.

Paralelamente à produção do software, acrescentou-se considerável carga de experiência no projeto e implementação de Sistemas Tutoriais. Este fato contribui positivamente para a continuidade e melhoria do trabalho ora realizado. Os textos constantes do Material Instrucional e as mensagens de erro precisam de freqüente aprimoramento.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Wenger, E., **Artificial Intelligence and Tutoring Systems**, Morgan Kaufmann Publishers Inc, 1987.
- [2] Ratzlaff, K.L., **Introduction to Computer-Assisted Experimentation**, John Wiley & sons, 1987.
- [3] Sleeman, D. & Brown, J.S., **Intelligent Tutoring Systems**, Academic Press Inc, 1982.
- [4] Borges, M.R.S., **Hipertexto e suas implicações**, XI Congresso Nacional da Sociedade Brasileira de Computação, Santos-SP, 1991.
- [5] Moreira, M.A., **Ensino e aprendizagem - enfoques teóricos**, Editora Moraes, 1985.
- [6] Jayant, N.S. & Noll, P. , **Digital Coding of Waveform**, Prentice-Hall, 1984.
- [7] Bulgren, W.G. , **Discret System Simulation** , Prentice-Hall , 1982.
- [8] Rabiner, L.R. & Gold, B. , **Digital Signal Processing** , Prentice-Hall, 1975.
- [9] Stanley, W.D., **Digital Signal Processing**, Reston Publishing Company, 1975.
- [10] Openheim, A.V. & Willsky, A.S., **Signals and Systems**, Prentice-Hall, 1983.

-
- [11] Hatori, M.T. & Pequeno, M.C., **FFT - Problemáticas de Testes**, XII CNMAC, S.J. do Rio Preto, SP, 1989.
- [12] **Programs for digital signal processing**, Digital signal processing committee, IEEE acoustic, speech, and signal processing society, 1979.
- [13] Lathi, B. P. , **Sistemas de comunicação**, Guanabara dois, 1979.
- [14] Parks, T.W. & Burrus, C.S., **DFT/FFT and Convolution Algorithms**, John Wiley & Sons, 1985.
- [15] Shneiderman, B. , **Desining the User interface : Strategies for effective Human-Computer Interaction**, Addison-Wesley Publishing Company, 1987.
- [16] Papoulis, A. , **Probability, Random Variables, and Stochastic Process**, Mcgraw-Hill, 1965.
- [17] Kennedy, W.J. & Gentle, J.E. , **Statistical Computing**, Marcel Dekker Inc, 1980.
- [18] Hector, J. , **DSN Un logiciel de manipulation de courbes**, Manuel de reference, Institut National Polytechnique, 1990.
- [19] Max, J. , **Traitement du Signal**, Masson & Cie, 1972.
- [20] Turbo Pascal, Reference guide , version 5.0, Borland International, 1989.
- [21] Turbo Pascal, Users guide, version 5.0, Borland International
- [22] Carlson, A. B. , **Communication systems**, McGraw-Hill, 1975.
- [23] Taub, H. & Shilling, D. L. , **Principles of communication systems**, McGraw-Hill, 1971.

APÊNDICE INSTALAÇÃO

Para instalar TUTSI, o usuário não necessita de procedimentos extras, basta estar de posse dos arquivos listados a seguir, cuja memória total ocupada é de 248212 bytes.

Tut.Exe
Tutor.Txt
CGAVGA.Bgi
EGA.Bgi
Herc.Bgi
Tutsi.Exe
Graphics.com

DESCRIÇÃO DOS ARQUIVOS

O arquivo Tut.exe é o arquivo executável que contém todas as rotinas do Tutsi, e que faz uso dos outros arquivos acima listados, exceto Tutsi.Exe.

O arquivo Tutor.Txt é um arquivo binário que contém todos os textos manipulados por Tutsi: Help, Resumo da teoria, Exemplos, Referências, e Exercícios. A ausência desse arquivo embora não impossibilite a execução de Tutsi, priva o usuário das informações valiosas do Help e do Material Instrucional.

Os arquivos com extensão Bgi são drivers gráficos necessários às rotinas de saídas gráficas. Sem esses arquivos é impossível qualquer operação gráfica.

Tutsi deteta a placa em uso para que o driver correspondente seja acionado no modo gráfico. Alistam-se aqui quatro drivers gráficos que correspondem às placas mais comumente usadas atualmente: Hércules, CGA, VGA, EGA. No entanto há outros drivers gráficos disponíveis para placa ATT400 e IBM8514, fornecidos junto com Turbo Pascal pela Borland International. Se for o caso, o usuário pode simplesmente instalar esses arquivos para que sejam utilizados pelas rotinas gráficas de Tutsi.

O arquivo Graphics.Com é um arquivo do DOS que permite uma saída da tela gráfica corrente, para a impressora, pelo acionamento da tecla PRINT SCREEN. Sem esse arquivo a opção 'Saída para impressora' dos gráficos que Tutsi produz, não funciona.

O arquivo Tutsi.Exe roda o programa correspondente ao arquivo Graphics.Com, deixando-o residente na memória, para tornar possível uma saída da tela gráfica para a impressora. Em seguida dá início a execução das rotinas de Tutsi do arquivo Tut.exe. Nesse procedimento os arquivos Tut.Exe e Graphics.Com constituem os processos filhos de Tutsi.Exe.