

TESE

838

ESCOLA FEDERAL DE ENGENHARIA DE ITAJUBÁ

Desenvolvimento de Técnicas Otimizadas para a
Gestão de Carga, Utilizando Programação
Dinâmica e Conjuntos Difusos

Eng. ROBERTO DEVIENNE FILHO

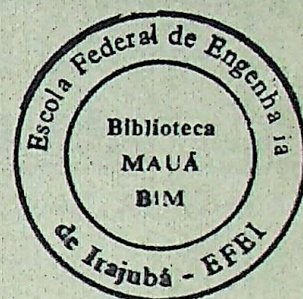
Itajubá, MG - Dez. - 1995



ESCOLA FEDERAL DE ENGENHARIA DE ITAJUBÁ

PRÓ-DIRETORIA DE PESQUISA E PÓS-GRADUAÇÃO - PPG
COORDENAÇÃO DOS CURSOS DE PÓS-GRADUAÇÃO EM
ENGENHARIA ELÉTRICA - CPG-E

Dissertação de Mestrado



“ Desenvolvimento de Técnicas Otimizadas para a
Gestão de Carga, Utilizando Programação
Dinâmica e Conjuntos Difusos ”

Autor : Eng. Roberto Devienne Filho

Orientador : Prof.Dr. Jamil Haddad

Co-Orientador : Prof. Dr. Afonso Henriques Moreira Santos

Itajubá - MG
Dezembro de 1995

CLASS. 6.21.3.012.3:681.31:519.6(043.2)

CUT: D 492 d

TOMBO. 838



Devienne, R.

Desenvolvimento de técnicas otimizadas para a gestão de carga, utilizando programação dinâmica e conjuntos difusos / Devienne, R. - 1.ed. - Itajubá MG : EFEI, 1995

p.131 : ilust.

1. Eng. Elétrica 2. Gestão de carga 3. Programação dinâmica 4. Conjuntos fuzzy

À minha esposa Ana Claudia, pelo apoio
e paciência nos momentos mais difíceis.

À meus pais Roberto e Maria Isabel e
minha irmã Daniela.

Ao amigo Carlos Gilberto de Figueiredo
que infelizmente não está mais junto de
nós.

AGRADECIMENTOS

À Deus, pelo dom da vida.

Aos meus orientadores, professor Dr. Jamil Haddad e professor Dr. Afonso Henriques Moreira Santos, pela orientação, amizade, dedicação, paciência e por acreditarem em minha capacidade.

À meus **eternos amigos** Ismael Mendonça Resende, Nelson Clodoaldo de Jesus, José Eugênio Pereira Ribeiro, Luís Octávio Mattos dos Reis (Pará), Antônio José do Couto Pitta, José Batista Siqueira Filho (Bubu) e Roubério Ribeiro Rolim, por se mostrarem realmente amigos nas horas mais difíceis, obrigado amigos !!

Aos amigos professor Dr. Germano Lambert Torres, professor Dr. Luís Eduardo Borges da Silva, e professor Dr. Ronaldo Rossi por sua amizade e estima.

Ao acadêmico Eduardo Pereira de castro pela ajuda no desenvolvimento dos programas computacionais.

Ao professor Dr. Geraldo Lucio Thiago Filho, professor Dr. Augusto Nelson Carvalho Viana, professor Dr. Zulcy de Souza, Professor José Eugênio Rios Ricci e aos funcionários Rita, Valéria, Zé bombeiro, Oswaldo, Wandir e Pedro, pela acolhida no LHPCH.

Aos amigos do LEPCH, José Manuel Esteves Vicente e José Carlos Grilo Rodrigues pelo incentivo.

Ao professor Dr. Carlos Alberto Dias Coelho pelo empenho em meu favor; e aos funcionários da PPG, Márcia, Cristina, Benê, Carlos e Luciara pela ajuda com toda a parte burocrática.

Aos funcionários da BIM, Luis Carlos, Maurício e Aluísio pela ajuda no “garimpo” de livros e periódicos.

Earth and Sun and Moon

In the morning we will wake up and take to the air
look back at the planet - I`m glued to my chair
southern half id burning as we climb through the sky
sea bird softly falling, smoke way up high

There`s the contours of the mountains, the deserts and plains
and a hurricane is blowing, and it turns once again
now there`s oil spills in the water where Columbus once sailed
and there`s history and misery and it`s rolling away

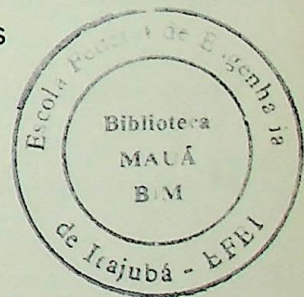
I wish you could see this great mystery
earth and sun and moon human tribe, thin blue line
earth and sun and moon will survive

Sediment is flowing from river to sea
now where are the mighty nations, no lines to be seen
an axe up the broken ground, the sigh of the trees
and its floating in the ether, it brings me to my knees

Too messed up to care
anyone got a wing and a prayer,

In the blink of an eye
thank you and good night

earth and sun and moon human tribe, thin blue line
earth and sun and moon will survive, will survive, we will survive



Resumo

Abstract

CAPÍTULO 1 - Introdução

1.1 Considerações Gerais.....01

1.2 Escopo do Trabalho.....02

CAPÍTULO 2 - Gestão de Carga Pelo Lado da Demanda

2.1 Introdução.....04

2.2 Principais transformações do setor elétrico.....04

2.3 Alguns aspectos sobre políticas de GLD.....05

2.4 Breve histórico da GLD no Brasil.....08

2.5 Situação atual e perspectivas da GLD no Brasil.....09

 2.5.1 Modulação de carga.....10

 2.5.2 Conservação.....13

2.6 Um caso real de implantação de políticas de GLD15

2.7 Controladores de demanda.....17

CAPÍTULO 3 - Determinação da Curva de Carga Típica

3.1 Introdução.....21

3.2 Coleta de Amostras.....22

3.3 Descrição da Metodologia.....	23
3.3.1 Critério de Exclusão Pelas Energias.....	23
3.3.2 Critério de Exclusão Pelo Fator de Carga.....	25
3.3.3 Critério de Seleção Pelos Desvios Médios Quadráticos.....	26
3.4 Aplicação da Metodologia.....	28
3.5 Sistema Dinâmico de Identificação de Curva de Carga Típica.....	31

CAPÍTULO 4 - Conjuntos Difusos

4.1 Introdução.....	36
4.2 Conceito intuitivo de conjuntos “Fuzzy”.....	36
4.3 Definição de conjuntos “Fuzzy”.....	37
4.4 Operações básicas com conjuntos “ Fuzzy ”.....	38
4.4.1 União.....	39
4.4.2 Interseção.....	39
4.4.3 Complementação.....	39
4.4.4 Igualdade.....	40
4.4.5 Soma disjuntiva.....	40
4.4.6 Produto e soma algébrica.....	40
4.4.7 Produto e soma limitada.....	41
4.4.8 Produto e soma drástico.....	41
4.4.9 Exponenciação.....	41

4.4.10 Produto por escalar.....	41
4.4.11 Concentração.....	42
4.4.12 Dilatação.....	43
4.4.13 Intensificação de contraste.....	43
4.4.14 Fuzzificação.....	44
4.5 Relação “ Fuzzy “.....	45
4.5.1 Definição.....	45
4.5.2 Operações utilizando relações “ Fuzzy “.....	46
4.5.2.1 União.....	46
4.5.2.2 Interseção.....	47
4.5.2.3 Produto e soma algébrica.....	47
4.5.2.4 Complementação.....	47
4.5.2.4 Soma disjuntiva.....	47
4.5.3 Composição de duas relações “ Fuzzy “.....	48
4.5.3.1 Definição.....	48
4.5.3.2 Composição MAX-MIN.....	48
4.5.3.3 Composição MAX-Produto.....	48
4.5.3.4 Conjuntos nível - α	48
4.6 Variáveis lingüísticas.....	50
4.6 Tomada de decisão utilizando conjuntos “ Fuzzy “.....	54
4.6.1 Introdução.....	54

4.6.2 Tomada de decisão utilizando técnicas “ Fuzzy “	54
4.6.3 Conceituação do problema.....	55
4.6.4 Objetivos e restrições “ Fuzzy “	57
4.6.5 Análise da decisão “ Fuzzy “	58

CAPÍTULO 5 - APLICAÇÃO DA PROGRAMAÇÃO DINÂMICA E DOS CONJUNTOS DIFUSOS À MODULAÇÃO DE CICLOS DE CARGA

5.1 Introdução.....	61
5.2 A técnica de programação dinâmica.....	62
5.2.1 Princípio da otimalidade de Bellman.....	63
5.3 Formulação do problema.....	65
5.4 Aplicação da TPD a modulação de um ciclo de carga típico.....	65
5.5 PDI - programação dinâmica por intervalos.....	74
5.6 Técnica de programação dinâmica por intervalos associada a teoria dos conjuntos fuzzy.....	86
5.7 Comparação das metodologias PDI e PDIF.....	92

CAPÍTULO 6 - CONCLUSÕES

6.1 Conclusões.....	93
6.2 Sugestão para trabalhos futuros.....	94

REFERÊNCIAS BIBLIOGRÁFICAS.....95

Anexos 1 e 2.....98

CAPÍTULO 2 - GESTÃO DE CARGA PELO LADO DA DEMANDA

TABELAS

tabela 2.1 - Evolução de resultados devido a políticas de GLD 15

tabela 2.2 - redução técnica de demanda 16

FIGURAS

figura 2.1 - custo marginal (CM), custo (C) e custo com GLD (C_{GLD}) 06

figura 2.2 - perfil da curva de carga após implementação das medidas de GLD 17

figura 2.3 - estrutura de sistema de controle de demanda central 19

figura 2.4 - esquema de controle de demanda central e lateral 20

CAPÍTULO 3 - METODOLOGIA PARA DETERMINAÇÃO DE CURVA DE CARGA TÍPICA

TABELAS

tabela 3.1 - exemplo de dados coletados durante um dia típico 23

tabela 3.2 - tabela para seleção de curva de carga típica 24

FIGURAS

figura 3.1 - conjunto de curvas de carga não típicas 25



INDICE DE TABELAS E FIGURAS

CAPÍTULO 2 - GESTÃO DE CARGA PELO LADO DA DEMANDA

TABELAS

tabela 2.1 - Evolução de resultados devido a políticas de GLD.....	10
tabela 2.2 - redução técnica de demanda.....	16

FIGURAS

figura 2.1 - custo marginal (CM), médio (C) e médio com GLD (C_{GLD}).....	06
figura 2.1 - perfil da curva de carga após implementação das medidas de GLD.....	17
figura 2.2 - esquema de sistema de controle de demanda central.....	19
figura 2.3 - esquema de controlador de demanda central e setorial.....	19

CAPÍTULO 3 - METODOLOGIA PARA DETERMINAÇÃO DE CURVA DE CARGA TÍPICA

TABELAS

tabela 3.1 - exemplo de dados coletados durante um dia típico.....	22
tabela 3.2 - tabela para seleção de curva de carga típica.....	26

FIGURAS

figura 3.1- conjunto de curvas de carga sob análise.....	28
figura 3.2 - situação das cargas restantes do primeiro critério de exclusão.....	29

figura 3.3 - situação das curvas restantes do segundo critério de exclusão.....	29
figura 3.4 - curva de carga resultante do critério de exclusão - curva típica.....	30
figura 3.5 - curva de carga típica resultante.....	30
figura 3.6 - esquema do controle dinâmico.....	32
figura 3.7 - exemplo da curva de carga com modulação dinâmica.....	32
figura 3.8 - fluxograma do sistema de identificação da curva de carga típica.....	33
figura 3.9 - fluxograma do sistema dinâmico de identificação da curva de carga típica.....	34
figura 3.10 - processo para determinação de curva típica não modulável.....	35

CAPÍTULO 4 - CONJUNTOS DIFUSOS

FIGURAS

figura 4.1 - representação gráfica dos conjuntos “ Fuzzy “ jovem e velho.....	38
figura 4.2 - representação gráfica da união dos conjuntos “ Fuzzy “ A e B.....	39
figura 4.3 - representação gráfica da interseção dos conjuntos “ Fuzzy “ A e B.....	39
figura 4.4 - representação gráfica do complemento do conjunto “ Fuzzy “.....	40
figura 4.5 - representação gráfica da operação de concentração.....	42
figura 4.6 - representação gráfica da operação de dilatação.....	43
figura 4.7 - algumas operações com conjuntos “ Fuzzy “.....	47
figura 4.8 - exemplos de conjuntos de nível- α	49
figura 4.9 - exemplo de objetivo e restrição “ Fuzzy “.....	57
figura 4.10 - exemplo de decisão “ Fuzzy “.....	57
figura 4.11 - designação de termos lingüísticos à decisão “ Fuzzy “.....	60

CAPÍTULO 5 - APLICAÇÃO DA PROGRAMAÇÃO DINÂMICA E DOS CONJUNTOS DIFUSOS À MODULAÇÃO DE CICLOS DE CARGA

TABELAS

tabela 5.1 - características das variáveis do problema proposto.....	69
tabela 5.2 - arranjo das variáveis da programação dinâmica para realização de cálculos....	70
tabela 5.3 - cálculos do estágio 1.....	70
tabela 5.4 - cálculos do estágio 2.....	71
tabela 5.5 - cálculos do estágio 3.....	71
tabela 5.6 - cálculos do estágio 4.....	72
tabela 5.7 - possibilidades de despacho a cada estágio.....	77
tabela 5.8 - controles viáveis do estágio 1.....	77
tabela 5.9 - controles viáveis do estágio 2.....	78
tabela 5.10 - controles viáveis do estágio 3.....	78
tabela 5.11 - controles viáveis do estágio 4.....	78
tabela 5.12 - modulação PDI para o estágio 1.....	79
tabela 5.13 - modulação PDI para o estágio 2.....	80
tabela 5.14 - modulação PDI para o estágio 3.....	81
tabela 5.15 - modulação PDI para o estágio 3 - continuação da tabela 5.14.....	82
tabela 5.16 - modulação PDI para o estágio 3 - continuação da tabela 5.14 e 5.15.....	79
tabela 5.17 - modulação PDI para o estágio 4.....	83
tabela 5.18 - modulação PDI para o estágio 1 - resultados ótimos.....	84
tabela 5.19 - modulação PDI para o estágio 2 - resultados ótimos.....	84
tabela 5.20 - modulação PDI para o estágio 3 - resultados ótimos.....	85
tabela 5.21 - modulação PDI para o estágio 4 - resultados ótimos.....	85
tabela 5.22 - tabela associativa com carregamento atual.....	88
tabela 5.23 - tabela associativa com carregamento otimizado.....	88

tabela 5.24 -modulação PDIF para o estágio 1 - resultados ótimos.....	90
tabela 5.25 -modulação PDIF para o estágio 2 - resultados ótimos.....	90
tabela 5.26 -modulação PDIF para o estágio 3 - resultados ótimos.....	90
tabela 5.27 -modulação PDIF para o estágio 4 - resultados ótimos.....	91

FIGURAS

figura 5.1 - representação de estratégia para a solução do problema	62
figura 5.2 - possíveis caminhos a serem percorridos.....	64
figura 5.3 -demonstração da escolha do menor custo das trajetórias.....	68
figura 5.4 -ciclo de carga típico a ser modulado.....	69
figura 5.5 - caminhos ótimos com menor perda total.....	72
figura 5.6 - curva de carga convertida (A).....	73
figura 5.7 - curva de carga convertida (B).....	73
figura 5.8 - curva de carga convertida (C).....	74
figura 5.9 - novos conceitos para PD por intervalos.....	75
figura 5.10 - busca do controle ótimo do pseudo-estado 15.....	76
figura 5.11 - curva de carga não modulável.....	76
figura 5.12 - curva de carga após modulação com PDI.....	86
figura 5.13 - resultado da modulação PDIF conforme carregamento das tabelas 5.24 à 5.27.....	91



RESUMO

Este trabalho tem por finalidade apresentar uma ferramenta que possa ser utilizada como uma política de gerenciamento de carga pelo lado da demanda (GLD). Assim sendo, foram definidos os conceitos do gerenciamento de carga pelo lado da demanda e o papel dos equipamentos controladores de demanda aplicados as políticas de GLD.

Na sequência foi desenvolvido um programa computacional dedicado, o qual, através de conceitos estatísticos, torna possível a determinação da curva de carga que melhor representa o comportamento da instalação, denominada por curva de carga típica. Foram elaboradas também conceituações sobre conjuntos difusos, bem como sobre as técnicas de programação dinâmica e a associação destes.

Finalmente, afim de comprovar a validade da proposta do trabalho, foi desenvolvido um outro programa computacional dedicado, o qual permite, através das técnicas de programação dinâmica e conjuntos difusos, a modulação dos ciclos de carga. Desta forma o desenvolvimento dessa nova técnica de modulação de curvas de carga, a qual pode ser incorporada a lógica dos controladores de demanda, permitindo usá-la como uma ferramenta extra para o gerenciamento de carga pelo lado da demanda e em prol da racionalização de energia elétrica.

ABSTRACT

This work presents an alternative tool that can be utilized as a demand side management policy (DSM). This way the demand side management concepts were defined, as soon as demand controllers concepts applied to that policies.

Afterwards it was developed a dedicated computational program with applied statistics in order to determine the better load curve that represents the installation behavior, this one is called by typical load curve. Dynamic programming technics, fuzzy sets concepts and their association were developed.

Finally, in order to complain the work proposal it was developed another dedicated computational program which through dynamic programming technic and fuzzy sets theory allows the load curve modulation.

The development of a load curve modulation technic was complained, the same one can be incorporated to demand controllers logic and so, they can be used as an extra tool for demand side management in order to reach electric energy saving.

1. Considerações Gerais

CAPÍTULO 1

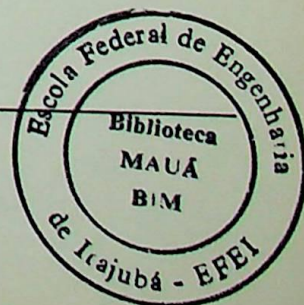
INTRODUÇÃO

1.1 Considerações Gerais

O gerenciamento de carga pelo lado da demanda (GLD), ou “ Demand Side Management - DSM ”, é sem dúvida um assunto que está em alta nos bastidores das concessionárias e no ambiente industrial. Isto se deve em grande parte às incertezas pelas quais o setor elétrico está atravessando, ou seja, falta de recursos para expansão do parque gerador e sistema de transmissão, bem como o possível esgotamento destas. Uma alternativa é racionalizar o uso da energia existente, já que esta é muito mau gerenciada devido a fatores tais como excesso de perdas nos usos finais (iluminação ineficiente, motores sobredimensionados, etc.) e má alocação das cargas ao longo do tempo. O aperfeiçoamento de sua aplicação pode reduzir as perdas. A introdução de tecnologias alternativas, a gestão do uso e o bom dimensionamento no uso final, são ferramentas importantes na conservação de energia.

A realização destas ações fazem parte do conceito da GLD, que orienta seu foco em direção aos consumidores, orientando e implementando ações que tornem possíveis a alteração da magnitude e forma de suas curvas de carga, trazendo consigo os benefícios desejados. A gestão de carga já vem sendo desenvolvida em alguns países europeus e nos EUA há mais ou menos 10 anos e, em alguns destes, estas políticas devem ser oferecidas pelas concessionárias a seus consumidores por força da legislação, que inclusive agregam estes conceitos aos seus planejamentos integrados de recursos (“integrated resources planning”). No caso brasileiro a GLD vem sendo discutida e estudada e ainda têm poucas aplicações significativas.

Neste contexto, pretende-se aqui desenvolver uma ferramenta que auxilie a execução das políticas de GLD. Esta ferramenta tem por função a modulação de curvas de carga de instalações industriais, afim de promover benefícios não só para a concessionária mas também à própria indústria. A execução desta tarefa demanda o conhecimento de várias características da instalação e do processo produtivo, portanto a performance da modulação dependerá da cooperação entre a concessionária e o consumidor.



1.2 Escopo do Trabalho

Afim de promover um completo estudo relacionado ao desenvolvimento de uma ferramenta que auxilie as políticas de GLD, esta dissertação foi dividida nos seguintes capítulos :

Capítulo 1 - Introdução

É apresentado neste capítulo a conceituação do problema, o que está sendo feito para sua solução, bem como descreve a proposta da dissertação.

Capítulo 2 - Gestão de carga pelo lado da demanda

Este capítulo faz uma breve abordagem sobre a história do GLD no contexto mundial e nacional discutindo, seus aspectos, sua situação atual e suas perspectivas, bem como a análise das ferramentas para sua execução. Mostra-se também um exemplo real da GLD executado pela CEMIG/PROCEL no vale do Jequitinhonha - MG e, finalmente, apresenta o funcionamento de um controlador de demanda.

Capítulo 3 - Metodologia para determinação da curva de carga típica

A proposta deste capítulo é obter uma curva de carga que melhor represente o comportamento de uma instalação industrial. Isto é conseguido através do uso de ferramentas estatísticas, bem como através do desenvolvimento de um programa computacional dedicado a esta ação. Um exemplo real é realizado para comprovação desta metodologia.

Capítulo 4 - Conjuntos Difusos

Como parte da proposta desta dissertação é a incorporação dos conjuntos fuzzy à programação dinâmica, dedicou-se este capítulo para descrever os conceitos, propriedades e definições desta teoria, afim de utiliza-lá na tomada de decisão sobre a modulação da curva de carga. Diferentemente da programação dinâmica, os conjuntos difusos merecem destaque, face ao pouco conhecimento desta teoria.

Capítulo 5 - Aplicação da programação dinâmica e dos conjuntos difusos à modulação de ciclos de carga

Este capítulo tem a finalidade de apresentar a técnica de programação dinâmica (PD) em sua forma convencional desenvolvida por R. Bellman, bem como o desenvolvimento de uma técnica dedicada ao problema em questão, designada por programação dinâmica por intervalos (PDI), e também apresentar a incorporação dos conjuntos difusos à PDI. São mostrados exemplos de aplicação das técnicas à modulação de ciclos de carga, exemplificando a proposta da dissertação através de programa computacional dedicado.

Capítulo 6 - Conclusão

Finalmente, este capítulo aborda os resultados encontrados através das diferentes metodologias desenvolvidas. Apresenta-se também sugestões para novos trabalhos.

Introdução

CAPÍTULO 2

GESTÃO DE CARGA

PELO LADO DA DEMANDA

2.1 Introdução

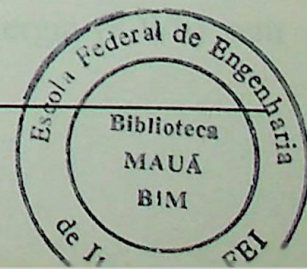
O setor elétrico brasileiro experimentou várias fases desde sua criação, destacando-se entre elas duas marcantes etapas, a primeira diz respeito ao período antes da crise do petróleo (1973-78), fase esta caracterizada pelo grande número de projetos e construções na geração, transmissão e distribuição (GT&D), que proporcionava grande oferta de energia elétrica; a segunda fase relevante ocorreu após 1978, e foi influenciada e marcada pela falta de recursos e as grandes mudanças tecnológicas, além das incertezas do mercado consumidor e ofertante.

Foi devido a tais aspectos que começaram a surgir várias mudanças tecnológicas e de gestão, alternativas essas que proporcionava menores custos e/ou riscos. Dentre essas, pode-se citar as pequenas centrais hidrelétricas (PCH'S), turbinas à gás com ciclo combinado e as atuais políticas de gestão de carga pelo lado da demanda (GLD).

2.2 Principais transformações do setor elétrico

As razões das reformas no setor elétrico, no caso de países em desenvolvimento, estão mais relacionadas com o problema de restrição de recursos para investimentos, do que com as preocupações técnicas com a expansão do setor [1]. O crescimento do número de produtores independentes de energia associado às inovações na área de geração, têm tornado atrativas as opções como co-geração e PCHS. Observa-se uma tendência de aumento da competição no nível da produção, elevando o risco para as empresas que operam no setor.

Neste novo padrão de concorrência [2], os ganhos de eficiência estão associados à flexibilidade da empresa energética para adaptar os seus serviços à características da demanda, à baixo custo. No contexto de elevação de riscos, a saída das concessionárias (principalmente nos EUA) para postergar investimentos na elevação de oferta e elevação das tarifas foi a gestão da demanda e a implementação de políticas de conservação de



energia. Neste caso, a cooperação com os clientes é essencial para a gestão da demanda, mais especificamente, da curva de carga.

As políticas de GLD, ou DSM(Demand Side Management), se inserem neste novo contexto [2] do setor elétrico. O paradigma tecnológico da eletrônica tem impacto importante; o desenvolvimento de medidores eletrônicos, novos meios de transmissão de dados, e a própria tecnologia da informática permitem um melhor acompanhamento e monitoramento da demanda de energia. Atualmente é possível para a concessionária saber a demanda de cada consumidor, a qualquer momento. Esta inovação permitiu também aos próprios usuários conhecerem e monitorarem a seu consumo de eletricidade, bem como a tarifa em tempo real. Nem todas essas novas tecnologias são acessíveis a baixo custo, fato que as torna ainda de difícil difusão no Brasil.

2.3 Alguns aspectos sobre políticas de GLD

A GLD pode ser considerada como uma inovação na forma de operação do setor elétrico. Administrar a demanda significa monitorar e atuar na utilização da energia nos mais diversos tipos de aplicação. O acompanhamento das formas de utilização da energia é uma atividade diferente das atividades produtivas tradicionais no setor e requer outras formas de capacitação tecnológica. Isto é : capacitação para pesquisar o mercado consumidor, realizar diagnósticos energéticos e otimizar a demanda. Portanto, ao invés de vender energia elétrica, a concessionária passa a oferecer serviços para otimização do consumo de seu cliente. Assim, as empresas produtoras de energia passam a prestar um novo serviço aos seus consumidores, uma vez que este serviço pode ser oferecido por esta, embora empresas independentes de serviços de energia também possam realizar este tipo de serviço (ESCO - energy service companies).

Várias concessionárias no Brasil e no exterior adotam atualmente algum tipo de política de GLD. Algumas políticas certamente contribuem para a elevação da rentabilidade das concessionárias. A gestão da curva de carga é um exemplo deste tipo de política. A modulação da curva de carga permite à concessionária vender mais energia (KWh) sem

eivar na mesma proporção os investimentos em capacidade de geração (KW). Ou seja, a empresa mantém o seu custo médio fixo e, portanto, aumenta a rentabilidade. Assim, as concessionárias têm um grande incentivo para modular a curva de carga de seus principais clientes, introduzindo inclusive incentivos tarifários. Tais incentivos, em geral, estão relacionados com uma forte tarifação sobre a demanda (KW), com a criação de patamares diferenciados para horários e estações de pico (tarifa horosazonal) e tarifas especiais para energia interruptível.

O objetivo destes vários tipos de tarifas é incentivar os consumidores a implementar políticas adequadas às características das concessionárias, transferindo parte da demanda para horários adequados.

Entretanto, existem outros tipos de políticas de GLD que resultam em conservação de energia que não necessariamente implicam na modulação da curva de carga. Políticas para difusão de equipamentos eficientes, campanhas de informação, podem acarretar numa elevação da eficiência no uso final de energia e na redução do ritmo de crescimento da demanda, e até na redução das vendas de eletricidade. Neste caso, a rentabilidade da conservação para a concessionária não é óbvia. Esta vai depender da relação entre custos e a tarifa de energia elétrica.

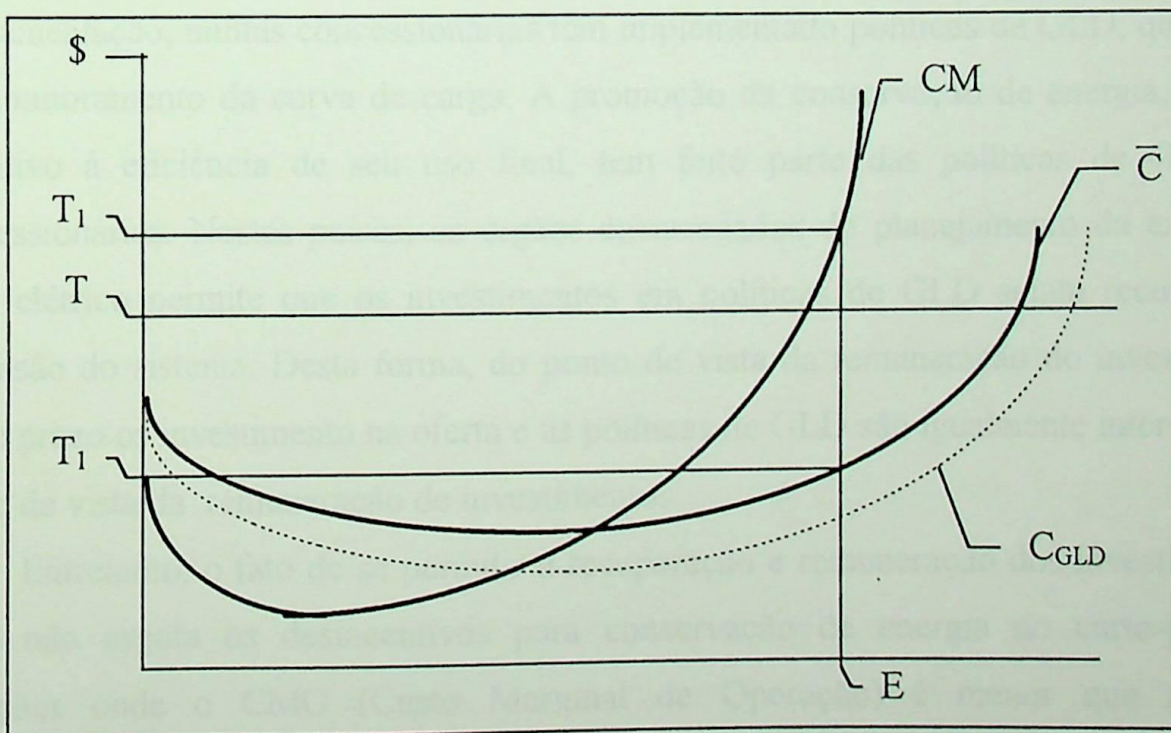


figura 2.1 - custo marginal (CM), médio (C) e médio com GLD (C_{GLD})

Pode-se dizer que uma concessionária terá uma tarifa (T) entre o custo marginal (CM) e o custo médio (C). Caso prevaleça a teoria de livre mercado, a tarifa tenderá ao custo marginal; no caso de se fixar a rentabilidade, ter-se-á o custo médio como base tarifária.

A GLD será interessante sempre que o seu custo for inferior ao custo marginal, embora isto possa reduzir as receitas bruta e líquida da empresa. Então, qual a vantagem da GLD? A resposta está na rentabilidade e no risco do investimento. Quando a GLD custa menos que o custo marginal de expansão, a rentabilidade da empresa cresce, mesmo que diminua seu lucro, pois reduz o investimento necessário.

Sob o ponto de vista de risco, a GLD, por estar associada a um elevado número de contratos com consumidores, promove uma diversificação, comparativamente a se ter todo o capital investido em uma só alternativa.

Porém, quando o custo da GLD for superior ao custo marginal de operação, e o sistema tiver folga entre demanda e oferta, não convém realizar tais contratos.

Esses casos se caracterizam apenas no curto prazo, enquanto os contratos de GLD são recomendados para médio e longo prazos.

Em muitos países industrializados, como os EUA e Canadá, por força da regulamentação, muitas concessionárias têm implementado políticas de GLD, que vão além do monitoramento da curva de carga. A promoção da conservação de energia, através do incentivo à eficiência de seu uso final, tem feito parte das políticas de GLD destas concessionárias. Nestes países, os órgãos encarregados do planejamento da expansão do setor elétrico permite que os investimentos em políticas de GLD sejam recuperados na expansão do sistema. Desta forma, do ponto de vista da remuneração do investimento no longo prazo os investimento na oferta e as políticas de GLD são igualmente interessantes do ponto de vista da remuneração de investimentos.

Entretanto, o fato de se permitir a recuperação e remuneração dos investimentos em GLD não esgota os desincentivos para conservação de energia no curto-prazo. Nas situações onde o CMO (Custo Marginal de Operação) é menor que a tarifa, a

concessionária tem incentivos para aumentar as vendas de energia. Desta forma, configura-se um conflito entre o lucro de curto prazo e a rentabilidade das ações de GLD.

Percebe-se que as políticas de GLD são discutíveis e dependem muito de como são aplicadas, o sucesso de tais políticas no entanto trazem a redução de riscos à concessionária, como já foi dito.

2.4 Breve histórico da GLD no Brasil

O Brasil não pode ser considerado um país exemplar na utilização de políticas de GLD. Entretanto, algumas ações de GLD importantes já foram implementadas no país. O programa de GLD que se destaca é a política para achatamento da curva de carga, através da estrutura tarifária (tarifa horosazonal). O PROCEL (Programa Nacional de Conservação de Energia Elétrica) é coordenado pela Eletrobrás e tem por objetivo a promoção de políticas de conservação de energia elétrica, embora não seja uma ação direta entre concessionária e consumidor. Esse já promoveu programas de conservação tais como : programas de auditoria energética na indústria, etiquetagem de geladeiras e motores elétricos, troca de lâmpadas e aumento da eficiência energética na iluminação pública, entre outros. Pode-se afirmar que a atuação do PROCEL está relacionado muito mais com o melhoramento do conhecimento disponível em relação às tecnologias e formas de utilização da eletricidade no país, do que com o próprio montante de energia conservada até o momento. O mesmo poderia evoluir financiando as concessionárias e consumidores para ações reais de GLD.

O planejamento do setor elétrico no país até hoje, tem dado pouca atenção à conservação de energia como um dos recursos para o atendimento às necessidades energéticas nacionais. Ao contrário do exemplo americano, não existe um monitoramento e uma fiscalização dos órgãos reguladores para garantir que as metas de conservação planejadas sejam atingidas pelas concessionárias. Desta forma, a conservação de energia elétrica no país tem dependido do voluntarismo e do real interesse de algumas concessionárias.

2.5 Situação atual e perspectivas da GLD no Brasil

As políticas de GLD têm, como objetivo primordial, a realização de ações que promovam a alteração na magnitude e forma da curva de carga de uma instalação, seja esta instalação residencial, comercial, industrial e da própria concessionária, de forma que advenham destas alterações benefícios financeiros e operacionais entre as partes envolvidas no processo. Sendo assim, pode-se dividir as políticas de GLD em duas categorias :

a) atividades de modulação de carga - ações que em suma pretendem deslocar carga para horários mais convenientes.

b) atividades de conservação - ações que promovem o uso mais eficiente de energia elétrica de maneira global.

Em ambas modalidades de GLD, o que deve ser caracterizado é que tais ações devem ser realizadas pelas duas partes, de forma que tenham planejamento completo e profundo, levando em conta parâmetros relacionados a financiamento para o desenvolvimento e dos projetos, assim como uma análise dos custos e benefícios da implantação das atividades de GLD, visando boa rentabilidade para ambos.

No caso isolado da concessionária, é necessário ressaltar que esta deve levar em conta as políticas de GLD para o cálculo de seus custos marginais de operação e expansão (CMO e CME). Nos EUA [3], estas políticas são incluídas nos programas de planejamento com mínimos custos (“Least Cost Planning”) por força da legislação, que entre outras, determinam que as tarifas praticadas pelas concessionárias, devem refletir os custos para diferentes horas do dia e que as mesmas deveriam oferecer aos consumidores algumas opções de modulação de carga.

Estas políticas estão caminhando para o planejamento integrado de recursos (“Integrated Resources Planning”), que é a tendência mundial das empresas de energia. Tal planejamento inclui questões tais como GLD, cogeração e conservação de energia elétrica.

Dentro dos cenários “privatista” e “competitivo”, uma concessionária brasileira [4] admite as hipóteses da tabela 2.1 como possíveis resultados da implantação de programas de GLD até o ano 2015.

	1995		2005		2015
			P	P	P
	0%		3%	4%	5%
	0%		1%	2%	3%

Tabela 2.1 - Evolução de resultados devido a políticas de GLD [4].

O setor industrial é o que melhor responde aos anseios da implantação de políticas de GLD. Para iniciar-se uma análise de qual política de GLD melhor se adapta a cada empresa, é necessário construir um banco de dados com as curvas de carga da mesma, de maneira que seja possível compreender o comportamento do sistema e correlacioná-lo com o comportamento do processo produtivo. Desta maneira é possível identificar as cargas responsáveis pelas pontas da curva de carga. Do ponto de vista da produção, as cargas de melhor e maior flexibilidade são aquelas não associadas a processos contínuos.

2.5.1 Modulação de carga

Foi com a introdução da tarifação horosazonal que pretendeu-se a modulação de carga, embora esta ação não tenha obtido grandes resultados. Verificou-se apenas algumas mudanças no horário de ponta da curva de carga do sistema interligado, mas não o suficiente para melhoria do fator de carga. Este tipo de tarifação, apresenta entre muitas, as seguintes características :

a) Um segmento horário fora de ponta e outro de ponta

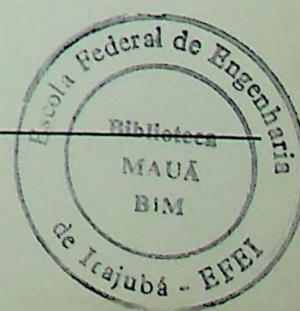
Estes segmentos horários tem embasamento no grande pico de demanda exigidos ao longo do dia, principalmente na faixa horária compreendida entre as 17:00 h e as 20:00 h. Como este tem sobrepreços de até 300%, deveria fazer com que os consumidores modulassem sua curva de carga, ceifando os picos de carga e preenchendo os vales, sempre que possível, o que na prática não ocorreu.

b) Um período seco e um período úmido

Estes períodos vem caracterizar a grande ou pouca disponibilidade de água nos reservatórios de água das plantas geradoras, que dizem respeito a disponibilidade, baixa ou alta, de energia elétrica para ser vendida. O período seco compreende os meses de abril a outubro e o período úmido compreende os meses de novembro a março, na região sudeste.

Baseado nestas características de tarifação do sistema elétrico, pretendia-se que o consumidor passasse a modular sua curva de carga, tendo em vista os sobrepreços, principalmente no horário de ponta. Desta forma, o consumidor passaria a estudar melhor o seu processo produtivo e remanejar cargas de seu processo, de forma que estas, preferencialmente, saíssem do horário de ponta, para o horário fora de ponta. Sendo assim, o consumidor, desfrutaria da vantagem da redução da fatura de energia elétrica; e a concessionária, da redução de suprimento no horário de ponta, redução das perdas em todo o sistema e menor risco de interrupção de energia elétrica; exigindo assim, menores investimentos para suprir estas deficiências.

Contudo, para que a modulação da curva de carga tenha sucesso, é preciso um detalhado estudo do processo produtivo em questão. Algumas das formas de se modular a curva de carga da instalação, são :



a) Uso de sistemas de aquecimento bi-energéticos

Estes são sistemas que empregam outras formas de energéticos mais baratas que a energia elétrica [4], e seus custos de operação variam de acordo com a eficiência dos equipamentos e os preços dos combustíveis empregados.

Um bom exemplo desta alternativa foi a economia de cerca de US\$ 500 mil obtida para a Autolatina com a utilização de energia temporária para substituição (ETST), no horário fora de ponta, em lugar de óleo BPF no ano de 1993, através da compra de 128 Gwh de ETST da ELETROPAULO, essa empresa deixou de queimar 10 mil Ton. de óleo. Em consequência deste contrato houve uma redução do consumo de óleo de 16,5 mil ton (1991) para 6,5 mil ton.

b) Re-alocação de processos

O objetivo desta modalidade de modulação não é economizar energia elétrica. Na verdade, esta visa tão somente deslocar cargas do período de ponta para o período fora de ponta, reduzindo os custos com a fatura de energia, permitindo desta maneira uma melhoria no fator de carga. Os processos produtivos a serem re-allocated, têm que ser flexíveis. Embora não seja o objetivo principal, a melhoria do fator de carga reduz as perdas no sistema.

Um exemplo bem sucedido de re-alocação de processos, foi concebido nas instalações da SKF&Dormer Tools S.A.[4] que, através da redistribuição de turnos de trabalho, possibilitou a remoção de operações no horário de ponta, trocando-as por horários de refeições. O grande atrativo desta ação são os relativos baixos custos de realização.

c) Auto-produção

A operação das instalações auto-produtoras pode auxiliar de maneira fundamental na modulação da curva de carga vista pela concessionária e do próprio consumidor, já que estas instalações poderiam entrar em funcionamento preferencialmente no horário de ponta, melhorando o seu fator de carga e liberando capacidade para a concessionária.

Outras formas de modulação podem e devem ser executadas de forma à atender ações de GLD, mas sempre levando-se em conta o aspecto custo versus benefício destas opções, bem como os riscos inerentes.

2.5.2 Conservação

As ações de conservação tem sido incentivadas, principalmente pelo PROCEL. O desperdício de energia elétrica no setor industrial é muito significativo, tanto para o consumidor quanto para a concessionária [5]. Algumas formas deste desperdício são :

- a) motores elétricos sobredimensionados;
- b) motores com baixo rendimento;
- c) uso de iluminação a níveis superiores ao necessário;
- d) uso de sistemas de iluminação ineficientes;
- e) desperdício de energia em sistemas de aquecimento;
- f) falta de isolamento térmico de fornos elétricos.

Como já foi citado, foi neste sentido que o PROCEL teve certa participação na conservação e racionalização de energia elétrica de forma global, principalmente a nível industrial, sendo algumas de suas principais ações, as seguintes :

- a) programa de substituição de lâmpadas;
- b) programa de etiquetagem de motores elétricos;
- c) manual de autodiagnóstico de desperdício de energia elétrica na indústria.

Estas ações contribuem também para a modulação, além de reduzir os custos de produção dos produtos, que ficam evidenciados nos processos energo-intensivos, sendo exemplo significativo as indústrias beneficiadoras de alumínio, onde os custos associados à energia elétrica, podem chegar à ordem de 70 %.

Nos EUA, estima-se o potencial de conservação de longo-prazo entre 24 e 38% [4]. O melhoramento da eficiência na força motriz seria o principal responsável por esta conservação.

Em um outro trabalho [6], é mostrado que a conservação de energia elétrica, através da substituição de lâmpadas incandescentes por outras de melhor eficiência, conduziria a uma economia de 3 TWh no consumo até o ano 2010.

Um recente projeto da CPFL, em conjunto com a UNICAMP [7], tratou sobre o desenvolvimento de pré-aquecedores solares. Estes equipamentos funcionam de forma a pré-aquecer a água destinada a chuveiros elétricos, fazendo com que estes gastem menor quantidade de energia para aquecer a água do que de maneira convencional. O desenvolvimento do projeto se justifica pelo grande impacto que o chuveiro representa sobre o horário de ponta. O pré-aquecedor foi testado em laboratório e em campo, através de projeto piloto que instalou tais equipamentos em algumas residências, associando-se registradores eletrônicos programáveis (REP'S) para que fosse monitorada a curva de carga diária e a redução do consumo de energia elétrica. Um dos pontos mais importantes a se destacar é o aproveitamento da energia solar, que é gratuita e abundante.

Um outro ponto a ser analisado pelo lado das concessionárias, e principalmente de toda sociedade, é a redução os impactos ambientais proporcionados por estas ações de conservação e racionalização de energia elétrica .

2.6 Um caso real de implantação de políticas de GLD

Para melhor elucidar as políticas de GLD, será descrito um caso real da aplicação destas ações, que compreendem a modulação da demanda bem como a conservação e racionalização de energia elétrica. O caso analisado foi objeto de um estudo feito pela parceria PROCEL/CEMIG [8].

A região alvo do estudo foi o Vale do Jequitinhonha, localizada no nordeste do estado de Minas Gerais, sendo uma região carente de recursos e de desenvolvimento industrial fraco. Os investimentos para o seu melhor atendimento estão esgotados e o reforço necessário à sua boa performance é da ordem de US\$ 25 milhões. Atualmente a citada região é atendida por uma linha de transmissão de 138 KV com capacidade para 42 MVA contínuos. Seu carregamento encontra-se em 123% ocasionando assim sérios problemas com níveis de tensão e suprimento.

Para melhorar o atendimento e até possibilitar a expansão do mercado com os mesmos recursos disponíveis, pretende-se reduzir a demanda de potência elétrica no horário de ponta em cerca de 9 MW. Dessa maneira se conseguiria postergar investimentos na transmissão, subtransmissão, subestações e na distribuição através de ações tais como :

- a) redução de carga na ponta;
- b) diluição da carga do horário de ponta para fora de ponta;
- c) conservação estratégica de energia;
- d) modulação de picos horosazonais (flexibilidade da curva de carga).

Para alcançar este intuito, foram analisadas as curvas de carga atuais por segmento de mercado, estudados o histórico de crescimento da região, assim como identificação dos consumidores e seus hábitos de consumo. Desta forma estabeleceram-se as seguintes ações a serem tomadas :

- a) substituição de lâmpadas incandescentes por fluorescentes compactas;
- b) desenvolvimento de estudos, visando à modulação de carga dos consumidores no horário de ponta e seu uso racional;
- c) gerenciador de carga no horário de ponta;
- d) aquecimento de água (aquecimento solar, aquecedor de baixa potência, 400 W - 200 litros cumulativo);
- e) avaliações energéticas, visando o uso racional de motores, iluminação, ar condicionado e refrigeração;
- f) quantificação de resultados;
- g) análise do custo/benefício.

Ainda foram implantadas as seguintes ações informativas :

- a) campanhas publicitárias;
- b) palestras, cursos e treinamentos.

Além disto foram implantados REP'S para monitoração da evolução do projeto piloto. Os gastos gerais com todo projeto foram da ordem de US\$ 1.430.000,00. A redução técnica divide-se conforme tabela 2.2.

SECTOR	ILUM	CHUVA			
	3 MW	1 MW	1,5 MW	0,5 MW	0,5 MW
TOTAL					

tabela 2.2 - redução técnica de demanda

Assim, obteve-se uma redução de 9 MW de demanda de potência, o que significa uma vantagem se comparado com US\$ 25 milhões para expansão do sistema. Na figura 2.2

pode-se constatar a curva de carga inicial bem como a curva resultante após as ações de GLD.

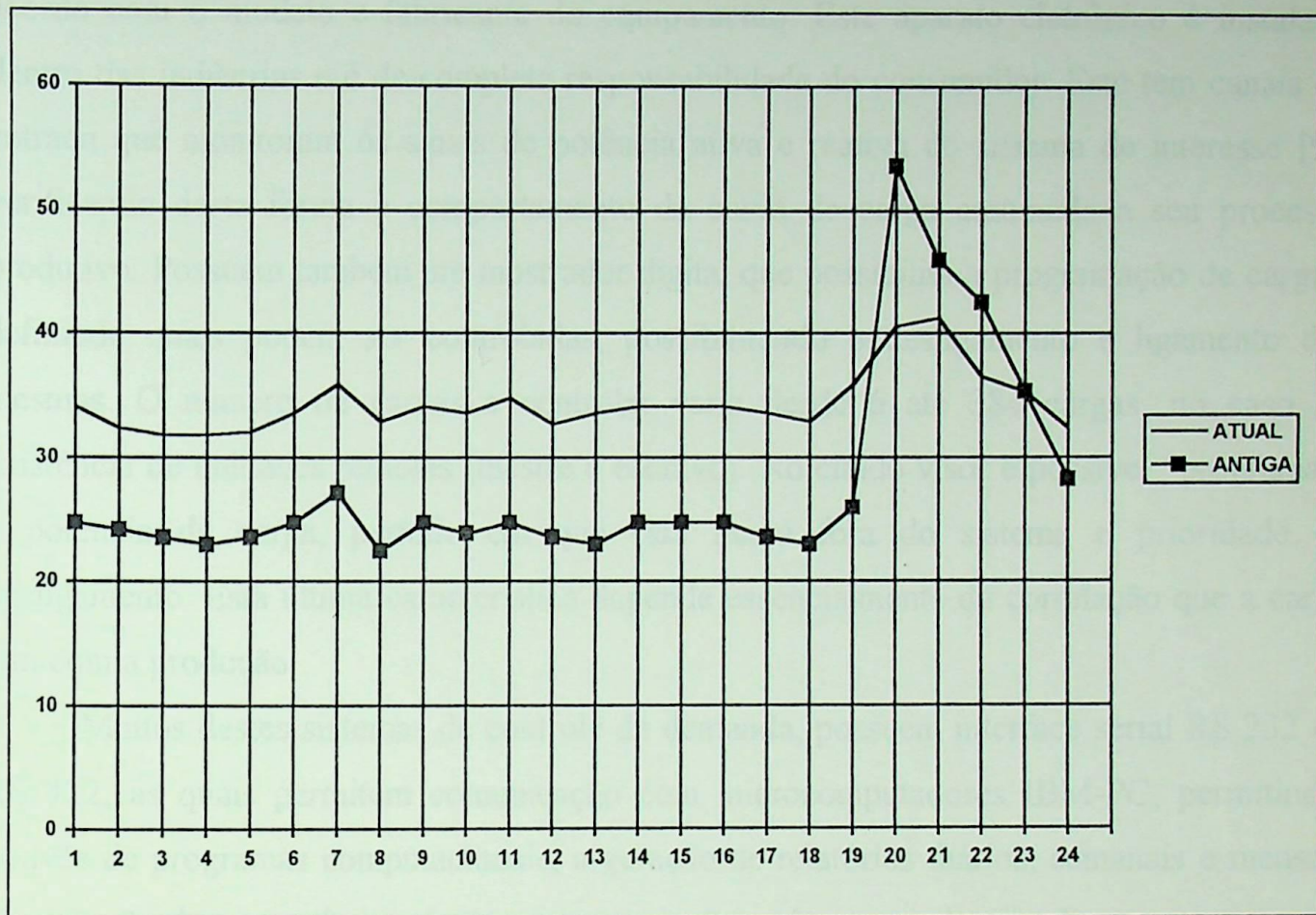


figura 2.2 - perfil da curva de carga após implementação das medidas de GLD

2.7 Controladores de demanda

Uma boa e eficiente maneira de realizar a modulação da curva de carga de instalações elétricas, se faz através do uso de controladores de demanda [9]. Estes equipamentos foram difundidos no Brasil, por volta da década de 80, visando atender as necessidades do mercado, e em consequência da implantação da tarifa horazonal, buscando evitar, dentre outros, a :

- ultrapassagem de valores contratados de demanda, no horário e fora de ponta;
- ultrapassagem de valores de consumo na ponta e fora de ponta;
- ultrapassagem da demanda contratada.

A filosofia de funcionamento de equipamentos controladores de demanda é bastante simplificada, sendo que sua topologia e capacidade de atendimento de cargas varia de acordo com o modelo e fabricante do equipamento. Este aparato eletrônico é instalado dentro das indústrias e é de completa responsabilidade do consumidor. Este tem canais de entrada que monitoram os sinais de potência ativa e reativa do sistema de interesse [9], verificando desta forma o comportamento da curva de carga associada a seu processo produtivo. Possuem também um mostrador digital que possibilita a programação de cargas, definindo quais podem ser controladas, possibilitando o desligamento e ligamento das mesmas. O número de cargas a controlar varia desde 6 até 384 cargas, no caso da existência de unidades remotas (mestre e escravo). No citado visor é possível “programar” a potência da carga, período em que esta ficará fora do sistema e prioridade de desligamento. Esta última característica depende essencialmente da correlação que a carga tem com a produção.

Muitos destes sistemas de controle de demanda, possuem interface serial RS 232 ou RS 422, as quais permitem comunicação com microcomputadores IBM-PC, permitindo, através de programas computacionais, a geração de relatórios diários, semanais e mensais da situação da curva de carga, com os principais parâmetros relacionados a esta, ou seja, dados acumulados por segmento horosazonal (ponta e fora de ponta) - consumo, demanda máxima, fator de potência e fator de carga de modo que se possa acompanhar a evolução do sistema e corrigir ou alterar característica de seu funcionamento.

Estes sistemas, podem ser instalados também setorialmente, monitorando diversas partes da planta industrial, sendo que estes estão interligados ao controlador principal [10]. Existe entre eles um sistema de comunicação, interligado por computador, que permite ações conjuntas ou isoladas, obtendo assim amplo acesso e controle das cargas a controlar. Nas figuras 2.3 e 2.4 são mostrados esquemas das possíveis maneiras de se instalar estes controladores de demanda.



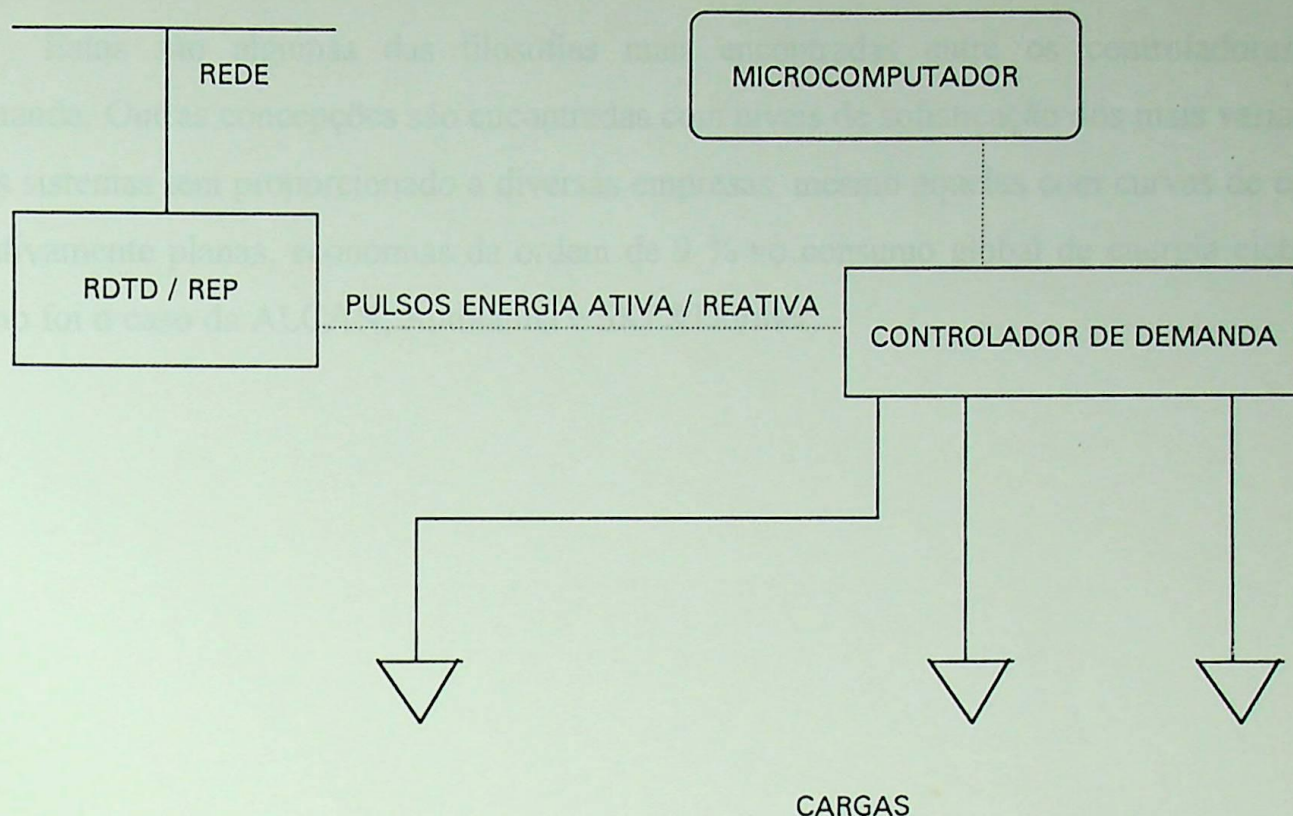


figura 2.2 - esquema de sistema de controle de demanda central

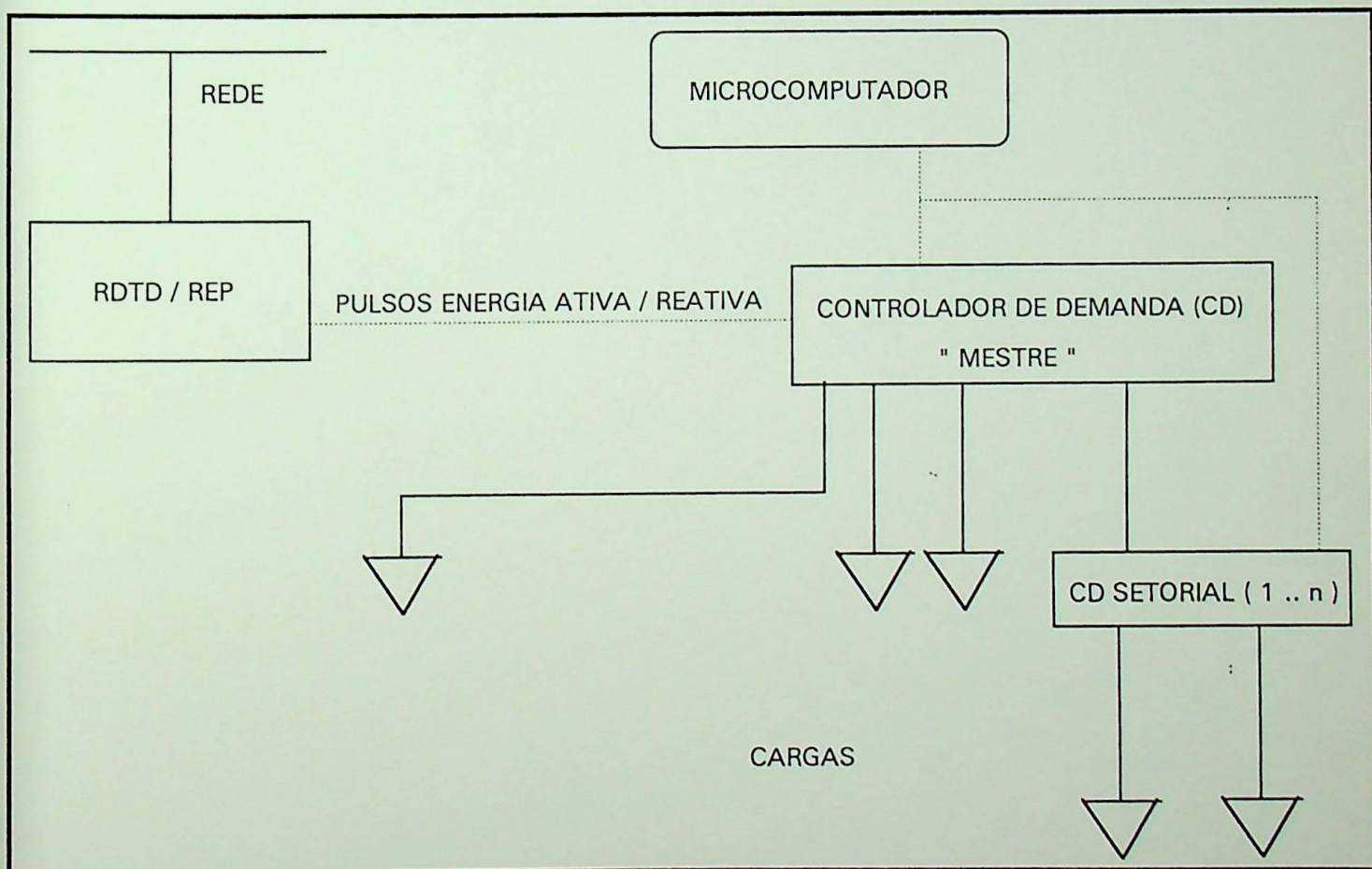


figura 2.3 - esquema de sistema de controle de demanda central e setorial

Estas são algumas das filosofias mais encontradas entre os controladores de demanda. Outras concepções são encontradas com níveis de sofisticação dos mais variados. Tais sistemas tem proporcionado a diversas empresas, mesmo aquelas com curvas de carga relativamente planas, economias da ordem de 9 % no consumo global de energia elétrica, como foi o caso da ALCAN, Autolatina e SKF&Dormer.

CAPÍTULO 3

METODOLOGIA PARA

DETERMINAÇÃO DA

CURVA DE CARGA TÍPICA

CAPÍTULO 3

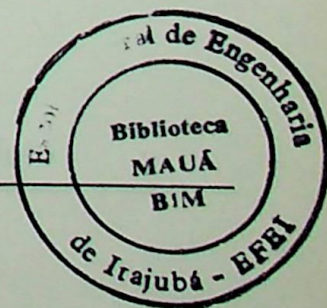
METODOLOGIA PARA DETERMINAÇÃO DA CURVA DE CARGA TÍPICA

3.1 Introdução

Neste capítulo propõe-se uma metodologia para a determinação da curva de carga típica, de forma que esta caracterize a operação de uma instalação e permita a alocação de carga otimizada. A metodologia proposta trata estatisticamente os dados obtidos a partir do levantamento de curvas de carga que pode ser executado pelo próprio consumidor. A coleta de dados é facilitada utilizando-se do registrador digital de tarifa diferenciada (RDTD), registrador digital de média tensão (RDMT), controladores de demanda (CD), analisador de energia (AEN) ou registrador eletrônico programável (REP). Estes, em suma, são equipamentos que acumulam pulsos num dado intervalo de tempo especificado [11], determinando a potência média e portanto, a energia no intervalo. Este valor de potência é alojado na memória de massa, seguindo tabelas horárias, diárias e mensais. Possuem memória de massa tipo RAM com bateria com autonomia de 120 horas, que atua quando há falta de energia elétrica.

Visto que pretende-se obter uma única curva de carga típica através de procedimentos estatísticos, é preciso levar-se em conta o número de amostras a ser analisado. Ressalta-se também que um sistema elétrico, na maioria dos casos, possui diferentes tendências de comportamento no que se refere a utilização de potência ao longo do tempo, ou seja, existem períodos onde este comportamento evolui de forma semelhante - Como exemplo, os dias úteis da semana, os sábados e os domingos formam três grupos de comportamento.

Note-se que naqueles sistemas onde o fator produção determina uma sazonalidade do uso de energia, é conveniente identificar e separar tais períodos em análise diferenciados; pode-se citar como representativo destes casos, as usinas de açúcar e destilarias de álcool, entre outros, onde o comportamento sazonal é pronunciado.



3.2 Coleta de Amostras

As amostras de curvas de carga diárias são coletadas das memórias de massa dos instrumentos de medição da própria concessionária instalados nos consumidores (RDTD, RDMT, CD, AEN, REP) ou por outro equipamento qualquer, apropriado para tal fim.

A tabela 3.1 exemplifica o formato dos dados contidos na memória de massa, sendo que se registra aí apenas a potência média e o fator de potência a cada 15 minutos. Nesta tabela tem-se os dados de um dia completo (24 horas).

KW	fp	KW	fp	KW	fp
42368.0	86.7	42368.0	86.7	47688.0	85.7
41528.0	87.0	42648.0	86.3	47128.0	85.7
40408.0	86.7	42648.0	86.3	46848.0	86.0
38168.0	87.0	42928.0	86.7	47128.0	85.7
37328.0	87.3	42088.0	86.3	46848.0	85.3
33688.0	88.7	42648.0	86.7	46288.0	85.3
33128.0	88.3	42368.0	87.0	46008.0	85.7
37328.0	88.7	42648.0	87.7	45448.0	85.7
37048.0	88.0	43208.0	87.3	45168.0	85.7
37608.0	88.0	42368.0	87.3	44888.0	86.0
33968.0	87.3	42368.0	87.3	45448.0	86.0
37048.0	87.0	42088.0	87.0	46008.0	86.0
42088.0	86.3	45728.0	85.6	45728.0	85.7
42088.0	86.4	48248.0	84.7	46848.0	86.0
41808.0	86.0	47688.0	85.0	46568.0	86.3
43208.0	86.0	47688.0	85.3	45728.0	86.0
43768.0	86.3	47968.0	85.0	45728.0	85.7
41808.0	87.0	46288.0	85.3	44888.0	85.3
40968.0	87.3	46288.0	86.0	45448.0	85.7
41528.0	87.0	48248.0	85.3	43488.0	86.7
41528.0	87.0	47968.0	85.3	45168.0	85.3
41808.0	86.7	47688.0	85.0	46568.0	85.3
40688.0	87.3	47408.0	85.7	46288.0	85.3
40968.0	87.0	46848.0	86.0	47688.0	85.3
41248.0	87.0	47408.0	85.7	47128.0	84.3
42088.0	86.0	46848.0	85.3	47408.0	85.0
41808.0	86.0	47968.0	85.3	47128.0	85.0
40968.0	87.0	47968.0	85.7	46848.0	84.7
41248.0	87.0	47688.0	85.7	47968.0	85.0
41248.0	87.3	47968.0	85.3	47688.0	84.7
40968.0	87.3	47688.0	85.7	46848.0	84.7
40968.0	87.3	47688.0	85.3	46568.0	85.3

tabela 3.1 - exemplo de dados coletados durante um dia completo

3.3 Descrição da Metodologia

Entende-se aqui como sendo a curva de típica do sistema, para um determinado conjunto de dias semelhantes, aquela que apresenta menor desvio em relação às demais.

Obtidas as curvas de carga diárias de um sistema durante um período pré-determinado, a curva de carga típica será determinada através de dois processos de exclusão e um de seleção [12]. Os critérios de exclusão são :

- a) o das energias;
- b) o do fator de carga.

O critério de seleção, aplicado sobre as curvas restantes dos critérios de exclusão, é :

- a) o do desvio médio quadrático.

3.3.1 Critério de Exclusão Pelas Energias

Este critério baseia-se no pressuposto que as energias ativa e reativa diárias se mantêm próximas, desde que os dias em análise tenham as mesmas características.

Desta forma, utilizando-se do histórico energético da instalação, tem-se :

$$\bar{C}_d = \frac{\sum_{i=1}^n C_{di}}{n} \quad (3.1)$$

Onde :

\bar{C}_d - consumo médio diário (ativo ou reativo) [KWh ou Kvarh]

C_{di} - consumo para o i-ésimo dia (ativo ou reativo) [KWh ou Kvarh]

n - número de dias

Assim, \bar{C}_d representará o consumo médio de energia ativa e reativa referente ao número de dias analisados. O desvio padrão (σ_{Cd}) da amostra será :

$$\sigma_{Cd} = \sqrt{\frac{\sum_{i=1}^n (C_{di} - \bar{C}_d)^2}{n-1}} \quad (3.2)$$

O critério adotado consiste em eliminar os ciclos de carga que estiverem fora do intervalo, definido por :

$$\bar{C}_d - M\sigma_{Cd} \leq C_d \leq \bar{C}_d + M\sigma_{Cd} \quad (3.3)$$

A variável M diz respeito ao número de desvios padrões a serem utilizados [13], sendo que estes compreendem intervalos de confiança para cada valor de M, ou seja :

M = 3 - o intervalo compreende 99,73 % dos casos dentro do limite.

M = 2 - o intervalo compreende 95,45 % dos casos dentro do limite.

M = 1 - o intervalo compreende 68,27 % dos casos dentro do limite.

Tal critério pode ser aplicado, como já se disse, às energias ativas e reativas, independentemente.

3.3.2 Critério de Exclusão Pelo Fator de Carga

Da mesma forma que do critério anterior, o fator de carga verifica também se as curvas de carga são semelhantes. Repetindo-se o procedimento anterior, tem-se :

$$\overline{FC}_d = \frac{\sum_{i=1}^n FC_{di}}{n} \quad (3.4)$$

Onde " FC_d " é o fator de carga para um determinado dia, calculado por :

$$FC_{di} = \frac{C_{di}}{24 \cdot D_{mdi}} \quad (3.5)$$

Onde :

C_d - consumo diário [KWh]

D_{mdi} - demanda máxima diária [Kw]

Desta forma \overline{FC}_d representa o fator de carga diário médio do período sob análise, sendo FC_{di} os fatores de carga de cada dia. Então, o desvio padrão (σ_{FCd}) será :

$$\sigma_{FCd} = \sqrt{\frac{\sum_{i=1}^n (FC_{di} - \overline{FC}_d)^2}{n-1}} \quad (3.6)$$

Onde :

FC_{di} - fator de carga do i-ésimo mês

\overline{FC}_d - fator de carga médio do período considerado

Assim, os ciclos de carga diários restantes do primeiro critério se submetem ao segundo. Aqueles fator de carga fora da faixa dada a seguir, deverão ser excluídos.

$$\overline{FC}_d - M\sigma_{FCd} \leq FC_d \leq \overline{FC}_d + M\sigma_{FCd} \quad (3.7)$$

3.3.3 Critério de Seleção Pelos Desvios Médios Quadráticos

Os ciclos restantes, após aplicados os critério de exclusão pelas energias e pelo fator de carga, se prestam à escolha da curva de carga típica, seja para as potências ativa, reativa ou aparente. Assim, em um conjunto de **n** curvas seleciona-se dentre elas uma única, afim de que esta seja representativa do comportamento típico da instalação. Seja um conjunto de 3 curvas de carga, onde deseja-se selecionar a curva típica. O critério adotado neste trabalho é se buscar aquela que tem menor desvio médio quadrático em relação às demais, assumindo valores discretos das curvas referentes ao mesmo instante. Segue-se, então, o seguinte procedimento :

a) Fixa-se uma curva como referência (por exemplo, $C_{ref} \equiv C_1$) ;

b) Calcula-se os desvios médios quadráticos, em todos intervalos de tempo, entre as curvas C_1 e C_2 , assim como para C_1 e C_3 .

Demanda→	$C_1(t)$	$C_2(t)$	$C_3(t)$	$[C_1(t) - C_2(t)]^2$	$[C_1(t) - C_3(t)]^2$
Intervalo de tempo (t) ↓					
1	5	2	1	9	16
2	2	3	4	1	4
...
...
96	4	5	7	1	9
				$\Sigma 1 =$	$\Sigma 2 =$
				$\sigma_1 = \Sigma 1 + \Sigma 2$	

tabela 3.2 - tabela para seleção de curva de carga típica

- c) soma-se os desvios, calculando-se a média
- d) repetir todo procedimento, colocando C_2 como referência
- e) repetir todo procedimento, colocando C_3 como referência
- f) escolher entre $\sigma_1, \sigma_2, \sigma_3$, aquele que apresentar menor valor numérico. Definindo assim a curva típica.

Genericamente, tem-se :

$$\sigma_i^2 = \frac{1}{n(k-1)} \left\{ \sum_{l=1}^n \left[\sum_{\substack{j=1 \\ j \neq i}}^k (C_i(t) - C_j(t)) \right]^2 \right\} \quad (3.8)$$

onde :

- σ_i^2 - desvio médio quadrático, assumindo a curva i como referência;
- n - número de intervalos de tempo;
- k - número de curvas, após a seleção inicial;
- $c_i(t), c_j(t)$ - potência ativa, reativa ou aparente a cada instante, associados aos dias i e j .

Logo, tem-se como critério de escolha da curva típica, aquela que apresentar o menor valor de σ_i^2 [14]. Pode-se buscar um critério único, combinando potência ativa com reativa. Para tanto, o objetivo é encontrar o menor desvio quadrático composto ($\sigma_i^2 C$), dado por :

$$\sigma_i^2 C = \sigma_i^2 P + \sigma_i^2 Q \quad (3.9)$$



3.4 Aplicação da Metodologia

Quando o número de curvas é elevado, a metodologia apresentada não é adequada, pois demandaria um grande esforço computacional. Um processo alternativo é apresentado por Halila [11], onde é utilizado o programa “cluster analysis”.

No presente caso, o número de curvas é no máximo 100, sendo suficiente amostras com 30 registros diários.

O programa desenvolvido inicialmente lê a memória de massa, construindo sua base de dados. Então, efetua os cálculos descritos neste capítulo, aplicando na sequência as lógicas dos critérios de exclusão e de seleção, determinando assim uma única curva, que é a curva de carga típica. Para melhor exemplificar, foi considerada uma instalação industrial e aquisitadas 29 curvas de carga da memória de massa do controlador de demanda. Com estes dados, primeiramente, tem-se as curvas de carga coletadas da memória de massa sob a forma gráfica (figura 3.1), onde é aplicado o critério de exclusão pela energia ativa.

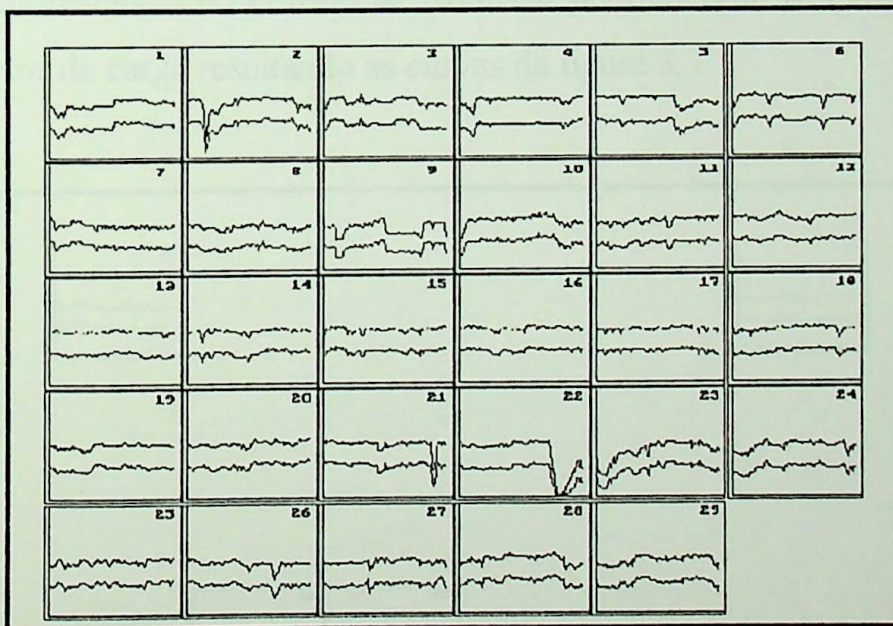


figura 3.1- conjunto de curvas de carga sob análise

Após a aplicação do critério de exclusão pela energia ativa, tem-se as curvas da figura 3.2.

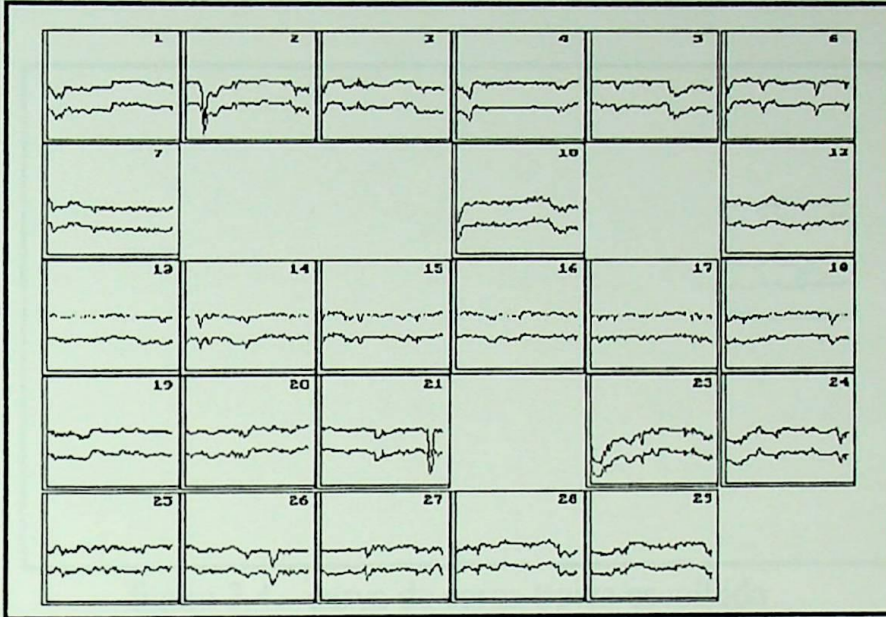


figura 3.2- situação das cargas restantes após o primeiro critério de exclusão

Das curvas restantes do critério de exclusão pela energia, é aplicado o critério de exclusão pelo fator de carga resultando as curvas da figura 3.3

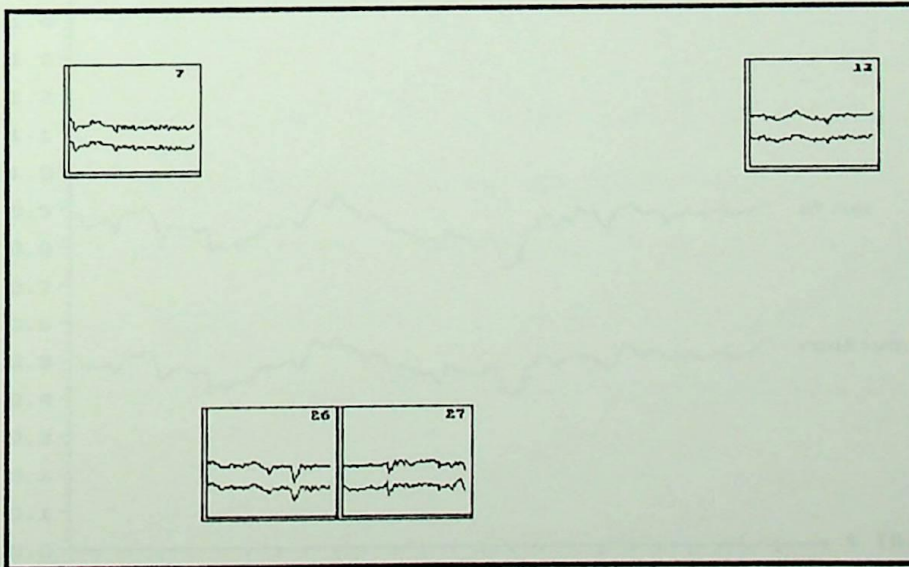


figura 3.3 - situação das curvas restantes após o segundo critério de exclusão

Finalmente, das curvas resultantes do critério de exclusão pelo fator de carga, aplica-se o critério de seleção pelos desvios médios quadráticos, obtendo-se a curva típica ilustrada na figura 3.4.

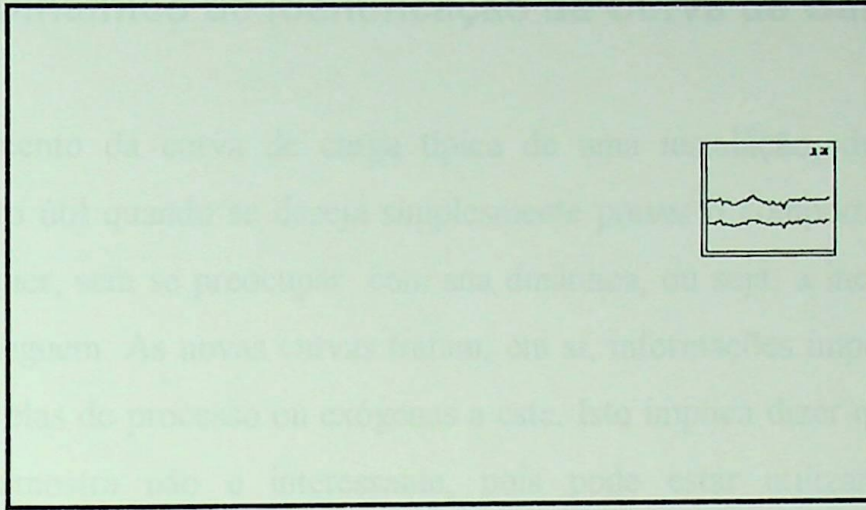


figura 3.4 - curva de carga típica escolhida

Esta mesma curva é apresentada de forma ampliada na figura 3.5.

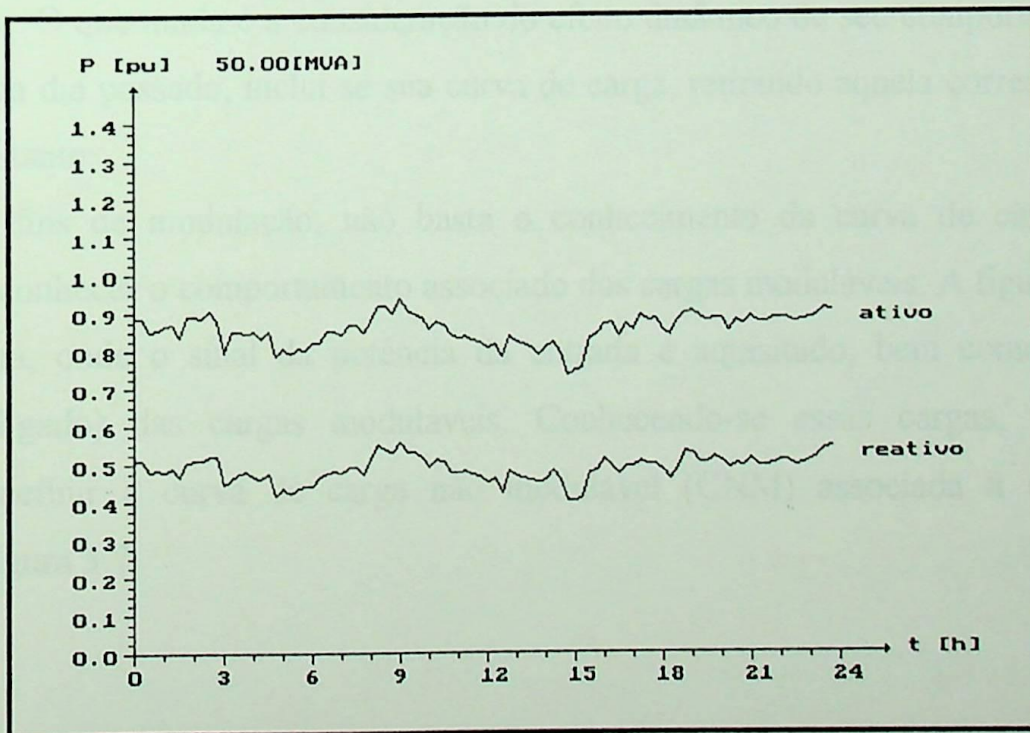


figura 3.5 - curva de carga típica resultante

Através da curva de carga típica, determinada pelo programa apresentado no anexo 1, o sistema sob estudo tem sua caracterização definida, possibilitando assim diversos tipos de estudos, tais como implantações de estratégias de GLD.

3.5 Sistema Dinâmico de Identificação da Curva de Carga Típica

O levantamento da curva de carga típica de uma instalação, descrito nos itens anteriores, é muito útil quando se deseja simplesmente prever o comportamento típico de um sistema qualquer, sem se preocupar com sua dinâmica, ou seja, a inclusão das curvas dos dias que se seguem. As novas curvas tratam, em si, informações importantes sobre as mudanças, sejam elas do processo ou exógenas a este. Isto implica dizer que simplesmente o aumento da amostra não é interessante, pois pode estar utilizando informações ultrapassadas quando se recorre as curvas associadas a dias longínquos. A princípio, as curvas não devem exceder um histórico de 30 a 60 dias.

Sendo assim, para a determinação da curva de carga típica dinâmica (CTD), necessita-se de uma nova lógica. Os passos para sua obtenção são os mesmos propostos inicialmente. O que muda é a consideração do efeito dinâmico de seu comportamento. Para tanto, a cada dia passado, inclui-se sua curva de carga, retirando aquela correspondente ao dia mais distante.

Para fins de modulação, não basta o conhecimento da curva de carga típica. É necessário conhecer o comportamento associado das cargas moduláveis. A figura 3.6 ilustra um esquema, onde o sinal da potência da entrada é aquisitado, bem como os estados (ligado/desligado) das cargas moduláveis. Conhecendo-se essas cargas, pode-se por subtração definir a curva de carga não modulável (CNM) associada à curva típica, conforme figura 3.7.

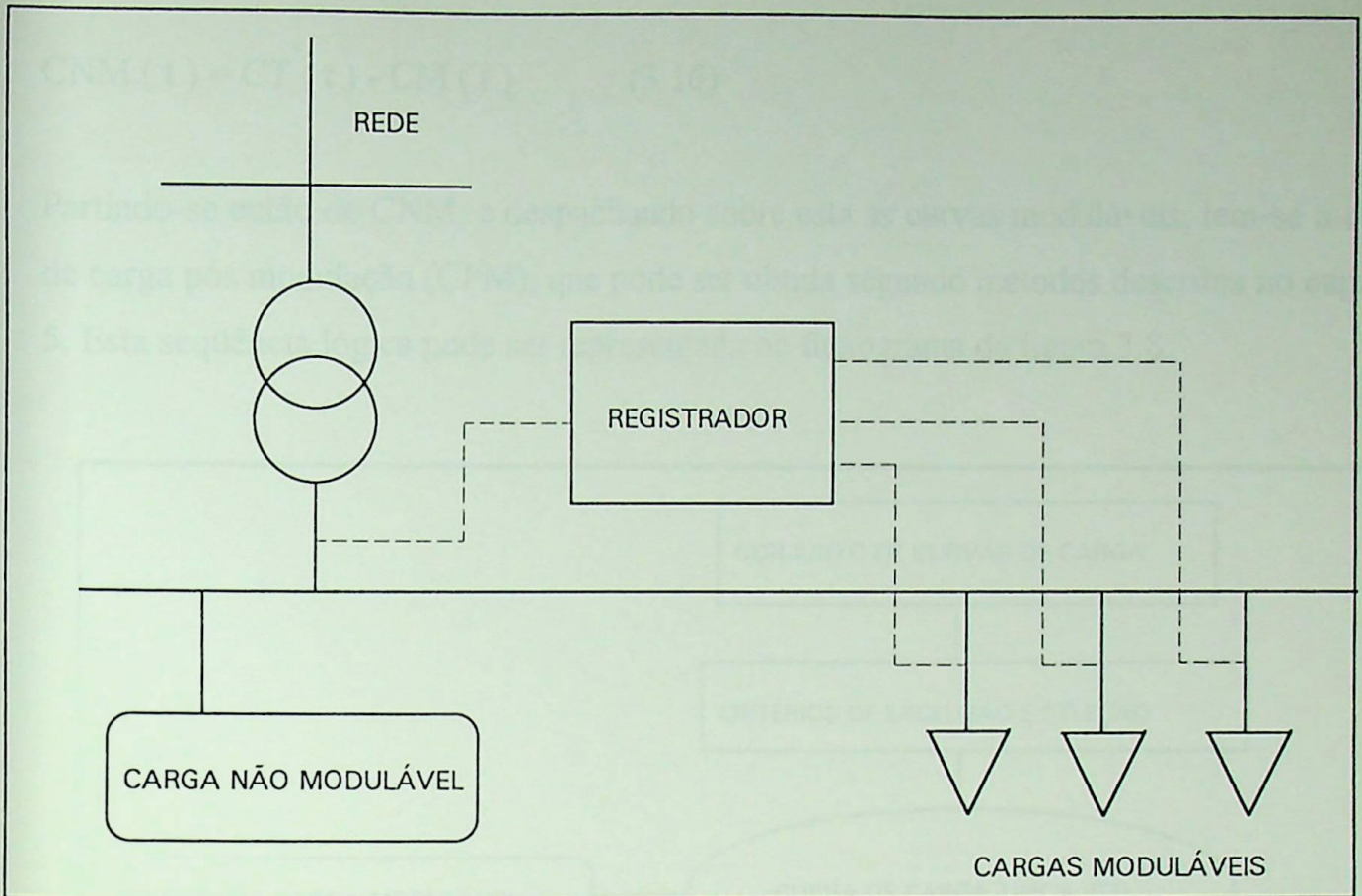


figura 3.6 - esquema do controle dinâmico

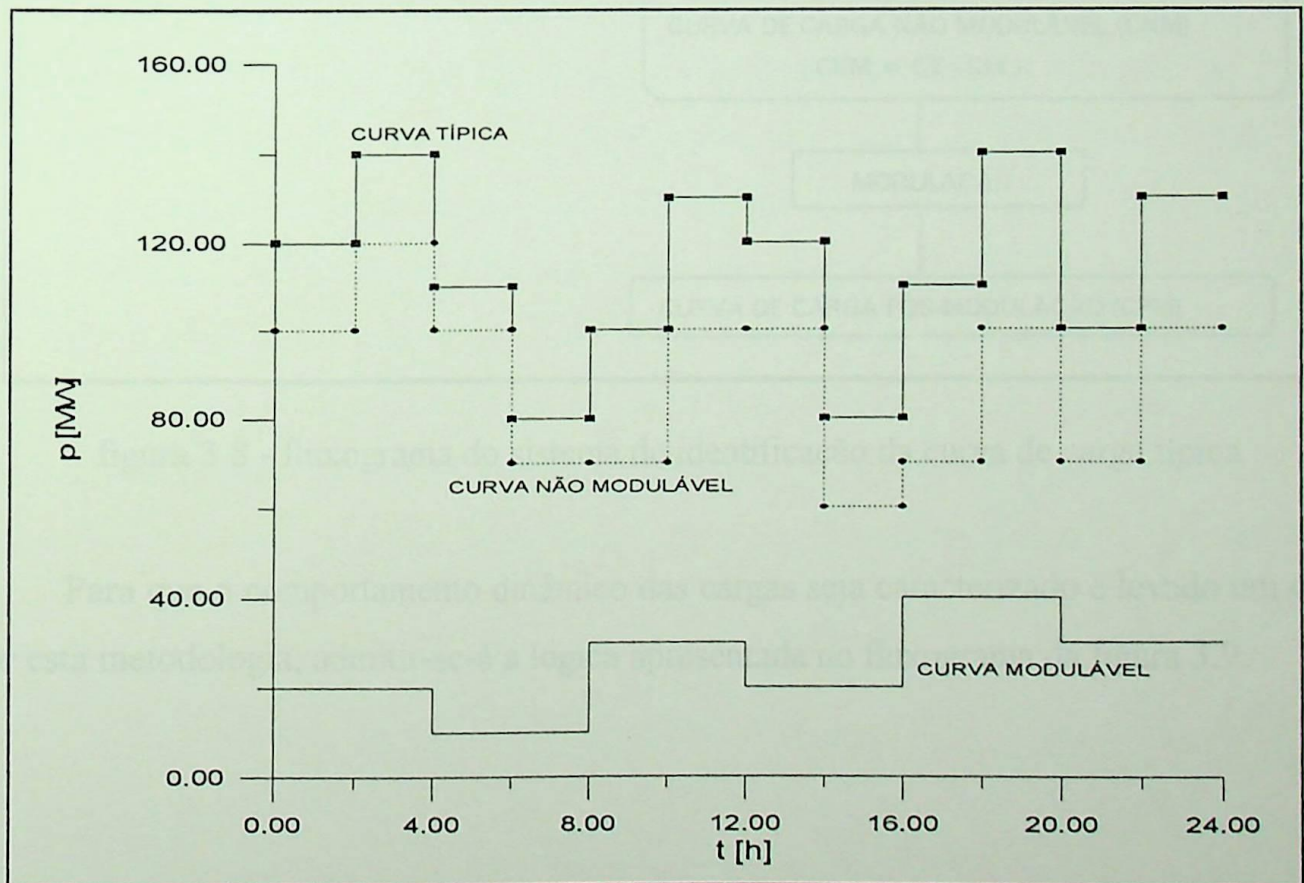


figura 3.7 - exemplo da curva de carga com modulação dinâmica

$$\text{CNM}(t) = \text{CT}(t) - \text{CM}(t) \quad (3.10)$$

Partindo-se então de CNM, e despachando sobre esta as curvas moduláveis, tem-se a curva de carga pós modulação (CPM), que pode ser obtida segundo métodos descritos no capítulo 5. Esta seqüência lógica pode ser representada no fluxograma da figura 3.8.

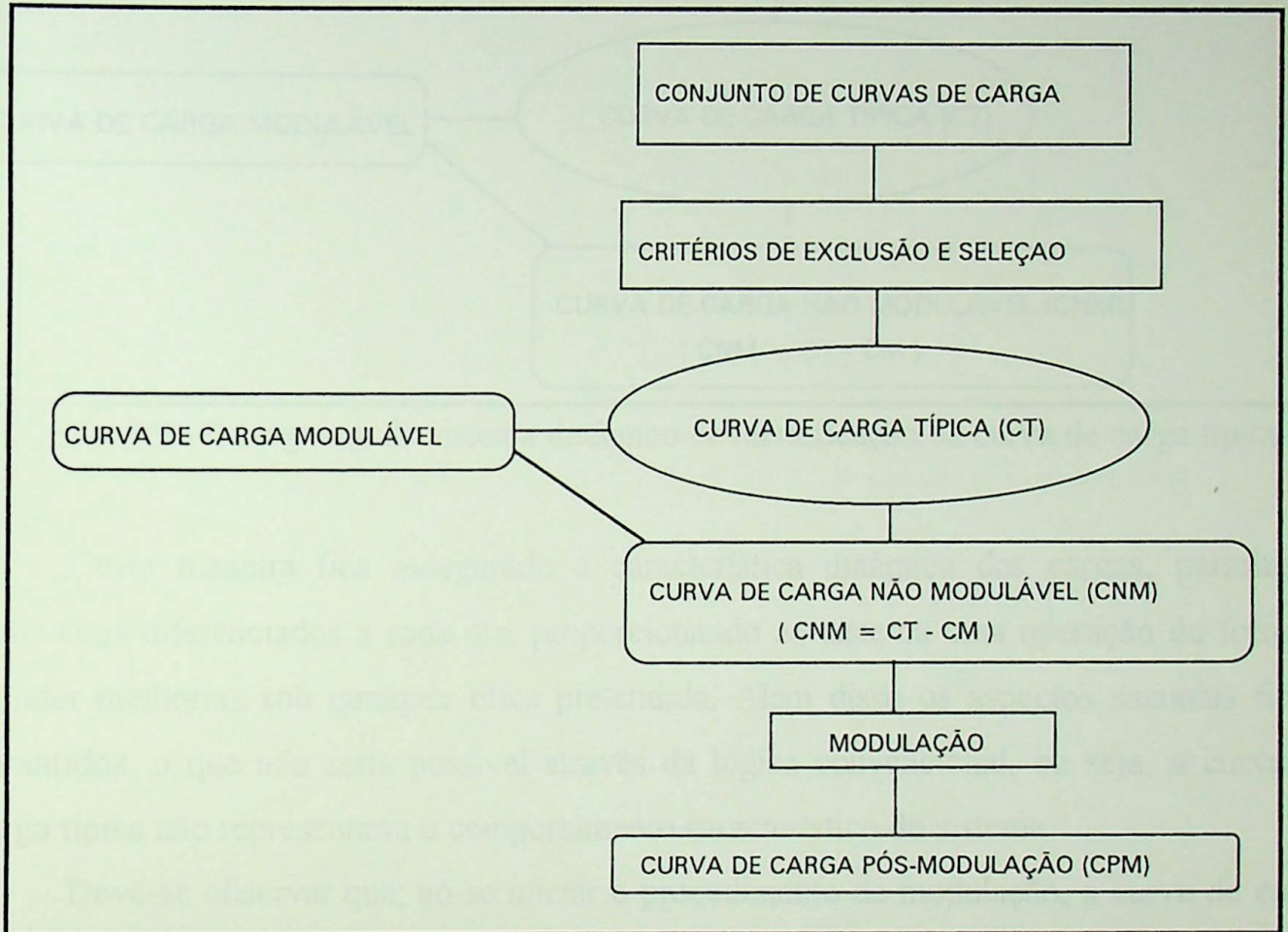


figura 3.8 - fluxograma do sistema de identificação da curva de carga típica

Para que o comportamento dinâmico das cargas seja caracterizado e levado em conta por esta metodologia, admitir-se-á a lógica apresentada no fluxograma da figura 3.9.

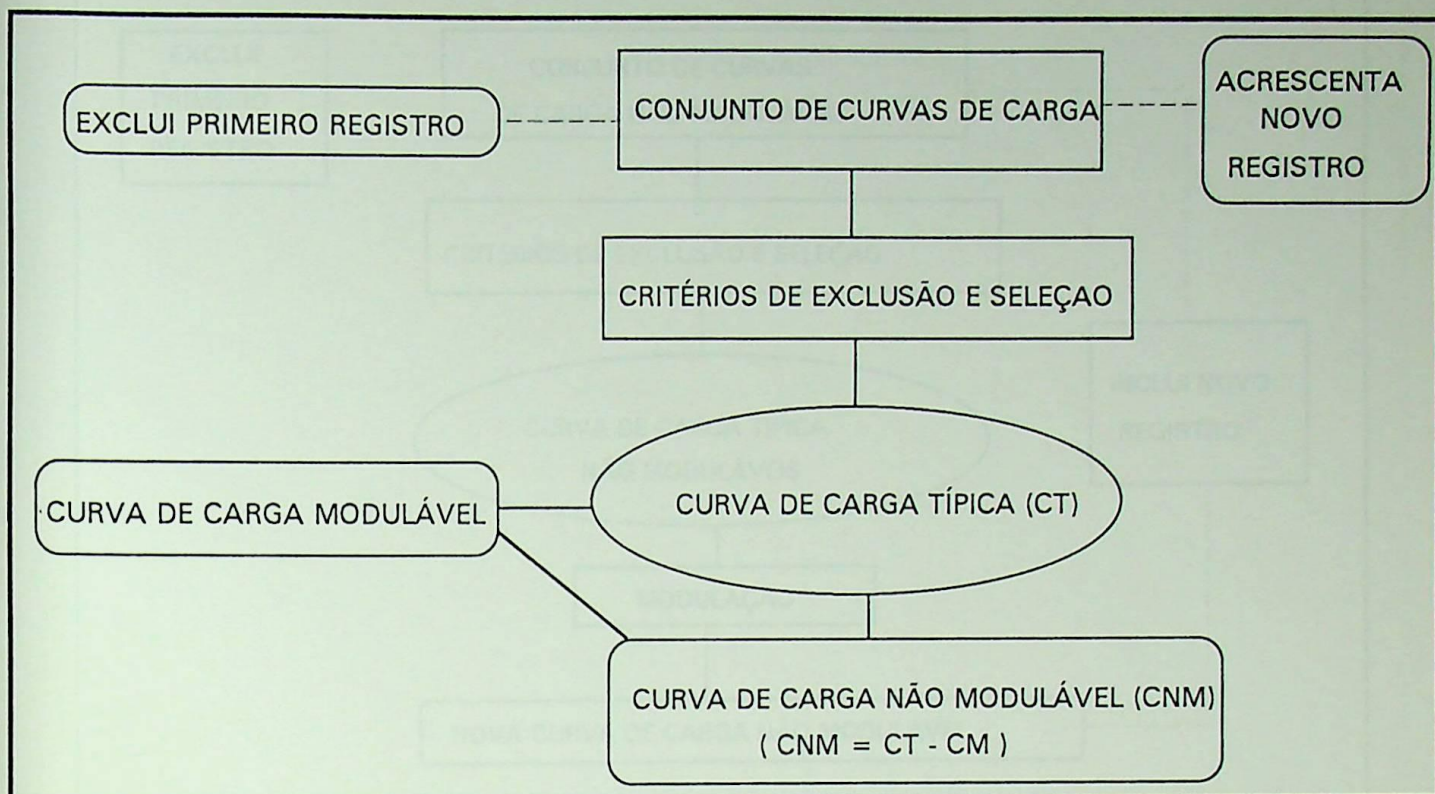


figura 3.9 - fluxograma do sistema dinâmico de identificação da curva de carga típica

Desta maneira fica assegurado a característica dinâmica das cargas, permitindo despachos diferenciados a cada dia, proporcionando ao sistema uma operação de forma a atender melhorias sob qualquer ótica pretendida. Além disso os aspectos sazonais ficam garantidos, o que não seria possível através da lógica convencional, ou seja, a curva de carga típica não representaria o comportamento característico do sistema.

Deve-se observar que, ao se iniciar o procedimento de modulação, a curva de carga não tem mais um comportamento natural, normalmente associado à produção. somente a carga não modulável permanece com tais características. Altera-se, então, o procedimento, buscando-se a curva de carga típica associada à parcela não modulável, conforme figura 3.10

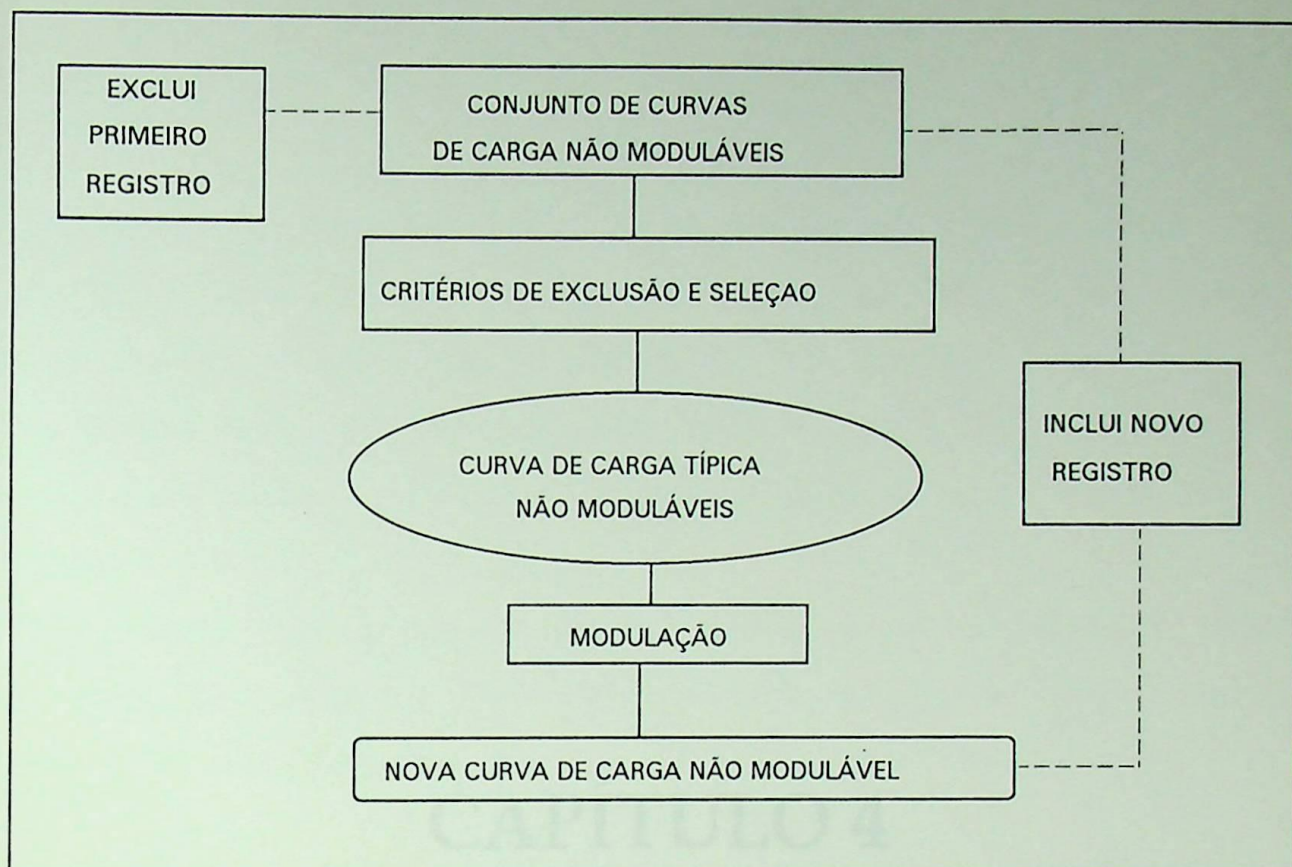
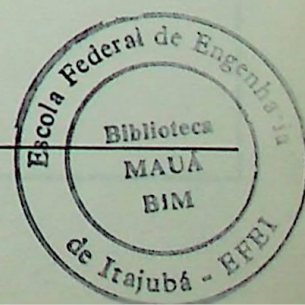


figura 3.10 - processo para determinação de curva típica não modulável



CAPÍTULO 4

CONJUNTOS DIFUSOS

4.1 Introdução

O conceito de conjuntos “Fuzzy” foi apresentado pela primeira vez por Lofti A. Zadeh [15]. A idéia clássica de conjunto obedece a uma lógica binária, ou seja, um elemento pertence ou não pertence a um conjunto, não existindo, desta maneira, outras opções intermediárias. A característica dos conjuntos “Fuzzy” é que a fronteira entre pertencer e não pertencer a ele não é bem definida, sugerindo assim, a idéia de graus de pertinência.

O presente capítulo procura conceituar estes conjuntos, caracterizando-os através das propriedades e operações, chegando-se a definição de relação “Fuzzy” e terminando com o conceito de variáveis lingüísticas.

4.2 Conceito intuitivo de conjuntos “Fuzzy”

Na chamada teoria clássica define-se conjunto, por exemplo, como uma coleção de “objetos” usualmente caracterizados pela enumeração de seus elementos ou por uma propriedade pertinente aos mesmos, assim:

$$C = \{ \text{azul, verde, amarelo} \} \quad (4.1)$$

pode ser definido como um conjunto de cores.

Com a finalidade de relacionar elemento com conjunto utiliza-se a relação de pertinência dizendo-se, por exemplo, que azul pertence ao conjunto C e a cor vermelha não pertence ao conjunto C, estabelecendo uma associação ou conjunto lógico de X em relação a C.

$$X \in C \quad (4.2)$$

$$X \notin C \quad (4.3)$$

ou em forma de uma função característica $\mu_C(x)$:

$$\mu_C(x) = 1 \text{ se } x \in C \quad (4.4)$$

$$\mu_C(x) = 0 \text{ se } x \notin C \quad (4.5)$$

Por outro lado existem várias tonalidades de verde, azul e amarelo que em primeira estância poderiam ser classificados em: fraca, média e forte. Percebe-se assim que pode existir ainda um elemento ponderador de caráter essencialmente subjetivo.

Estes conceitos estão presentes no chamado conjunto “Fuzzy”, ou seja, “são conjuntos sem limites precisos em que a transição entre pertencer e não pertencer ao conjunto é gradual e não abrupta” [16]; e qualquer conjunto em que a fonte de imprecisão seja a ausência de critérios bem definidos para avaliar se um elemento pertence ou não ao conjunto.

4.3 Definição de conjuntos “Fuzzy”

Um conjunto “Fuzzy” A de um universo de discurso U pode ser caracterizado por uma função de pertinência $\mu_A(x)$ também conhecida por função característica ou função de compatibilidade que atribui a cada $x \in U$ um grau de associação. Portanto um conjunto “Fuzzy” A é um conjunto de pares ordenados do tipo:

$$\left\{ \left(x, \mu_A(x) \right) \right\}, \forall x \in U \quad (4.6)$$

Como exemplo, considere o conjunto universo formado pelos números inteiros de 0 até 50:

$$U = \{ 0, 1, 2, \dots, 49, 50 \}$$

Seja agora o conjunto “Fuzzy” A constituído pelas pessoas jovens. Pode-se, então, tomando o conjunto M no intervalo $[0, 1]$ atribuir graus aos elementos de U como, por exemplo:

$$\mu_a (15) = 0,95$$

$$\mu_a (30) = 0,70$$

$$\mu_a (50) = 0,2$$

Neste exemplo, os graus de associação são subjetivos no sentido de que não derivam de resultados provenientes de fórmulas exatas. Para o mesmo universo de discurso U é possível associar mais de um conjunto “Fuzzy” e representá-los graficamente como, por exemplo:

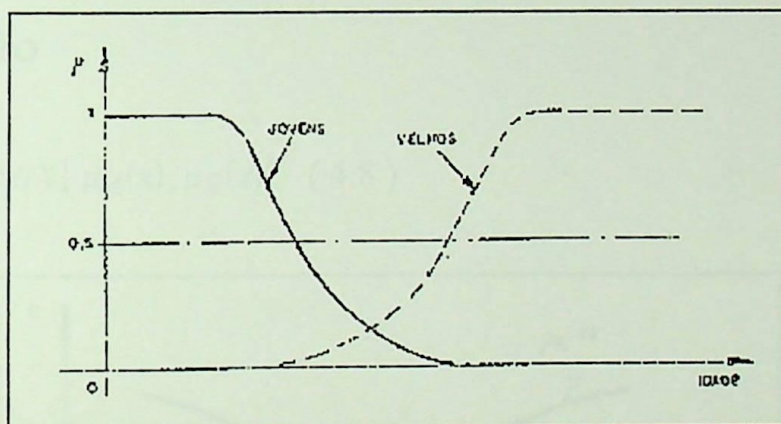


figura 4.1 - representação gráfica dos conjuntos “ Fuzzy “ jovem e velho

4.4 Operações básicas com conjuntos “ Fuzzy ”

Considere dois conjuntos “Fuzzy” A e B de um universo de discurso U com funções características $\mu_a (x)$ e $\mu_b (x)$, respectivamente. as operações básicas para os conjuntos “Fuzzy”, podem ser definidos através das seguintes funções características.

4.4.1 União

$$\forall x \in U: \mu_{A \cup B}(x) = \text{MAX}\{\mu_A(x), \mu_B(x)\} \quad (4.7)$$

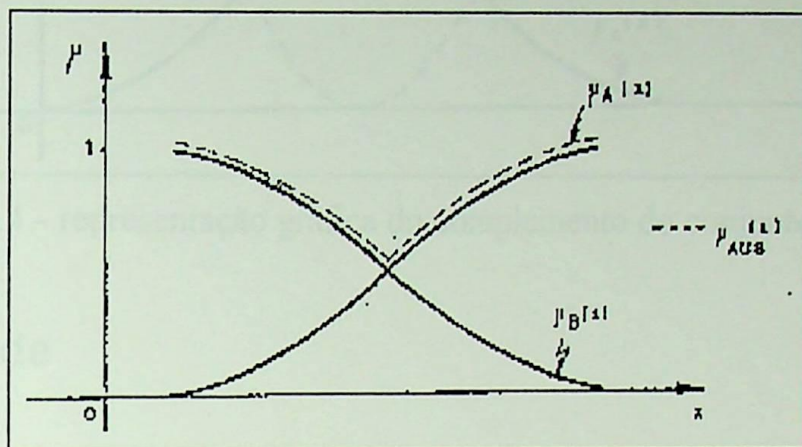


figura 4.2 - representação gráfica da união dos conjuntos "Fuzzy" A e B

4.4.2 Interseção

$$\forall x \in U: \mu_{A \cap B}(x) = \text{MIN}\{\mu_A(x), \mu_B(x)\} \quad (4.8)$$

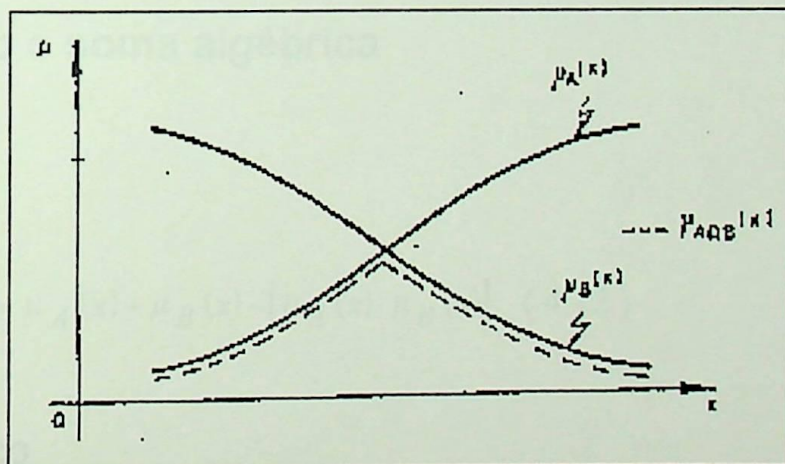


figura 4.3 - representação gráfica da interseção dos conjuntos "Fuzzy" A e B

4.4.3 Complementação

$$\forall x \in U: \mu_{\bar{A}}(x) = 1 - \mu_A(x) \quad (4.9)$$

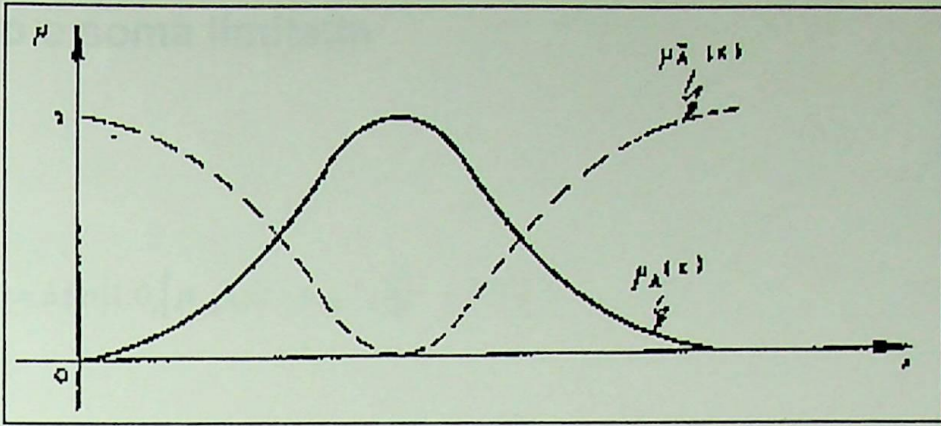


figura 4.4 - representação gráfica do complemento do conjunto "Fuzzy"

4.4.4 Igualdade

$$\forall x \in U: \mu_A(x) = \mu_B(x) \quad (4.10)$$

4.4.5 Soma disjuntiva

$$A \wedge B = [A \cap \bar{B}] \cup [\bar{A} \cap B] \quad (4.11)$$

4.4.6 Produto e soma algébrica

Soma algébrica

$$\forall x \in U: \mu_{A \hat{+} B}(x) = \mu_A(x) + \mu_B(x) - [\mu_A(x) \cdot \mu_B(x)] \quad (4.12)$$

Produto algébrico

$$\forall x \in U: \mu_{A \cdot B}(x) = \mu_A(x) \cdot \mu_B(x) \quad (4.13)$$

4.4.7 Produto e soma limitada

Soma limitada

$$\forall x \in U: \mu_{A \oplus B}(x) = \text{Min}\{1, 0, [\mu_A(x) + \mu_B(x)]\} \quad (4.14)$$

Produto limitado

$$\forall x \in U: \mu_{A \otimes B}(x) = \text{Max}\{0, [\mu_A(x) + \mu_B(x) - 1, 0]\} \quad (4.15)$$

4.4.8 Produto e soma drástico

Soma drástica

$$\forall x \in U: \mu_{A \dot{\cup} B}(x) = \begin{cases} \mu_A(x) & \text{se } \mu_B(x) = 0 \\ \mu_B(x) & \text{se } \mu_A(x) = 0 \\ 1 & \text{se } \mu_A(x), \mu_B(x) > 0 \end{cases} \quad (4.16)$$

Produto drástico

$$\forall x \in U: \mu_{A \dot{\cap} B}(x) = \begin{cases} \mu_A(x) & \text{se } \mu_B(x) = 1 \\ \mu_B(x) & \text{se } \mu_A(x) = 1 \\ 1 & \text{se } \mu_A(x), \mu_B(x) < 0 \end{cases} \quad (4.17)$$

4.4.9 Exponenciação

$$\forall x \in U: \mu_{A^\alpha}(x) = (\mu_A(x))^\alpha \quad \alpha \in \mathbb{R}^+ \quad (4.18)$$

4.4.10 Produto por escalar

$$\forall x \in U: \mu_{\beta A}(x) = \beta \cdot \mu_A(x) \quad \beta \in \mathbb{R}^{0,+} \quad (4.19)$$

4.4.11 Concentração

A ação desta operação sobre um conjunto “Fuzzy” A , em um universo de discurso U , com uma redução relativamente pequena nos graus de associação dos elementos de U que tem um alto grau de associação em A , e uma redução relativamente grande nos graus de associação dos elementos de U que têm um baixo grau de associação em A . Este resultado pode ser obtido elevando-se, por exemplo, a função de pertinência $\mu_A(x)$ ao quadrado. Assim, a operação de concentração sobre um conjunto “Fuzzy” A , denotada por $\text{CONC}(A)$, calcula-se, por exemplo, através da seguinte expressão:

$$\forall x \in U: \mu_{\text{CONC}(A)}(x) = (\mu_A(x))^2 \quad (4.20)$$

Utilizando-se a definição da operação de exponenciação, pode-se representar esta operação de concentração por: $\text{CONC}(A) = A^2$. A figura seguinte ilustra esta operação:

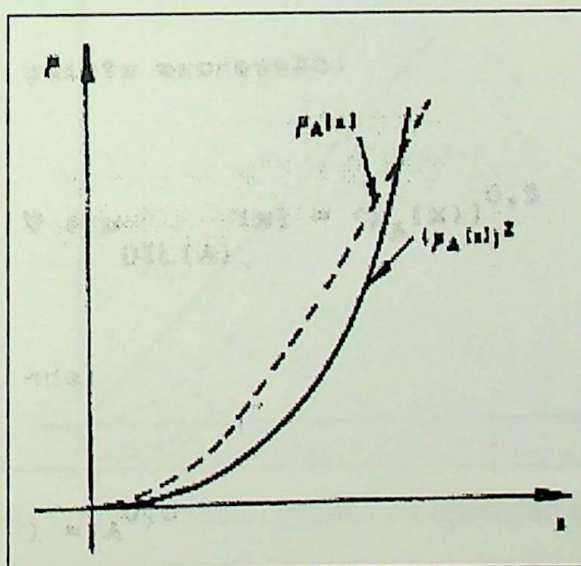


figura 4.5 - representação gráfica da operação de concentração

Esta operação apresenta as propriedades distributivas sobre a união, a interseção e o produto entre conjuntos “Fuzzy” e também, pode ser composta com ele próprio: $\text{CONC}(\text{CONC}(A)) = A^4$. Assim, genericamente para α inteiro maior ou igual a um tem-se:

$$\text{CONC}^\alpha(A) = A^{2^\alpha}$$

4.4.12 Dilatação

Esta operação apresenta o efeito contrário ao da concentração, ou seja, sua aplicação sobre um conjunto “Fuzzy” A , em um universo de discurso U , produz como resultado um subconjunto “Fuzzy”, com um aumento relativamente pequeno nos graus de associação dos elementos que possuem um alto grau de associação em A e um aumento relativamente grande nos graus de associação dos elementos que possuem um baixo grau de associação em A . Assim a operação de dilatação sobre um conjunto “Fuzzy” A , denotado por $DIL(A)$, calcula-se, por exemplo, através da seguinte expressão:

$$\forall x \in U: \mu_{DIL(A)}(x) = (\mu_A(x))^{0,5} \quad (4.21)$$

A figura seguinte ilustra esta operação:

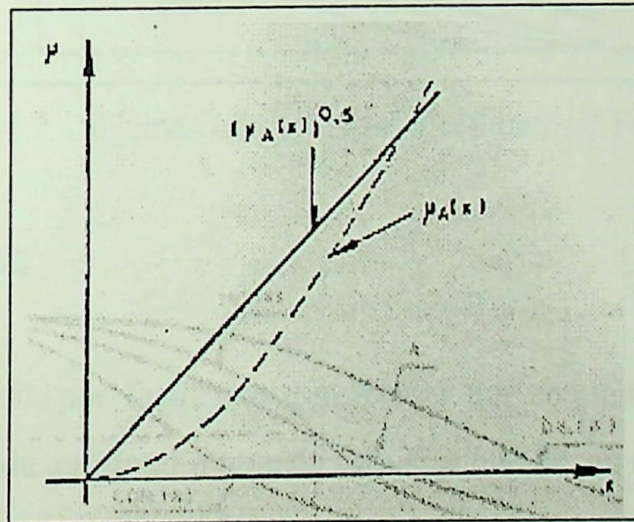


figura 4.6 - representação gráfica da operação de dilatação

4.4.13 Intensificação de contraste

Esta operação quando aplicada a um conjunto “Fuzzy” A , em um universo de discurso U , produz um aumento nos graus de associação $\mu_A(x)$ que estão acima de 0,5 e um decréscimo nos graus de associação $\mu_A(x)$ que estão abaixo de 0,5. Assim, a operação de

intensificação de contraste sobre um conjunto “Fuzzy” A, denotado por INT (A), calcula-se, através da seguinte expressão:

$$\mu_{INT(A)}(x) \begin{cases} = 2(\mu_A(x))^2 & \text{se } 0 < \mu_A(x) < 0,5 \\ = 1 - 2[1 - \mu_A(x)]^2 & \text{se } 0,5 < \mu_A(x) < 1,0 \end{cases} \quad (4.22)$$

Nota-se, portanto, que esta operação tem o efeito de reduzir a nebulosidade do conjunto “Fuzzy”. A figura seguinte mostra as operações de concentração, dilatação e intensificação de contraste para um conjunto “Fuzzy” A.

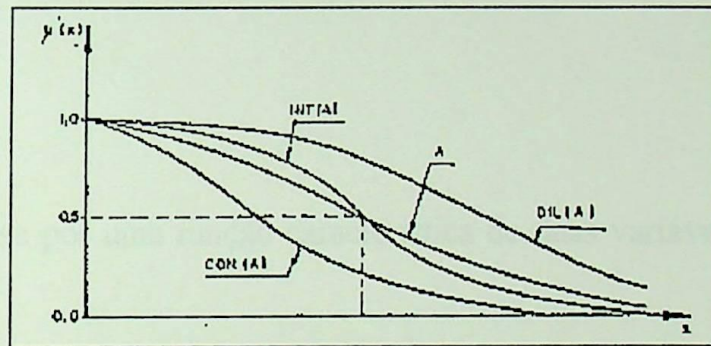


figura 4.7 - algumas operações com conjuntos “Fuzzy”

4.4.14 Fuzzificação

Esta operação tem por finalidade transformar um conjunto não “Fuzzy” em um conjunto “Fuzzy” ou ainda aumentar o grau de nebulosidade de um conjunto “Fuzzy” [17].

Pode-se representar um conjunto A “Fuzzificando” por $F(A)$ ou \underline{A} , onde o símbolo sob o conjunto representa o “Fuzzificador”. Um “Fuzzificador” F é caracterizado pelo seu Kernel $K(y)$ que é o conjunto “Fuzzy” resultante da aplicação de F a um Singleton $1/y$ em U, ou seja, $k(y) \overset{\Delta}{=} \underline{1/y}$, o que mostra a dependência de \underline{y} sobre y. Geralmente $K(y)$ é um intervalo “Fuzzy” cuja função associação $\mu_{K(y)}(Z)$ é uma função não crescente à distância entre Z e Y sendo $Y \in U$ e também $Z \in U$. A operação de Fuzzificação é

importante para a definição dos chamados delimitadores lingüísticos como, por exemplo, “mais” ou “menos”, “ligeiramente”, “muito”.

4.5 Relação “ Fuzzy “

4.5.1 Definição

Uma relação “ Fuzzy “ R de dois conjuntos A e B defini-se como um subconjunto “ Fuzzy “ do produto cartesiano $A \times B$. $A \times B$ é a coleção de pares ordenados (x, y) com as seguintes relações de pertinência :

$$x \in A \text{ e } y \in B$$

R caracteriza-se por uma função característica de duas variáveis $\mu_R(x, y)$ expressa por :

$$R = \int_{A \times B} \frac{\mu_R(x, y)}{(x, y)} \quad (4.23)$$

Uma relação “ Fuzzy “ em $A \times B$ também pode ser representado por :

$$x \in A, y \in B : x R y \quad (4.24)$$

Que significa “ x se relaciona com y através da relação R “. Como ilustração, considere dois conjuntos A e B :

$$A = \{ x_1, x_2, x_3 \} \text{ e } B = \{ y_1, y_2 \}$$

Uma relação binária “ Fuzzy “ entre membros de A e B pode ser expressa, por exemplo, por :

$$R = \{ (0,8 / (x_1, y_1)); (0,2 / (x_1, y_2)); (0,3 / (x_2, y_2)); \\ (0,1 / (x_2, y_2)); (0,7 / (x_3, y_1)); (0,3 / (x_3, y_2)) \}$$

Genericamente, para uma relação “Fuzzy” n-ária R, ou seja, um subconjunto “Fuzzy” de $A_1 \times A_2 \times \dots \times A_n$ tem-se por definição :

$$R = \int_{A_1 \times \dots \times A_n} \mu_R \left(\frac{x_1, x_2, \dots, x_n}{x_1, x_2, \dots, x_n} \right) \quad (4.25)$$

4.5.2 Operações utilizando relações “Fuzzy”

Considere dois conjuntos “Fuzzy” A e B e as relações R1 e R2 que se caracterizam pelas expressões :

$$x \in A, y \in B : x R_1 y$$

$$x \in A, y \in B : x R_2 y$$

e pelas funções características $\mu_{R_1}(x, y)$ e $\mu_{R_2}(x, y)$

4.5.2.1 União

$$\mu_{R_1 \cup R_2}(x, y) = \text{Max} [\mu_{R_1}(x, y), \mu_{R_2}(x, y)] \quad (4.26) \\ = \mu_{R_1}(x, y) \vee \mu_{R_2}(x, y)$$

Para n relações, tem-se :

$$\mu_{R_1 \cup R_2 \cup \dots \cup R_n}(x, y) = \bigvee_{R_i} \mu_{R_i}(x, y) \quad (4.27)$$

4.5.3 Composição de duas relações "Fuzzy"

4.5.2.2 Interseção

4.5.2.2.1 Definição

$$\begin{aligned} \mu_{R_1 \cap R_2}(x, y) &= \text{Min} [\mu_{R_1}(x, y), \mu_{R_2}(x, y)] \\ &= \mu_{R_1}(x, y) \wedge \mu_{R_2}(x, y) \end{aligned} \quad (4.28)$$

Para n relações, tem-se :

$$\mu_{R_1 \cup R_2 \cup \dots \cup R_n}(x, y) = \bigwedge_{R_i} \mu_{R_i}(x, y) \quad (4.29)$$

4.5.2.3 Produto e soma algébrica

Produto algébrico

$$\mu_{R_1 \cdot R_2}(x, y) = \mu_{R_1}(x, y) \cdot \mu_{R_2}(x, y) \quad (4.30)$$

Soma algébrica

$$\mu_{R_1 + R_2}(x, y) = \mu_{R_1}(x, y) + \mu_{R_2}(x, y) - [\mu_{R_1}(x, y) \cdot \mu_{R_2}(x, y)] \quad (4.31)$$

4.5.2.4 Complementação

$$\forall (x, y) \in A \times B : \mu_{\bar{R}}(x, y) = 1 - \mu_R(x, y) \quad (4.32)$$

4.5.2.4 Soma disjuntiva

$$R_1 + R_2 = (R_1 \cap R_2) \cup (R_1 \cap R_2) \quad (4.33)$$

4.5.3 Composição de duas relações “ Fuzzy “

4.5.3.1 Definição

Dadas as relações “ Fuzzy “ R_1 e R_2 em $A \times B$ e $B \times C$ respectivamente, a composição das mesmas é uma relação “ Fuzzy “ denotada por $R_1 \cdot R_2$ formada por pares ordenadas (x, z) onde $x \in A$ e $z \in C$ e definida por uma expressão que emprega um ou mais operadores.

4.5.3.2 Composição MAX-MIN

$$\begin{aligned} \mu_{R_1 \cdot R_2}(x, z) &= \underset{y}{\text{Max}} \left[\underset{y}{\text{Min}} \left(\mu_{R_1}(x, y) ; \mu_{R_2}(y, z) \right) \right] \\ &= \underset{y}{\vee} \left[\mu_{R_1}(x, y) \wedge \mu_{R_2}(y, z) \right] \end{aligned} \quad (4.34)$$

onde : $x \in A$, $y \in B$ e $z \in C$

$$R_1 \subset A \times B \quad ; \quad R_2 \subset B \times C$$

4.5.3.3 Composição MAX-Produto

$$\mu_{R_1 \cdot R_2}(x, z) = \underset{y}{\vee} \left[\mu_{R_1}(x, y) \cdot \mu_{R_2}(y, z) \right] \quad (4.35)$$

4.5.3.4 Conjuntos nível - α

O conjunto ordinário de nível- α de um dado conjunto “ Fuzzy “ A defini-se como sendo aquele onde o grau da pertinência ultrapassa o valor α [17]. Representando-se este conjunto por S_α , tem-se :

$$S_\alpha = \{x \mid \mu_A(x) \geq \alpha\} \quad (4.36)$$

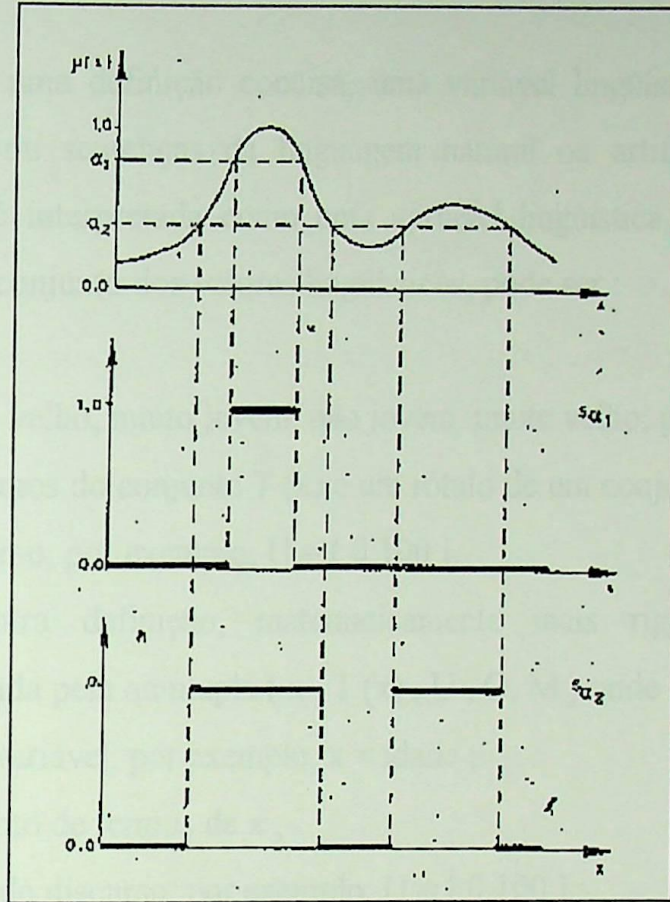


figura 4.8 - exemplos de conjuntos de nível- α

Para estes conjuntos de nível pode-se escrever a seguinte propriedade :

$$se \alpha_1 \geq \alpha_2 \rightarrow S_{\alpha_1} \subset S_{\alpha_2} \quad (4.37)$$

Qualquer conjunto " Fuzzy " A pode ser decomposto em um produto de conjuntos ordinários por escalares α da seguinte forma :

$$A = \bigcup_{\alpha \in [0,1]} \alpha S_{\alpha} \quad (4.38)$$

4.6 Variáveis lingüísticas

Empregando-se uma definição concisa, uma variável lingüística x , é aquela cujos valores são palavras ou sentenças da linguagem natural ou artificial [17]. Assim por exemplo, se "idade" é interpretada como uma variável lingüística, então o conjunto de termos $T(x)$, que é, o conjunto dos valores lingüísticos, pode ser :

$$T(x) = \{ \text{jovem, velho, muito jovem, não jovem, muito velho, pouco jovem, ...} \}$$

Cada um dos termos do conjunto $T(x)$ é um rótulo de um conjunto "Fuzzy" relativo a um universo de discurso, por exemplo, $U = [0,100]$.

Empregando outra definição, matematicamente mais rigorosa, uma variável lingüística é caracterizada pela quintupla $(x, T(x), U, G, M)$ onde :

- x é o nome da variável, por exemplo, $x = \text{idade}$;
- $T(x)$ é o conjunto de termos de x ;
- U é o universo de discurso, por exemplo, $U = [0,100]$;
- G é uma regra sintática ("Grammar") que define a formação das sentenças em $T(x)$
- M é uma regra semântica ("Meaning") que associa a cada valor lingüístico v o seu significado $M(v)$, onde $M(v)$ é um conjunto "Fuzzy" de U .

Um "termo primário" (ou "termo atômico") em $T(x)$ é um termo cujo significado pode ser definido *a priori* e serve como uma base para o cálculo do significado dos termos não primários (ou termos compostos) em $T(x)$. Os termos atômicos dividem-se em quatro categorias:

1. Termos primários que servem de rótulos para um conjunto "Fuzzy" específico de um dado universo de discurso ;
2. Negação ("não") e conectivos ("e", "ou") ;
3. Delimitadores, como por exemplo, muito, mais ou menos, e ligeiramente ;
4. Marcadores, como por exemplo o "()" (parênteses) .

Por exemplo, os termos primários são jovem e velho, cujos significados podem ser definidos pelas suas respectivas "funções de compatibilidade" μ_{jovem} e μ_{velho} , que

associam a cada valor u de $U = [0,100]$ suas compatibilidades com os termos jovem e velho. Diante do exposto, o significado ou as funções de compatibilidade dos termos não primários podem ser calculadas pela aplicação de uma regra semântica como por exemplo:

$$\mu_{\text{muito jovem}} = (\mu_{\text{jovem}})^2$$

$$\mu_{\text{não muito jovem}} = 1 - (\mu_{\text{jovem}})^2$$

Com a composição dos termos atômicos, surge o problema de se encontrar o significado do termo composto. De uma forma geral, pode-se escrever $x = hv$, onde h é um delimitador e v é um termo com significado específico, como por exemplo, $x = \text{homem muito alto}$ onde $h = \text{muito}$ e $v = \text{homem alto}$.

Este delimitador lingüístico "h" pode ser definido como um operador que transforma o conjunto "Fuzzy" $M(v)$ (significado de v) em um conjunto "Fuzzy" $M(hv)$. Logo, tem a função de gerar um conjunto de valores para as variáveis lingüísticas, a partir de uma coleção de termos primários. Por exemplo, o termo primário "alto", as conjunções "e" e "não" e o delimitador "muito", podem gerar por exemplo, muito alto, muito muito alto, não muito alto, alto e não muito alto. O operando sobre o qual um delimitador lingüístico age deve possuir um significado "Fuzzy". para definir um delimitador como um operador é conveniente empregar algumas das operações básicas apresentadas anteriormente, como por exemplo, concentração, dilatação e fuzziificação. São exemplos deste tipo de delimitador; muito, mais ou menos e ligeiramente.

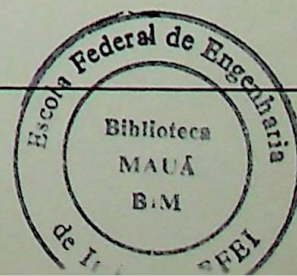
Assim, o delimitador "muito" age como um intensificador do significado do seu operando. A operação que tem esta propriedade é a concentração. Considere um conjunto "Fuzzy" A em um universo de discurso U representando o significado de um termo y e A^* o conjunto "Fuzzy" resultante da aplicação da operação de concentração sobre A , com significado y^* :

$$A^* = \text{CONC}(A)$$

$$A^* = A^2$$

e substituindo-se o termo pelo seu significado, tem-se:

$$y^* = \text{muito } y = y^2$$



Se $y = \mu_1(x_1) / x_1 + \mu_2(x_2) / x_2 + \dots + \mu_n(x_n) / x_n$

$\forall x \in U$, então:

muito $y = \mu_1^2(x_1) / x_1 + \mu_2^2(x_2) / x_2 + \dots + \mu_n^2(x_n) / x_n$

O delimitador "muito" pode ser composto com ele próprio :

$$\text{muito muito} = x^4$$

Não aconselha-se escrever, por exemplo, $x = \text{muito não exato}$, pois a frase pode se tornar incompreensível. Substituí-se então "não exato" por "inexato", resultando:

$$x = \text{muito inexato} = \text{inexato}^2 = (\text{não exato})^2$$

Já os delimitadores "mais" e "menos" podem ser chamados de acentuadores e desacentuadores, respectivamente, e fornecem graus mais suaves de concentração e dilatação que aqueles associados às operações CONC e DIL. Eles podem ser definidos como :

$$\text{mais } x = x^{1,25}$$

$$\text{menos } x = x^{0,75}$$

Os expoentes foram escolhidos com o propósito de fazer valer a identidade aproximada :

$$\text{mais mais } x \sim \text{menos muito } x$$

outra aplicação importante destes foram delimitadores é na definição dos delimitadores naturais, cujo significado difere ligeiramente de outros, como por exemplo :

$$\text{altamente} = \text{mais muito ou, altamente} = \text{menos muito muito}$$

Existem outros delimitadores que necessitam do emprego de Algoritmos "Fuzzy" para a caracterização das suas ações. Estes Algoritmos "Fuzzy" podem ser representados por uma seqüência de instruções ordenadas que fornecem uma representação aproximada de uma grande variedade de fenômenos complexos. Nestes Algoritmos encontram-se instruções dos seguintes tipos :

1. declarações de atribuição :

- x é aproximadamente igual a 5 ;

- x é grande ;

- x é positivo ;
- x não é grande e x não é pequeno ;
- $x = 3$.

2. Declarações condicionais “ Fuzzy “ :

- se x é pequeno, então y é grande, senão y não é grande ;
- se x é positivo, então decresça y ligeiramente ;
- se x é muito maior que 5, então pare ;
- se x é muito pequeno, então vá para 7 .

3. Declarações incondicionais ;

- multiplique x por x ;
- decresça ligeiramente x ;
- ‘delete’ primeiro algumas ocorrências de 2 ;
- vá para 5 ;
- pare ;
- escreva x .

Estes Algoritmos podem ser classificados de acordo com o tipo de aplicação, podendo ser utilizados para a definição e geração de conjuntos “ Fuzzy “, para a descrição do comportamento de sistemas e de estratégias ou regras de decisão.

A execução de uma instrução-” Fuzzy “, apresenta como resultado um conjunto “ Fuzzy “, mas se for necessário a escolha de um único elemento deste conjunto como resposta, pode-se adotar várias estratégias para obtê-la. A estratégia mais comum é escolher o elemento de maior grau de associação no conjunto e no caso de ser mais de um, pode-se escolher um elemento aleatoriamente, ou ainda utilizar um outro critério qualquer. Esta alternativa recebe o nome de Regra da Alternativa Preponderante.

4.6 Tomada de decisão utilizando conjuntos “ Fuzzy “

4.6.1 Introdução

Os problemas de tomada de decisão multicritério ou multiobjetivo apresentam geralmente metas conflitantes com características de incerteza e imprecisão na sua quantificação. Para resolver tais problemas uma variedade de métodos matemáticos foram desenvolvidos e têm sido aplicados em áreas tão diversificadas como engenharia, políticas governamentais e econômicas, negócios e ciência sociais entre outras. Assim, pode-se, encontrar questões envolvendo, por exemplo, planejamento de transportes e gerenciamento de tráfegos, opções de investimento de capital, gerenciamento de recursos hidráulicos, fixação de tarefas e administração pública.

Como elementos que incorporam incertezas, pode-se mencionar, as previsões econômicas e sociais de longo prazo, a natureza abstrata de algumas informações subjetivas e a heterogeneidade de dados utilizados em alguns modelos de análise. Considerando estas circunstâncias, a teoria dos conjuntos “ Fuzzy “ torna-se uma ferramenta metodologicamente capaz de tratar sistematicamente a incerteza presente em tais situações.

4.6.2 Tomada de decisão utilizando técnicas “ Fuzzy “

Em vários métodos de tomada de decisão multiobjetivo os resultados são influenciados pela natureza das informações fornecidas pelo decisor. Assim, em algumas situações os objetivos, as restrições e conseqüentemente as possíveis decisões não são precisas, podendo ainda serem inconsistentes e sujeitas a erros. Para quantificar esta imprecisão pode-se utilizar os conceitos e técnicas da teoria da probabilidade, entendendo-se que desta forma que a imprecisão pode ser relacionada com a aleatoriedade. Outros [18] julgam, ser melhor associar esta imprecisão aos conjuntos “ Fuzzy “ pois enquanto aleatoriedade trata com as incertezas da pertinência ou não de um elemento em relação a

um conjunto ordinário, o grau de subjetividade ou “nebulosidade” reflete os valores intermediários entre a pertinência total e a não pertinência. Assim, como ilustração a frase “A empresa X tem uma moderna perspectiva” é imprecisa devido a característica “Fuzzy” do termo “moderna perspectiva”.

Já a frase “A probabilidade da empresa X estar operando com prejuízo é 0,7” é uma medida da incerteza proveniente da mesma se encontrar na classe de conjuntos não “Fuzzy” das empresas que estão operando com prejuízo. Como consequência, as técnicas matemáticas para tratar com esta nebulosidade são diferentes daquelas da teoria da probabilidade. De outra forma pode-se também dizer que, enquanto a teoria da probabilidade lida com a aleatoriedade referente a incerteza da ocorrência ou não de algum evento, a imprecisão dos conjuntos “Fuzzy” trata com a pertinência ou não pertinência de um objeto em um conjunto com limites não bem delimitados ou imprecisos.

Os processos de decisão envolvendo conjuntos “Fuzzy” podem ser classificados sob diversos aspectos, como por exemplo, envolvendo um ou vários decisores ou através de um ou mais estágios do processo de decisão. As ferramentas metodológicas abrangem, entre outras, programação linear, programação dinâmica, programação inteira, podendo ainda ser ou não interativa.

Entre os vários trabalhos publicados sobre o assunto, geralmente, cita-se o de Zimmermann [19] publicado como sendo o pioneiro em programação multicritério a tratar da teoria dos conjuntos “Fuzzy”. O mesmo envolvendo apenas problemas lineares e o processo não-interativo, mostra, entre outros aspectos que as soluções obtidas são diferenciadas pelos vários graus de sofisticação obtidos.

4.6.3 Conceituação do problema

Assim como na teoria convencional sobre tomada de decisão pode-se analisar o problema de tomada de decisão “Fuzzy” através dos seguintes pontos: as variáveis de decisão, as restrições e os objetivos. A noção de nebulosidade pode então ser introduzida em todos estes elementos básicos. Geralmente as variáveis de decisão são consideradas

como permanecendo determinísticos (ou, no máximo, probabilísticos) e apenas os objetivos e restrições constituem classes de alternativas cujos limites não estão muito bem definidos.

Segundo Bellman [18] a função objetivo e restrição " Fuzzy " podem ser caracterizados por sua função de pertinência. A finalidade é satisfazer ambos os objetivos e as restrições; portanto, uma decisão " Fuzzy " é considerada como a interseção das restrições e os objetivos " Fuzzy ". Pode-se definir a estrutura geral deste problema como se segue.

Seja A um conjunto de possíveis ações de decisão alternativa. Um objetivo " Fuzzy " G é um subconjunto " Fuzzy " em A caracterizado por sua função de pertinência dado por $\mu_g : A \rightarrow [0,1]$. Uma restrição " Fuzzy " C é um subconjunto " Fuzzy " em A, caracterizado por sua função de pertinência $\mu_c : A \rightarrow [0,1]$.

A decisão " Fuzzy " D resultante do objetivo G e a restrição C é a interseção de ambos, logo:

$$D = G \cap C$$

e utilizando-se a expressão 13 tem-se para a função de pertinência, com $X \in A$:

$$\mu_d(x) = \text{Min} \{ \mu_g(x), \mu_c(x) \}$$

A decisão D resultante de n objetivos " Fuzzy " $G_1 \dots G_n$ restrições " Fuzzy " $C_1 \dots C_m$ pode ser calculada através da expressão:

$$D = G_1 \cap G_2 \cap \dots \cap G_n \cap C_1 \cap C_2 \cap \dots \cap C_m$$

Por exemplo, considere um objetivo G e uma restrição C expressos como:

G: x deve ser muito maior que 5.

C: deve estar perto de 10.

Que estão ilustrados na figura 09 e possuindo as seguintes funções de pertinência:

$$\mu_G(x) = \begin{cases} 1 - \left(1 + (0.2(x-5))^2\right)^{-1} & \text{para } x > 5 \\ 0 & \text{para } x \leq 5 \end{cases} \quad (4.39)$$

$$\mu_C(x) = \left(1 + (x-10)^2\right)^{-1}$$

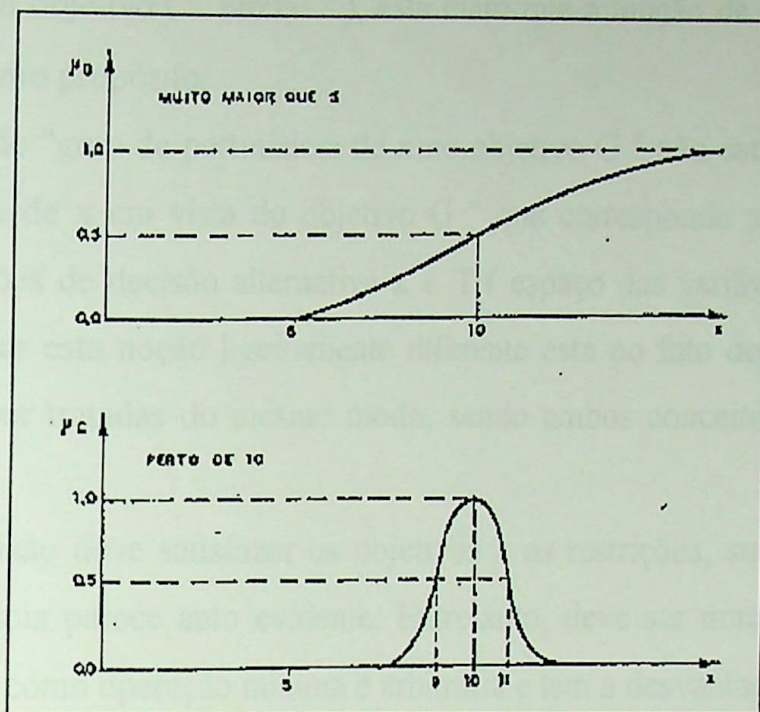


figura 4.9 - exemplo de objetivo e restrição "Fuzzy".

A decisão "Fuzzy" D é a interseção de ambos os conjuntos "Fuzzy" ($D = G \cap C$) e está mostrada na figura seguinte 4.10.

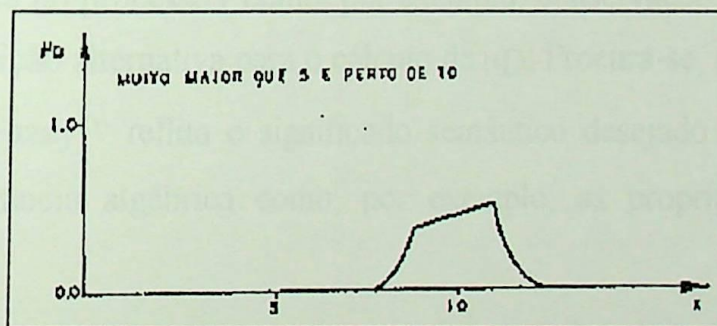


figura 4.10 - exemplo de decisão "Fuzzy".

4.6.4 Objetivos e restrições "Fuzzy"

Nos processos de tomada de decisão convencional a direção na qual o processo deve se desenvolver pode ser induzida por uma função objetivo ou de desempenho, como por exemplo, a função de utilidade. Esta função ordena o conjunto de alternativas de acordo com a preferência do tomador de decisão e, embora não haja menção de uma função

objetivo mas de um objetivo (" Fuzzy "), está claro que a função de pertinência, $\mu_g(x)$, pode servir ao mesmo propósito.

O conceito de "grau de pertinência de x no objetivo G " não está longe do conceito de "nível de aceite de x em vista do objetivo G " que corresponde a uma ordenação de preferência das ações de decisão alternativa $x \in T$ (espaço das variáveis de decisão). A razão para se adotar esta noção ligeiramente diferente está no fato de que os objetivos e restrições podem ser tratadas do mesmo modo, sendo ambos conceitos matematicamente idênticos.

Como a decisão deve satisfazer os objetivos e as restrições, sua definição como a interseção destes dois parece auto evidente. Entretanto, deve ser notado que a definição usual de interseção como operação mínima é arbitrária e tem a desvantagem de uma falta de interdependência. Em outras palavras, enquanto $\mu_A \geq \mu_B$, a interseção $\text{Min}(\mu_A, \mu_B)$ é absolutamente independente do valor de μ_A . Se for desejável que a

decisão " Fuzzy " reflita sempre os valores dos objetivos e restrições envolvidos e alguma condição de interesse no processo (como, por exemplo, conservação de energia), pode-se introduzir uma definição alternativa para o cálculo de μ_D . Procura-se, ainda, que a definição de uma decisão " Fuzzy " reflita o significado semântico desejado mas, possuindo pelo menos alguma elegância algébrica como, por exemplo, as propriedades associativa e distributiva.

4.6.5 Análise da decisão " Fuzzy "

Embora possa haver situações onde o decisor esteja satisfeito com soluções na forma de conjuntos " Fuzzy ", como por exemplo, " x é muito maior que 7" e " x aproximadamente entre 10 e 12", é aceitável que ocorram muitas situações onde a decisão final a ser tomada tenha que ser exata, bem definida, portanto, não " Fuzzy ". A questão que se coloca é como deve se avaliar a decisão " Fuzzy " resultante para se obter uma decisão não " Fuzzy " representativa. Um modo de se avaliar uma decisão " Fuzzy " D é pela divisão do conjunto

“ Fuzzy “ em seus conjuntos de nível- α . Um conjunto de nível- α (ou corte α) S_α de um conjunto “ Fuzzy “ D em T pode ser definido por:

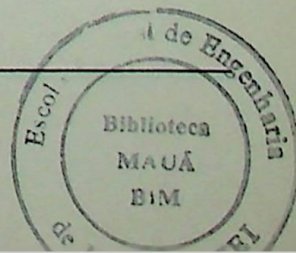
$$S_\alpha = \{x \in T \mid \mu_D(x) \geq \alpha\} \quad (4.40)$$

Este conjunto de nível- α é obviamente um conjunto não “ Fuzzy “. Por meio deste conceito pode-se construir uma série de conjuntos de acordo com seus níveis mas ainda não leva a uma decisão simples em particular. Um modo muito fácil é procurar aquela resposta onde a decisão “ Fuzzy “ consegue sua função pertinência máxima. Assim pode-se definir o conjunto decisão M ótimo (ou máximo) do objetivo “ Fuzzy “ G por :

$$M = \left\{ X_0 \in T \mid \mu_G(X_0) = \max_{x \in T} \mu_G(x) \right\} \quad (4.41)$$

Note que restrições adicionais devem ser impostas sobre G para assegurar que haja apenas $X_0 \in M$. Na maior parte, M consiste de mais de um elemento. Uma avaliação final bastante simples poderia consistir em se tomar a média X_M de todos $X_0 \in M$. e X_M representar o conjunto “ Fuzzy “ G . Este procedimento de avaliação é arbitrário e tem a desvantagem de que depende apenas do máximo do conjunto “ Fuzzy “. O restante deste conjunto - sua forma ou contorno - não é considerado.

Uma maneira diferente pode ser que, ao invés de procurar um valor simples e exato, designe-se um valor lingüístico aquela decisão “ Fuzzy “. Assim, dados um conjunto de valores lingüísticos básicos como "grande", "pequeno", etc; um conjunto de limites lingüísticos como "muito", "bastante", "mais ou menos", e o conjunto de conceitos usuais "e", "ou", "não", pode-se construir um procedimento de avaliação que designe a resultante decisão “ Fuzzy “ um termo lingüístico tal como "nem pequeno e nem muito grande". Iterativamente, o termo gerado é comparado com a decisão “ Fuzzy “ real, e de acordo com alguns critérios, o termo o é designado à decisão ou não [20]. Kaufmann [21] apresenta outros critérios para designação dos termos lingüísticos às decisões “ Fuzzy “, conforme figura 4.11.



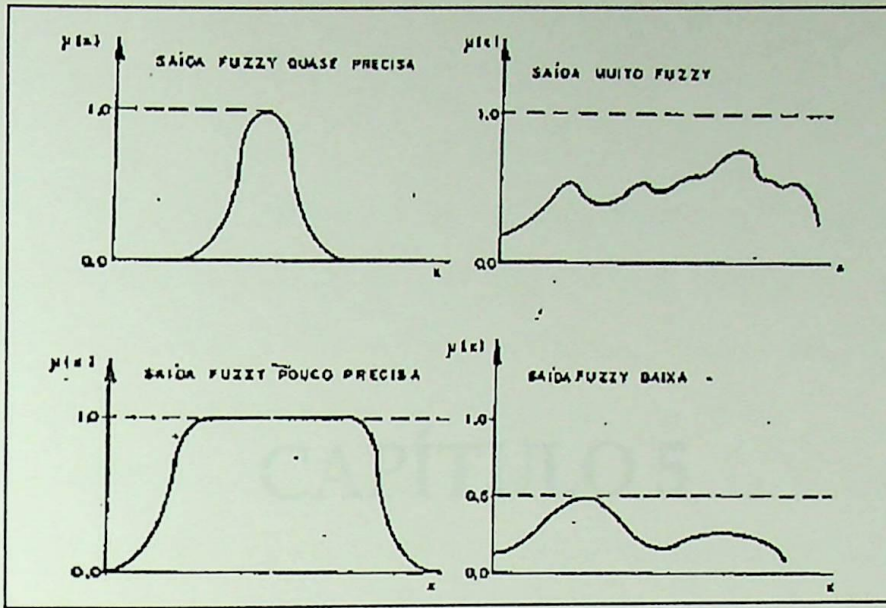


figura 4.11 - designação de termos lingüísticos à decisão "Fuzzy"

APLICAÇÃO DA PROGRAMAÇÃO
 DINÂMICA E DOS CONJUNTOS
 DIFUSOS À MODULAÇÃO DE
 CICLOS DE CARGA

CAPÍTULO 5

APLICAÇÃO DA PROGRAMAÇÃO DINÂMICA E DOS CONJUNTOS DIFUSOS À MODULAÇÃO DE CICLOS DE CARGA

5.1 Introdução

Desde os primórdios até hoje em dia a humanidade vem experimentando evoluções em diversos segmentos da ciência e, por conseqüência, vem evoluindo também a complexidade dos problemas. Na luz destas incertezas, os procedimentos de tomada de decisão devem ser flexíveis, adaptativos e até interativos.

No caso da engenharia elétrica, especificamente tratando-se de sistemas elétricos em geral, a avalanche tecnológica associada ao crescimento da concorrência e exigências da sociedade, fazem com que estes sistemas tenham que oferecer cada vez mais qualidade, rapidez e eficácia dos serviços energéticos em geral. Neste contexto pode-se destacar a participação de empresas de energia junto a seus consumidores. O mercado consumidor de energia elétrica da atualidade está em constante expansão e, logo exigindo maior demanda. Como as concessionárias de energia têm carência de recursos financeiros para incrementar a oferta de energia elétrica, é possível solucionar este problema através de alternativas que possam suprir o mercado, no curto prazo a baixo custo. Alternativas tais como a cogeração, entrada de capital estrangeiro no setor elétrico, gerações termelétrica, solar, nuclear ou, ainda, atitudes complementares como a racionalização e conservação estratégica de energia. São alguns dos pensamentos atuais para lidar com estes problemas. O mesmo ocorre com sistemas industriais de transformação, os quais devem otimizar seu consumo energético, de forma a prestar melhor atendimento e qualidade a baixo custo aos consumidores finais.

Contudo a humanidade, principalmente grande parte dos cientistas, está com os olhos voltados para a tríade: fontes energéticas alternativas(renováveis e "limpas"), racionalização de energia e meio-ambiente. A técnica desenvolvida neste capítulo busca contribuir na solução desses problemas. Ela se presta à conservação de energia, mas, ainda mais que isto ao uso racional da energia, entendendo-se, aí, a utilização nos momentos corretos. Este tipo de tarefa é árdua pois leva em conta diversos fatores desde perdas elétricas até hábitos de consumo. A figura 5.1 ilustra este cenário.

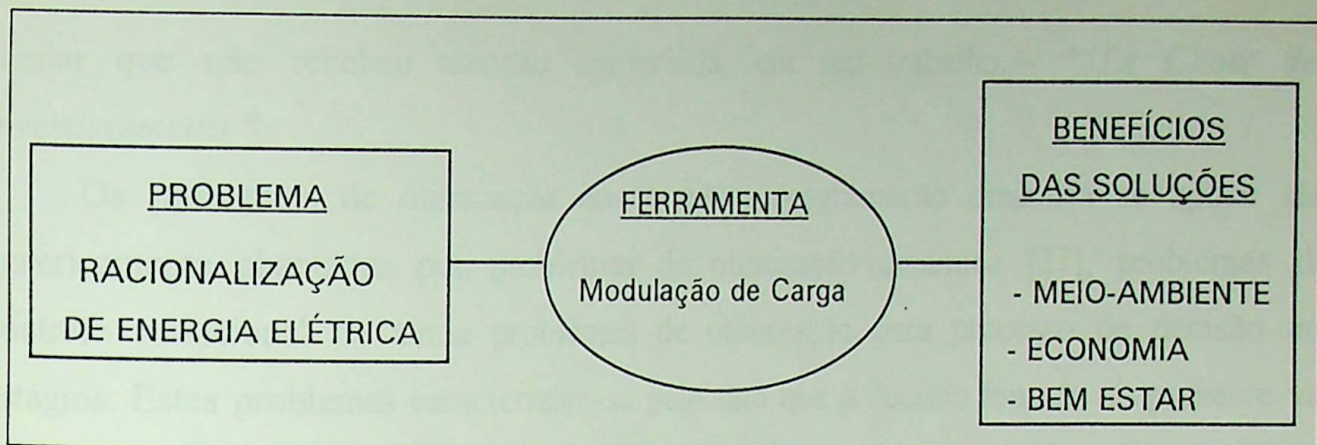


figura 5.1 - representação de estratégia para a solução do problema

Para alcançar os benefícios oferecidos pela metodologia da modulação da curva de carga, é preciso lançar mão de ferramentas matemáticas dedicadas à otimização de sistemas. Diversas são estas ferramentas, desde o método da tentativa e erro até os inovadores algoritmos genéticos(GA) incluindo a combinação de várias destas. Para este caso escolheu-se a técnica de programação dinâmica [22], bem como a associação desta aos conjuntos difusos [23].

Este capítulo abordará os princípios da programação dinâmica e uma variação desta, chamada aqui de programação dinâmica por intervalos, bem como a associação destas aos conjuntos difusos. Finalmente exemplos de aplicações serão mostrados afim de ilustrar a eficácia destes métodos.

5.2 A técnica de programação dinâmica

A programação dinâmica, daqui por diante designada por PD, é uma técnica matemática geralmente útil para a realização de decisão de processos interrelacionados. Esta permite o desenvolvimento de procedimentos sistemáticos para a determinação de decisões combinadas as quais fornecem um mapa ótimo de ações a serem tomadas, segundo políticas previamente estabelecidas. Esta técnica, foi desenvolvida originalmente por Richard Bellman, na década de 50, esta técnica ganhou reconhecimento como uma ferramenta poderosa na solução e otimização de uma ampla variedade de situações problemáticas [24 e 25]. Vale ressaltar que em 1946, P. Massé [26] utilizou um método

similar que não recebeu atenção apropriada, em seu trabalho - "*Le Choix des Investissements*".

Os problemas de otimização aos quais a programação dinâmica se aplica são genericamente chamados por problemas de otimização dinâmica [27], problemas de controle variacional ou ainda problemas de otimização para processo de decisão em estágios. Estes problemas caracterizam-se pelo fato que a decisão tomada no presente vai afetar o comportamento do sistema no futuro, de tal forma que a solução ideal é uma seqüência de decisões, durante um período de controle, e não apenas uma solução para um momento determinado.

Esta classe de problemas, aos quais a programação dinâmica conduz a uma solução ideal, é muito grande. As equações do sistema dinâmico podem ser não-lineares e variáveis no tempo. O critério de performance, no qual a otimização é baseada, pode ser bastante arbitrário. As restrições mais diversas são permitidas, e a solução será sempre um ótimo absoluto, e nunca um máximo ou mínimo local. Além disso, a programação dinâmica pode considerar efeitos estocásticos ou randômicos, fornecendo um mapa de decisões ótimas, a serem adotadas conforme o comportamento do sistema. Diversas são as suas aplicações, desde economia até diversas aplicações de engenharia.

5.2.1 Princípio da otimalidade de Bellman

O coração da programação dinâmica é o princípio da otimalidade de Bellman. É este princípio que torna possível elaborar um procedimento computacional capaz de resolver complexos problemas de otimização [27], sem perder as vantagens de uma solução que analisa todas as possibilidades factíveis.

Para introduzir o teorema da otimalidade de Bellman, pode-se fazer uso de um simples problema de multiestágios. A figura 5.2 representa o processo de decisão.

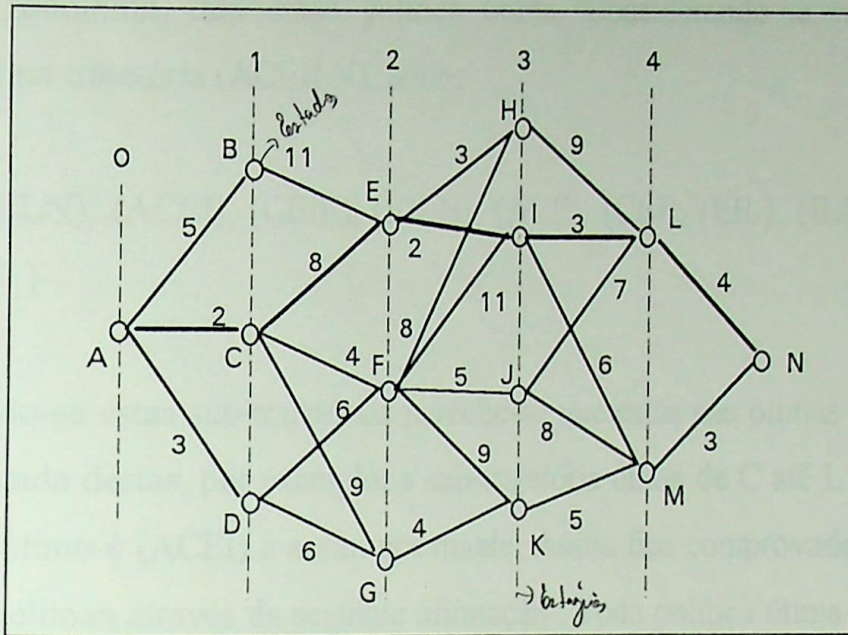


figura 5.2 - possíveis caminhos a serem percorridos

Na figura acima, as linhas verticais tracejadas da figura representam os **estágios**, bem como os círculos vazios representam os **estados** de cada estágio e as linhas cheias representam os possíveis caminhos a serem percorridos com exceção das linhas mais escuras que representam a solução ótima.

Iniciando-se no estágio zero, partindo-se do ponto **A**, pode-se ir até os pontos(estados) **B**, **C** e **D** os quais estão situados no estágio 1, da mesma maneira, no estágio 1 pode-se ir aos estados **E**, **F** e **G** do estágio 2, provendo assim trajetórias entre estes pontos, e da mesma maneira pode-se efetuar o mesmo procedimento através dos estágios 3, 4 e 5 de maneira a alcançar o ponto **N**(destino).

Para cada trajetória entre estados de diferente estágios pode-se associar um **custo**, por exemplo, para ir de **C** até **F** o custo vale 4, de **J** até **M** o custo vale 8, e assim por diante.

Qualquer trajetória entre os pontos **A** e **N** é chamada de uma política. Assim (**ADFILN**) é uma política. Os trechos desta trajetória são chamado de subpolíticas, assim, (**DFK**), (**CEIM**), (**HL**), e (**CGKMN**) são subpolíticas.

Enumerando-se todas as possíveis políticas, percebe-se que a política (**ACEILN**) é a que apresenta menor custo total, este vale 19. Se a pretensão é minimizar os custos, esta

será a política escolhida, dita então política ótima. Considerando-se todas trajetórias parciais contidas na trajetória (ACEILN), temos :

{ (ACEIL), (CEILN), (ACEI), (CEIL), (EILN), (ACE), (CEI), (EIL), (ILN), (AC), (CE), (EI), (IL), e (LN) }

Examinando-se estas sub-trajetórias percebe-se que estas são ótimas entre os pontos de partida e chegada destas, por exemplo, a sub-trajetória ótima de C até L é (CEIL), de A até I o caminho ótimo é (ACEI) e assim por diante. Assim fica comprovado o princípio da otimalidade de Bellman através da seguinte afirmação : Toda política ótima contém única e exclusivamente subpolíticas ótimas.

5.3 Formulação do problema

Através da curva de carga típica de um sistema elétrico genérico, é possível efetuar a sua modulação. Modular a curva de carga significa rearranjar as suas cargas em outros segmentos horários de forma a alterar seu aspecto inicial para se obter melhorias, por exemplo, como o seu fator de carga. Tal tarefa demanda o conhecimento profundo do comportamento das cargas, e assim "movê-las" pela curva até encontrar-se o melhor *locus* para sua operação. Esta técnica de tentativa e erro leva a um procedimento demorado e impreciso. Sendo assim, propõe-se o uso da técnica de programação dinâmica em um algoritmo desenvolvido através da linguagem de programação Turbo Pascal 7.0, conforme anexo 2, o qual fará a modulação otimizada do ciclo de carga típico, trazendo assim precisão e velocidade a este processo de decisão.

5.4 Aplicação da PD à modulação de um ciclo de carga típico

Através de uma simples analogia com o princípio da otimalidade de Bellman, pode-se comprovar a aplicabilidade da PD na modulação de um ciclo de carga típico. Inicialmente é

necessário definir as variáveis da programação dinâmica quando aplicada ao caso da modulação do ciclo de carga típico.

1. Estágios (i) - são os intervalos de tempo envolvidos na análise para o caso sob estudo. Estes intervalos compreendem 24 horas, que podem ser divididos em quantos intervalos se desejar, por exemplo, intervalos de 5 minutos, 15 minutos, 1 hora, 2 horas, 6 horas ou outro valor qualquer;

2. Estados ($X_{i,j}$) - cada estágio tem seus respectivos estados, que representam a energia despachada do início do ciclo até aquele momento. Pode-se trabalhar com energia ou potência média;

3. Controle (u) - estes são as potências despachadas em cada estágio e dizem respeito as cargas que podem ser alocadas no ciclo de carga em questão;

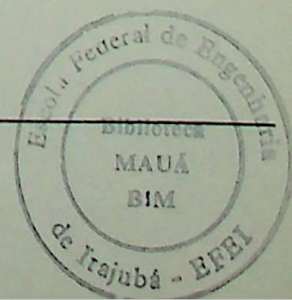
4. Equação recursiva - esta equação tem por finalidade descrever o comportamento do sistema como um todo. Ela faz a ligação dos estados de um estágio com aqueles do estágio anterior. No presente caso, tem-se :

$$X_{i,j} = X_{i-1,k} + u_N \quad (5.1)$$

$\begin{matrix} \nearrow \text{estado} \\ \downarrow \text{est\u00e1gio} \end{matrix}$

onde :

- X - estado (energia despachada acumulada)
- j,k - índice dos estados
- i - estágio sob estudo
- $i - 1$ - estágio anterior
- u - controle (carga despachada no estágio)



5. Custo associado ao controle ($C_{i,j}$) - Adotou-se, neste exemplo, como critério de custo a potência perdida naquele estágio, associada à carga despachada. Assim,

$$C_{i,j} = k X_{i,j}^2 \quad (5.2)$$

onde k é uma constante que pode ser adotada como 1, pois não interfere no resultado.

5. Custo total ($CT_{i,j}$) - O custo total de um estado é o custo associado ao controle mais o custo total do estado anterior.

$$CT_{i,j} = CT_{i-1,k} + C_{i,j} \quad (5.3)$$

6. restrições - as restrições, como o próprio nome diz, servem para delinear as fronteiras de atuação da programação dinâmica. Estas no caso do exemplo sob estudo são os valores mínimos e máximos de carga a cada estágio. Os valores mínimos de carga dizem respeito ao carregamento mínimo que o sistema possui, ou seja são cargas que não podem ser realocadas, no processo. Os valores máximos de carga são aqueles que representam as cargas que podem ser realocadas para aquele estágio.

O controle ótimo para cada estado de cada estágio é aquele o apresentar menor custo total. Obviamente o menor custo total, ao final de toda análise, implica na solução ótima, pois representa a menor perda.

A figura 5.3 ilustra a busca do controle ótimo do estado D.

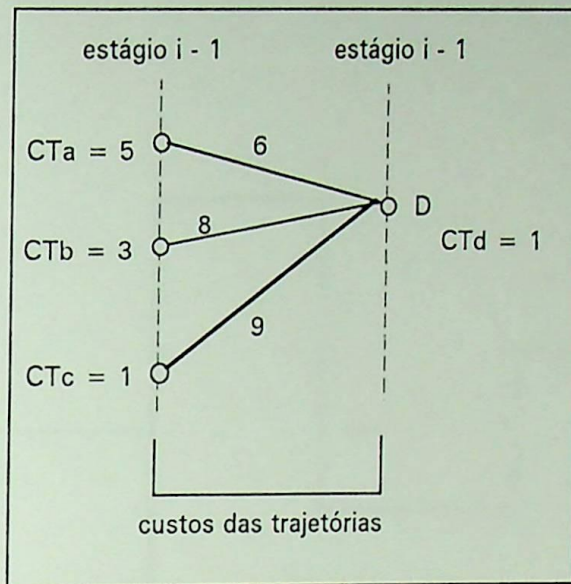


figura 5.3 - demonstração da escolha do menor custo das trajetórias

Definidas as variáveis do problema proposto, bem como as equações que regem seu comportamento, pode-se agora mostrar o exemplo a seguir, que é baseado na curva de carga simplificada da figura 5.4.

O exemplo em questão possui 4 estágios, onde cada um destes compreende um intervalo de 6 horas. O objetivo é despachar a mesma energia mas de maneira a se ter menos perdas. Existe, entretanto, um limite a essa flexibilidade, indicada pelos limites máximos e mínimos. Isto é, estes limites indicam as restrições ao despacho de carga a cada intervalo.

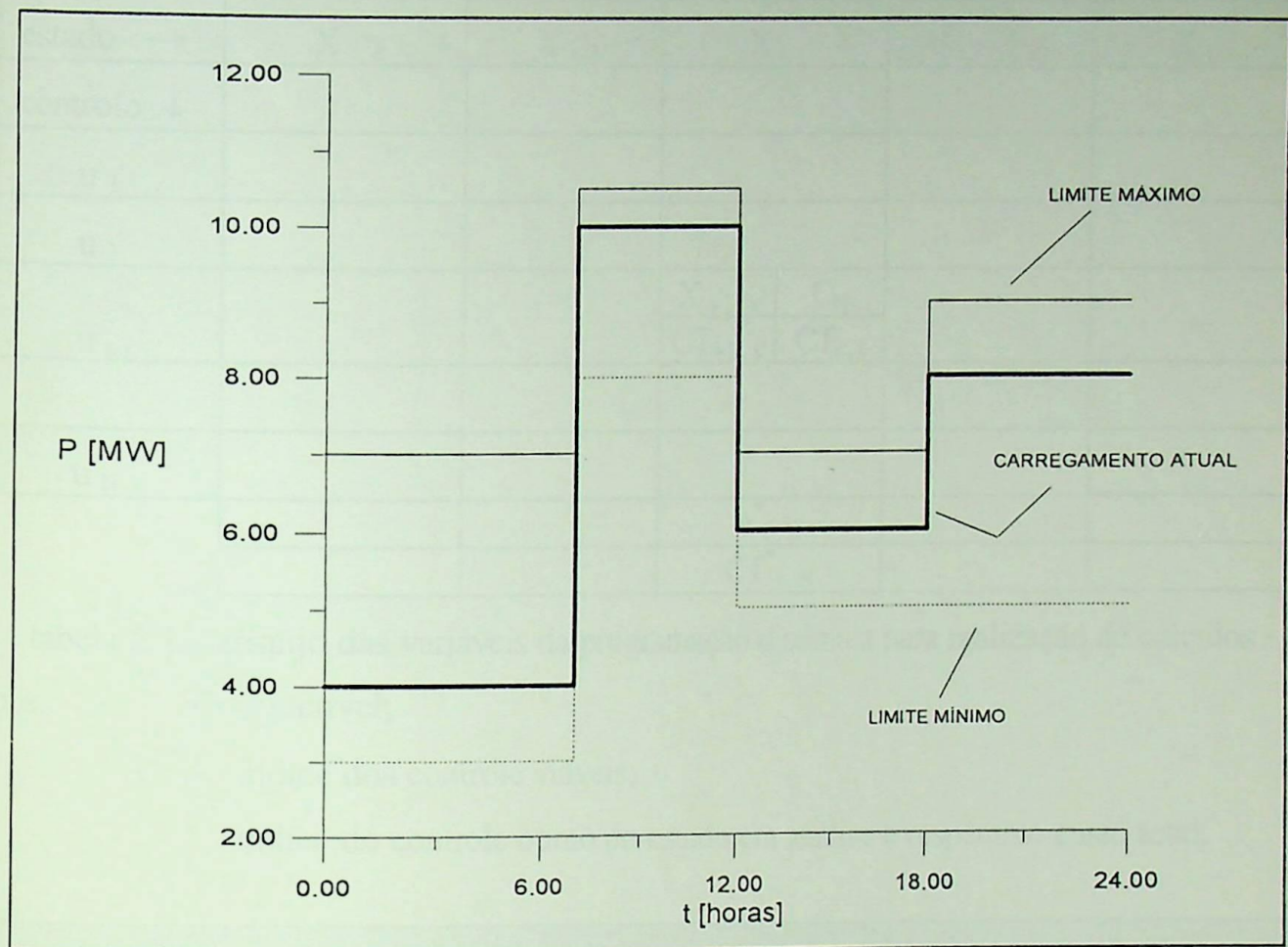


figura 5.4 - ciclo de carga típico a ser modulado

ESTÁGIOS (h)	limite mínimo (MW)	limite máximo (MW)	CONTROLES (MW)
1	3	7	3 ; 4 ; 5 ; 6 ; 7
2	8	10	8 ; 9 ; 10
3	5	7	5 ; 6 ; 7
4	5	9	5 ; 6 ; 7 ; 8 ; 9

tabela 5.1 - características das variáveis do problema proposto

obs. : cada estágio compreende um período de seis horas.

A tabela 5.2 apresenta o arranjo a ser adotado no exemplo, para a apresentação dos resultados nos diferentes estágios conforme tabelas 5.3, 5.4, 5.5 e 5.6

Estágio i					
estado →	$X_{i,1}$	$X_{i,2}$	$X_{i,j}$		$X_{i,J}$
controle ↓					
u_1					
u_2					
u_n			$X_{i-1,k}$	C_N	
			$CT_{i-1,k}$	$CT_{i,j}$	
u_N					
				$u_{i,N}^*$	
				$CT_{i,N}^*$	

tabela 5.2 - arranjo das variáveis da programação dinâmica para realização de cálculos

- obs.:
- - inactivél;
 - n - índice dos controle viáveis;
 - * - índice do controle ótimo do estado em análise e respectivo custo total.

Estágio 1										
estado →	3		4		5		6		7	
controle ↓										
3	0	9	1	•	2	•	3	•	4	•
	0	9	•	•	•	•	•	•	•	•
4	-1	•	0	16	1	•	2	•	3	•
	•	•	0	16	•	•	•	•	•	•
5	-2	•	-1	•	0	25	1	•	2	•
	•	•	•	•	0	25	•	•	•	•
6	-3	•	-2	•	-1	•	0	36	1	•
	•	•	•	•	•	•	0	36	•	•
7	-4	•	-3	•	-2	•	-1	•	0	49
	•	•	•	•	•	•	•	•	0	49
	3		4		5		6		7	
	9		16		25		36		49	

tabela 5.3 - cálculos do estágio 1

Estágio 2															
estado →	11		12		13		14		15		16		17		
controle ↓															
8	3	64	4	64	5	64	6	64	7	64	8	•	9	•	
	9	73	16	80	25	89	36	100	49	113	•	•	•	•	
9	2	•	3	81	4	81	5	81	6	81	7	81	8	•	
	•	•	9	90	16	97	25	106	36	117	49	130	•	•	
10	1	•	2	•	3	100	4	100	5	100	6	100	7	100	
	•	•	•	•	9	109	16	116	25	125	36	136	49	149	
		8		8		8		8		8		9		10	
		73		80		89		100		113		130		149	

tabela 5.4 - cálculos do estágio 2

Estágio 3															
estado →	16		17		18		19		20		21		22		
controle ↓															
5	11	25	12	25	13	25	14	25	15	25	16	25	17	25	
	73	98	80	105	89	114	100	125	113	138	130	155	149	174	
6	10	•	11	36	12	36	13	36	16	36	15	36	16	36	
	•	•	73	109	80	116	89	125	100	136	113	149	130	166	
7	9	•	10	•	11	49	12	49	17	49	14	49	15	49	
	•	•	•	•	73	122	80	129	89	138	100	149	113	162	
		5		5		5		5/6		6		6/7		7	
		98		105		114		125		136		149		162	
estado →	23		24												
controle ↓															
5	18	•	19	•											
	•	•	•	•											
6	17	36	18	•											
	149	185	•	•											
7	16	49	17	49											
	130	179	149	198											
		7		7											
		179		198											

tabela 5.5 - cálculos do estágio 3

Estágio 4		
estado →	28	
controle ↓		
5	23	25
	179	204
6	22	36
	162	198
7	21	49
	149	198
8	20	64
	136	200
9	19	81
	125	206
	6/7	
	198	

tabela 5.6 - cálculos do estágio 4

De posse das tabelas 5.3, 5.4, 5.5 e 5.6, pode-se recuperar os caminhos ótimos, partindo-se do menor custo total ótimo. A figura 5.5 ilustra esta afirmação.

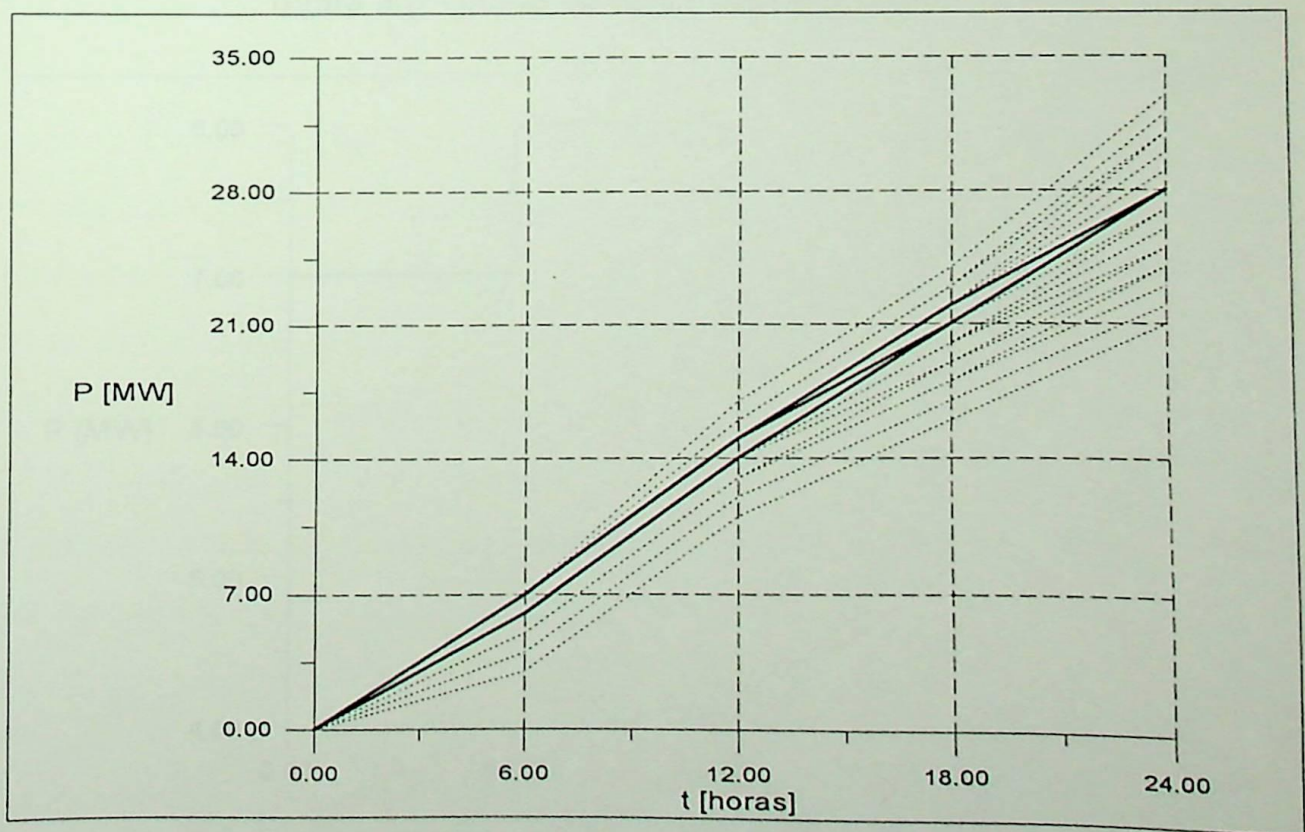


figura 5.5 caminhos ótimos com menor perda total

A figura 5.5 ilustra todos as possíveis trajetórias a serem percorridas segundo a equação recursiva e o critério de performance. As linhas mais escuras representam as 3 trajetórias ótimas resultantes da discretização. Desta forma, é possível então converter tais trajetórias em curvas de carga, as quais estão representadas pelas figuras 5.6, 5.7 e 5.8.

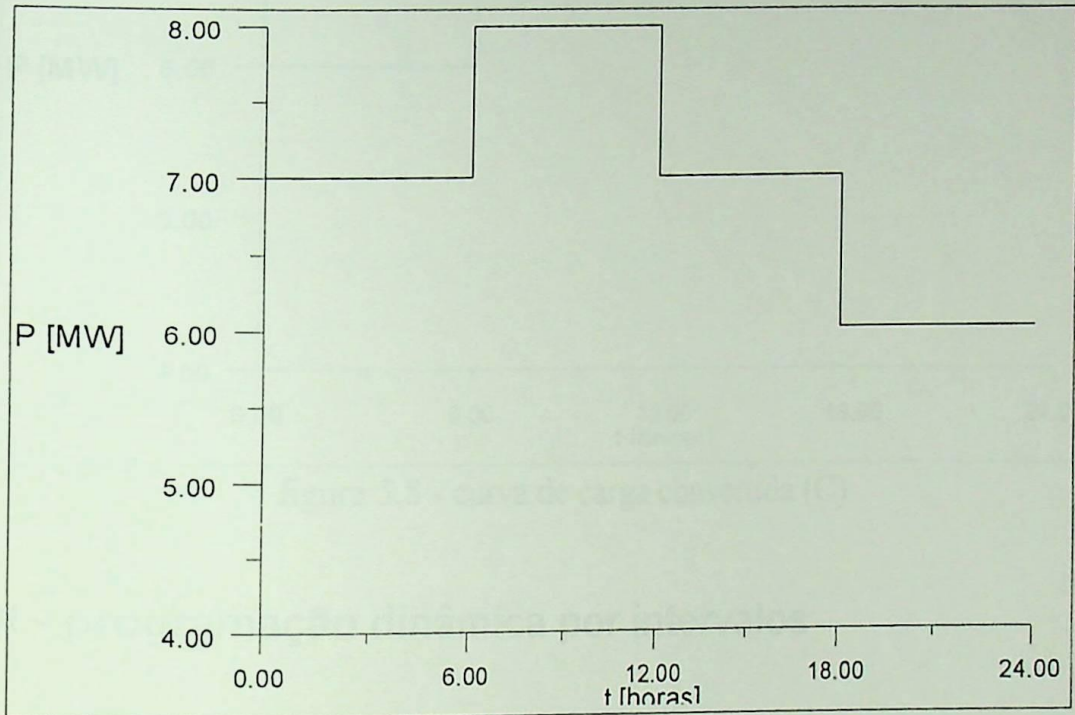


figura 5.6 - curva de carga convertida (A)

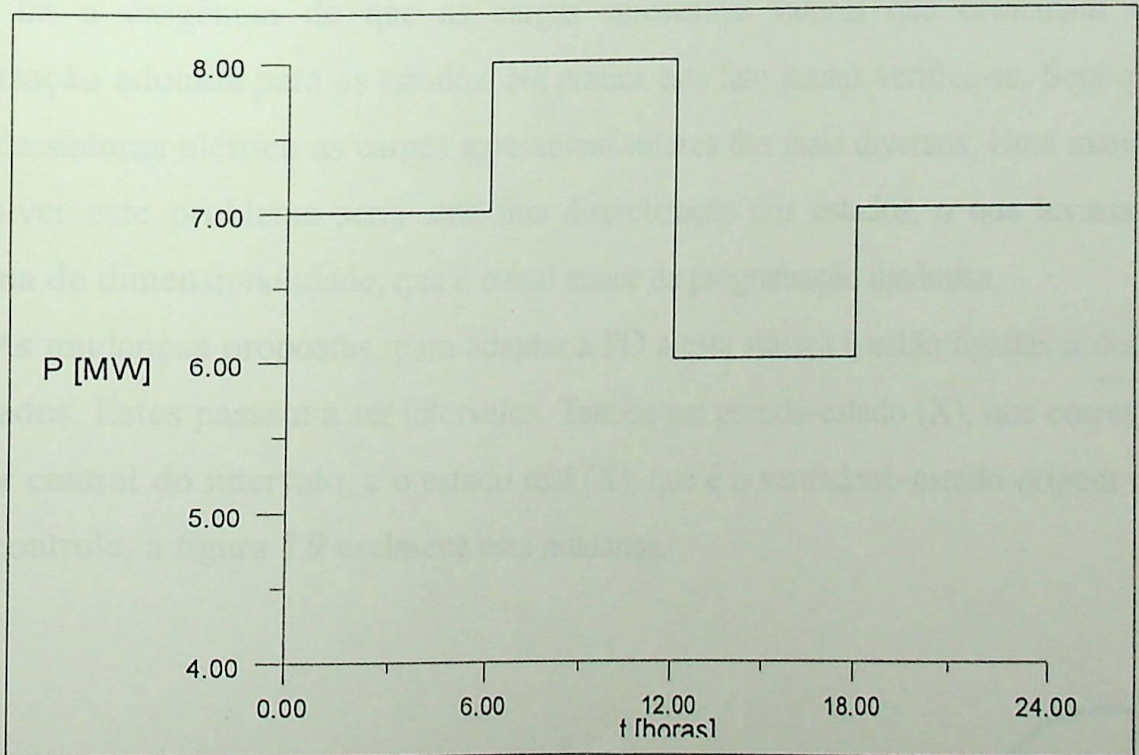


figura 5.7 - curva de carga convertida (B)

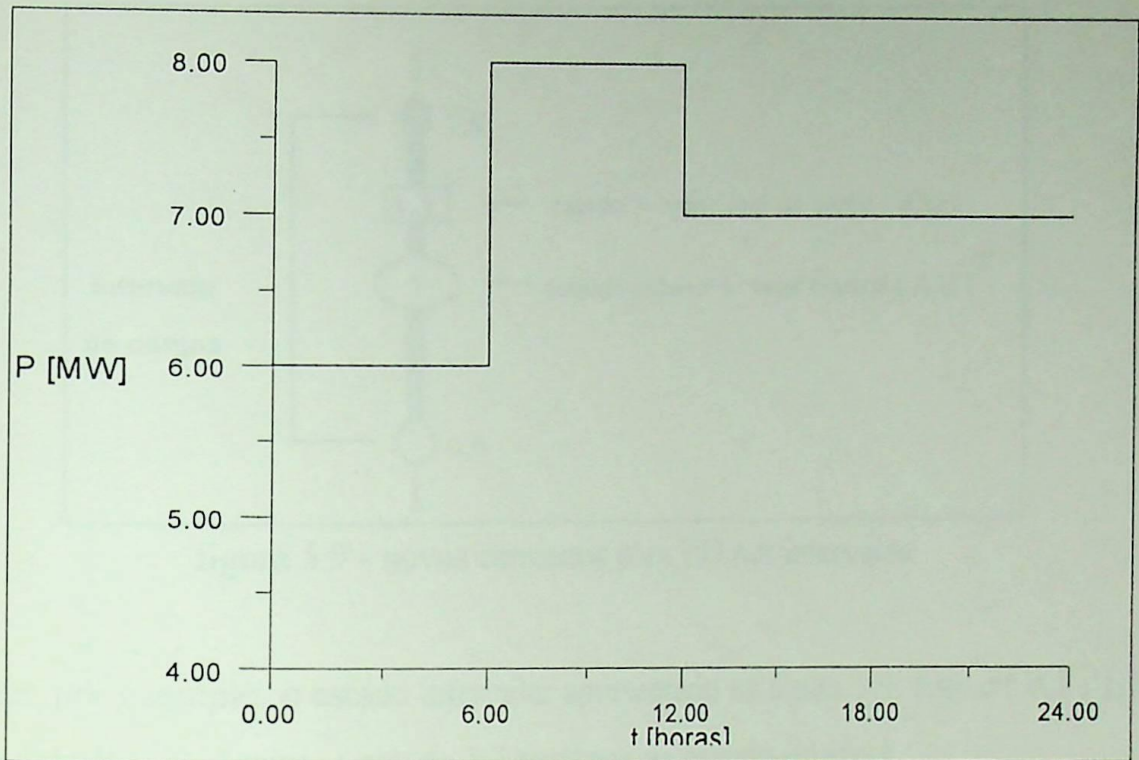
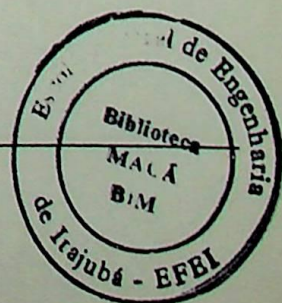


figura 5.8 - curva de carga convertida (C)

5.5 PDI - programação dinâmica por intervalos

A programação dinâmica em sua forma convencional apresenta bons resultados, porém há a exigência de que as cargas apresentem valores que coincidam com a discretização adotada para os estados. Na prática este fato jamais verifica-se. Seja qual for o tipo de sistema elétrico as cargas apresentam valores dos mais diversos. Uma maneira de se resolver este problema seria uma fina discretização dos estados, o que levaria a um problema de dimensionalidade, que é o mal maior da programação dinâmica.

As mudanças propostas, para adaptar a PD a esta situação, estão ligadas à definição dos estados. Estes passam a ser intervalos. Tem-se um pseudo-estado (X), que corresponde ao valor central do intervalo, e o estado real (X), que é o verdadeiro estado origem ou fim de um controle, a figura 5.9 esclarece esta mudança.



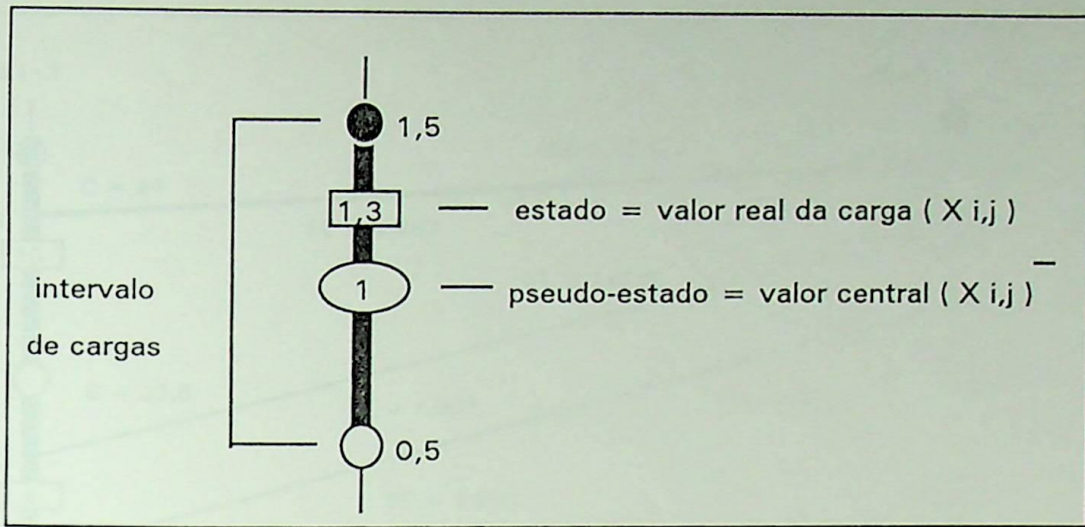


figura 5.9 - novos conceitos para PD por intervalos

Seja, por exemplo, o estado intervalar apresentado na figura 5.9. Este é $(0,5 ; 1,5]$ e o pseudo-estado é 1. Assim, o estado 1,3 pertence ao pseudo-estado 1.

Tem-se como meta escolher o controle ótimo associado a cada pseudo-estado. Entretanto, estará sendo feita, muitas vezes, uma comparação entre estados diferentes, dentro do mesmo pseudo-estado. Portanto a comparação do custo não é suficiente. Propõe-se aqui, o uso de um decisor (PE), que é a relação entre o quadrado da carga total despachada e o respectivo custo total.

$$PE = \frac{X_{i,j}^2}{CT_{i,j}} \quad (5.4)$$

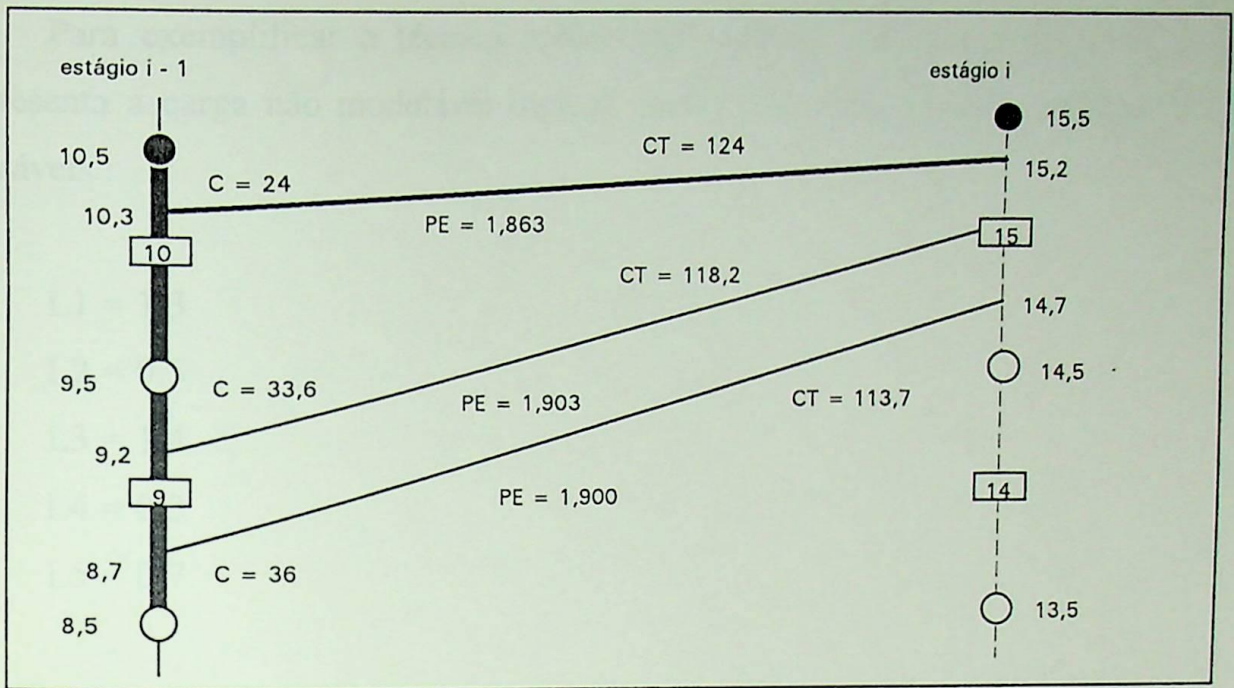


figura 5.10 -busca do controle ótimo do pseudo-estado 15

A figura 5.10 ilustra o uso do PE, na tomada de decisão. A trajetória superior tem maior custo (124), mas está associada a um maior estado (15,2).

Tem-se, agora, que armazenar, para cada pseudo-estado, o controle ótimo, o custo total, e o estado real. O princípio da PDI é a eliminação de estados próximos, de maneira a reduzir sua dimensão.

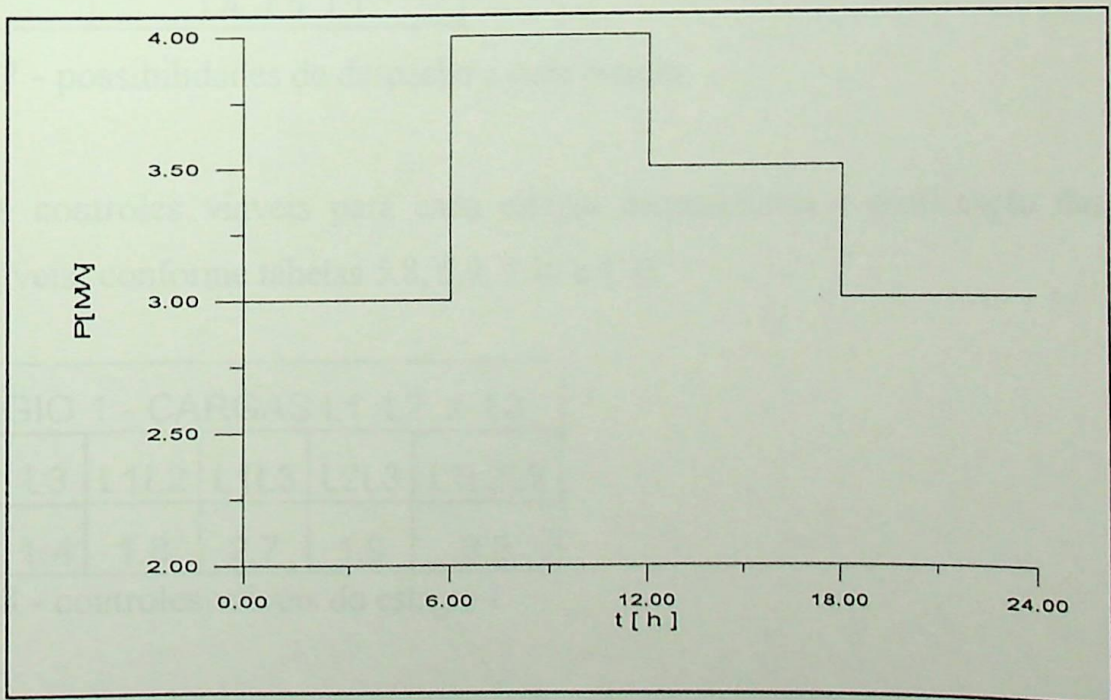


figura 5.11 - curva de carga não modulável (típica)

Para exemplificar a técnica apresentada, utiliza-se da curva da figura 5.11, que representa a carga não modulável (típica). Sobre esta serão alocadas as seguintes cargas alocáveis :

$$L1 = 1.3$$

$$L2 = 0.5$$

$$L3 = 1.4$$

$$L4 = 0.3$$

$$L5 = 0.7$$

Entretanto, estas cargas não podem ser despachadas em qualquer instante. As possibilidades de despacho estão na tabela 5.7

ESTÁGIO	Cargas despacháveis
1	L1, L2, L3
2	L1, L3
3	L4, L5, L2
4	L4, L5, L1

tabela 5.7 - possibilidades de despacho a cada estágio

Os controles viáveis para cada estágio correspondem à combinação das cargas despacháveis, conforme tabelas 5.8, 5.9, 5.10 e 5.11

ESTÁGIO 1 - CARGAS L1 L2 e L3						
L1	L2	L3	L1L2	L1L3	L2L3	L1L2L3
1.3	0.5	1.4	1.8	2.7	1.9	3.2

tabela 5.8 - controles viáveis do estágio 1

ESTÁGIO 2 - CARGAS L1 e L3		
L1	L3	L1L3
1.3	1.4	2.7

tabela 5.9 - controles viáveis do estágio 2

ESTÁGIO 3 - CARGAS L4 L5 e L2						
L4	L5	L2	L4L5	L4L2	L5L2	L4L5L2
0.3	0.7	0.5	1	0.8	1.2	1.5

tabela 5.10 - controles viáveis do estágio 3

ESTÁGIO 4 - CARGAS L4 L5 e L2						
L4	L5	L1	L4L5	L4L1	L5L1	L4L5L1
0.3	0.7	0.3	1	1.6	2	2.3

tabela 5.11 - controles viáveis do estágio 4

Os pseudo-estados corresponderão a intervalos de amplitude igual a um, aberto em limite inferior e fechado no superior. As tabelas 5.12, 5.13, 5.14, 5.15, 5.16 e 5.17 apresentam o estudo em pauta.

ESTÁGIO	1	INTERVALO	3 À 6	CARGA MÍNIMA	3
CARGAS VARIÁVEIS			L1 L2 L3 e suas combinações		

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	V	INTERVALO	C_t	PE	Cargas
0	0	0	3	3	0	-0.5 <> 0.5	9	1	0
1.3	0	0	3	•	-1.3	-1.8 <> -0.8	•	∅	•
0.5	0	0	3	3.5	-0.5	-1 <> 0	12.25	1	L2
1.4	0	0	3	•	-1.4	-1.9 <> -0.9	•	∅	•
1.8	0	0	3	•	-1.8	-2.3 <> -1.3	•	∅	•
2.7	0	0	3	•	-2.7	-3.2 <> -2.2	•	∅	•
1.9	0	0	3	•	-1.9	-2.4 <> -1.4	•	∅	•
3.2	0	0	3	•	-3.2	-3.7 <> -2.7	•	∅	•
0	0	0	4	•	1	0.5 <> 1.5	•	∅	•
1.3	0	0	4	4.3	-0.3	-0.8 <> 0.2	18.49	1	L1
0.5	0	0	4	•	0.5	0 <> 1	•	∅	•
1.4	0	0	4	4.4	-0.4	-0.9 <> 0.1	19.36	1	L3
1.8	0	0	4	•	-0.8	-1.3 <> -0.3	•	∅	•
2.7	0	0	4	•	-1.7	-2.2 <> -1.2	•	∅	•
1.9	0	0	4	•	-0.9	-1.4 <> -0.4	•	∅	•
3.2	0	0	4	•	-2.2	-2.7 <> -1.7	•	∅	•
0	0	0	5	•	2	1.5 <> 2.5	•	∅	•
1.3	0	0	5	•	0.7	0.2 <> 1.2	•	∅	•
0.5	0	0	5	•	1.5	1 <> 2	•	∅	•
1.4	0	0	5	•	0.6	0.1 <> 1.1	•	∅	•
1.8	0	0	5	4.8	0.2	-0.3 <> 0.7	23.04	1	L1L2
2.7	0	0	5	•	-0.7	-1.2 <> 1.2	•	∅	•
1.9	0	0	5	4.9	0.1	-0.4 <> 0.6	24.01	1	L2L3
3.2	0	0	5	•	-1.2	-1.7 <> 0.7	•	∅	•
0	0	0	6	•	3	2.5 <> 3.5	•	∅	•
1.3	0	0	6	•	1.7	1.2 <> 2.2	•	∅	•
0.5	0	0	6	•	2.5	2 <> 3	•	∅	•
1.4	0	0	6	•	1.6	1.1 <> 2.1	•	∅	•
1.8	0	0	6	•	1.2	0.7 <> 1.7	•	∅	•
2.7	0	0	6	5.7	0.3	-0.3 <> 0.8	32.49	1	L1L3
1.9	0	0	6	•	1.1	0.6 <> 1.6	•	∅	•
3.2	0	0	6	6.2	-0.2	-0.7 <> 0.3	38.44	1	L1L2L3

tabela 5.12 - modulação PDI para o estágio 1

ESTÁGIO	2	INTERVALO	7 À 10	CARGA MÍNIMA	4
CARGAS VARIÁVEIS			L1 L3 e suas combinações		

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	V	INTERVALO	Ct	PE	Cargas
0	3	3	7	7	3	2.5 <> 3.5	25	1.96	0/0
	3	3.5	7	7.5	3	2.5 <> 3.5	28.25	1.991	L2/0
1.3	•	•	7	•	1.7	1.2 <> 2.2	•	∅	•
1.4	•	•	7	•	1.6	1.1 <> 2.1	•	∅	•
2.7	•	•	7	•	0.3	-0.2 <> 0.8	•	∅	•
0	4	4.3	8	8.3	4	3.5 <> 4.5	34.49	1.997	L1/0
	4	4.4	8	8.4	4	3.5 <> 4.5	35.36	1.995	L3/0
1.3	3	3	8	8.3	2.7	2.2 <> 3.2	37.09	1.857	0/L1
1.4	3	3	8	8.4	2.6	2.1 <> 3.1	38.16	1.849	0/L3
2.7	•	•	8	•	1.3	0.8 <> 1.8	•	∅	•
0	5	4.8	9	8.8	5	4.5 <> 5.5	39.04	1.983	L1L2/0
	5	4.9	9	8.9	5	4.5 <> 5.5	40.01	1.979	L2L3/0
1.3	3	3.5	9	8.8	3.7	3.2 <> 4.2	40.34	1.919	L2/L1
1.4	3	3.5	9	8.9	3.6	3.1 <> 4.1	41.41	1.913	L2/L3
2.7	•	•	9	•	2.3	1.8 <> 2.8	•	∅	•
0	6	5.7	10	9.7	6	5.5 <> 6.5	48.49	1.940	L1L3/0
	6	6.2	10	10.2	6	5.5 <> 6.5	54.44	1.911	L1L2L3/0
1.3	•	L1	10	•	4.7	4.2 <> 5.2	•	∅	•
	4	4.4	10	9.7	4.7	4.2 <> 5.2	47.45	1.983	L3/L1
	•	L1L2	10	•	4.7	4.2 <> 5.2	•	∅	•
	5	4.9	10	10.2	4.7	4.2 <> 5.2	52.1	1.977	L2L3/L1
1.4	4	4.3	10	9.7	4.6	4.1 <> 5.1	47.65	1.975	L1/L3
	•	L3	10	•	4.6	4.1 <> 5.1	•	∅	•
	5	4.8	10	10.2	4.6	4.1 <> 5.1	52.2	1.973	L1L2/L3
	•	L2L3	10	•	4.6	4.1 <> 5.1	•	∅	•
2.7	3	3	10	9.7	3.3	2.8 <> 3.8	53.89	1.746	0/L1L3
	3	3.5	10	10.2	3.3	2.8 <> 3.8	57.14	1.821	L2/L1L3

tabela 5.13 - modulação PDI para o estágio 2

ESTÁGIO	3	INTERVALO	10 À 15	CARGA MÍNIMA	3.5
CARGAS VARIÁVEIS			L4 L5 L2 e suas combinações		

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	v	INTERVALO	Ct	PE	Cargas
0	7	7	10	10.5	6.5	6 <> 7	37.25	2.960	0/0/0
0.3	•	•	10	•	6.2	5.7 <> 6.7	•	∅	•
0.7	•	•	10	•	5.8	5.3 <> 6.3	•	∅	•
0.5	•	•	10	•	6	5.5 <> 6.5	•	∅	•
1	•	•	10	•	5.5	5 <> 6	•	∅	•
0.8	•	•	10	•	5.7	5.2 <> 6.2	•	∅	•
1.2	•	•	10	•	5.3	4.8 <> 5.8	•	∅	•
1.5	•	•	10	•	5	4.5 <> 5.5	•	∅	•
0	7	7.5	11	11	7.5	7 <> 8	40.5	2.988	L2/0/0
0.3	7	7	11	10.8	7.2	6.7 <> 7.7	39.44	2.957	0/0/L4
	7	7.5	11	11.3	7.2	6.7 <> 7.7	42.69	2.991	L2/0/L4
0.7	7	7	11	11.2	6.8	6.3 <> 7.3	42.64	2.942	0/0/L5
0.5	7	7	11	11	7	6.5 <> 7.5	41	2.951	0/0/L2
	•	L2	11	•	7	6.5 <> 7.5	•	∅	•
1	7	7	11	11.5	6.5	6 <> 7	45.25	2.923	0/0/L4L5
0.8	7	7	11	11.3	6.7	6.2 <> 7.2	43.49	2.936	0/0/L4L2
1.2	•	•	11	•	6.3	5.8 <> 6.8	•	∅	•
1.5	•	•	11	•	6	5.5 <> 6.5	•	∅	•
0	8	8.3	12	11.8	8.5	8 <> 9	46.74	2.979	L1/0/0
	8	8.4	12	11.9	8.5	8 <> 9	47.61	2.974	L3/0/0
	9	8.8	12	12.3	8.5	8 <> 9	51.29	2.950	L1L2/0/0
	9	8.9	12	12.4	8.5	8 <> 9	52.26	2.942	L2L3/0/0
0.3	8	8.3	12	12.1	8.2	7.7 <> 8.7	48.93	2.992	L1/0/L4
	8	8.4	12	12.2	8.2	7.7 <> 8.7	49.8	2.989	L3/0/L4
0.7	7	7.5	12	11.7	7.8	7.3 <> 8.3	45.89	2.983	L2/0/L5
	8	8.3	12	12.5	7.8	7.3 <> 8.3	52.13	2.997	L1/0/L5
0.5	8	8.3	12	12.3	8	7.5 <> 8.5	50.49	2.996	L1/0/L2
	8	8.4	12	12.4	8	7.5 <> 8.5	51.36	2.994	L3/0/L2
1	7	7.5	12	12	7.5	7 <> 8	48.5	2.969	L2/0/L4L5
0.8	•	L2/0	12	•	7.7	7.2 <> 8.2	•	∅	•
1.2	7	7	12	11.7	7.3	6.8 <> 7.8	47.09	2.907	0/0/L5L2
	•	L2/0	12	•	7.3	6.8 <> 7.8	•	∅	•
1.5	7	7	12	12	7	6.5 <> 7.5	50	2.880	0/0/L4L5L2
	•	L2/0	12	•	7	6.5 <> 7.5	•	∅	•

tabela 5.14 - modulação PDI para o estágio 3

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	V	INTERVALO	Ct	PE	Cargas
0	10	9.7	13	13.2	9.5	9 < > 10	59.7	2.918	L3/L1/0
	10	10.2	13	13.7	9.5	9 < > 10	64.35	2.917	L2L3/L1/0
1	8	8.3	13	12.8	8.5	8 < > 9	57.74	2.837	L1/0/L4L5
	8	8.4	13	12.9	8.5	8 < > 9	55.61	2.992	L3/0/L4L5
	9	8.8	13	13.3	8.5	8 < > 9	59.29	2.983	L1L2/0/L4L5
	9	8.9	13	13.4	8.5	8 < > 9	60.26	2.980	L2L3/0/L4L5
0.8	8	8.3	13	12.6	8.7	8.2 < > 9.2	52.98	2.997	L1/0/L4L2
	8	8.4	13	12.7	8.7	8.2 < > 9.2	83.85	2.923	L3/0/L4L2
	•	L1L2	13	•	8.7	8.2 < > 9.2	•	∅	•
	•	L2L3	13	•	8.7	8.2 < > 9.2	•	∅	•
0.3	9	8.8	13	12.6	9.2	8.7 < > 9.7	53.48	2.968	L1L2/0/L4
	9	8.9	13	12.7	9.2	8.7 < > 9.7	54.45	2.016	L2L3/0/L4
	10	9.7	13	13.5	9.2	8.7 < > 9.7	61.89	2.945	L3/L1/L4
0.7	8	8.4	13	12.6	8.8	8.3 < > 9.3	53	2.995	L3/0/L5
	9	8.8	13	13	8.8	8.3 < > 9.3	56.68	2.982	L1L2/0/L5
	9	8.9	13	13.1	8.8	8.3 < > 9.3	57.65	2.977	L2L3/0/L5
0.5	•	L1L2	13	•	9	8.5 < > 9.5	•	∅	•
	•	L2L3	13	•	9	8.5 < > 9.5	•	∅	•
1.2	8	8.3	13	13	8.3	7.8 < > 8.8	56.68	2.982	L1/0/L5L2
	8	8.4	13	13.1	8.3	7.8 < > 8.8	57.45	2.987	L3/0/L5L2
	•	L1L2	13	•	8.3	7.8 < > 8.8	•	∅	•
1.5	8	8.3	13	13.3	8	7.5 < > 8.5	59.49	2.973	L1/0/L4L5L2
	8	8.4	13	13.4	8	7.5 < > 8.5	60.36	2.975	L3/0/L4L5L2
0	10	10.2	14	13.7	10.5	10 < > 11	64.35	2.917	L2L3/L1/0
0.3	10	10.2	14	14	10.2	9.7 < > 10.7	66.54	2.946	L2L3/L1/L4
0.7	10	9.7	14	13.9	9.8	9.3 < > 10.3	65.09	2.968	L3/L1/L5
	10	10.2	14	14.4	9.8	9.3 < > 10.3	69.74	2.973	L2/L3/L1
0.5	10	9.7	14	13.7	10	9.5 < > 10.5	63.45	2.958	L3/L1/L2
	•	L2L3	14	•	10	9.5 < > 10.5	•	∅	•
1	10	9.7	14	14.2	9.5	9 < > 10	67.7	2.978	L3/L1/L4L5
0.8	10	9.7	14	14	9.7	9.2 < > 10.2	65.94	2.972	L3/L1/L4L2
	•	L2L1	14	•	9.7	9.2 < > 10.2	•	∅	•
1.2	•	L2L3	14	•	9.3	8.8 < > 9.8	•	∅	•
	10	9.7	14	14.4	9.3	8.8 < > 9.8	69.54	2.982	L3/L1/L5L2
1.5	•	L1L2	14	•	9	8.5 < > 9.5	•	∅	•
	•	L2L3	14	•	9	8.5 < > 9.5	•	∅	•

tabela 5.15 - modulação PDI para o estágio 3 - continuação da tabela 5.14

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	V	INTERVALO	Ct	PE	Cargas
0	•	•	15	∉	11.5	11 <> 12	•	∃	•
0.3	•	•	15	∉	11.2	10.7 <> 11.7	•	∃	•
0.7	•	•	15	∉	10.8	10.3 <> 11.3	•	∃	•
1.5	•	•	15	∉	11	10.5 <> 11.5	•	∃	•
1	10	10.2	15	14.7	10.5	10 <> 11	72.35	2.987	L2L3/L1/L4L5
0.8	•	•	15	∉	10.7	10.2 <> 11.2	•	∃	•
1.2	•	L2L3	15	•	10.3	9.8 <> 10.8	•	∃	•
1.5	10	9.7	15	14.7	10	9.5 <> 10.5	72.45	2.983	L3/L1/L4L5L2
	•	L2L3	15	•	•	•	•	∃	•

tabela 5.16 - modulação PDI para o estágio 3 - continuação da tabela 5.14 e 5.15

ESTÁGIO	4	INTERVALO	18	CARGA MÍNIMA	3
CARGAS VARIÁVEIS			L4 L5 L1 e suas combinações		

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	V	INTERVALO	Ct	PE	Cargas
0	14	14.7	18	17.7	15	14.5 <> 15.5	81.35	3.851	L2L3/L1/L4L5/0
0.3	14	14.4	18	17.7	14.7	14.2 <> 15.2	80.43	3.895	L3/L1/L5L2/L4
0.7	•	L4L5	18	•	14.3	13.8 <> 14.8	•	∃	•
1.3	•	•	18	18.5	13.7	13.2 <> 14.2	•	∃	•
1	•	•	18	18.4	14	13.5 <> 14.5	•	∃	•
1.6	•	•	18	17.6	13.4	12.9 <> 13.9	•	∃	•
2	13	12.7	18	17.7	13	12.5 <> 13.5	78.85	3.973	L3/0/L4L2/L5L1
2.3	•	L1L5	18	•	12.7	12.2 <> 13.2	•	∃	•

tabela 5.17 - modulação PDI para o estágio 4

Os quadros 5.18, 5.19, 5.20 e 5.21, apresentam o resumo do estudo, com os valores ótimos de cada estado.

ESTÁGIO	1	INTERVALO	3 À 6	CARGA MÍNIMA	3
CARGAS VARIÁVEIS			L1 L2 L3 e suas combinações		

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	V	INTERVALO	Ct	PE	Cargas
0	0	0	3	3	0	-0.5 <> 0.5	9	1	0
0.5	0	0	3	3.5	-0.5	-1 <> 0	12.25	1	L2
1.3	0	0	4	4.3	-0.3	-0.8 <> 0.2	18.49	1	L1
1.4	0	0	4	4.4	-0.4	-0.9 <> 0.1	19.36	1	L3
1.8	0	0	5	4.8	0.2	-0.3 <> 0.7	23.04	1	L1L2
1.9	0	0	5	4.9	0.1	-0.4 <> 0.6	24.01	1	L2L3
2.7	0	0	6	5.7	0.3	-0.3 <> 0.8	32.49	1	L1L3
3.2	0	0	6	6.2	-0.2	-0.7 <> 0.3	38.44	1	L1L2L3

tabela 5.18 - modulação PDI para o estágio 1- resultados ótimos

ESTÁGIO	2	INTERVALO	7 À 10	CARGA MÍNIMA	4
CARGAS VARIÁVEIS			L1 L3 e suas combinações		

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	V	INTERVALO	Ct	PE	Cargas
0	3	3	7	7	3	2.5 <> 3.5	25	1.960	0/0
	3	3.5	7	7.5	3	2.5 <> 3.5	28.25	1.991	L2/0
0	4	4.3	8	8.3	4	3.5 <> 4.5	34.49	1.997	L1/0
	4	4.4	8	8.4	4	3.5 <> 4.5	35.36	1.995	L3/0
0	5	4.8	9	8.8	5	4.5 <> 5.5	39.04	1.983	L1L2/0
	5	4.9	9	8.9	5	4.5 <> 5.5	40.01	1.979	L2L3/0
	4	4.4	10	9.7	4.7	4.2 <> 5.2	47.45	1.983	L3/L1
	5	4.9	10	10.2	4.7	4.2 <> 5.2	52.1	1.977	L2L3/L1

tabela 5.19 - modulação PDI para o estágio 2- resultados ótimos

ESTÁGIO	3	INTERVALO	10 À 15	CARGA MÍNIMA	3.5
CARGAS VARIÁVEIS			L4 L5 L2 e suas combinações		

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	v	INTERVALO	Ct	PE	Cargas
0	7	7	10	10.5	6.5	6 <> 7	37.25	2.960	0/0/0
0.3	7	7	11	10.8	7.2	6.7 <> 7.7	39.44	2.957	0/0/L4
	7	7.5	11	11.3	7.2	6.7 <> 7.7	42.69	2.991	L2/0/L4
	8	8.4	12	12.2	8.2	7.7 <> 8.7	49.8	2.989	L3/0/L4
	8	8.3	12	12.5	7.8	7.3 <> 8.3	52.13	2.997	L1/0/L5
0.8	8	8.3	13	12.6	8.7	8.2 <> 9.2	52.98	2.997	L1/0/L4L2
	8	8.4	13	12.7	8.7	8.2 <> 9.2	83.85	2.993	L3/0/L4L2
0.7	10	9.7	14	13.9	9.8	9.3 <> 10.3	65.09	2.968	L3/L1/L5
	10	9.7	14	14.4	9.3	8.8 <> 9.8	69.54	2.982	L3/L1/L5L2
1	10	10.2	15	14.7	10.5	10 <> 11	72.35	2.987	L2L3/L1/L4L5
1.5	10	9.7	15	14.7	10	9.5 <> 10.5	72.45	2.983	L3/L1/L4L5L2

tabela 5.20 - modulação PDI para o estágio 3- resultados ótimos

ESTÁGIO	4	INTERVALO	18	CARGA MÍNIMA	3
CARGAS VARIÁVEIS			L4 L5 L1 e suas combinações		

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	v	INTERVALO	Ct	PE	Cargas
0	14	14.7	18	17.7	15	14.5 <> 15.5	81.35	3.851	L2L3/L1/L4L5/0
0.3	14	14.4	18	17.7	14.7	14.2 <> 15.2	80.43	3.895	L3/L1/L5L2/L4
2	13	12.7	18	17.7	13	12.5 <> 13.5	78.85	3.973	L3/0/L4L2/L5L1

tabela 5.21 - modulação PDI para o estágio 4- resultados ótimos

A curva final, após a modulação é apresentada na figura 5.12.

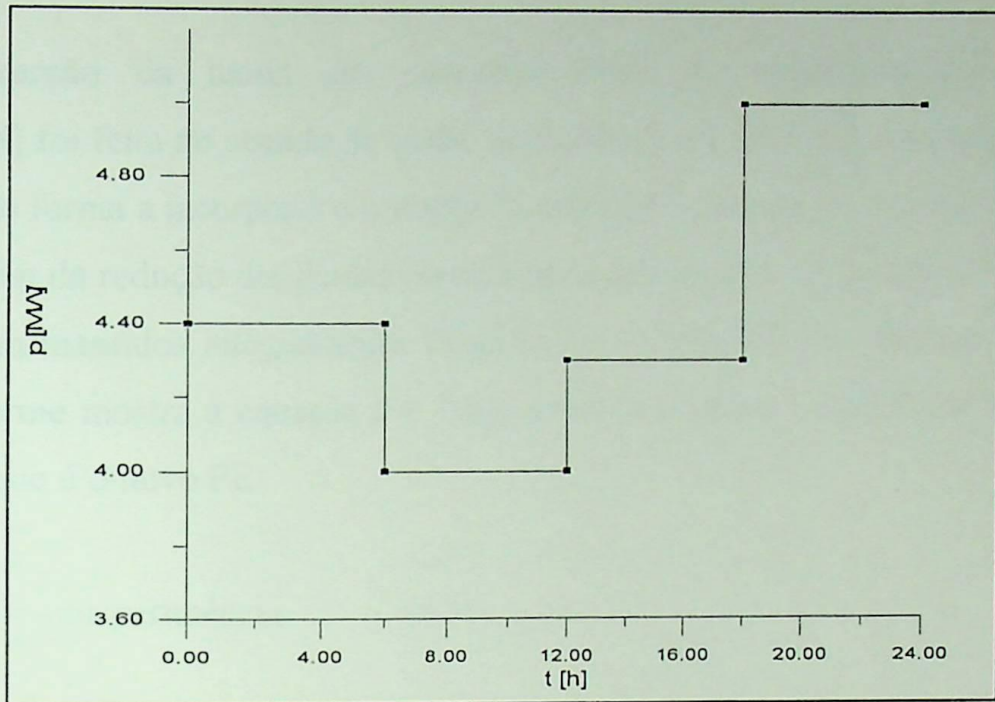
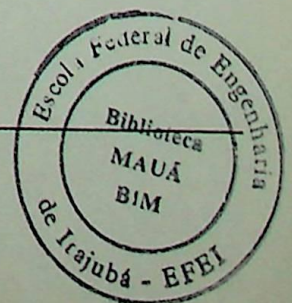


figura 5.12 - curva de carga após modulação com PDI

O que difere a programação dinâmica por intervalos da convencional é que durante o processo de cálculos busca-se apenas, um valor de estado real dentro de cada pseudo-estado (intervalo), desprezando-se as demais possibilidades isto, com certeza, não levará ao ótimo verdadeiro, mas, como acontece com a própria discretização, conduzirá a um valor muito próximo a esse ótimo

5.6 Técnica de programação dinâmica por intervalos associada a teoria dos conjuntos fuzzy

A associação dos conjuntos fuzzy, ou conjuntos difusos, à técnica de programação dinâmica é perfeitamente viável. Esta junção já foi objeto de várias aplicações, como descrito em [23] e [29]. A mesma pode ser feita de várias maneiras, desde o uso de variáveis lingüísticas, *fuzzificação* de equações recursivas e posterior *desfuzzificação*, ou mesmo o uso de algumas de suas propriedades, descritas no capítulo 4.



A inserção da teoria dos conjuntos difusos à programação dinâmica por intervalos[28] foi feita no sentido de ajudar na decisão de escolha entre as várias trajetórias existentes, de forma a incorporar o conceito de menor carregamento possível nos horários de ponta, além da redução das perdas buscada no modelo anterior. Os demais procedimentos da PDI foram mantidos integralmente. Desta forma, o PE da PDI foi substituído pelo PE-fuzzy, conforme mostra a equação 5.4. Logo a melhor trajetória é identificada pela maior pertinência que é o novo PE.

$$Pe_fuzzy = \text{maior pertinência} \quad (5.5)$$

Assim sendo, o procedimento adotado foi o de separar as cargas passíveis de realocação no processo sob estudo. Estas cargas, divididas agora em conjuntos, podem receber graus de pertinência, associando períodos preferenciais para seu funcionamento. Tais períodos podem ser classificados como bom, ruim, regular, ótimo, muito bom, etc.

Por exemplo, a iluminação de um pátio no período das 7:00 às 17:00 horas é considerado péssimo, assim pode-se atribuir um baixo grau de pertinência à mesma. Logo $\mu_{iluminação} = 0.05$, e assim por diante. Desta maneira pode-se ter os seguintes graus de pertinência associados a diversas situações.

Muito ruim	- pertinência 0.2
Ruim	- pertinência 0.4
Regular	- pertinência 0.6
Bom	- pertinência 0.7
Muito bom	- pertinência 0.8
Ótimo	- pertinência 0.9

A tabela 5.22 mostra as cargas passíveis de deslocamento no processo, seu horários de funcionamento e o grau de pertinência atribuído a estas, assim como a tabela 5.23 mostra uma alternativa melhor para a alocação destas cargas, sempre respeitando o processo produtivo.

CARGAS OU COMBINAÇÕES DE CARGAS	DESCRIÇÃO	POTÊNCIA (MVA)	HORÁRIO ATUAL DE FUNCIONAMENTO	PERTINÊNCIA
L1	FORNO 1	2	17:00 - 20:00	0.4
L2	FORNO 2	3	20:00 - 22:00	0.2
L3	MOTOR 1	0,5	17:00 - 20:00	0.4
L4	ESTUFA 1	2	17:00 - 20:00	0.6
L5	MOTOR 2	0,5	20:00 - 22:00	0.4
L6	ESTUFA 2	1	20:00 - 22:00	0.6
<hr/>				
L2 - L5 - L6		4,5	20:00 - 22:00	0.808
L2 - L5		3,5		0.520
L2 - L6		4,0		0.680
L5 - L6		1,5		0.760
<hr/>				
L1 - L3 - L4		4,5	17:00 - 20:00	0.856
L1 - L3		2,5		0.640
L1 - L4		4,0		0.760
L3 - L4		2,5		0.760

tabela 5.22 - tabela associativa carregamento atual

CARGAS OU COMBINAÇÕES DE CARGAS	DESCRIÇÃO	POTÊNCIA (MVA)	HORÁRIO PROPOSTO DE FUNCIONAMENTO	PERTINÊNCIA
L1	FORNO 1	2	11:00 - 14:00	0.7
L2	FORNO 2	3	14:00 - 16:00	0.8
L3	MOTOR 1	0,5	11:00 - 14:00	0.8
L4	ESTUFA 1	2	11:00 - 14:00	0.9
L5	MOTOR 2	0,5	14:00 - 16:00	0.6
L6	ESTUFA 2	1	14:00 - 16:00	0.9
<hr/>				
L2 - L5 - L6		4,5	14:00 - 16:00	0.992
L2 - L5		3,5		0.920
L2 - L6		4,0		0.980
L5 - L6		1,5		0.960
<hr/>				
L1 - L3 - L4		4,5	11:00 - 14:00	0.994
L1 - L3		2,5		0.940
L1 - L4		4,0		0.970
L3 - L4		2,5		0.980

tabela 5.23 - tabela associativa carregamento otimizado

Através das citadas tabelas pode-se observar que quando existe associação de cargas a serem realocadas, o conjunto resultante tem grau de pertinência que pode ser calculado através da equação 5.5. Esta é uma das propriedades dos conjuntos difusos, denominada por *soma algébrica*.

$$\mu_{A\hat{+}B} = \mu_A(x) + \mu_B(x) - \mu_A(x) \cdot \mu_B(x) \quad (5.6)$$

Assim, por exemplo, a associação das cargas L3 e L4, no segmento horário das 11:00 às 14:00 da tabela 5.23 tem a seguinte pertinência :

$$\mu_{A\hat{+}B} = 0,8 + 0,9 - 0,8 \cdot 0,9$$

$$\mu_{A\hat{+}B} = 0,980$$

Este resultado (0,980) poderia ser interpretado como “muito além de ótimo”, de modo a qualificar a operação deste conjunto de cargas (L3-L4 = motor 1 + estufa 1) no citado segmento horário.

Desta forma, a programação dinâmica por intervalo fuzzy (PDIF) permite uma maneira alternativa de modulação de ciclos de carga. Tal modelo foi também implementado em microcomputador através da linguagem turbo pascal 7.0, conforme anexo 2. A mesma curva típica do exemplo anterior foi considerada para modulação, onde o programa escolhe as trajetórias ótimas empregando o PE_fuzzy.

As tabelas 5.24, 5.25, 5.26 e 5.27 apresentam a síntese deste estudo e a figura 5.13 a curva de carga resultante da modulação PDIF.

ESTÁGIO	1	INTERVALO	3 À 6	CARGA MÍNIMA	3
CARGAS VARIÁVEIS			L1 L2 L3 e suas combinações		

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	V	INTERVALO	Ct	PE_fuzzy	Cargas
0.5	0	0	3	3.5	-0.5	-1 <> 0	12.25	0.890	L2
1.3	0	0	4	4.3	-0.3	-0.8 <> 0.2	18.49	0.920	L1
1.8	0	0	5	4.8	0.2	-0.3 <> 0.7	23.04	0.820	L1L2
3.2	0	0	6	6.2	-0.2	-0.7 <> 0.3	38.44	0.740	L1L2L3

tabela 5.24 - modulação PDIF para o estágio 1- resultados ótimos

ESTÁGIO	2	INTERVALO	7 À 10	CARGA MÍNIMA	4
CARGAS VARIÁVEIS			L1 L3 e suas combinações		

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	V	INTERVALO	Ct	PE_fuzzy	Cargas
0	3	3.5	7	7.5	3	2.5 <> 3.5	28.25	0.800	L2/0
0	4	4.3	8	8.3	4	3.5 <> 4.5	34.49	0.840	L1/0
0	5	4.8	9	8.8	5	4.5 <> 5.5	39.04	0.710	L1L2/0
0	6	6.2	10	10.2	6	5.5 <> 6.5	54.44	0.940	L1L2L3/0

tabela 5.25 - modulação PDIF para o estágio 2- resultados ótimos

ESTÁGIO	3	INTERVALO	10 À 15	CARGA MÍNIMA	3.5
CARGAS VARIÁVEIS			L4 L5 L2 e suas combinações		

u	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	V	INTERVALO	Ct	PE_fuzzy	Cargas
0.3	7	7.5	11	11.3	7.2	6.7 <> 7.7	42.69	0.860	L2/0/L4
0.5	8	8.3	12	12.3	8	7.5 <> 8.5	50.49	0.790	L1/0/L2
1.2	8	8.3	13	13	8.3	7.8 <> 8.8	56.68	0.840	L1/0/L5L2
1.0	10	10.2	15	14.7	10.5	10 <> 11	72.35	0.680	L2L3/L1/L4L5

tabela 5.26 - modulação PDIF para o estágio 3- resultados ótimos

ESTÁGIO	4	INTERVALO	18	CARGA MÍNIMA	3
CARGAS VARIÁVEIS			L4 L5 L1 e suas combinações		

D_i	\bar{X}_{i-1}	X_{i-1}	\bar{X}_i	X_i	V	INTERVALO	C_t	PE_fuzzy	Cargas
0	14	14.7	18	17.7	15	14.5 <> 15.5	81.35	0.960	L2L3/L1/L4L5/0

tabela 5.27 - modulação PDF para o estágio 4- resultados ótimos

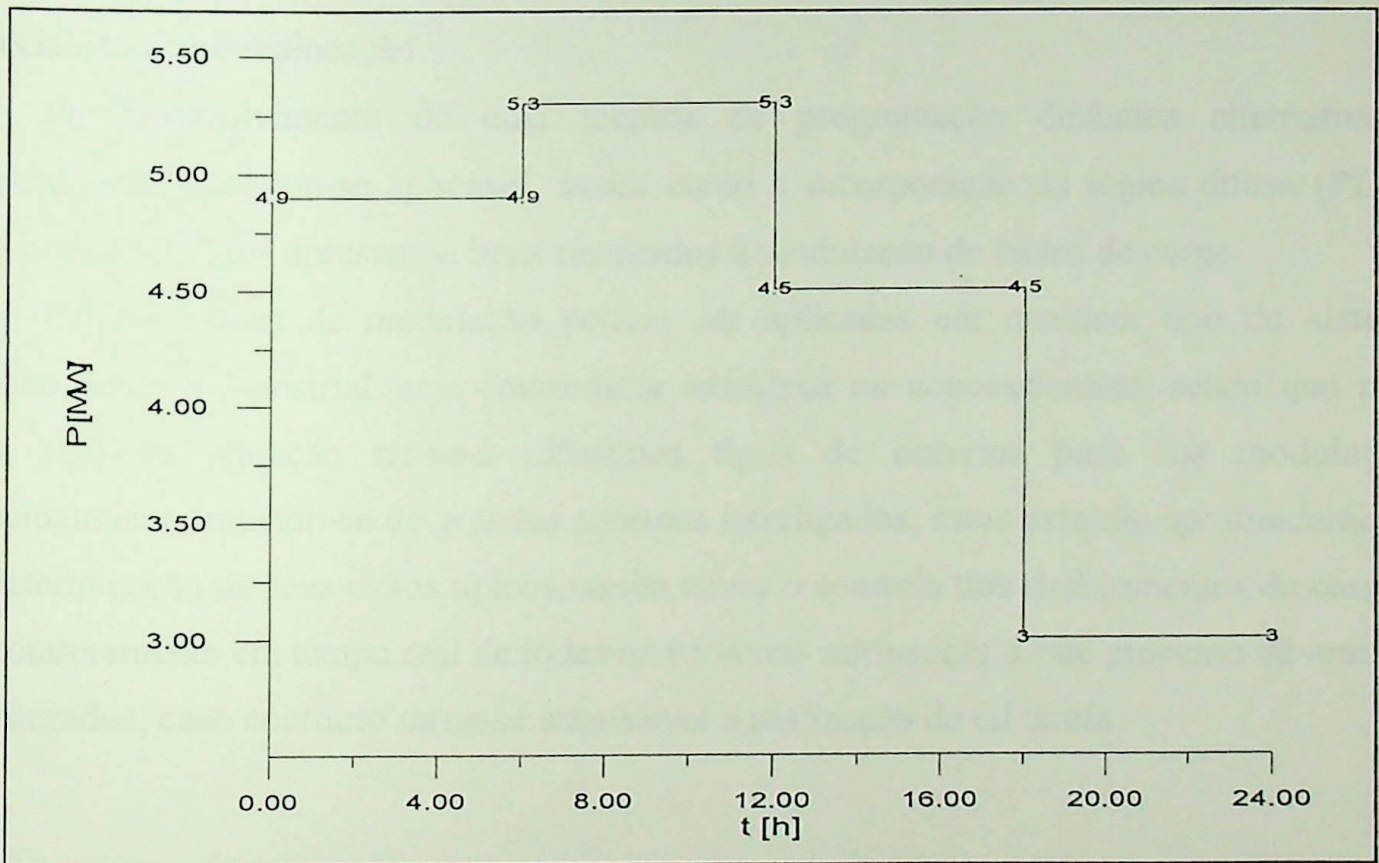


figura 5.13 - resultado da modulação PDIF conforme carregamento das tabela 5.24 à 5.27



5.7 Comparação das metodologias PDI e PDIF

Ambas metodologias, PDI e PDIF, mostram-se eficazes para a solução de problemas relacionados à modulação de ciclos de carga, porém cada uma possui melhor aplicabilidade, dependendo do tipo de modulação pretendida.

Percebe-se claramente que a curva de carga modulada através da PDI tornou-se bem achatada. Com o uso da PDIF pode-se, manter o mesmo sucesso, face a opinião do especialista sobre realocação.

O desenvolvimento de uma técnica de programação dinâmica alternativa à convencional mostrou-se aplicável, assim como a incorporação da lógica difusa (PDIF) com bons resultados apresentou bons resultados à modulação de ciclos de carga.

Estas técnicas de modulação podem ser aplicadas em qualquer tipo de sistema elétrico como o industrial, seja consumidor industrial ou concessionário, sendo que para cada tipo de situação ter-se-á diferentes tipos de critérios para sua modulação. Principalmente tratando-se de grandes sistemas interligados, estes exigirão aprofundamento na determinação de seus ciclos típicos, assim como o controle dos desligamentos de cargas. O monitoramento em tempo real de todas as variáveis intrínsecas a este processo devem ser monitorados, caso contrário torna-se impossível a realização de tal tarefa.

5.1 Conclusões

Esta dissertação tem como objetivo principal a análise da influência da mobilidade de capital de risco, bem como a possibilidade de aumento da produtividade das empresas brasileiras de tecnologia de ponta.

Os dados são provenientes de pesquisas realizadas com investidores de venture capitalists de sucesso.

Os resultados são os seguintes:

CAPÍTULO 6

CONCLUSÕES

6.1 Conclusões

Esta dissertação procurou abordar o desenvolvimento de técnicas dedicadas à modulação de curvas de carga, com enfoque aos sistemas consumidores, embora também seja possível atender ao sistema concessionário.

Os erros que poderiam advir na determinação da curva de carga, devido a mudança das características da mesma, são atenuados através do desenvolvimento do sistema dinâmico de identificação da curva de carga.

Mostrou-se que o desenvolvimento de uma técnica de programação dinâmica, denominada por programação dinâmica por intervalos, para atender às características da carga, é viável, e pode ser verificado através de um exemplo.

Mais adiante mostrou-se a incorporação de conjuntos difusos a esta última, afim de auxiliar no processo de decisão da realocação de cargas. Sua aplicabilidade na modulação de curvas de carga também foi ilustrada por um exemplo prático.

Diversos aspectos podem ser incorporados na função objetivo, tais como, perdas, tarifação, custos de interrupção, embora se tenha incorporado apenas as perdas.

Assim sendo, estas duas implementações apresentam-se como ferramentas úteis para alcançar resultados relacionados à racionalização de energia e redução da fatura e, com certeza, podem fazer parte do contexto das políticas de gerenciamento de carga do lado da demanda.

Ressalta-se também que o conhecimento das características da carga é quesito fundamental para que se obtenha sucesso nesta tarefa, assim como a cooperação concessionária e consumidor.

Conclui-se que os programas desenvolvidos possuem grande flexibilidade, para que o usuário escolha a melhor forma de modular a curva de carga de sua instalação. O processamento computacional é relativamente rápido e preciso, permitindo assim segurança nas decisões.

6.2 Sugestão para trabalhos futuros

Assim como todos trabalhos científicos, este abriu uma gama de opções a serem investigadas a título complementar. Sumariza-se alguns pontos a serem estudados e que provavelmente apontarão para outros :

1. implementação computacional para a modulação de curva de carga para grandes sistemas elétricos de potência;
 2. desenvolvimento de processo de decisão para modulação de curva de carga utilizando, somente conjuntos difusos;
 3. desenvolvimento de estudo relacionado ao uso de instalações cogeneradoras de grandes consumidores industriais, afim de verificar qual o instante ótimo para funcionamento desta em relação aos sistemas tarifários horo-sazonais e a tarifação em tempo real;
 4. aplicação de redes neurais e algoritmos genéticos à modulação de curvas de carga, com posterior comparação com a metodologia aplicada neste trabalho, tanto para sistemas industriais como para sistemas elétricos de potência.
-

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] - Fagundes, E.L.A.; " Políticas de Demand Side Management (DSM) no Setor Elétrico : Barreiras e Perspectivas para o Brasil "; **II Encontro de Economistas de Língua Portuguesa**; Vol.9 - Economia e Energia; Rio de Janeiro - RJ; Setembro de 1995.
- [2] - HIRST, E.; " How Competition Might Affect Electric-Utility DSM Programs "; **Energy**; Vol.19, Nº 12; pp 1193-1203; Elsevier Science Ltd.; Great Britain; 1994
- [3] - WIRL, F.; " On the Unprofitability of Utility Demand-Side Management Conservation Programmes "; **Energy Economics**; Vol.16; Nº 1; pp.46-53; Butterworth-Heinemann Ltd.; England; 1994.
- [4] - BAJAY, S.V. ; et al ; " Planejamento da Expansão da Geração na Eletropaulo "; Capítulo 7 - Modulação de Carga; Relatório Final NIPE/PRDU/UNICAMP; Campinas - SP; Julho de 1994.
- [5] L.A.H. Nogueira; " Em Busca do Uso Racional de Energia Elétrica no Brasil "; **in Mesa Redonda sobre desperdícios no Setor Elétrico : Causas e Soluções**; Instituto de Engenharia; São Paulo - SP; Agosto de 1995;
- [6] RECHE, A.L.R.S.; JANUZZI, G.M.; " Curva de Suprimento de Potencial de Conservação de Energia Elétrica em Sistemas de Iluminação do Setor Terciário ".
- [7] UNICAMP/CPFL; " Pré-aquecedor Solar " ;
- [8] CEMIG/PROCEL; " Gerenciamento Pelo Lado da Demanda - GLD na Região do Vale do Jequitinhonha" **in Seminário Internacional de Estratégias de Conservação de Energia Elétrica**; 26 à 28 de Outubro de 1994; Rio de Janeiro, RJ, Brasil;
- [9] CCK Automação; " Manual Descritivo de Gerenciamento de Energia Elétrica" São Paulo - SP; 1995;
- [10] ELO Sistemas Eletrônicos; " Catálogo Geral de Medidores e Registradores de Energia Elétrica e Demanda" Porto Alegre - RS; 1995;
- [11] Vieira, C.A.H. " Estabelecimento de Curva Diária de Carga de Consumidores Residenciais "; **Dissertação de Mestrado - Universidade de São Paulo - USP**, 1993;
-

- [12] Santos A.H.M et al" Tarifação horosazonal, carregamento e rendimento cíclico de transformadores : Uma solução e compromisso " - **9º CBA - Congresso Brasileiro de Automática** - 1992 - pp.975-986;
- [13] - SPIEGEL,M.R.; " Estatística "; 2ª Edição; McGraw-Hill; São Paulo - SP; 1984.
- [14] L.L. Chao; " Statistics - Methods and analyses "; 2nd Edition; McGrawHill, Tokyo, Japan; 1974;
- [15] D.Dubois; H. Prade; " Fuzzy sets and systems - Theory and Applications "; Vol. 144, Academic Press, Inc., Georgia Institute of Technology, 1980;
- [16] R.F. Calusdian; " Fuzzy Reality "; **IEEE Potentials**; Vol. 14 No.2 April/MAY 1995; pp 20-22;
- [17] Haddad J., " Uma contribuição à análise de conservação de energia elétrica utilizando a teoria dos conjuntos Fuzzy " **Tese de Doutorado - Universidade Estadual de Campinas** - UNICAMP - Abril/1993;
- [18] Bellman, R. E. ; Zadeh, L. A. ; " Decision-Making in a Fuzzy Environment " **Management Science**, Vol.17, No. 4, pp. 141-164, 1970.
- [19] Zimmerma, H.J. "Fuzzy Programming and Linear Programming with Several Objective Functions " ; **Fuzzy Sets and Systems**, Vol. 1, No. 1, pp. 44-56, 1978.
- [20] Wenstop, F. ; " Deductive Verbal Models of Organizations "; **International Journal of Man-Machine Studies**, Vol. 8, pp. 293-311, 1976.
- [21] Kaufmann, A. ; " Introduction à la théorie des sous-ensembles flous. Thome 3 : Applications à la classification et à la reconnaissance des formes ", **aux automates et aux systèmes**, aux choix des critères. Masson, Paris, 1975
- [22] R.E. Larson ; J.L.Casti " Principles of Dynamic Programming Part I Basic Analytic and Computational Methods " Marcel Dekker Inc. NY(1978);
- [23] Esogbue A.O., Bellman R.E., " Fuzzy Dynamic Programming and Its Extensions ", **TIMS/ Studies in the Management Sciences**, 20, 1984, pp.147-167
-

- [24] Partanen J., " A Modified Dynamic Programming Algorithm for Sizing, Locating and Timing of Feeder Reinforcements " , IEEE Transactions on Power Delivery, Vol.5, No.1, January 1990, pp.277-283;
- [25] Adams R., Laughton M. , " A Dynamic Programming Network Flow Procedure for Distribution System " Proceedings Power Industry Computer Applications Conference, pp 348-354,1973;
- [26] A. Kaufmann; " Graphs, Dynamic Programming and Finite Games "; Vol. 36, Academic Press, Inc., University of Southern California, 1967;
- [27] F.S. Hillier; G.J. Lieberman ; " Operations Research "; 2nd Edition; Holden Day Inc.; 1974; San Francisco - USA;
- [28] Santos, A.H.M; Devienne, R.; Haddad, J.; " Controladores de Demanda com Programação Otimizada" - FL/STC/17 - XIII SNPTEE - Balneário Camboriu - SC - Brasil, 01 - 05 de Outubro/1995.
- [29] J. Kacprzyk; R.A. Francelin; F.A.C. Gomide; " Multistage Fuzzy Control : Problem Classes and Their Solution Via Dynamic Programming, Branch-And-Bound, Neural Networks And Genetic Algorithms "; VI IFSA WORLD CONGRESS, São Paulo, Brazil, 1995, Volume II, pp 213-216;

ANEXOS

1 e 2



ANEXOS

1 e 2

```
program curva_tipica e modulação;
```

```
{*****}
Este programa foi baseado em critérios de programação
dinamica e conjuntos fuzzy
{*****}
```

```
uses crt,graph,BgiDriv,BgiFont;
```

```
type
  rtd      = record
    a :real;
    fp :real;
  end;
  rtd2     = array[1..96] of rtd;
  rtd_file = file of rtd2;
  tip_nome = string[20];
  tip_c    = record
    a :real;
    r :real;
    fc :real;
  end;
  limite  = record
    inf :real;
    sup :real;
  end;
  tip_lim = record
    a :limite;
    r :limite;
    fc :limite;
  end;
  tip_cur = array [1..96] of real;
  tip_cur_ok = array[1..30] of integer;
  tip_d   = record
    a :real;
    r :real;
    dem :real;
  end;
  tip_med_dia = array [1..100] of tip_d;
  tip_carg    = record
    num :integer;
    descr :string[15];
    valor :real;
    pert :real;
  end;
  v_carg      = array[1..400] of tip_carg;
  vet_carg    = record
    num_carg :integer;
    carg     :v_carg;
  end;
  tip_interv = record
    li :real;
    ls :real;
```

```
end;
vet_carg_desp = record
  num_carg :integer;
  carg     :v_carg;
  i1      :tip_interv;
  i2      :tip_interv;
end;
vet_c_d_file = file of vet_carg_desp;
combi       = ^comb;
comb        = record
  v :integer;
  next :combi;
end;
combina     = ^combin;
combin      = record
  a :combi;
  next :combina;
end;
triang      = ^trian;
trian       = record
  b :combina;
  next :triang;
end;
tip_interv_r = record
  li :real;
  ls :real;
end;
tip_interv_i = record
  li :integer;
  ls :integer;
end;
carg_tab    = array[1..400] of integer;
pos_carg_tab = array[1..400] of integer;
tip_tab     = record
  u :real;
  xi_l_ :real;
  xi_l  :real;
  xi_   :real;
  xi    :real;
  v     :real;
  interv :tip_interv_r;
  cti   :real;
  cti_l :real;
  num_carg :integer;
  pos_carg :pos_carg_tab;
 argas :carg_tab;
  pert  :real;
end;
tab_file    = file of tip_tab;
var
  arq_ld :vet_c_d_file;
  arq_l  :file of vet_carg;
  c1     :vet_carg;
  c2     :vet_carg_desp;
  kva_base :real;
```

Anexos 1 e 2

```

drv_virtual :char;
text_tab :text;

{*****}
*****}

procedure menu_principal(var ch :char);

const MatMax = 50;
  DGPattern : array [1..5] of word =
($FFFF,$FFFC,$8888,$FF00,$F020);
  DGMaxPnt : word = 400;
  AUTOMATIC = TRUE;
  MANUAL = FALSE;
  _COLOR = TRUE;
  _MONO = FALSE;

var
  EE : integer;
  GM, GD, OP,iI,jJ: Integer;
  Kk,PA, EA, cCH, DH, RA, BHM,EH,REQ, REQ2,
  PEA, RANL, DC, RANEL : Real;
  RI : Array [1..20] of Real;
  R : Array [1..20,1..20] of Real;
  Rm : real;
  nao:string;
(*-----*)

procedure janela(x1,y1,x2,y2 :integer);
begin
  clrscr;
  window(1,1,80,25);
  textbackground(7);
  textcolor(0);
  clrscr;
  gotoxy(25,2);
  write("DSM DINAMICA / FUZZY");
  window(x1+2,y1+1,x2+2,y2+1);
  textbackground(0);
  clrscr;
  window(x1,y1,x2,y2);
  textbackground(1);
  textcolor(15);
  clrscr;
end;

procedure Efei(X,Y,M : integer);

var A : real;
  I,J : integer;
  Ri,Re : ArcCoordsType;
  Pattern : FillPatternType;

begin
  GetFillPattern(Pattern);
  SetFillPattern(Pattern,White);
  { DESENHA A "ENGRENAGEM" }
  SetColor(White);
  Circle(X,Y,Round(3.2*M));
  A:=360/32;
  Arc(X,Y,0,Round(A),Round(4.1*M));
  GetArcCoords(Ri);
  repeat
    Arc(X,Y,Round(A),Round(A+360/16),Round(5*M));
    GetArcCoords(Re);
    Line(Ri.Xend,Ri.Yend,Re.Xstart,Re.Ystart);
    A:=A+360/16;
    Arc(X,Y,Round(A),Round(A+360/16),Round(4.1*M));
    GetArcCoords(Ri);
    Line(Re.Xend,Re.Yend,Ri.Xstart,Ri.Ystart);
    A:=A+360/16;
  until A>360/16*15;
  { DESENHA O "E" }
  I:=Round(2.25*M);
  J:=Round(0.9*M);
  MoveTo(X+I,Y+I);
  LineTo(X-I,Y+I);
  LineTo(X-I,Y-I);
  LineTo(X+I,Y-I);
  LineTo(X+I,Y-I+J);
  LineTo(X-I+J,Y-I+J);
  LineTo(X-I+J,Y-I+2*J);
  LineTo(X+I-J,Y-I+2*J);
  LineTo(X+I-J,Y-I+3*J);
  LineTo(X-I+J,Y-I+3*J);
  LineTo(X-I+J,Y-I+4*J);
  LineTo(X+I,Y-I+4*J);
  LineTo(X+I,Y-I+5*J);
  FloodFill(X,Round(Y+3.6*M),White);
  FloodFill(X,Y,White);
  SetFillPattern(Pattern,LightBlue);
  FloodFill(X-I-1,Y,White);
  FloodFill(X+I+1,Y,White);
  FloodFill(X,Y-I-1,White);
  FloodFill(X,Y+I+1,White);
  { DESENHA O RAIO }
  SetFillPattern(Pattern,Red);
  SetColor(Red);
  MoveTo(X+Round(1.516438344*M),Y-
Round(4.55416455*M));
  LineTo(X-I,Y+I);
  LineTo(X-Round(0.193333333*M),Y-I+3*J+1);
  LineTo(X-I+Round(1.5*M),Y-I+3*J+1);
  LineTo(X+Round(2.339927257*M),Y-
Round(4.191030951*M));
  LineTo(X+Round(1.516438344*M),Y-
Round(4.55416455*M));
  MoveTo(X+Round(0.193333333*M),Y-I+2*J-1);
  LineTo(X+I,Y-I);
  LineTo(X-
Round(1.560786206*M),Y+Round(3.918981221*M));
  LineTo(X-
Round(1.389684611*M),Y+Round(3.943055080*M));
  LineTo(X-
Round(1.920951276*M),Y+Round(4.398857374*M));

```

Anexos 1 e 2

```

LineTo(X-
Round(1.787292050*M),Y+Round(3.711736416*M));
LineTo(X-
Round(1.681797166*M),Y+Round(3.848579888*M));
LineTo(X+I-Round(1.6*M),Y-I+2*J-1);
LineTo(X+Round(0.193333333*M),Y-I+2*J-1);
FloodFill(X+Round(1.516438344*M),Y-
Round(4*M),Red);
FloodFill(X+Round(0.5*M),Y-I+2*J-2,Red);
end;

```

```

procedure TELA1;

```

```

var
  Gd,Gm,x,y,x1,y1 :integer;
  Ch : Char;
begin
  ClrScr;
  Gd:=detect;
  InitGraph(gd,gm,"");
  if GraphResult <> grok then Halt(1);
  SetColor(15);           {Moldura da Tela}
  rectangle(0,0,GetMaxX,GetMaxY);
  Setfillstyle(SolidFill,LightBlue);
  bar(0,0,GetMaxX,GetMaxY);
  rectangle(0,0,GetMaxX,GetMaxY);
  rectangle(4,4,GetMaxX-4,GetMaxY-4);
  SetColor(15);
  SetTextStyle(TriplexFont,HorizDir,9);

  EFEI(100,100,15);
  SetColor(Black);

  SetColor(0);
  SetTextStyle(1,HorizDir,3);
  OutTextXy(60 ,180,'          DSM          ');
  OutTextXy(60 ,218,'          DINAMICA / FUZZY
');
  OutTextXy(60 ,250,'          ');
  SetTextStyle(2,HorizDir,5);
  SETCOLOR(WHITE);
  outtextxy(300,380,'ORIENTADORES :
');
  OutTextXy(300,400,'AFONSO HENRIQUES M.
SANTOS E JAMIL HADDAD          ');
  OutTextXy(300,425,'DESENVOLVIMENTO :
');
  OutTextXY(300,450,'ROBERTO DEVIENNE FILHO
');
  Readkey;
  CloseGraph;
end;

```

```

procedure sub_menu(var ch:char);
begin
  ClrScr;

```

```

GD := 3;
GM := 1;
InitGraph(GD,GM,"");
RECTANGLE(0,0,600,300);
setfillstyle(solidfill,1);
bar (0,0,600,300);
settextstyle(3,0,1);
outtextxy (150,20,' MENU DE OPCOES:');
settextstyle(1,0,1);
outtextxy (0,45,");
outtextxy (0,65,");
outtextxy (0,85,");
outtextxy (0,105,");
outtextxy (0,125,");
outtextxy (0,145,");
outtextxy (0,165,");
outtextxy (0,185,");
outtextxy (0,205,");
outtextxy (0,225,");
settextstyle(3,0,1);
outtextxy (150,260,");
gotoxy (50,20);
ch:=upcase(readkey);
CloseGraph;
end;

```

```

{*****
*****}

```

```

begin
  EE:= RegisterBGIDriver(@EGAVGADriverProc);
  EE:= RegisterBGIDriver(@HercDriverProc);
  EE:= RegisterBGIDriver(@ATTDriverProc);
  EE:= RegisterBGIDriver(@PC3270DriverProc);

```

```

  EE:= RegisterBGIfont(@GothicFontProc);
  EE:= RegisterBGIfont(@SansSerifFontProc);
  EE:= RegisterBGIfont(@SmallFontProc);
  EE:= RegisterBGIfont(@TriplexFontProc);
  TextMode(CO80);
  Tela1;
  { repeat
    sub_menu(ch);
    until(ch=#27);}
end;

```

```

{*****
*****}

```

```

function FileExists(FileName: String): Boolean;
var
  F: file;

```

Anexos 1 e 2

```

begin
  {$I-}
  Assign(F, FileName);
  Reset(F);
  Close(F);
  {$I+}
  FileExists:=(IOResult=0);
end;

function pow(base,expoente:real):real;
var
  res :real;
begin
  if base<>0
  then res:=Exp(Ln(Abs(Base))*Expoente)
  else res:=0;
  if (base<0) and Odd(Round(Expoente)) then Res:=-Res;
  pow:=res;
end;

{*****}
{*****}

procedure plt_cur(cur :integer;
  curval :tip_cur;
  curva2 :tip_cur;
  x,y :integer;
  cor1,cor2 :integer);
var
  i,j :integer;
  s :string[5];
  ganho :real;
begin
  ganho:=0.02;
  setcolor(white);
  line(x,y,x+104,y);
  line(x,y-94,x+104,y-94);
  line(x,y,x,y-94);
  line(x+104,y,x+104,y-94);
  setcolor(lightred);
  line(x+3,y-3,x+103,y-3);
  line(x+3,y-3,x+3,y-93);
  setcolor(cor1);
  moveto(x+4,y-3-trunc(curval[1]/ganho));
  for i:=1 to 96 do
    lineto(x+i+3,y-3-trunc(curval[i]/ganho));
  setcolor(cor2);
  moveto(x+3,y-3-trunc(curva2[1]/ganho));
  for i:=1 to 96 do
    lineto(x+i+3,y-3-trunc(curva2[i]/ganho));
  str(cur,s);
  setcolor(white);
  outtextxy(x+85,y-90,s);
end;

```

```

procedure novo_hst(nome :tip_nome);
var
  st :text;
  ca,cr,dm,fp,fc :real;
  i,k,err :integer;
begin
  clrscr;
  writeln('Entrada de dados (historico Energetico
Anual:');
  writeln;
  gotoxy(1,10);
  writeln(' Consumo Ativo :');
  writeln(' Consumo Reativo :');
  writeln(' Demanda Media :');
  writeln(' Fator de Potencia :');
  writeln(' Fator de Carga :');
  assign(st,nome);
  rewrite(st);
  for i:=1 to 12 do
    begin
      gotoxy(10,7);
      writeln('Mes: ',i);
      gotoxy(24,10);
      readln(ca);
      gotoxy(24,11);
      readln(cr);
      gotoxy(24,12);
      readln(dm);
      gotoxy(24,13);
      readln(fp);
      gotoxy(24,14);
      readln(fc);
      writeln(st,ca:10:1,cr:10:1,dm:10:1,fp:10:3,fc:10:3);
      gotoxy(24,10);
      write(' ');
      gotoxy(24,11);
      write(' ');
      gotoxy(24,12);
      write(' ');
      gotoxy(24,13);
      write(' ');
      gotoxy(24,14);
      write(' ');
    end;
  close(st);
end;

procedure novo_rtd(nome :tip_nome);
begin
end;

procedure novo_arq_rtd_hst(var nome :tip_nome);
var
  i :integer;

```

Anexos 1 e 2

```

ch :char;
begin
repeat
  clrscr;
  writeln('  Analise e Modulacao de Curva de Carga');
  writeln;
  writeln('<F1> - Novo Arquivo do Historico
Energetico');
  writeln('<F2> - Novo Arquivo de Curvas De Cargas
Diarias');
  writeln('<ESC> - Sair');
repeat
  repeat
    ch:=readkey
  until(ch in [#0,#27]);
  if (ch<>#27) then ch:=readkey;
until(ch in [';', '<', #27]);
if (ch<>#27)
then
  begin
    writeln;
    write('Entre com o nome do arquivo [.dat]: ');
    readln(nome);
    i:=pos('.', nome);
    if (i=0) then nome:=nome+'.dat'
      else
        begin
          delete(nome,i,4);
          nome:=nome+'.dat';
        end;
    for i:=1 to 20 do
      nome[i]:=upcase(nome[i]);
    gotoxy(37,3);
    write(nome, ' ');
    case ch of
      ';': novo_hst(nome);
      '<': novo_rtd(nome);
    end;
  end;
until(ch=#27);
end;

procedure list_hist_energ(nome:tip_nome);
var
  st :text;
  ca,cr,dm,fp,fc :real;
  ch :char;
begin
  clrscr;
  textcolor(white);
  writeln(' Historico Energetico Anual:');
  writeln;
  writeln;
  writeln(' CA CR DM FP FC');
  writeln;
  nome:=nome+'.hst';
  assign(st,nome);
  reset(st);

```

```

while (not(eof(st))) do
  begin
    readln(st,ca,cr,dm,fp,fc);
    writeln(ca/kva_base:10:6,cr/kva_base:10:6,dm:10:1,fp:1
0:3,fc:10:3);
    end;
  close(st);
end;

procedure calc_ano( nome :tip_nome;
var lim :tip_lim);
var
  cm :tip_c;
  dp :tip_c;
  st :text;
  i :integer;
  ca,cr,dm,fp,fc :real;
begin
  nome:=nome+'.hst';
  assign(st,nome);
  reset(st);
  cm.a:=0;
  cm.r:=0;
  cm.fc:=0;
  for i:=1 to 12 do
    begin
      readln(st,ca,cr,dm,fp,fc);
      cm.a:=cm.a+ca;
      cm.r:=cm.r+cr;
      cm.fc:=cm.fc+fc;
    end;
  cm.a:=cm.a/12;
  cm.r:=cm.r/12;
  cm.fc:=cm.fc/12;
  reset(st);
  dp.a:=0;
  dp.r:=0;
  dp.fc:=0;
  for i:=1 to 12 do
    begin
      readln(st,ca,cr,dm,fp,fc);
      dp.a:=dp.a+sqr(ca-cm.a);
      dp.r:=dp.r+sqr(cr-cm.r);
      dp.fc:=dp.fc+sqr(fc-cm.fc);
    end;
  dp.a:=sqrt(dp.a/11);
  dp.r:=sqrt(dp.r/11);
  dp.fc:=sqrt(dp.fc/11);
  lim.a.inf:=(cm.a-2*dp.a)/kva_base;
  lim.a.sup:=(cm.a+2*dp.a)/kva_base;
  lim.r.inf:=(cm.r-2*dp.r)/kva_base;
  lim.r.sup:=(cm.r+2*dp.r)/kva_base;
  lim.fc.inf:=cm.fc-2*dp.fc;
  lim.fc.sup:=cm.fc+2*dp.fc;
  close(st);
end;

```

Anexos 1 e 2

```

procedure calc_dia( nome :tip_nome;
                  var energ :tip_med_dia;
                  var dias :integer);
var
  st :rtd_file;
  curva :rtd2;
  i,x,y :integer;
  s :real;
  ea15 :real;
  er15 :real;
  fp15 :real;
  fc15 :real;
  dem15 :real;
  ca_dia :real;
  er_dia :real;
  dem_max :real;
begin
  x:=wherex;
  y:=wherey;
  textcolor(143);
  write('Aguarde');
  assign(st,nome+'.cr');
  dias:=0;
  reset(st);
  while (not(cof(st))) do
    begin
      inc(dias);
      ca_dia:=0;
      er_dia:=0;
      dem_max:=0;
      read(st,curva);
      for i:=1 to 96 do
        begin
          ca_dia:=ca_dia+curva[i].a;
          s:=curva[i].a/(curva[i].fp/100);
          er15:=sqrt(s*s-curva[i].a*curva[i].a);
          er_dia:=er_dia+er15;
          dem15:=curva[i].a;
          if (dem15>dem_max) then
            dem_max:=dem15;
          end;
          energ[dias].a:=ca_dia/96;
          energ[dias].r:=er_dia/96;
          energ[dias].dem:=dem_max/1000;
        end;
      gotoxy(x,y);
      textcolor(white);
      write(' ');
    end;

  procedure imp_limites(lim:tip_lim);
  begin
    textcolor(white);
    writeln;
    writeln;
    writeln('      Limite Inferior      Limite Superior');
    writeln;

```

```

    writeln('Ativa ',lim.a.inf:13:6,lim.a.sup:21:6);
    writeln('Reativa:',lim.r.inf:13:6,lim.r.sup:21:6);
    writeln('F.C. ',lim.fc.inf:13:3,lim.fc.sup:21:3);
    readkey;
  end;

```

```

  procedure menu1(var ch :char;
                 nome :tip_nome);
  begin
    clrscr;
    textcolor(white);
    gotoxy(1,15);
    write('Arquivo: ');
    if (nome='') then write('<nenhum>')
      else write(nome);
    gotoxy(1,1);
    writeln(' Analise e Modulacao de Curva de Carga');
    writeln;
    writeln('<F1> - Novo Arquivo');
    writeln('<F2> - Abrir Arquivo');
    writeln('<F3> - Determinacao da Curva Tipica');
    writeln('<F4> - Realocacao de Carga');
    writeln('<F5> - Verificar Resultados');
    writeln('<ESC> - Sair');
    writeln;
    write('Opcao: ');
    repeat
      repeat
        ch:=readkey
      until(ch in [#0,#27]);
      if (ch<>#27) then ch:=readkey;
    until(ch in [',','<','=','>','?','#27]);
  end;

```

```

  procedure menu2(var ch :char);
  var
    x,y :integer;
  begin
    clrscr;
    textcolor(white);
    writeln(' Analise e Modulacao de Curva de Carga');
    writeln;
    writeln('<F1> - Curva Tipica');
    writeln('<F2> - Curva Tipica Linearizada');
    writeln('<F3> - Curva de Temperatura');
    writeln('<F4> - Curva de Perda de Vida');
    writeln('<F5> - Curva de Rendimento');
    writeln('<ESC> - Sair');
    writeln;
    write('Opcao: ');
    y:=wherey;
    x:=wherex;
    repeat
      repeat
        ch:=readkey

```

Anexos 1 e 2

```

until(ch in [#0,#27]);
if (ch<>#27) then ch:=readkey;
until(ch in [','<','>','?','#27]);
end;

procedure le_nome(var nome :tip_nome;
                 var fun2 :boolean);
begin
  clrscr;
  write('Entre com o nome do arquivo [.dat]: ');
  readln(nome);
  if (nome<>'') then fun2:=true;
end;

procedure arq_rtd(nome:tip_nome);
var
  st1,st2 :text;
  c       :array [1..96] of record
    c :real;
    f :real;
    fc :real;
  end;

  i,k :integer;
begin
  clrscr;
  writeln('Verificando o arquivo ',nome,'.RTD');
  writeln;
  writeln;
  write('Dias: ');
  assign(st1,nome+'.rtd');
  assign(st2,nome+'.tmp');
  reset(st1);
  rewrite(st2);
  k:=0;
  readln(st1,kva_base);
  repeat
    i:=0;
    repeat
      inc(i);
      readln(st1,c[i].c,c[i].f,c[i].fc);
    until((i=96) or eof(st1));
    if (i=96) then
      begin
        inc(k);
        write(k:3);
        if (k=15) then
          begin
            writeln;
            write(' ');
          end;
        for i:=1 to 96 do
          begin
            writeln(st2,c[i].c,c[i].f,c[i].fc);
          end;
        end;
      until(eof(st1));
      writeln;
      writeln;

```

```

textcolor(143);
write('Aguarde');
reset(st2);
rewrite(st1);
writeln(st1,kva_base:0:2);
while (not(eof(st2))) do
  begin
    readln(st2,c[1].c,c[1].f,c[1].fc);
    writeln(st1,c[1].c:10:1,c[1].f:10:1,c[1].fc:10:3);
  end;
close(st1);
close(st2);
clrscr;
end;

```

```

procedure plota_cur_lin(x,y :integer;
                      ganho :real;
                      curva :tip_cur);

var
  i :integer;
begin
  moveto(x,y-trunc(curva[1]*ganho));
  for i:=2 to 96 do
    begin
      lineto(x+(i-1)*4,y-trunc(curva[i-1]*ganho));
      lineto(x+(i-1)*4,y-trunc(curva[i]*ganho));
    end;
end;

```

```

procedure plota_cur(x,y :integer;
                  ganho :real;
                  curva :tip_cur);

var
  i :integer;
begin
  moveto(x,y-trunc(curva[1]*ganho));
  for i:=2 to 96 do
    lineto(x+(i-1)*4,y-trunc(curva[i]*ganho));
  end;
end;

```

```

procedure eixo(x,y :integer;
              f :integer;
              cor :integer);

begin
  setcolor(cor);
  line(x,y,x+400,y);
  line(x,y,x,y-300);
end;

```

```

procedure axes1(cx,cy :integer;
               cor :integer);

var
  i,aux :integer;
  s :string;
begin
  setcolor(cor);
  line(cx,cy,cx+410,cy);

```

Anexos 1 e 2

```

line(cx,cy,cx,cy-300);

line(cx,cy-300,cx-2,cy-298);
line(cx,cy-300,cx+2,cy-298);
line(cx-2,cy-298,cx+2,cy-298);

line(cx+410,cy,cx+408,cy-2);
line(cx+410,cy,cx+408,cy+2);
line(cx+408,cy-2,cx+408,cy+2);

i:=0;
aux:=0;
repeat
  if ((i/12)<>trunc(i/12))
  then line(cx+i*4,cy,cx+i*4,cy+3)
  else
  begin
    line(cx+i*4,cy,cx+i*4,cy+5);
    str(aux,s);
    outtextxy(cx+i*4-3,cy+9,s);
    aux:=aux+3;
  end;
  i:=i+4;
until(i>96);
i:=0;
aux:=0;
repeat
  if ((i/4)<>trunc(i/4))
  then line(cx,cy-i*6,cx-3,cy-i*6)
  else
  begin
    line(cx,cy-i*6,cx-5,cy-i*6);
    str(aux,s);
    case length(s) of
      1:s:=' '+s;
      2:s:=' '+s;
    end;
    outtextxy(cx-33,cy-i*6-2,s);
    aux:=aux+5;
  end;
  i:=i+1;
until(i>48);
end;

procedure axes2( cx,cy :integer;
  cur :integer;
  var _val_inic :integer;
  var ganho :real);

procedure _qual_curva( cur :integer;
  var _incr :integer;
  var _dist_lin :integer;
  var _limite :integer;
  var _escala :real;
  var bb :integer;
  var _val_inic :integer);

begin
  case cur of
    0:begin
      _incr:=1;
      _dist_lin:=5;
      _limite:=56;
      _escala:=0.1;
      bb:=1;
      _val_inic:=0;

      ganho:=(_val_inic+_incr*_limite/4)/(_incr*_limite/4)*1
      99;
    end;
    1:begin
      _incr:=1;
      _dist_lin:=5;
      _limite:=56;
      _escala:=0.1;
      bb:=1;
      _val_inic:=0;

      ganho:=(_val_inic+_incr*_limite/4)/(_incr*_limite/4)*1
      99;
    end;
    2:begin
      _incr:=10;
      _dist_lin:=6;
      _limite:=48;
      _escala:=1;
      bb:=0;
      _val_inic:=0;
    end;
    3:begin
      _incr:=10;
      _dist_lin:=7;
      _limite:=40;
      _escala:=0.00002;
      bb:=4;
      _val_inic:=0;

      ganho:=(_val_inic+_incr*_limite/4)/(_incr*_limite/4)*1
      40000
    end;
    4:begin
      _incr:=2;
      _dist_lin:=7;
      _limite:=40;
      _escala:=1;
      bb:=0;
      _val_inic:=80;

      ganho:=(_val_inic+_incr*_limite/4)/(_incr*_limite/4)*2.
      80
    end;
  end;
end;
var

```

Anexos 1 e 2

```

i,aux :integer;
s :string;
_incr :integer;
_dist_lin :integer;
_limite :integer;
_escala :real;
bb :integer;
begin

_qual_curva(cur,_incr,_dist_lin,_limite,_escala,bb,_val_inic);
line(cx,cy,cx+410,cy);
line(cx,cy,cx,cy-300);

line(cx,cy-300,cx-2,cy-298);
line(cx,cy-300,cx+2,cy-298); {procedura para cixos e
divisoos destes}
line(cx-2,cy-298,cx+2,cy-298);

line(cx+410,cy,cx+408,cy-2);
line(cx+410,cy,cx+408,cy+2);
line(cx+408,cy-2,cx+408,cy+2);

i:=0;
aux:=0;
repeat
if ((i/12)<>trunc(i/12))
then line(cx+i*4,cy,cx+i*4,cy+3)
else
begin
line(cx+i*4,cy,cx+i*4,cy+5);
str(aux,s);
outtextxy(cx+i*4-3,cy+9,s);
aux:=aux+3;
end;
i:=i+4;
until(i>96);
i:=0;
aux:=_val_inic;
repeat
if ((i/4)<>trunc(i/4))
then line(cx,cy-i*_dist_lin,cx-3,cy-i*_dist_lin)
else
begin
line(cx,cy-i*_dist_lin,cx-5,cy-i*_dist_lin);
str(_escala*aux:7:bb,s);
case length(s) of
1:s:=' '+s;
2:s:=' '+s;
end;
outtextxy(cx-64,cy-i*_dist_lin-2,s);
aux:=aux+_incr;
end;
i:=i+1;
until(i>_limite);
end;

procedure_qual_curva( cur :integer;

```

```

var_incr :integer;
var_dist_lin :integer;
var_limite :integer;
var_escala :real;
var_bb :integer;

begin
case cur of
0:begin
_incr:=1;
_dist_lin:=10;
_limite:=28;
_escala:=0.2;
bb:=1;
end;
1:begin
_incr:=1;
_dist_lin:=10;
_limite:=28;
_escala:=0.2;
bb:=1;
end;
end;
end;

procedure_axes3( cx,cy :integer;
cur :integer);

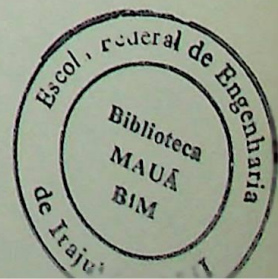
var
i,aux :integer;
s :string;
_incr :integer;
_dist_lin :integer;
_limite :integer;
_escala :real;
bb :integer;
begin
_qual_curva(cur,_incr,_dist_lin,_limite,_escala,bb);
line(cx,cy,cx+410,cy);
line(cx,cy,cx,cy-300);

line(cx,cy-300,cx-2,cy-298);
line(cx,cy-300,cx+2,cy-298); {procedura para cixos e
divisoos destes}
line(cx-2,cy-298,cx+2,cy-298);

line(cx+410,cy,cx+408,cy-2);
line(cx+410,cy,cx+408,cy+2);
line(cx+408,cy-2,cx+408,cy+2);

i:=0;
aux:=0;
repeat
if ((i/12)<>trunc(i/12))
then line(cx+i*4,cy,cx+i*4,cy+3)
else
begin
line(cx+i*4,cy,cx+i*4,cy+5);

```



Anexos 1 e 2

```

str(aux,s);
outtextxy(cx+i*4-3,cy+9,s);
aux:=aux+3;
end;
i:=i+4;
until(i>96);
i:=0;
aux:=0;
repeat
if ((i/4)<>trunc(i/4))
then line(cx,cy-i*_dist_lin,cx-3,cy-i*_dist_lin)
else
begin
line(cx,cy-i*_dist_lin,cx-5,cy-i*_dist_lin);
str(_escala*aux:7:bb,s);
case length(s) of
1:s:=' '+s;
2:s:=' '+s;
end;
outtextxy(cx-64,cy-i*_dist_lin-2,s);
aux:=aux+_incr;
end;
i:=i+1;
until(i>_limite);
end;

```

```

procedure plota_pv_rend_lin( cur :integer;
cx,cy :integer;
ganho :real;
var curva :tip_cur);
procedure mens(cur :integer;
a :real);

```

```

var
s :string;
begin
case cur of
3:begin
outtext(' Perda de Vida');
outtextxy(cx+400,cy-322,'Perda de Vida');
outtextxy(cx+400,cy-312,' Diaria');
str(a:7:5,s);
s:='('+s+')';
outtextxy(cx+412,cy-302,s);
outtextxy(cx+416,cy-2,' t [h]');
outtextxy(cx+7,cy-302,' pv [%]');
end;
4:begin
outtext(' Rendimento');
outtextxy(cx+416,cy-2,' t [h]');
outtextxy(cx+7,cy-302,' rendimento');
end;
end;
end;
end;

```

```

var
i :integer;
cor4 :integer;
aux :integer;
s :string;
_au :real;
somat :real;
s1,s2 :string;
_val_inic :integer;
begin
cleardevice;
cor4:=white;
axes2(cx,cy,cur,_val_inic,ganho);
setcolor(cor4);
_au:=(curva[1]-_val_inic)*ganho;
somat:=curva[1];
moveto(cx,cy-trunc(_au));
lineto(cx+4,cy-trunc(_au));
for i:=2 to 96 do
begin
somat:=somat+curva[i];
_au:=(curva[i]-_val_inic)*ganho;
lineto(cx+(i-1)*4,cy-trunc(_au));
lineto(cx+i*4,cy-trunc(_au));
end;
mens(cur,somat);
end;

```

```

procedure plota_lin(cx,cy :integer;
nome :tip_nome;
ch1 :char;
ch2 :char);

```

```

procedure mens(cur :integer);
var
s :string;
begin
str(kva_base/1000:0:2,s);
s:=s+'[MVA]';
case cur of
0:begin
outtextxy(cx+420,cy-5,' t [h]'); {testando}
outtextxy(cx-35,cy-315,' P [pu] '+s); {testando}
end;
1:begin
outtextxy(cx+420,cy-5,' t [h]'); {testando}
outtextxy(cx-35,cy-315,' P [pu] '+s); {testando}
end;
end;
end;
end;

```

```

var
st :rtd_file;
curva :rtd2;
fp :real;
fc :real;
s :real;

```

Anexos 1 e 2

```

curva1 :tip_cur;
curva2 :tip_cur;
ij :integer;
cor :integer;
dr,md :integer;
_val_inic :integer;
ganho :real;
k :integer;
begin
case ch1 of
':begin
assign(st,nome+'.cr');
k:=1;
end;
'<:begin
assign(st,nome+'.mod');
k:=0;
end;
end;
reset(st);
dr:=detect;
initgraph(dr,md,' '); {c:\tp\bgi}
cleardevice;
case ch2 of
':begin
axes2(cx,cy,0,_val_inic,ganho);
mens(0);
cor:=white;
while (not(cof(st))) do
begin
setcolor(cor);
inc(cor);
if cor=16 then cor:=2;
read(st,curva);
de potencias}
for i:=1 to 96 do
begin
curva1[i]:=curva[i].a;
if (k=1) then
begin
s:=curva[i].a/(curva[i].fp/100);
curva2[i]:=sqrt(s*s-curva[i].a*curva[i].a);
end;
end;
plota_cur(cx,cy,ganho,curva1);
outtext(' ativo');
if (k=1) then
begin
plota_cur(cx,cy,ganho,curva2);
outtext(' reativo');
end;
end;
readkey;
end;
'<:begin
cleardevice;
axes2(cx,cy,1,_val_inic,ganho);
mens(1);

```

```

reset(st);
cor:=2;
while (not(cof(st))) do
begin
setcolor(white{cor});
{inc(cor);
if cor=16 then cor:=2;}
read(st,curva);
for i:=1 to 96 do
begin
curva1[i]:=curva[i].a;
if (k=1) then
begin
s:=curva[i].a/(curva[i].fp/100);
curva2[i]:=sqrt(s*s-curva[i].a*curva[i].a);
end;
end;
plota_cur_lin(cx,cy,ganho,curva1);
outtext(' ativo');
if (k=1) then
begin
plota_cur_lin(cx,cy,ganho,curva2);
outtext(' reativo');
end;
end;
close(st);
readkey;
end;
end;
closegraph;
end;

procedure cria_arq_cr( nome :tip_nome;
var cur_ok :tip_cur_ok;
var num_cur_ok :integer);

var
st1 :text;
st2 :rtd_file;
curva :rtd2;
a :tip_cur;
i :integer;
begin
assign(st1,nome+'.rtd');
assign(st2,nome+'.cr');
reset(st1);
rewrite(st2);
clrscr;
textcolor(white);
writeln;
writeln;
writeln('Criando arquivo ',nome,'.CR');
writeln;
writeln;
textcolor(143);
write('Aguarde');
readln(st1,kva_base);
while (not cof(st1)) do

```

Anexos 1 e 2

```

begin
  for i:=1 to 96 do
    begin
      readln(st1,curva[i].a,curva[i].fp,a[i]);
      curva[i].a:=curva[i].a/kva_base;
    end;
  write(st2,curva);
end;
num_cur_ok:=filesize(st2);
for i:=1 to num_cur_ok do
  cur_ok[i]:=i;
close(st1);
close(st2);
clrscr;
end;

procedure criterio_1( nome      :tip_nome;
                    lim       :tip_lim;
                    energ     :tip_med_dia;
                    var dias   :integer;
                    var cur_ok  :tip_cur_ok;
                    var num_cur_ok :integer);

var
  st1,st2 :rtd_file;
  curva   :rtd2;
  dia,i,k  :integer;
  x,y      :integer;
  s        :real;
  ca15    :real;
  cr15    :real;
  fp15    :real;
  fc15    :real;
  ea_dia  :real;
  er_dia  :real;
begin
  assign(st1,nome+'.cr');
  assign(st2,nome+'.tmp');
  clrscr;
  textcolor(white);
  writeln('Critério 1');
  writeln;
  writeln(' Eng.At. Lim.Inf. Lim.Sup Eng.Rc.
Lim.Inf. Lim.Sup');
  writeln;
  num_cur_ok:=0;
  for i:=1 to dias do
    begin
      write(energ[i].a:10:3,lim.a.inf:10:3,lim.a.sup:10:3);
      write(energ[i].r:10:3,lim.r.inf:10:3,lim.r.sup:10:3);
      if (((energ[i].a>=lim.a.inf) and
(energ[i].a<=lim.a.sup)) and
((energ[i].r>=lim.r.inf) and
(energ[i].r<=lim.r.sup)))
      then
        begin
          cur_ok[num_cur_ok+1]:=i;
          inc(num_cur_ok);
          writeln(' OK');

```

```

end
else writeln;
if (i=15) then
  begin
    readkey;
    clrscr;
    writeln('Critério 1');
    writeln;
    writeln(' Eng.At. Lim.Inf. Lim.Sup Eng.Rc.
Lim.Inf. Lim.Sup');
    writeln;
    end;
  end;
  writeln;
  writeln;
  writeln;
  x:=wherex;
  y:=wherey;
  writeln(' Criando Arquivo ',nome,'.CR');
  reset(st1);
  rewrite(st2);
  dia:=1;
  for i:=1 to num_cur_ok do
    begin
      while (cur_ok[i]<>dia) do
        begin
          inc(dia);
          read(st1,curva);
        end;
        read(st1,curva);
        write(st2,curva);
        inc(dia);
      end;
      dias:=num_cur_ok;
      reset(st2);
      rewrite(st1);
      while (not eof(st2)) do
        begin
          read(st2,curva);
          write(st1,curva);
        end;
      close(st1);
      close(st2);
      gotoxy(x,y);
      textcolor(lightred);
      writeln(' Arquivo ',nome,'.CR criado ');
      readkey;
    end;

procedure criterio_2( nome      :tip_nome;
                    lim       :tip_lim;
                    energ     :tip_med_dia;
                    var dias   :integer;
                    cur_ok_1  :tip_cur_ok;
                    var cur_ok  :tip_cur_ok;
                    var num_cur_ok :integer);

var
  st1,st2 :rtd_file;

```

Anexos 1 e 2

```

curva :rtd2;
dia,i,k,x,y :integer;
fc :real;
ca15 :real;
fp15 :real;
fc15 :real;
begin
assign(st1,nome+'.cr');
assign(st2,nome+'.tup');
clrscr;
textcolor(white);
writeln('Critério 2');
writeln;
writeln(' Fat.C. Lim.Inf. Lim.Sup');
writeln;
num_cur_ok:=0;
for i:=1 to dias do
begin
{ fc:=energ[i].a/(24*energ[i].dem);
fc:=fc/22;}
fc:=(energ[i].a)/(energ[i].dem*1000);

write(fc:10:3,lim.fc.inf:10:3,lim.fc.sup:10:3);
if ((fc>=lim.fc.inf) and (fc<=lim.fc.sup))
then
begin
cur_ok[num_cur_ok+1]:=i;
inc(num_cur_ok);
writeln(' OK');
end
else writeln;
if (i=15) then
begin
readkey;
clrscr;
writeln('Critério 2');
writeln;
writeln(' Fat.C. Lim.Inf. Lim.Sup');
writeln;
end;
end;
writeln;
writeln;
writeln;
x:=wherex;
y:=wherey;
writeln(' Criando Arquivo ',nome,'.CR');
reset(st1);
rewrite(st2);
dia:=1;
for i:=1 to num_cur_ok do
begin
while (cur_ok[i]<>dia) do
begin
inc(dia);
read(st1,curva);
end;
read(st1,curva);

```

```

write(st2,curva);
inc(dia);
end;
dias:=num_cur_ok;
reset(st2);
rewrite(st1);
while (not cof(st2)) do
begin
read(st2,curva);
write(st1,curva);
end;
close(st1);
close(st2);
gotoxy(x,y);
textcolor(lightred);
writeln(' Arquivo ',nome,'.CR criado ');
for i:=1 to dias do
cur_ok[i]:=cur_ok_1[cur_ok[i]];
readkey;
end;

procedure le_curva(var st :rtd_file;
a :integer;
var ca15 :tip_cur;
var cr15 :tip_cur);
var
curva :rtd2;
i,k :integer;
s :real;
fp15 :real;
fc15 :real;
begin
reset(st);
for i:=1 to a do
begin
read(st,curva);
for k:=1 to 96 do
begin
ca15[k]:=curva[k].a;
s:=curva[k].a/(curva[k].fp/100);
cr15[k]:=sqrt(s*s-curva[k].a*curva[k].a);
end;
end;
close(st);
end;

procedure criterio_3( nome :tip_nome;
energ :tip_med_dia;
var dias :integer;
cur_ok_2 :tip_cur_ok;
var cur_ok :tip_cur_ok;
var num_cur_ok :integer);
var
a,b :real;
st1,st2 :rtd_file;
curva :rtd2;
i,j,k :integer;
ca15_1 :tip_cur;

```

Anexos 1 e 2

```

cr15_1 :tip_cur;
ca15_2 :tip_cur;
cr15_2 :tip_cur;
fp15 :real;
fc15 :real;
dist_a :array[1..30] of real;
dist_r :array[1..30] of real;
dist :array[1..30] of real;
menor :integer;
begin
assign(st1,nome+'.cr');
assign(st2,nome+'.tmp');
for i:=1 to dias do
begin
dist_a[i]:=0;
dist_r[i]:=0;
le_curva(st1,i,ca15_1,cr15_1);
for j:=1 to dias do
if i<>j then
begin
gotoxy(60,2);
writeln(i:3,j:3);
le_curva(st1,j,ca15_2,cr15_2);
for k:=1 to 96 do
begin
dist_a[i]:=dist_a[i]+(ca15_1[k]-
ca15_2[k])*(ca15_1[k]-ca15_2[k]);
dist_r[i]:=dist_r[i]+(cr15_1[k]-
cr15_2[k])*(cr15_1[k]-cr15_2[k]);
end;
end;
a:=sqrt(dist_a[i])/energ[cur_ok_2[i]].a;
b:=sqrt(dist_r[i])/energ[cur_ok_2[i]].r;
dist[i]:=a*a+b*b;
end;
menor:=1;
for i:=2 to dias do
if (dist[i]<dist[menor]) then menor:=i;
rewrite(st2);
reset(st1);
i:=0;
repeat
inc(i);
if (i=menor)
then
begin
read(st1,curva);
write(st2,curva);
i:=dias;
end
else
read(st1,curva);
until(i=dias);
reset(st2);
rewrite(st1);
while (not eof(st2)) do
begin
read(st2,curva);

```

```

write(st1,curva);
end;
close(st1);
close(st2);
clrscr;
writeln('Distancias Quadraticas:');
writeln;
writeln;
writeln('      Ativo      Reativo');
writeln;
for i:=1 to dias do
begin
write('Dia ',i:3,':
',dist_a[i]:15:3,dist_r[i]:15:3,dist[i]:15:5);
if (i=menor)
then writeln(' OK')
else writeln;
end;
cur_ok[1]:=cur_ok_2[menor];
num_cur_ok:=1;
readkey
end;

procedure plota_curvas(nome :tip_nome;
cur_ok :tip_cur_ok;
num_cur_ok :integer;
tot_cur :integer);
var
dr,md :integer;
_cur,cur :integer;
st :rtd_file;
curva1 :tip_cur;
curva2 :tip_cur;
x,y :integer;
begin
dr:=detect;
initgraph(dr,md,' ');
assign(st,nome+'.cr');
reset(st);
x:=3;
y:=95;
_cur:=1;
dr:=filesize(st);
close(st);
for cur:=1 to tot_cur do
begin
if (cur=cur_ok[_cur])
then
begin
le_curva(st,_cur,curva1,curva2);
plt_cur(cur,curva1,curva2,x,y,lightgreen,lightgray);
inc(_cur);
end;
x:=x+106;
if x=639 then
begin
y:=y+96;

```

Anexos 1 e 2

```

x:=3;
end;
end;
readkey;
closegraph;
end;
{barata}
procedure le_uma_carga( x,y :integer;
                      cor1 :integer;
                      cor2 :integer;
                      j :integer;
                      var alter :boolean;
                      var carga :tip_carg);
var
s :string;
err :integer;
begin
alter:=false;
repeat
textcolor(cor1);
gotoxy(x,y); write('Carga 'j:3,' - No.: ');
textcolor(cor2);
readln(s);
if (s<>") then val(s,carga.num,err);
until((err=0) or (s="));
if (s<>") then
begin
alter:=true;
textcolor(cor1);
gotoxy(x+25,y); write('Descricao: ');
textcolor(cor2);
readln(carga.descr);
repeat
textcolor(cor1);
gotoxy(x,y+1); write('Valor : ');
textcolor(cor2);
readln(s);
val(s,carga.valor,err);
until(err=0);
repeat
textcolor(cor1);
gotoxy(x,y+2); write('Pertinencia: ');
textcolor(cor2);
s:='1';
val(s,carga.pert,err);
until(err=0);
end;

textcolor(black);
gotoxy(x,y); write('
');
gotoxy(x,y+1); write('
');
gotoxy(x,y+2); write('
');

end;

```

```

procedure entra_cargas_tec(nome :tip_nome);
var
st :file of vet_carg;
i,j :integer;
s :string;
err :integer;
x,y :integer;
cor0,cor1 :integer;
cor2 :integer;
alter :boolean;
begin
cor0:=white;
cor1:=lightgray;
cor2:=yellow;
x:=10;
y:=7;
assign(st,nome+'.l');
rewrite(st);
for i:=1 to 96 do
begin
j:=0;
repeat
inc(j);
clrscr;
textcolor(cor0);
writeln('Entrada para Cargas Variaveis:');
writeln;
writeln;
write(' Intervalo ');
textcolor(cor2);
write(i);
textcolor(cor0);
write(':');
le_uma_carga(10,7,cor1,cor2,j,alter,c1.carg[j]);
until(not alter);
c1.num_carg:=j-1;
write(st,c1);
end;
close(st);
end;

procedure verificar(nome :tip_nome);

procedure mens1( cor0 :integer;
                cor1 :integer;
                i :integer);
begin
textbackground(cor0);
textcolor(cor1);
clrscr;
write(' Cargas Despachaveis');
gotoxy(10,3); write('Intervalo ',i+1,':');
gotoxy(50,22); write('Tecla [F1] para Alterar');
gotoxy(50,23); write('Tecla [F2] para Incluir');
gotoxy(50,24); write('Tecla [F3] para Retirar');
gotoxy(50,25); write('Tecla [F4] Define o Intervalo');
gotoxy(1,5);

```

Anexos 1 e 2

```

write(' Quant. Carga      Valor  Pert.
Descricao');
end;

```

```

procedure str_carga( carga :tip_carg;
                    var s   :string);

```

```

var
  i,j :integer;
begin
  str(carga.num,s);
  s:='L'+s;
  i:=length(s);
  for j:=1 to 10-i do
    s:=s+' ';
  end;
end;

```

```

procedure novo_interv( x,y :integer;
                      cor1 :integer;
                      cor2 :integer;
                      cor3 :integer;
                      var i  :integer);

```

```

var
  s :string;
  err :integer;
begin
  repeat
    gotoxy(x,y);
    textcolor(cor1);
    textbackground(cor3);
    write('Digite o Intervalo Desejado: ');
    textcolor(cor2);
    gotoxy(x+29,y);
    readln(s);
    val(s,i,err);
    dec(i);
  until((err=0) and (i>=0) and (i<=96));
  textcolor(black);
  textbackground(black);
  gotoxy(x,y);
  write(' ');
end;

```

```

procedure qual_carga( x,y :integer;
                     cor1 :integer;
                     cor2 :integer;
                     cor3 :integer;
                     mens :string;
                     var num :integer;
                     max_c :integer);

```

```

var
  s :string;
  err :integer;
begin
  repeat
    textcolor(cor1);
    textbackground(cor3);
    gotoxy(x,y);
    writeln(mens);

```

```

gotoxy(x,y+1);
write('Digite qual Carga: ');
textcolor(cor2);
gotoxy(x+20,y+1);
readln(s);
if (s='') then
  begin
    num:=0;
    err:=0;
  end
else
  val(s,num,err);
until((err=0) and (num>=0) and (num<=max_c));
textcolor(black);
textbackground(black);
gotoxy(x,y);
write(' ');
end;

```

```

procedure que_carga( x,y :integer;
                    cor1 :integer;
                    cor2 :integer;
                    cor3 :integer;
                    mens :string;
                    var num :integer;
                    max_c :integer);

```

```

var
  s :string;
  err :integer;
begin
  repeat
    textcolor(cor1);
    textbackground(cor3);
    gotoxy(x,y);
    writeln(mens);
    gotoxy(x,y+1);
    write('E|||||||'»);
    gotoxy(x,y+2);
    write('° Digite qual Carga: ° ');
    gotoxy(x,y+3);
    write('E|||||||'¼');
    textcolor(cor2);
    gotoxy(x+20,y+2);
    readln(s);
    if (s='') then
      begin
        num:=0;
        err:=0;
      end
    else
      val(s,num,err);
  until((err=0) and (num>=0) and (num<=max_c));
  textcolor(black);
  textbackground(black);
  gotoxy(x,y);
  write(' ');
  gotoxy(x,y+1);
  write(' ');
end;

```

Anexos 1 e 2

```

gotoxy(x,y+2);
write(' ');
end;

var
  st :file of vet_carg;
  i,j :integer;
  alter :boolean;
  s1 :string;
  ch :char;
  cor0,cor1 :integer;
  cor2,cor3 :integer;
  cor4 :integer;
  l1,l2 :integer;
  k,h :integer;
  num_carga :integer;
begin
  cor0:=black;
  cor1:=white;
  cor2:=yellow;
  cor3:=lightblue;
  cor4:=lightcyan;
  assign(st,nome+'.1');
  reset(st);
  i:=0;
  repeat
    seek(st,i);
    read(st,c1);
    alter:=false;
    j:=0;
    k:=0;
    mens1(cor0,cor1,i);
    ch:=#0;
    while ((j<c1.num_carg)) and (ch<>#27)) do
      begin
        if (c1.num_carg>0) then
          begin
            str_carga(c1.carg[j+1],s1);
            gotoxy(1,k+7);
            textcolor(cor1);
            textbackground(cor0);

            write(j+1:5,s1:14,c1.carg[j+1].valor:10:3,c1.carg[j+1].pc
            rt:10:2,c1.carg[j+1].descr:20,' ');
            inc(j);
            inc(k);
          end;
          if ((k=12) or ((j+1)=c1.num_carg+1) or
            (c1.num_carg=0)) then
            begin
              ch:=uppercase(readkey);
              case ch of
                #0:begin
                  ch:=uppercase(readkey);
                  case ch of
                    ':begin

```

```

if (c1.num_carg>0) then
  begin

```

```

    qual_carga(10,23,white,lightcyan,lightblue,'Alteracao:',n
    um_carga,c1.num_carg);
    if (num_carga>0) then
      begin

```

```

        le_uma_carga(1,21,cor1,cor2,num_carga,alter,c1.carg[n
        um_carga]);

```

```

          if alter then

```

```

            begin
              seek(st,i);
              write(st,c1);
            end;

```

```

          end;

```

```

          mens1(cor0,cor1,i);

```

```

          if (k=12) then j:=j-12
            else j:=j-k;

```

```

          k:=0;

```

```

          end;

```

```

        end;

```

```

      '<':begin

```

```

        le_uma_carga(1,21,cor1,cor2,c1.num_carg+1,alter,c1.ca
        rg[c1.num_carg+1]);

```

```

        if alter then

```

```

          begin

```

```

            inc(c1.num_carg);

```

```

            seek(st,i);

```

```

            write(st,c1);

```

```

          end;

```

```

          mens1(cor0,cor1,i);

```

```

          if (k=12) then j:=j-12

```

```

            else j:=j-k;

```

```

          k:=0;

```

```

          end;

```

```

        '=':begin

```

```

          if (c1.num_carg>0) then

```

```

            begin

```

```

              qual_carga(10,23,white,lightcyan,lightblue,'Exclusao:',n
              um_carga,c1.num_carg);

```

```

              if (num_carga<>0) then

```

```

                begin

```

```

                  for h:=num_carga to

```

```

                    c1.num_carg-1 do

```

```

                      c1.carg[h]:=c1.carg[h+1];

```

```

                      dec(c1.num_carg);

```

```

                      seek(st,i);

```

```

                      write(st,c1);

```

```

                end;

```

```

                mens1(cor0,cor1,i);

```

```

                if (k=12) then j:=j-12

```

```

                  else j:=j-k;

```

```

                k:=0;

```

```

                end;

```

```

              end;

```

Anexos 1 e 2

```

        '>':begin
novo_interv(10,23,white,lightcyan,lightbluc,i);
        mensl(cor0,cor1,i);
        seek(st,i);
        read(st,c1);
        j:=0;
        k:=0;
        end;
    end;
end;
else
begin
    mensl(cor0,cor1,i);
    k:=0;
    if (c1.num_carg=0)
        then j:=1;
    end;
end;
end;
inc(i);
if (i=96) then i:=0;
until(ch=#27);
close(st);
end;

procedure le_cargas( nome :tip_nome;
                    var ch :char);

procedure sub_menu(var ch :char;
                  nome :tip_nome);

begin
    clrscr;
    gotoxy(1,15);
    write('Arquivo: ');
    if (fileexists(nome+'.l'))
        then write(nome+'.l')
        else write('<nenhum>');
    gotoxy(1,1);
    writeln('Arquivo de Cargas Relocaveis e Pertinencias');
    writeln;
    writeln('<F1> - Novo');
    writeln('<F2> - Abrir');
    writeln('<F3> - Modulacao');
    writeln('<ESC> - Sair');
    writeln;
    write('Opcao:');
    repeat
        repeat
            ch:=readkey
        until(ch in [#0,#27]);
        if (ch<>#27) then ch:=readkey;
    until(ch in [',','<','=',#27]);
end;

```

```

begin
    repeat
        sub_menu(ch,nome);
        if (ch<>#27) then
            begin
                case ch of
                    '!':entra_cargas_tec(nome);
                    '<':verificar(nome);
                end;
            end;
        until(ch in [#27,'=']);
    end;

function get_interv(l :real;
                   fat1 :real;
                   fat2 :real):real;

var
    valor :real;
begin
    valor:=round(1*fat1)/fat1;
    if (l=valor) then
        valor:=valor-fat2;
    get_interv:=valor;
end;

procedure calc_interv2(var interv1 :tip_interv;
                      var interv2 :tip_interv;
                      fat1 :real;
                      fat2 :real);

begin
    interv2.li:=get_interv(interv1.li,fat1,fat2);
    interv2.ls:=get_interv(interv1.ls,fat1,fat2);
end;

procedure cria_despac(var nome :tip_nome;
                      fat1 :real;
                      fat2 :real);

var
    st1 :file of vet_carg;
    { st1_ :file of vet_carg; }
    st2 :file of vet_carg_desp;
    st3 :rtd_file;
    curva :rtd2;
    i,j,cur :integer;
    teste :boolean;
begin
    clrscr;
    assign(st3,nome+'.cr_');
    reset(st3);
    read(st3,curva);
    close(st3);
    { assign(st1,nome+'.l'); }
    assign(st1,nome+'.lpu'); { .tmp }
    { reset(st1_); }

```

Anxos 1 e 2

```

rewrite(st1);
while (not eof(st1_)) do
  begin
    read(st1_,c1);
    for i:=1 to c1.num_carg do
      c1.carg[i].valor:=c1.carg[i].valor/kva_base;
      write(st1,c1);
    end;}
assign(st2,nome+'.ld');
reset(st1);
rewrite(st2);
read(st1,c1);
c2.num_carg:=c1.num_carg;
c2.carg:=c1.carg;
c2.i1.li:=curva[1].a;
c2.i1.ls:=curva[1].a;
for i:=1 to c1.num_carg do
  c2.i1.ls:=c2.i1.ls+c2.carg[i].valor; {
c2.i1.ls:=c2.i1.ls+c1.carg[i].valor;}
calc_interv2(c2.i1,c2.i2,fat1,fat2);
write(st2,c2);
cur:=2;
while (not eof(st1)) do
  begin
    read(st1,c1);
    for i:=1 to c1.num_carg do
      begin
        teste:=true;
        for j:=1 to c2.num_carg do
          if (c1.carg[i].num=c2.carg[j].num) then
teste:=false;
          if teste then
            begin
              inc(c2.num_carg);
              c2.carg[c2.num_carg]:=c1.carg[i];
              c2.carg[i].valor:=c2.carg[i].valor;
            end;
          end;
        c2.i1.li:=0;
        for i:=1 to cur do
          c2.i1.li:=c2.i1.li+curva[i].a;
        c2.i1.ls:=c2.i1.li;
        for i:=1 to c2.num_carg do
          c2.i1.ls:=c2.i1.ls+c2.carg[i].valor;
        calc_interv2(c2.i1,c2.i2,fat1,fat2);
        write(st2,c2);
        inc(cur);
      end;
    close(st1);
    close(st2);
  { close(st1_);}
end;

procedure lista_desp(var nome :tip_nome);
var
  st :file of vet_carg_desp;
  i :integer;
begin

```

```

assign(st,nome+'.ld');
reset(st);
while (not eof(st)) do
  begin
    clrscr;
    read(st,c2);
    for i:=1 to c2.num_carg do
      begin
        write(c2.carg[i].num:3,c2.carg[i].descr:17,c2.carg[i].valo
r:8:4,c2.carg[i].pert:5:2);

        writeln(c2.i1.li:9:4,c2.i1.ls:9:4,c2.i2.li:9:4,c2.i2.ls:9:4);
        end;
        readkey;
        end;
        close(st);
      end;

procedure ter;
var
  st :file of vet_carg;
  a :vet_carg;
  i,j,k :integer;
  c :array[1..40] of real;
begin
  assign(st,'empresa.l');
  rewrite(st);
  for i:=1 to 40 do
    c[i]:=250+250*random(12);
  for i:=1 to 96 do
    begin
      a.num_carg:=1+random(4);
      for j:=1 to a.num_carg do
        begin
          a.carg[j].num:=j+(j-1)*3+random(j*3);
          a.carg[j].valor:=c[a.carg[j].num];
          a.carg[j].descr:='Motor';
          a.carg[j].pert:=1;
        end;
      write(st,a);
    end;
  close(st);
end;

procedure plota_aquec( cx,cy :integer;
                       ganho :real;
                       var curva1 :tip_cur;
                       var curva2 :tip_cur;
                       var curva3 :tip_cur);
var
  i :integer;
  cor1,cor2,cor3 :integer;
  aux :integer;

```



Anexos 1 e 2

```

s      :string;
s1,s2  :string;
_val_inic  :integer;
ch      :char;
begin
cleardevice;
cor1:=white;
cor2:=white;
cor3:=white;
axes2(cx,cy,2,_val_inic,ganho);
outtextxy(cx+416,cy-2,' t [h]'); {nome dos eixos de
temp x hora}
outtextxy(cx+7,cy-302,' T [oC]');
setcolor(cor1);
moveto(cx,cy-trunc(curva1[1]*ganho));
for i:=2 to 96 do
  lincto(cx+(i-1)*4,cy-trunc(curva1[i]*ganho));
outtext(' Ponto Mais Quente');
setcolor(cor2);
moveto(cx,cy-trunc(curva2[1]*ganho));
for i:=2 to 96 do
  lincto(cx+(i-1)*4,cy-trunc(curva2[i]*ganho));
outtext(' Enrolamento');
setcolor(cor3);
moveto(cx,cy-trunc(curva3[1]*ganho));
for i:=2 to 96 do
  lincto(cx+(i-1)*4,cy-trunc(curva3[i]*ganho));
outtext(' Oleo');
end;

procedure aquec(nome  :tip_nome;
               ch    :char);
var
rp,n,m,dton,dten,tto,tte,pnom    :real;
dtoi,dtei,dtof,dtef,dto,dte     :real;
t,tpq,ta,teta,b,pv              :real;
fc                               :real;
st2                             :rtd_file;
curva                          :rtd2;
curva1,curva2,curva3,curva4,curva5 :tip_cur;
i                               :integer;
dr,md                          :integer;
po                              :real;
a                               :real;
pjn                             :real;
rend                            :real;

begin
rp:=3.2;
n:=0.8;
m:=1.6;
dton:=55;
dten:=25;
tto:=3;
tte:=0.08;

```

```

pnom:=kva_base;
ta:=20;

po:=1000/kva_base;
pjn:=2000/kva_base;

dr:=detect;
initgraph(dr,md,"");

assign(st2,nome+'.cr');
reset(st2);
read(st2,curva);
for i:=1 to 96 do
  begin
    fc:=curva[i].a;

    dtoi:=Dton*(POW((((Sqr(Fc))*(Rp+1))/(Rp+1)),n));
    dtei:=Dten*(POW((Fc),(m)));{para tempt inicial =
0}

    dtof:=Dton*(POW((((Sqr(Fc))*(Rp+1))/(Rp+1)),n));
    dtef:=Dten*(POW((Fc),(m)));

    Dto:=(Dtof*(1-(exp(-0.25/TTo))))+(Dtoi*exp(-
0.25/TTo));
    Dtc:=(Dtef*(1-(exp(-0.25/TTe))))+(Dtei*exp(-
0.25/TTe));

    tpq:=dte+dto+ta;

    b:=6927.15;
    a:=-14.133;
    teta:=273.15+tpq;
    pv:=exp((-2.3)*(a+(b/teta)))*100;
    a:=fc*curva[i].a*kva_base;
    b:=fc*fc*pjn*kva_base;
    rend:=( a / (a+b+po*kva_base) ) * 100;

    curva1[i]:=tpq;
    curva2[i]:=dto;
    curva3[i]:=dte;
    curva4[i]:=pv;
    curva5[i]:=rend;

  end;
case ch of
'=':plota_aquec(80,400,2.42,curva1,curva2,curva3);
'>':plota_pv_rend_lin(3,80,400,140000,curva4);
'?':plota_pv_rend_lin(4,80,400,5.60,curva5);
end;
ch:=readkey;
if (ch=#0) then ch:=readkey;
close(st2);
closegraph;
end;

procedure cria_1_pu(nome  :tip_nome);

```

Anexos 1 e 2

```

var
  aux :file of vet_carg;
  a :vet_carg;
  i :integer;
begin
  assign(arq_1,nome+'.l');
  assign(aux,nome+'.lpu');
  rreset(arq_1);
  rewrite(aux);
  while (not eof(arq_1)) do
    begin
      read(arq_1,a);
      for i:=1 to a.num_carg do
        a.carg[i].valor:=a.carg[i].valor/kva_basc;
      write(aux,a);
    end;
  close(arq_1);
  close(aux);
end;

procedure prim_v(var vfirst :combi;
                 num_term :integer);
var
  vnext,vlast :combi;
  h :integer;
begin
  new(vnext);
  vnext^.next:=nil;
  vlast:=vnext;
  vfirst:=vnext;
  vnext^.v:=1;
  for h:=2 to num_term do
    begin
      new(vnext);
      vnext^.next:=nil;
      vnext^.v:=vlast^.v+1;
      vlast^.next:=vnext;
      vlast:=vnext;
    end;
end;

procedure demais_v(var vfirst :combi;
                  alast :combina;
                  num_term :integer;
                  num_cargas :integer);
var
  vnext,vlast :combi;
  p :combi;
  h,g,k,a :integer;
  teste :boolean;
begin
  new(vnext);
  vnext^.next:=nil;
  vlast:=vnext;
  vfirst:=vnext;
  p:=alast^.a;
  vnext^.v:=p^.v;
  for h:=2 to num_term do

```

```

begin
  new(vnext);
  vnext^.next:=nil;
  vlast^.next:=vnext;
  vlast:=vnext;
  p:=p^.next;
  vnext^.v:=p^.v;
end;
g:=num_cargas;
k:=num_term;
teste:=false;
repeat
  p:=vfirst;
  for h:=2 to k do
    p:=p^.next;
  inc(p^.v);
  if (p^.v>g)
    then begin
      dec(k);
      dec(g);
    end
  else begin
    for h:=k+1 to num_term do
      begin
        a:=p^.v;
        p:=p^.next;
        p^.v:=a+1;
      end;
      teste:=true;
    end;
  until(teste);
end;

procedure get_comb( cargas :v_carg;
                  num_comb :integer;
                  num_term :integer;
                  num_cargas :integer;
                  var cmb :combina);
var
  anext,alast,afirst :combina;
  vfirst :combi;
  i :integer;
begin
  new(anext);
  anext^.next:=nil;
  alast:=anext;
  afirst:=anext;
  prim_v(vfirst,num_term);
  anext^.a:=vfirst;
  for i:=2 to num_comb do
    begin
      new(anext);
      anext^.next:=nil;
      alast^.next:=anext;
      demais_v(vfirst,alast,num_term,num_cargas);
      anext^.a:=vfirst;
      alast:=anext;
    end;
end;

```

Anexos 1 e 2

```

cmb:=afirst;
end;

function fat(x :integer):real;
var
  f :real;
begin
  if (x=0)
  then fat:=1
  else
  begin
    f:=x;
    fat:=f*fat(x-1);
  end;
end;

procedure get_num_comb( n,i :integer;
  var num_comb :integer);
begin
  num_comb:=trunc(fat(n)/(fat(i)*fat(n-i)));
end;

procedure relaciona_cargas(tab :triang;
  cargas :v_carg);
var
  aux1 :triang;
  aux2 :combina;
  aux3 :combi;
begin
  aux1:=tab^.next;
  aux2:=aux1^.b;
  aux3:=aux2^.a;
  repeat
  repeat
    aux3^.v:=cargas[aux3^.v].num;
    aux3:=aux3^.next;
  until(aux3=nil);
  aux2:=aux2^.next;
  aux3:=aux2^.a;
  until(aux2=nil);
  aux1:=aux1^.next;
  aux2:=aux1^.b;
  aux3:=aux2^.a;
  until(aux1=nil);
end;

procedure combinacoes( num_cargas :integer;
  {num_carg}
  var cargas :v_carg;
  var tab :triang);

var
  i,j,k,h,f :integer;
  anext :combina;
  vnext :combi;

```

```

first :triang;
next :triang;
last :triang;
num_ter :integer;
num_comb :integer;
t1,t2 :boolean;
cmb :combina;
begin
  new(first);
  first^.next:=nil;
  next:=first;
  last:=first;
  new(anext);
  anext^.next:=nil;
  new(vnext);
  vnext^.next:=nil;
  vnext^.v:=0;
  anext^.a:=vnext;
  next^.b:=anext;
  for i:=1 to num_cargas do
  begin
    get_num_comb(num_cargas,i,num_comb);
    get_comb(cargas,num_comb,i,num_cargas,cmb);
    new(next);
    next^.next:=nil;
    last^.next:=next;
    last:=next;
    next^.b:=cmb;
  end;
  tab:=first;
  if (tab^.next<>nil) then
    relaciona_cargas(tab,cargas);
end;

procedure arquivo_cb(var st :text;
  tab :triang;
  inter :integer);

var
  aux1 :triang;
  aux2 :combina;
  aux3 :combi;
begin
  writeln(st,'I',inter);
  aux1:=tab;
  aux2:=aux1^.b;
  aux3:=aux2^.a;
  repeat
  repeat
  repeat
    write(st,aux3^.v:4);
    aux3:=aux3^.next;
  until(aux3=nil);
  aux2:=aux2^.next;
  aux3:=aux2^.a;
  writeln(st,0:4);
  until(aux2=nil);
  aux1:=aux1^.next;
  aux2:=aux1^.b;

```

Anexos 1 e 2

```

aux3:=aux2^.a;
until(aux1=nil);
end;

procedure list_comb(tab :triang;
inter :integer);
var
l :integer;
aux1 :triang;
aux2 :combina;
aux3 :combi;
begin
l:=0;
clrscr;
writeln('Intervalo ',inter:3,' ');
writeln;
aux1:=tab;
aux2:=aux1^.b;
aux3:=aux2^.a;
repeat
repeat
write(aux3^.v:4);
aux3:=aux3^.next;
until(aux3=nil);
aux2:=aux2^.next;
aux3:=aux2^.a;
writeln;
inc(l);
if (l=20) then begin
l:=0;
readkey;
clrscr;
writeln('Intervalo ',inter:3,' ');
writeln;
end;
until(aux2=nil);
aux1:=aux1^.next;
aux2:=aux1^.b;
aux3:=aux2^.a;
writeln;
until(aux1=nil);
readkey;
clrscr;
end;

procedure calc_combinacoes(var nome :tip_nome);
var
i,n :integer;
tab :triang;
st1 :file of vet_carg;
st2 :text;
p :pointer;
begin
i:=1;
assign(st1,nome+'.lpu');
assign(st2,nome+'.cb');

```

```

reset(st1);
rewrite(st2);
while (not eof(st1)) do
begin
read(st1,c1);
mark(p);
combinacoes(c1.num_carg,c1.carg,tab);
arquivo_cb(st2,tab,i);
{ list_comb(tab,i); }
release(p);
inc(i);
end;
close(st1);
close(st2);
end;

procedure arq_virtual(nome :tip_nome);
var
arq_l_aux :file of vet_carg;
arq_ld_aux :vet_c_d_file;
arq_cb :text;
arq_cb_aux :text;
ch :char;
begin
assign(arq_ld,nome+'.ld');
assign(arq_ld_aux,drv_virtual+'\'+nome+'.ld');
reset(arq_ld);
rewrite(arq_ld_aux);
while (not eof(arq_ld)) do
begin
read(arq_ld,c2);
write(arq_ld_aux,c2);
end;
close(arq_ld);
close(arq_ld_aux);

assign(arq_l,nome+'.lpu');
assign(arq_l_aux,drv_virtual+'\'+nome+'.lpu');
reset(arq_l);
rewrite(arq_l_aux);
while (not eof(arq_l)) do
begin
read(arq_l,c1);
write(arq_l_aux,c1);
end;
close(arq_l);
close(arq_l_aux);

assign(arq_cb,nome+'.cb');
assign(arq_cb_aux,drv_virtual+'\'+nome+'.cb');
reset(arq_cb);
rewrite(arq_cb_aux);
while (not eof(arq_cb)) do
begin
read(arq_cb,ch);
write(arq_cb_aux,ch);
end;
end;

```

Anexos 1 e 2

```

close(arq_cb);
close(arq_cb_aux);

end;

procedure imp_linha(t      :tip_tab;
                   estagio :integer);
var
  i,j :integer;
  aux :integer;
begin
  with t do
    begin
write(pert:6:4,u:7:4,xi_1_:7:4,xi_1:7:4,xi_:7:4,xi:7:4);
write(v:7:4,interv.li:7:2,interv.ls:7:2,' ',cti:8:4);
aux:=0;
write(' ');
for i:=1 to estagio do
  begin
    if (i>1) then aux:=aux+t.pos_carg[i-1];
    for j:=1 to t.pos_carg[i] do
      begin
        write(cargas[aux+j]);
        if (j<>t.pos_carg[i]) then write('.');
      end;
      if (t.pos_carg[i]=0) then write('0');
      if (i<>estagio) then write('-');
    end;
    writeln;
  end;
end;

procedure imp_linha_text(t      :tip_tab;
                       estagio :integer);
var
  i,j :integer;
  aux :integer;
begin
  with t do
    begin
{
write(text_tab,pert:6:4,u:7:4,xi_1_:7:4,xi_1:7:4,xi_:7:4,x
i:7:4);
write(text_tab,v:7:4,interv.li:7:2,interv.ls:7:2,'
',cti:8:4);

write(pert:6:4,u:7:4,xi_1_:7:4,xi_1:7:4,xi_:7:4,xi:7:4);
write(v:7:4,interv.li:7:2,interv.ls:7:2,' ',cti:9:4);}

write(text_tab,pert:6:5,u:12:8,xi_1_:9:5,xi_1:9:5,xi_:9:5,
xi:9:5);
write(text_tab,v:8:4,interv.li:8:4,interv.ls:8:4,'
',cti:8:4);

write(pert:6:5,u:12:9,xi_1_:9:5,xi_1:9:5,xi_:9:5,xi:9:5);
write(v:8:4,interv.li:8:4,interv.ls:8:4,' ',cti:8:4);

```

```

aux:=0;
write(text_tab,' ');
write(' ');
for i:=1 to estagio do
  begin
    if (i>1) then aux:=aux+t.pos_carg[i-1];
    for j:=1 to t.pos_carg[i] do
      begin
        write(text_tab,cargas[aux+j]);
        write(cargas[aux+j]);
        if (j<>t.pos_carg[i]) then
          begin
            write(text_tab,' ');
            write('.');
          end;
        end;
      if (t.pos_carg[i]=0) then
        begin
          write(text_tab,'0');
          write('0');
        end;
      if (i<>estagio) then
        begin
          write(text_tab,'-');
          write('-');
        end;
      end;
    writeln(text_tab);
    writeln;
  end;
end;

procedure imp_linha_reduzida(t      :tip_tab;
                             estagio :integer);
var
  i,j :integer;
  aux :integer;
begin
  with t do
    begin
write(pert:2:2,u:5:2,xi_1_:5:2,xi_1:5:2,xi_:5:2,xi:5:2);
write(v:5:2,interv.li:5:2,interv.ls:5:2,' ',cti:6:5);
aux:=0;
write(' ');
for i:=1 to estagio do
  begin
    if (i>1) then aux:=aux+t.pos_carg[i-1];
    for j:=1 to t.pos_carg[i] do
      begin
        write(cargas[aux+j]);
        if (j<>t.pos_carg[i]) then write('.');
      end;
      if (t.pos_carg[i]=0) then write('0');
      if (i<>estagio) then write('-');
    end;

```

Anexos 1 e 2

```

writeln;
end;
end;

procedure imp_tab(var arq :tab_file;
                  estagio :integer);

var
  t :tip_tab;
begin
  writeln('Estagio: ',estagio);
  writeln('pert u  Xi-1/ Xi-1 Xi/ Xi  V  Int.Inf.
Int.Sup.  ');
  reset(arq);
  while (not eof(arq)) do
    begin
      read(arq,t);
      imp_linha_reduzida(t,estagio);
    {  imp_linha(t,estagio);}
    end;
  writeln;
end;

procedure imp_tab_text(var arq :tab_file;
                       estagio :integer);

var
  t :tip_tab;
begin
  writeln(text_tab,'Estagio: ',estagio);
  writeln(text_tab,'pert u  Xi-1/ Xi-1 Xi/
Xi  V  L.I  L.S  Ct  CARGAS');
  writeln('Estagio: ',estagio);
  writeln('pert u  Xi-1/ Xi-1 Xi/ Xi
V  L.I  L.S  Ct  CARGAS ');
  reset(arq);
  while (not eof(arq)) do
    begin
      read(arq,t);
      imp_linha_text(t,estagio);
    {  imp_linha_reduzida(t,estagio);}
    {  imp_linha(t,estagio);}
    end;
  writeln(text_tab);
  writeln;
end;

procedure define_ultimo_criterio(var x :integer;
                                  var ch :char);

begin
  clrscr;
  writeln('Escolha do Criterio:');
  writeln;
  writeln('<F1> - Heuristica Dinamica');
  writeln('<F2> - Heuristica Dinamica/Fuzzy');
  writeln('<ESC> - Sair');
  writeln;

```

```

write('Opcao');
repeat
  ch:=upcase(readkey);
  if (ch=#0) then ch:=upcase(readkey);
until(ch in [#27,',' '<']);
case ch of
  ':x:=0;
  '<':x:=1;
end;
end;

procedure get_carga_simp( a :integer;
                          var aux_carg :real);

var
  i :integer;
  teste :boolean;
begin
  teste:=true;
  for i:=1 to cl.num_carg do
    begin
      if ((a=c1.carg[i].num) and teste) then
        begin
          teste:=false;
          aux_carg:=c1.carg[i].valor;
        end;
    end;
end;

procedure get_u( nome :tip_nome;
                 estagio :integer;
                 i :integer;
                 var t :tip_tab);

var
  arq_cb :text;
  str1,str2 :string;
  k,a :integer;
  aux_pert :real;
  aux_carg :real;
begin
  assign(arq_l,drv_virtual+'\\'+nome+'.lpu');
  assign(arq_cb,drv_virtual+'\\'+nome+'.cb');
  reset(arq_l);
  reset(arq_cb);
  seek(arq_l,estagio-1);
  read(arq_l,c1);
  readln(arq_cb,str1);
  str(estagio,str2);
  str2:='I'+str2;
  while ((not eof(arq_cb)) and (str1 <>str2)) do
    readln(arq_cb,str1);
  for k:=1 to i-1 do
    begin
      readln(arq_cb,str1);
    end;
  k:=t.num_carg;

```

Anexos 1 e 2

```

seek(arq_1,estagio-1);
read(arq_1,c1);
t.u:=0;
read(arq_cb,a);
if (a<>0) then
  begin
    get_carga_simp(a,aux_carg);
    t.u:=t.u+aux_carg; {}{}{}{}{}{}{}
    inc(k);
  end;
while (a<>0) do
  begin
    read(arq_cb,a);
    if (a<>0) then
      begin
        inc(k);
        get_carga_simp(a,aux_carg);
        t.u:=t.u+aux_carg;
      end;
  end;
close(arq_1);
close(arq_cb);
end;

procedure get_cargas( nome :tip_nome;
                     estagio :integer;
                     i :integer;
                     var t :tip_tab);

var
  arq_cb :text;
  str1,str2 :string;
  k,a :integer;
  aux_pert :real;
begin
  assign(arq_1,drv_virtual+'\'+nome+'.lpu');
  assign(arq_cb,drv_virtual+'\'+nome+'.cb');
  reset(arq_1);
  reset(arq_cb);
  for k:=1 to estagio do
    read(arq_1,c1);
    readln(arq_cb,str1);
    str(estagio,str2);
    str2:='I'+str2;
    while ((not eof(arq_cb)) and (str1<>str2)) do
      readln(arq_cb,str1);
    for k:=1 to i-1 do
      begin
        readln(arq_cb,str1);
      end;
    k:=t.num_carg;
    seek(arq_1,estagio-1);
    read(arq_1,c1);
    read(arq_cb,a);
    if (a<>0) then
      begin
        inc(k);
        t.cargas[k]:=a;

```

```

end;
while (a<>0) do
  begin
    read(arq_cb,a);
    if (a<>0) then
      begin
        inc(k);
        t.cargas[k]:=a;
      end;
  end;
t.num_carg:=k;
close(arq_1);
close(arq_cb);
end;

procedure calc_linha( estagio :integer;
                     cm :real;
                     i :integer;
                     nome :tip_nome;
                     var t :tip_tab;
                     var retorno :boolean;
                     fat2 :real);

begin
  retorno:=false;
  get_u(nome,estagio,i,t);
  get_cargas(nome,estagio,i,t);
  t.pos_carg[estagio]:=t.num_carg;
  if (estagio>1) then
    t.pos_carg[estagio]:=t.pos_carg[estagio]-
t.pos_carg[estagio-1];
  t.v:=t.xi_cm-t.u;
  t.interv.li:=t.v-fat2;
  t.interv.ls:=t.v+fat2;
  if ((t.xi_1>t.interv.li) and (t.xi_1<=t.interv.ls))
  then
    then
      begin
        t.xi:=t.xi_1+cm+t.u;
        t.cti:=sqr(t.u+cm)+t.cti_1;
        retorno:=true;
      end;
end;

procedure pert_carga( carga :integer;
                     estagio :integer;
                     var pert :real);

var
  i :integer;
  teste :boolean;
begin
  pert:=0;
  i:=1;
  teste:=false;
  reset(arq_1);
  seek(arq_1,estagio-1);
  read(arq_1,c1);
  repeat
    if (carga=c1.carg[i].num) then

```

Anexos 1 e 2

```

begin
  pert:=c1.carg[i].pert;
  teste:=true;
end;
inc(i);
until(teste or (i>c1.num_carg));
close(arq_l);
end;

procedure calc_pert(var t :tip_tab;
                   estagio :integer);

var
  i,j :integer;
  aux_cont :integer;
  aux_pert :real;
  pertin :real;
  pert_ant :array[1..96] of real;
begin
  aux_cont:=0;
  aux_pert:=0;
  if (t.pos_carg[1]=0)
  then aux_pert:=0
  else
    begin
      pert_carga(t.cargas[1],1,aux_pert);
      for j:=2 to t.pos_carg[1] do
        begin
          pert_carga(t.cargas[j],1,pertin);
          aux_pert:=aux_pert+pertin-aux_pert*pertin;
        end;
      end;
    pert_ant[1]:=aux_pert;
    for i:=2 to estagio do
      begin
        aux_cont:=aux_cont+t.pos_carg[i-1];
        aux_pert:=0;
        if (t.pos_carg[i]=0)
        then
          aux_pert:=0
        else
          begin
            pert_carga(t.cargas[aux_cont+1],i,aux_pert);
            for j:=2 to t.pos_carg[i] do
              begin
                pert_carga(t.cargas[aux_cont+j],i,pertin);
                aux_pert:=aux_pert+pertin-aux_pert*pertin;
              end;
            end;
            pert_ant[i]:=aux_pert;
          end;
        aux_pert:=0;
        aux_cont:=0;
        for i:=1 to estagio do
          begin
            if (pert_ant[i]<>0) then
              begin
                aux_pert:=aux_pert+pert_ant[i];

```

```

inc(aux_cont);
end;
end;
if (aux_cont=0) then t.pert:=0
else t.pert:=aux_pert/aux_cont;
end;

procedure calc_vezes( estagio :integer; {numero de
linhas para cada semi-tabela}
                    var nome :tip_nome;
                    var vezes :integer);

var
  arq_cb :text;
  s1,s2 :string;
begin
  str(estagio,s2);
  s2:='I'+s2;
  assign(arq_cb,drv_virtual+'I'+nome+'.cb');
  reset(arq_cb);
  readln(arq_cb,s1);
  while ((not eof(arq_cb)) and (s1<>s2)) do
    readln(arq_cb,s1);
  vezes:=0;
  readln(arq_cb,s1);
  while ((not eof(arq_cb)) and (s1[1]<>'I')) do
    begin
      inc(vezes);
      readln(arq_cb,s1);
    end;
  close(arq_cb);
end;

procedure calc_semi_tab(var st :tab_file;
                       var t :tip_tab;
                       cm :real;
                       estagio :integer;
                       var nome :tip_nome;
                       cv :integer;
                       fat2 :real);

var
  str1 :string;
  str2 :string;
  i,vezes :integer;
  aux_num_carg :integer;
  teste1 :boolean;
  aux :integer;
begin
  aux_num_carg:=t.num_carg;
  rewrite(st);
  calc_vezes(estagio,nome,vezes);
  for i:=1 to vezes do
    begin
      if (cv=0) then t.num_carg:=0
      else
        begin
          t.num_carg:=aux_num_carg;
          if (estagio>1) then

```

Anexos 1 e 2

```

begin
    aux:=t.pos_carg[estagio-1];
    t.pos_carg[estagio-1]:=t.num_carg;
end;
end;
calc_linha(estagio,cm,i,nome,t,teste1,fat2);
if (estagio>1) then
    t.pos_carg[estagio-1]:=aux;
calc_pert(t,estagio);
if teste1 then
    write(st,t);
end;
{ clrscr;
imp_tab(st,cstagio); {}
end;

procedure elimina(var arq_temp1 :tab_file;
    posi :integer);
var
    arq_temp2 :tab_file;
    t1 :tip_tab;
    i :integer;
begin
    reset(arq_temp1);
    assign(arq_temp2,drv_virtual+'\'+temp.$$$');
    rewrite(arq_temp2);
    i:=1;
    while (not eof(arq_temp1)) do
        begin
            read(arq_temp1,t1);
            if (i<>posi) then
                write(arq_temp2,t1);
            inc(i);
        end;
    reset(arq_temp2);
    rewrite(arq_temp1);
    while (not(eof(arq_temp2))) do
        begin
            read(arq_temp2,t1);
            write(arq_temp1,t1);
        end;
    close(arq_temp2);
end;

procedure acha_iguais(var teste1 :boolean;
    var posi :integer;
    var arq_temp1 :tab_file);
var
    i :integer;
    t1,t2 :tip_tab;
begin
    reset(arq_temp1);
    teste1:=false;
    if (not eof(arq_temp1)) then
        begin
            read(arq_temp1,t1);
            for i:=2 to filesize(arq_temp1) do
                begin

```

```

read(arq_temp1,t2);
if (t1.xi=t2.xi) then
    begin
        teste1:=true;
        posi:=i;
    end;
end;
end;

procedure crit_tab_0(var arq_temp1 :tab_file);
var
    arq_temp2 :tab_file;
    t :tip_tab;
    i,j :integer;
    teste :boolean;
begin
    assign(arq_temp2,drv_virtual+'\'+temp2.tmp');
    rewrite(arq_temp2);
    reset(arq_temp1);
    while (not eof(arq_temp1)) do
        begin
            read(arq_temp1,t);
            teste:=false;
            for i:=1 to t.num_carg-1 do
                for j:=i+1 to t.num_carg do
                    if (t.cargas[i]=t.cargas[j]) then
                        teste:=true;
                if (not teste) then
                    write(arq_temp2,t);
            end;
            reset(arq_temp2);
            rewrite(arq_temp1);
            while (not eof(arq_temp2)) do
                begin
                    read(arq_temp2,t);
                    write(arq_temp1,t);
                end;
            close(arq_temp2);
        end;

procedure crit_tab_1(var arq_temp1 :tab_file);
var
    arq_temp2 :tab_file;
    t :tip_tab;
begin
    reset(arq_temp1);
    assign(arq_temp2,drv_virtual+'\'+temp2.tmp');
    rewrite(arq_temp2);
    while (not eof(arq_temp1)) do
        begin
            read(arq_temp1,t);
            if ((t.xi_1>t.interv.li) and (t.xi_1<=t.interv.ls))
                then write(arq_temp2,t);
            end;
        reset(arq_temp2);

```

Anexos 1 e 2

```

rewrite(arq_temp1);
while (not eof(arq_temp2)) do
  begin
    read(arq_temp2,t);
    write(arq_temp1,t);
  end;
close(arq_temp2);
end;

procedure crit_tab_2(var arq_temp1 :tab_file);
var
  arq_temp2 :tab_file;
  arq_temp3 :tab_file;
  t1,t2 :tip_tab;
  i,j,num :integer;
  posi :integer;
  teste1 :boolean;
begin
  reset(arq_temp1);
  assign(arq_temp2,drv_virtual+'\'+'temp2.tmp');
  assign(arq_temp3,drv_virtual+'\'+'temp3.tmp');
  rewrite(arq_temp2);
  rewrite(arq_temp3);
  while (not eof(arq_temp1)) do
    begin
      acha_iguais(teste1,posi,arq_temp1);
      if (not teste1)
        then
          begin
            reset(arq_temp1);
            read(arq_temp1,t1);
            write(arq_temp3,t1);
            elimina(arq_temp1,1);
          end
        else
          begin
            reset(arq_temp1);
            read(arq_temp1,t1);
            seek(arq_temp1,posi-1);
            read(arq_temp1,t2);
            if (t1.cti<t2.cti) then
              begin
                seek(arq_temp1,posi-1);
                write(arq_temp1,t1);
              end;
            elimina(arq_temp1,1);
          end;
      reset(arq_temp1);
    end;
  reset(arq_temp3);
  rewrite(arq_temp1);
  while (not eof(arq_temp3)) do
    begin
      read(arq_temp3,t1);
      write(arq_temp1,t1);
    end;
end;

```

```

close(arq_temp2);
close(arq_temp3);
end;

procedure crit_tab_3(var arq_temp1 :tab_file);
var
  arq_temp2 :tab_file;
  t :tip_tab;
  maior,aux :real;
  i,posi :integer;
begin
  assign(arq_temp2,drv_virtual+'\'+'temp2.tmp');
  rewrite(arq_temp2);
  reset(arq_temp1);
  while ((filesize(arq_temp2)<2) and (not
eof(arq_temp1))) do
    begin
      read(arq_temp1,t);
      maior:=((t.xi*t.xi)/t.cti);
      posi:=1;
      i:=1;
      while (not eof(arq_temp1)) do
        begin
          inc(i);
          read(arq_temp1,t);
          aux:=((t.xi*t.xi)/t.cti);
          if (maior<aux) then
            begin
              maior:=aux;
              posi:=i;
            end;
          end;
        seek(arq_temp1,posi-1);
        read(arq_temp1,t);
        write(arq_temp2,t);
        elimina(arq_temp1,posi);
        reset(arq_temp1);
      end;
      reset(arq_temp2);
      rewrite(arq_temp1);
      while (not eof(arq_temp2)) do
        begin
          read(arq_temp2,t);
          write(arq_temp1,t);
        end;
      close(arq_temp2);
    end;

procedure crit_tab_4(var arq_temp1 :tab_file);
var
  arq_temp2 :tab_file;
  t :tip_tab;
  maior,aux :real;
  i,posi :integer;
begin

```

Anexos 1 e 2

```

assign(arq_temp2,drv_virtual+'\\'+temp2.tmp');
rewrite(arq_temp2);
reset(arq_temp1);
while ((filesize(arq_temp2)<2) and (not
cof(arq_temp1))) do
begin
read(arq_temp1,t);
maior:=t.pert;
posi:=1;
i:=1;
while (not cof(arq_temp1)) do
begin
inc(i);
read(arq_temp1,t);
aux:=t.pert;
if (maior<aux) then
begin
maior:=aux;
posi:=i;
end;
end;
seek(arq_temp1,posi-1);
read(arq_temp1,t);
write(arq_temp2,t);
elimina(arq_temp1,posi);
reset(arq_temp1);
end;
reset(arq_temp2);
rewrite(arq_temp1);
while (not cof(arq_temp2)) do
begin
read(arq_temp2,t);
write(arq_temp1,t);
end;
close(arq_temp2);
end;

procedure crit_tabe(var arq :tab_file; {c1: 0 -
princiro estagio}
c1 :integer; { 1 - demias estagios}
c2 :integer; {}
estagio :integer);
begin
if (c1=1)
then
begin
crit_tab_0(arq);
{ imp_tab(arq,estagio); {}
end
else
begin
crit_tab_1(arq);
{ imp_tab(arq,estagio); {}
end;
crit_tab_2(arq);
{ imp_tab(arq,estagio); {}
if (c2=0)
then

```

```

begin
crit_tab_3(arq);
{ imp_tab(arq,estagio); {}
end
else
begin
crit_tab_4(arq);
{ imp_tab(arq,estagio); {}
end;
end;

procedure calc_tabela( nome :tip_nome;
ult_crit :integer;
fat1 :real;
fat2 :real);
var
arq_cr :rtd_file;
arq_tab :tab_file;
arq_tab_ax1 :tab_file;
arq_tab_ax2 :tab_file;
arq_tab_ant :tab_file;
t :tip_tab;
a_cr :rtd2;
estag :string;
a :string;
carga :integer;
cm :real;
estagio :integer;
teste :boolean;
xi_ini :integer;
xi_fin :integer;
xi_cont :integer;
h :integer;
begin
assign(arq_ld,drv_virtual+'\\'+nome+'.ld');
assign(arq_cr,nome+'.cr_');
assign(arq_tab,drv_virtual+'\\'+nome+'.tab');
assign(arq_tab_ax1,drv_virtual+'\\'+nome+'.tp1');
assign(arq_tab_ax2,drv_virtual+'\\'+nome+'.tp2');
assign(arq_tab_ant,drv_virtual+'\\'+nome+'.ant');
reset(arq_ld);
reset(arq_cr);
read(arq_cr,a_cr);
close(arq_cr);
rewrite(arq_tab);
rewrite(arq_tab_ax1);
rewrite(arq_tab_ant);
estag:='I1';
estagio:=1;
t.xi_1:=0;
t.xi_1:=0;
t.cti_1:=0;
t.num_carg:=0;
t.pert:=0;
for h:=1 to 400 do

```

Anexos 1 e 2

```

t.pos_carg[h]:=0;

cm:=a_cr[1].a;
seek(arq_ld,0);
read(arq_ld,c2);
xi_ini:=round(c2.i2.li*fat1);
xi_fin:=round(c2.i2.ls*fat1);
for xi_cont:=xi_ini to xi_fin do
begin
seek(arq_ld,0);
read(arq_ld,c2);
t.xi_:=xi_cont/fat1;

calc_semi_tab(arq_tab_ax1,t,cm,estagio,nome,0,fat2);{xi
_cont-trunc(c2.i2.li/1000)+1;}
crit_tabe(arq_tab_ax1,0,ult_crit,estagio);
reset(arq_tab_ax1);
while (not eof(arq_tab_ax1)) do
begin
read(arq_tab_ax1,t);
write(arq_tab,t);
write(arq_tab_ant,t);
end;
end;
clrscr;
imp_tab_text(arq_tab,estagio);

for h:=1 to 95 do      {for h:=1 to 94 do  obs1}
begin
reset(arq_tab_ant);
rewrite(arq_tab_ax1);
rewrite(arq_tab);
inc(estagio);
cm:=a_cr[estagio].a;
seek(arq_ld,h);
read(arq_ld,c2);
xi_ini:=trunc(c2.i2.li*fat1);
xi_fin:=trunc(c2.i2.ls*fat1);
for xi_cont:=xi_ini to xi_fin do
begin
reset(arq_tab_ant);
rewrite(arq_tab_ax2);
while (not eof(arq_tab_ant)) do
begin
seek(arq_ld,0);
read(arq_ld,c2);
read(arq_tab_ant,t);
t.xi_1_:=t.xi_;
t.xi_1:=t.xi;
t.cti_1:=t.cti;
t.xi_:=xi_cont/fat1;

calc_semi_tab(arq_tab_ax1,t,cm,estagio,nome,1,fat2);{xi
_cont-trunc(c2.i2.li/1000)+1;}
reset(arq_tab_ax1);
while (not eof(arq_tab_ax1)) do
begin
seek(arq_ld,0);
read(arq_ld,c2);
read(arq_tab_ant,t);
t.xi_1_:=t.xi_;
t.xi_1:=t.xi;
t.cti_1:=t.cti;
t.xi_:=xi_cont/fat1;

calc_semi_tab(arq_tab_ax1,t,cm,estagio,nome,1,fat2);
reset(arq_tab_ax1);
while (not eof(arq_tab_ax1)) do
begin
read(arq_tab_ax1,t);
write(arq_tab_ax2,t);
end;
end;
crit_tabe(arq_tab_ax2,1,ult_crit,estagio);
reset(arq_tab_ax2);
while (not eof(arq_tab_ax2)) do
begin
read(arq_tab_ax2,t);
write(arq_tab,t);
end;
end;
clrscr;
reset(arq_tab);
rewrite(arq_tab_ant);
while (not eof(arq_tab)) do
begin
read(arq_tab,t);
write(arq_tab_ant,t);
end;
imp_tab_text(arq_tab,estagio);
end;

{ obs1 }
{ h:=95;
reset(arq_tab_ant);
rewrite(arq_tab_ax1);
rewrite(arq_tab);
inc(estagio);
cm:=a_cr[estagio].a;
seek(arq_ld,h);
read(arq_ld,c2);
xi_ini:=trunc(c2.i2.li*fat1);
xi_fin:=trunc(c2.i2.ls*fat1);
xi_cont:=xi_fin;
reset(arq_tab_ant);
rewrite(arq_tab_ax2);
while (not eof(arq_tab_ant)) do
begin
read(arq_tab_ant,t);
t.xi_1_:=t.xi_;
t.xi_1:=t.xi;
t.cti_1:=t.cti;
t.xi_:=xi_cont/fat1;

calc_semi_tab(arq_tab_ax1,t,cm,estagio,nome,1,fat2);
reset(arq_tab_ax1);
while (not eof(arq_tab_ax1)) do
begin
read(arq_tab_ax1,t);
write(arq_tab_ax2,t);
end;
end;
crit_tabe(arq_tab_ax2,1,ult_crit,estagio);
reset(arq_tab_ax2);
while (not eof(arq_tab_ax2)) do
begin
read(arq_tab_ax2,t);
write(arq_tab,t);
end;
end;
clrscr;
reset(arq_tab);
rewrite(arq_tab_ant);
while (not eof(arq_tab)) do
begin
read(arq_tab,t);
write(arq_tab_ant,t);
end;
imp_tab_text(arq_tab,estagio);
end;

```

Anexos 1 c 2

```

begin
  read(arq_tab_ax2,t);
  write(arq_tab,t);
end;
clrscr;}
{ imp_tab_text(arq_tab,cstagio);
} close(arq_ld);
close(arq_tab);
close(arq_tab_ax1);
close(arq_tab_ant);
close(arq_tab_ax2);
end;

procedure get_carga( a :integer;
                    var carga :real);
var
  teste :boolean;
  i :integer;
begin
  reset(arq_l);
  teste:=false;
  while ((not eof(arq_l)) and (not teste)) do
    begin
      read(arq_l,c1);
      for i:=1 to c1.num_carg do
        begin
          if (a=c1.carg[i].num) then
            begin
              teste:=true;
              carga:=c1.carg[i].valor;
            end;
          end;
        end;
      end;
    end;
  end;

procedure plt_cur_final(cx,cy :integer;
                      cr :rtd2);
var
  i :integer;
  ganho :real;
begin
  ganho:=200.5;
  moveto(cx,cy-trunc(cr[1].a*ganho));
  lineto(cx+4,cy-trunc(cr[1].a*ganho));
  for i:=2 to 96 do
    begin
      lineto(cx+(i-1)*4,cy-trunc(cr[i].a*ganho));
      lineto(cx+i*4,cy-trunc(cr[i].a*ganho));
    end;
  lineto(cx+i*4,cy);
end;

procedure grafi(cx,cy :integer;
               cr :rtd2);
var
  dr,md :integer;

```

```

begin
  dr:=detect;
  initgraph(dr,md,"");
  axes3(cx,cy,0);
  plt_cur_final(cx,cy,cr);
  readkey;
  closegraph;
end;

procedure cur_resp_detect( nome :tip_nome;
                          var t :tip_tab);
var
  arq_rtd :rtd_file;
  arq_tab :tab_file;
  curva :rtd2;
  i :integer;
  s :real;
  tab :integer;
  menor :real;
begin
  assign(arq_rtd,nome+'.cr');
  assign(arq_tab,nome+'.tab');
  reset(arq_rtd);
  reset(arq_tab);
  read(arq_rtd,curva);
  s:=0;
  for i:=1 to 96 do
    s:=s+curva[i].a;
  i:=0;
  tab:=0;
  read(arq_tab,t);
  menor:=abs(s-t.xi);
  while (not eof(arq_tab)) do
    begin
      read(arq_tab,t);
      inc(i);
      if (abs(s-t.xi)<menor) then
        begin
          menor:=s-t.xi;
          tab:=i;
        end;
      end;
    seek(arq_tab,tab);
    read(arq_tab,t);
    close(arq_rtd);
    close(arq_tab);
  end;

procedure cria_cur_resp( nome :tip_nome);
var
  cr_ant :rtd2;
  cr_resp :rtd2;
  arq_mod :rtd_file;
  arq_cr :rtd_file;
  t :tip_tab;
  i :integer;

```

Anexos 1 e 2

```

aux      :integer;
aux_tot  :integer;
estagio  :integer;
carga    :real;
aux_carg :real;
ch       :char;
begin
  clrscr;
  assign(arq_cr,nome+'.cr_');
  assign(arq_mod,nome+'.mod');
  assign(arq_l,nome+'.lpu');
  filemode:=2;
  reset(arq_cr);
  read(arq_cr,cr_ant);
  close(arq_cr);
  reset(arq_l);
  { read(arq_tab,t);}
  cur_resp_detect(nome,t);
  estagio:=96;
  cr_resp[estagio].a:=t.xi;
  repeat
    aux:=0;
    for i:=1 to estagio-1 do      {passar de 3 para 95}
      aux:=aux+t.pos_carg[i];
    aux_tot:=aux+t.pos_carg[estagio];
    aux_carg:=0;
    for i:=aux+1 to aux_tot do
      begin
        get_carga(t.cargas[i],carga);
        aux_carg:=aux_carg+carga;
      end;
    aux_carg:=aux_carg+cr_ant[estagio].a;
    cr_resp[estagio-1].a:=cr_resp[estagio].a-aux_carg;
    dec(estagio);
  until(estagio=1);
  for i:=96 downto 2 do
    cr_resp[i].a:=cr_resp[i].a-cr_resp[i-1].a;
  rewrite(arq_mod);
  write(arq_mod,cr_resp);
  close(arq_mod);
  close(arq_l);
end;

procedure _arq_virtual(nome :tip_nome);
var
  arq_tab  :tab_file;
  arq_aux  :tab_file;
  t       :tip_tab;
begin
  assign(arq_aux,nome+'.tab');
  assign(arq_tab,drv_virtual+':\'+nome+'.tab');
  reset(arq_tab);
  rewrite(arq_aux);
  while (not eof(arq_tab)) do
    begin
      read(arq_tab,t);
      write(arq_aux,t);
    end;
end;

close(arq_tab);
close(arq_aux);
end;

procedure det_cur_tipica( nome :tip_nome;
                          var fun3 :boolean);
var
  lim      :tip_lim;
  energ    :tip_med_dia;
  dias     :integer;
  cur_ok_0 :tip_cur_ok;
  cur_ok_1 :tip_cur_ok;
  cur_ok_2 :tip_cur_ok;
  cur_ok_3 :tip_cur_ok;
  num_cur_ok_0 :integer;
  num_cur_ok_1 :integer;
  num_cur_ok_2 :integer;
  num_cur_ok_3 :integer;
begin
  arq_rtd(nome);
  cria_arq_cr(nome,cur_ok_0,num_cur_ok_0);
  list_hist_energ(nome);
  calc_ano(nome,lim);
  imp_limite(lim);
  calc_dia(nome,energ,dias);

  plota_curvas(nome,cur_ok_0,num_cur_ok_0,num_cur_o
k_0);

  criterio_1(nome,lim,energ,dias,cur_ok_1,num_cur_ok_1
);

  plota_curvas(nome,cur_ok_1,num_cur_ok_1,num_cur_o
k_0);

  criterio_2(nome,lim,energ,dias,cur_ok_1,cur_ok_2,num
cur_ok_2);

  plota_curvas(nome,cur_ok_2,num_cur_ok_2,num_cur_o
k_0);

  criterio_3(nome,energ,dias,cur_ok_2,cur_ok_3,num_cur
_ok_3);

  plota_curvas(nome,cur_ok_3,num_cur_ok_3,num_cur_o
k_0);
  fun3:=true;
end;

procedure result_cur_tip(nome :tip_nome;
                          fun4 :boolean);

procedure sub_menu(var ch :char;
                  fun4 :boolean);
begin
  clrscr;
  writeln('Resultados');
  writeln;

```

Anexos 1 e 2

```

writeln('<F1> - Curva Tipica');
writeln('<F2> - Curva Tipica Modulada');
writeln;
writeln('Opcao:');
repeat
  repeat
    ch:=readkey
    until(ch in [#0,#27]);
    if (ch<>#27) then ch:=readkey;
    if ((ch='<') and (not fun4)) then ch:=#0;
    until(ch in [';', '<', #27]);
  end;
end;

var
  ch1, ch2 :char;
begin
  repeat
    sub_menu(ch1, fun4);
    if (ch1<>#27) then
      begin
        repeat
          menu2(ch2);
          if (ch2<>#27) then
            begin
              case ch2 of
                '..<':plota_lin(90,400,nome,ch1,ch2);
                '..?':aquec(nome,ch2);
              end;
            end;
          until(ch2 in [#27,#0]);
        end;
      until(ch1=#27);
    end;
  end;

  procedure subtrai_cur_tip(nome :tip_nome);
  var
    _cr :rtd_file;
    _lpu :file of vet_carg;
    _cr_ :rtd_file;
    curva :rtd2;
    vet_c :vet_carg;
    i,k :integer;
    tot_carg :real;
  begin
    assign(_cr,nome+'.cr');
    assign(_lpu,nome+'.lpu');
    assign(_cr_,nome+'.cr_');
    reset(_cr);
    reset(_lpu);
    rewrite(_cr_);
    read(_cr,curva);
    for k:=1 to 96 do
      begin
        read(_lpu,vet_c);
        tot_carg:=0;
        for i:=1 to vet_c.num_carg do
          tot_carg:=tot_carg+vet_c.carg[i].valor;
          curva[k].a:=curva[k].a-tot_carg;
        end;
      end;
    until(ch=#27);
  end;

  procedure modulacao( nome :tip_nome;
    var fun4 :boolean;
    fat1 :real;
    fat2 :real);
  var
    cr_ant :rtd2;
    cr_resp :rtd2;
    ch :char;
    ult_crit :integer;
  begin
    assign(text_tab,nome+'.txt');
    repeat
      lc_cargas(nome,ch);
      if (ch='=') then
        begin
          define_ultimo_criterio(ult_crit,ch);
          clrscr;
          if (ch<>#27)
            then
              begin
                cria_l_pu(nome);
                cria_despac(nome,fat1,fat2); {calcula os
                intervalos}
                calc_combinacoes(nome);
                arq_virtual(nome);

                rewrite(text_tab);
                calc_tabela(nome,ult_crit,fat1,fat2);
                close(text_tab);

                _arq_virtual(nome);
                cria_cur_resp(nome);
                fun4:=true;
              end
            else ch:=#0;
          end;
        until(ch=#27);
      end;
    end;

  var
    nome :tip_nome;
    ch :char;
    fun2, fun3 :boolean;
    fun4 :boolean;
  begin
    { cria_cur_resp('ba');
    result_cur_tip('ba',true);

```

Anexos 1 e 2

```
exit;}
drv_virtual:='d';
nome:="";
fun2:=false;
fun3:=false;
fun4:=false;
menu_principal(ch);
repeat
  menu1(ch,nome);
  case ch of
    '|': novo_arq_rtd_hst(nome);
    '<': le_nome(nome,fun2);
    '=': if fun2 then det_cur_tipica(nome,fun3);
    '>': if fun3 then
      begin
        subtrai_cur_tip(nome);
        modulacao(nome,fun4,100,0.005);
      end;
    '?': if (fun4 or fun3) then result_cur_tip(nome,fun4);
  end;
until(ch=#27);
end.
```

