

UNIVERSIDADE ESTADUAL DE CAMPINAS  
FACULDADE DE ENGENHARIA ELÉTRICA  
DEPARTAMENTO DE ENGENHARIA DA COMPUTAÇÃO E AUTOMAÇÃO INDUSTRIAL

SISTEMAS DE RACIOCÍNIO APROXIMADO  
E SUA UTILIZAÇÃO EM CONTROLADORES INDUSTRIAIS:  
UM ESTUDO SOBRE OS  
CONTROLADORES BASEADOS EM REGRAS NEBULOSAS

por: Aluizio Saiter Mota <sup>nº</sup>  
orientador: Prof. Dr. Marcio Luiz A. Netto <sup>de</sup>

191

Tese apresentada à Faculdade de  
Engenharia Elétrica FEE - UNICAMP  
como parte dos requisitos exigidos  
para obtenção do título de MESTRE  
EM ENGENHARIA.

25 de setembro 1991

Este exemplar corresponde à redação final da  
defendida por Aluizio Saiter Mota  
pela Comissão  
Julgadora em 02/12/91.

  
Orientador

UNICAMP  
BIBLIOTECA CENTRAL

BRUNO

Este trabalho contou com o apoio financeiro do  
Conselho Nacional de Desenvolvimento Científico e Tecnológico  
CNPq

Dedico este trabalho aos meus pais,  
meus irmãos e à Regina.

## AGRADECIMENTOS

Aos amigos de "república": Thomas, José e Wolfi pela amizade durante todo este tempo.

Aos amigos Zig e Mirian.

À Holz, que á "distancia" torceu pela realização deste trabalho.

Ao amigo Anirio pelo apoio e amizade.

À Regina pelo incentivo e apoio na realização deste trabalho, e cuidado na confecção dos desenhos deste texto.

Ao prof. Luis Gimeno, que recebeu-me como aluno na UNICAMP.

Ao corpo docente do Departamento de Engenharia de Computação e Automação Industrial da Faculdade de Engenharia Elétrica da UNICAMP pela convivência em todo este tempo.

À Marcio Luiz pela orientação deste trabalho e amizade dedicada em todos estes anos.

*"Se você não consegue resolver um problema, você pode mudar sua estratégia e tentar resolvê-lo de outra forma."*

*Marvin Minsky*

## RESUMO

Os sistemas de controle de processos industriais são baseados em uma teoria bem desenvolvida: controle de sistemas. Esta teoria permite-nos tratar vários problemas complicados do cotidiano destes sistemas. Entretanto, a teoria de controle de sistemas tem como condição básica a existência de um modelo matemático preciso do sistema. Esta condição tem apresentado limitações e inconvenientes.

Este trabalho estuda uma abordagem alternativa: a utilização de sistemas especialistas realizando a função de controladores. Esta abordagem é discutida em dois tipos de ambientes. O primeiro, com o sistema especialista desempenhando diretamente a função do controlador. O segundo, com o sistema especialista em um ambiente hierárquico atuando como um módulo de supervisão de um controlador convencional. São apresentados os resultados obtidos em simulação na implementação destes dois sistemas e apontados alguns pontos a serem considerados em projeto.

## ABSTRACT

The industrial process control systems are based on a well founded theory: control system. This theory allow us to work with a variety of complicated daily problems of process system. However the theory of control system has as primary condition: the existence of a mathematical model. Such condition in many occasions has presented inconveniences and limitations.

This work studies an alternative approach to this problem: the use of expert systems. This approach is made on two kinds of environment. The first acts as the controller of the plant. The second studies an hierarchal environment as if it was acting as a supervising module. The result of the simulation of these two systems are presented and points to be considered in design are discussed.

# ÍNDICE

Tabela de Símbolos	1v
Índice de Figuras	v
Prefácio	1x

## PARTE I

### Fundamentos Teóricos

#### Capítulo 1

Introdução aos Conjuntos e a Lógica Nebulosa	1
1.1 - Definição de Conjunto Nebuloso	2
1.2 - Algumas Operações Básicas com Conjuntos Nebulosos	4
1.3 - Definição de Lógica Nebulosa	7
1.4 - Operações Básicas da Lógica Nebulosa	9

#### Capítulo 2

Alguns Aspectos a serem Considerados sobre Sistemas Baseados em Conhecimento	15
2.1 - Conhecimento e sua Representação	16
2.2 - O uso da Lógica Nebulosa para Representação de Senso Comum	19
2.3 - Modelamento Lingüístico	23

Conclusão da Parte I	26
----------------------	----

## PARTE II

### Análise e Síntese de Controladores Baseados em Regras Nebulosas

#### Capítulo 3

Análise de Controladores Baseados em Regras Nebulosas	27
3.1 - Arquitetura de um Controlador Baseado em Regras Nebulosas	28

3.2 - Componentes de um Controlador Baseado em Regras Nebulosas	30
3.2.1 - Amplificadores	30
3.2.2 - Conversão do Sinal Determinístico para Sinal Nebuloso	31
3.2.3 - Regras e Inferências	34
3.2.3.1 - Algoritmo de Inferência para Sistemas de Uma Entrada e Uma Saída	35
3.2.3.2 - Algoritmo de Inferência para Sistemas de Duas Entradas e Uma Saída	39
3.2.4 - Conversão de Sinal Nebuloso para Sinal Determinístico	43
<b>Capítulo 4</b>	
<b>Estudo de um Modelo para o Controlador Baseado em Regras Nebulosas</b>	45
4.1 - Um Modelo Baseado em Heurística	46
4.2 - A Relação entre o Controlador Baseado em Regras Nebulosas e o Controlador PI	48
<b>Capítulo 5</b>	
<b>Implementação de um Controlador Baseado em Regras Nebulosas</b>	54
5.1 - Seleção dos Parâmetros do Controlador Baseado em Regras Nebulosas e da Linguagem de Controle	55
5.2 - Projeto do Conjunto de Regras	60
5.3 - Ajuste dos parâmetros	62
5.4 - Exemplo de Implementação de um Controlador Baseado em Regras Nebulosas	63
<b>Conclusão da Parte II</b>	68

## PARTE III

### O Controlador Baseado em Regras Nebulosas como Módulo de Supervisão

#### Capítulo 6

#### Análise e Projeto de um Supervisor Utilizando Regras Nebulosas

6.1 - Arquitetura do Sistema	72
6.2 - O Algoritmo de Inferência	75
6.3 - Implementação do Supervisor e Resultados de Simulação	81
6.4 - A Utilização de um Conjunto de Meta-Regras de Controle para Monitoramento de um Sistema de Segunda Ordem	90

Conclusão da Parte III	94
------------------------	----

Conclusão Geral do Trabalho	97
-----------------------------	----

#### Apendice A

Uma Versão Rápida do Algoritmo de Inferência	98
--	----

#### Apendice B

O Método de Inferência de Zadeh	103
---------------------------------	-----

#### Apendice C

Listagem dos Conjuntos de Regras dos Capítulos 5 e 6	108
--	-----

Bibliografia	114
--------------	-----

## TABELA DOS PRINCIPAIS SÍMBOLOS UTILIZADOS

$A, B, C, \dots$	- Conjuntos ou subconjuntos não-nebulosos.
$\underline{A}, \underline{B}, \underline{C}, \dots$	- Conjuntos nebulosos.
$\bar{A}, \bar{B}, \bar{C}, \dots$	- Valores lingüísticos.
$x, y, z, \dots$	- Variáveis lingüísticas.
$\mu_A(a)$	- Função de pertinência do elemento $a$ em relação ao conjunto nebuloso $A$ .
$\in$	- Símbolo de pertence de um elemento em relação a um conjunto não-nebuloso.
$\notin$	- Símbolo de não-pertence de um elemento em relação a um conjunto não-nebuloso.
$\vee$	- Operador max (dado dois números retorna o maior deles).
$\wedge$	- Operador min (dado dois números retorna o menor deles).
$A \circ B$	- Regra de inferência de composição entre $A$ e $B$ .
$U \times V$	- Produto cartesiano entre $U$ e $V$ .
$A \rightarrow B$	- Implicação (se $A$ então $B$ ).
$\neg A$	- Complemento do conjunto $A$ .
$R_{AB}$	- Relação entre dois conjuntos $A$ e $B$ .
$\cup$	- União entre conjuntos.
$\cap$	- Interseção entre conjuntos.
$+$	- Adição.
$-$	- Subtração.
$f : M \rightarrow N$	- Mapeamento de $M$ em relação a $N$ através de $f$ .
$\forall$	- Quantificador universal (para todo).
$\exists$	- Quantificador existencial (existe ao menos um).
$\subset$	- Inclusão (é um subconjunto de).

## ÍNDICE DE FIGURAS

<b>Figura 2.1</b>		
Bases de conhecimento como modelo do domínio.		17
<b>Figura 2.2</b>		
Arquitetura de um sistema inteligente.		18
<b>Figura 2.3</b>		
Conjunto nebuloso para o predicado jovem.		21
<b>Figura 3.1</b>		
Sistema de controle utilizando um CBRN.		28
<b>Figura 3.2</b>		
Arquitetura do CBRN.		29
<b>Figura 3.3</b>		
Procedimento de normalização do sinal.		31
<b>Figura 3.4</b>		
Ex. de um grupo de conjuntos nebulosos.		32
<b>Figura 3.5</b>		
Conjuntos nebulosos discretizados.		33
<b>Figura 3.6</b>		
Ex. de um grupo de regras de controle.		34
<b>Figura 3.7</b>		
Forma canônica do conjunto de regras do tipo $A \rightarrow B$ .		35
<b>Figura 3.8</b>		
Ex. geométrico de inferência para regras $A \rightarrow B$ .		38
<b>Figura 3.9</b>		
Forma canônica do conjunto de regras do tipo $(A \wedge B) \rightarrow C$ .		39
<b>Figura 3.10</b>		
Ex. geométrico de inferência de duas regras $(A \wedge B) \rightarrow C$ interligadas pelo conectivo $\vee$ .		42
<b>Figura 3.11</b>		
Comparação de três procedimentos de conversão.		44
<b>Figura 4.1</b>		
Arquitetura de um controlador PI.		46

<b>Figura 4.2</b>	Arquitetura funcional de um CBRN.	47
<b>Figura 4.3</b>	Regras na forma de matriz de controle.	49
<b>Figura 4.4</b>	Efeito do aumento do número de conjuntos nebulosos na matriz de controle.	50
<b>Figura 4.5</b>	Efeito na matriz de controle quando o número de conjuntos nebulosos tende ao infinito.	50
<b>Figura 4.6</b>	Inclinação C da política de controle.	51
<b>Figura 5.1</b>	Conjuntos nebulosos quanto a sua resolução.	56
<b>Figura 5.2</b>	Ex. de conversão de um domínio contínuo para outro discreto.	58
<b>Figura 5.3</b>	Ex. de um conjunto de regras de controle.	59
<b>Figura 5.4</b>	Ex. do efeito da escolha do domínio do $e$ e da $e^\circ$ na resposta do sistema.	64
<b>Figura 5.5</b>	Conjuntos nebulosos dos sinais $e$ , $e^\circ$ e do sinal $y$ .	64
<b>Figura 5.6</b>	Resposta do sistema com um CBRN.	65
<b>Figura 5.7</b>	Resposta do sistema com um PI.	65
<b>Figura 5.8</b>	Conjunto nebulosos modificados, não sobrepondo-se.	66
<b>Figura 5.9</b>	Resposta do sistema com os conjunto nebulosos não sobrepondo-se.	66
<b>Figura 5.10</b>	Conjunto nebulosos modificados, sobrepondo-se mais.	67

<b>Figura 5.11</b>	Resposta do sistema com os conjuntos nebulosos sobrepondo-se mais.	67
<b>Figura 6.1</b>	Arquitetura do sistema de controle utilizando o supervisor.	72
<b>Figura 6.2</b>	Arquitetura funcional do supervisor.	73
<b>Figura 6.3</b>	Matriz de controle MC 5x5.	77
<b>Figura 6.4</b>	Conjuntos nebulosos utilizados na classificação do $e$ e da $e^\circ$ .	82
<b>Figura 6.5</b>	Conjuntos nebulosos utilizados na classificação do sinal de saída P.	82
<b>Figura 6.6</b>	Conjuntos nebulosos utilizados na classificação do sinal de saída I.	83
<b>Figura 6.7</b>	Resposta do sistema sem o supervisor ao sinal degrau.	84
<b>Figura 6.8</b>	Resposta do sistema com o supervisor ao sinal degrau.	85
<b>Figura 6.9</b>	Respostas do sistema com e sem o supervisor ao sinal rampa.	85
<b>Figura 6.10</b>	Respostas do sistema com e sem o supervisor ao sinal senoidal.	86
<b>Figura 6.11</b>	Resposta do sistema com e sem o supervisor ao sinal degrau com $T_p = 4.0\text{seg.}$ .	86
<b>Figura 6.12</b>	Resposta do sistema com e sem o supervisor ao sinal rampa com $T_p = 4.0\text{seg.}$ .	87

<b>Figura 6.13</b>	
Resposta do sistema com e sem o supervisor ao sinal senoidal com $T_p = 4.0\text{seg.}$	87
<b>Figura 6.14</b>	
Sensibilidade do sistema a $T_p$ , com $T_d = 0.32\text{seg.}$	88
<b>Figura 6.15</b>	
Sensibilidade do sistema a $T_d$ , com $T_p = 4.0\text{seg.}$	88
<b>Figura 6.16</b>	
Resposta do sistema a uma planta de segunda ordem.	89
<b>Figura 6.17</b>	
Arquitetura do supervisor utilizando um conjunto de meta-regras de controle.	91
<b>Figura 6.18</b>	
Resposta ao sinal degrau para o novo supervisor.	92
<b>Figura 6.19</b>	
Resposta ao sinal degrau do novo supervisor com $G_p(s)$ alterado, $T_{p2} = 6.0\text{seg.}$	93
<b>Figura 6.20</b>	
Estrutura de um sistema auto-organizado.	95
<b>Figura A.1</b>	
Exemplo geométrico de inferência para regras $A \rightarrow B$ com o novo algoritmo.	100
<b>Figura A.2</b>	
Exemplo geométrico de inferência para regras $(A \wedge B) \rightarrow C$ utilizando o novo algoritmo.	102
<b>Figura B.1</b>	
Exemplo geométrico de inferência para regras do tipo $(A \wedge B) \rightarrow C$ utilizando o método de Zadeh.	106
<b>Figura B.2</b>	
Resposta ao degrau do CBRN utilizando o método de Zadeh.	107

## PREFÁCIO

Em 1965 Lofit A. Zadeh estabeleceu os fundamentos da teoria dos conjuntos nebulosos com a publicação do trabalho "Fuzzy Sets" ([Zade65]). Desde então, esta disciplina tem tido grandes avanços e uma crescente aplicação nas mais diversas áreas: reconhecimento de padrões, sistemas especialistas, visão, otimização de sistemas, linguagem de consulta para comunicação homem-máquina, controle de sistemas entre outros. Este trabalho estuda uma destas áreas em especial: os controladores industriais.

## ALGUNS COMENTÁRIOS SOBRE A TEORIA DA LÓGICA E DOS CONJUNTOS NEBULOSOS

A teoria clássica dos conjuntos é baseada na lógica bivalor, a qual permite a uma proposição possuir apenas um dos valores: falso ou verdadeiro. Esta abordagem não nos permite flexibilidade suficiente para representar conceitos que possuam sentido vago ou de senso comum. "Nós vemos os objetos em tons de cinza, não apenas em preto e branco" ([Zade84b]).

A idéia chave na abordagem utilizando os conjuntos nebulosos é associar a cada elemento de um dado conjunto um grau de pertinência a este conjunto, representado neste texto como  $\mu$ :

$$\mu \longrightarrow [0, 1] .$$

A diferença básica entre os conjuntos ordinários e os

nebulosos reside em que no primeiro  $\mu$  toma valores no conjunto  $\{0, 1\}$ , ou seja, uma proposição é falsa ou verdadeira, ou um elemento pertence ou não a um conjunto. Já nos conjuntos nebulosos  $\mu$  toma valores no intervalo fechado  $[0, 1]$ . Então, uma proposição não possui apenas os valores falso ou verdadeiro, mas pode ser, também, parcialmente verdadeira dependendo do valor de  $\mu$ .

Como exemplo do exposto anteriormente consideremos o conjunto nebuloso "alto". Os elementos deste conjunto são homens, e seus graus de pertinências,  $\mu$ , dependem de suas alturas. Por exemplo: um homem com 1,20m de altura pode ter  $\mu = 0.0$ , um homem com 1,40m pode ter  $\mu = 0.4$  e um homem com 1m90 pode ter  $\mu = 1.0$ .

Os conjuntos ordinários possuem um preciso significado, tendo uma transição abrupta entre seus atributos: pertence e não-pertence. Os conjuntos nebulosos, como visto anteriormente, permitem uma transição gradual entre estes dois atributos. Usando esta idéia, informações qualitativas e vagas podem ser manipuladas adequadamente.

## A TEORIA DOS CONJUNTOS NEBULOSOS E SUA APLICAÇÃO AOS CONTROLADORES INDUSTRIAIS

O controle de processos industriais é baseado em uma bem desenvolvida teoria: controle de sistemas. Esta teoria pode trabalhar com vários problemas complicados do cotidiano de um sistema de processos. Por exemplo: sistemas lineares multivariáveis podem ser estudados usando-se a conhecida técnica do lugar das raízes; o problema de estimar o estado de um sistema que sofre perturbações de ruído pode ser solucionado usando-se o filtro de Kalman. De forma similar, o princípio do máximo pode ser utilizado para resolver problemas de controle ótimo. Certos tipos de sistemas

não-lineares podem ser solucionados usando-se de técnicas estocásticas. Todas estas técnicas têm sido utilizadas na prática com relativo sucesso. Entretanto, estas técnicas têm um aspecto básico em comum: fazem uso de um modelo real e acurado do processo.

Em muitas situações este modelo matemático é impossível de se obter: alguns não permitem medidas precisas das condições que o envolvem, como os processos de reações químicas, outros não são factíveis de obter devido a sua complexidade, como o processo existente em um forno de fabricação de cimento ou em um alto-forno de uma siderurgica.

Além do exposto, muitos tipos de processos têm uma grande quantidade de informações disponíveis apenas de forma qualitativa e não quantitativa (numérica), como exemplo: informações sobre o desempenho do sistema.

O fato é que estes processos existem e são controlados. Isto é feito por meio da utilização de experientes operadores humanos, os quais descrevem a sua ação de controle através de regras heurísticas expressas na forma lingüística, vaga e imprecisa.

Vejamos como exemplo o processo de fabricação de cimento. Um forno de cimento é um enorme recipiente, em geral cilíndrico e com algumas dezenas de metros de comprimento, no qual são depositados pedras e argila, as quais reagirão com outros componentes em uma temperatura entre  $1000^{\circ}$  e  $1400^{\circ}$ C. O processo é difícil de controlar porque existem apenas alguns poucos estados possíveis de medidas no forno e devido à complexidade da reação química. Entretanto, um experiente operador pode controlar o sistema e, com razoável sucesso, maximizar a produção, minimizar o consumo de combustível e manter o torque de giro do forno em limites aceitáveis.

O operador monitora este sistema utilizando-se de quatro variáveis: a porcentagem de oxigênio, o conteúdo de cal (uma medida da qualidade do produto), a mudança no torque (uma indicação da temperatura do forno) e o conteúdo de carbono existente nos gases expelidos. O operador aplica um conjunto típico de 40-50 regras para manter o desempenho do sistema em

um nível desejado. Uma regra típica é: "Se a porcentagem de oxigênio é alta e o teor de cal e o torque estão normais, diminua o fluxo de gás e reduza um pouco a taxa de combustível" ([Zade84a]).

Além dos problemas mencionados, o operador muitas vezes entra em conflito sobre qual regra irá aplicar. Estes fatos possibilitam-nos a abordagem por meio dos conjuntos nebulosos. Estes sistemas simulam a ação de experientes operadores na tarefa de controlar este tipo de processo, ou seja, eles são sistemas especialistas utilizando de raciocínio aproximado.

Os sistemas que utilizam esta abordagem são de fácil implementação, possuem algoritmos simples e rápidos, e não enfrentam problemas de convergência numérica ao contrário dos sistemas convencionais.

## HISTÓRICO SOBRE A EVOLUÇÃO DOS CONTROLADORES BASEADOS EM REGRAS NEBULOSAS

O primeiro artigo descrevendo uma aplicação da teoria dos conjuntos nebulosos ao problema de controle de sistemas foi publicado por Ebrahim Mamdani, Queen Mary College em Londres. Os resultados de sua pesquisa foram publicados no trabalho "Application of Fuzzy Algorithms for Control of Simple Dynamic Plant" ([Mamd74]).

Neste trabalho Mamdani desenvolveu em laboratório um sistema de controle para uma máquina de vapor. O sistema tinha como tarefa controlar o conjunto de válvulas regulando o fluxo de vapor, a temperatura e a velocidade da máquina. Mamdani implementou um conjunto de 24 regras lingüísticas de controle na forma: "SE erro está alto E variação do erro é positiva ENTÃO diminua a pressão". Este conjunto de regras traduzia a experiência do operador em controlar o sistema.

Este sistema desenvolvido por Mamdani foi comparado com um sistema de controle digital direto (c.d.d) obtendo um melhor desempenho.

Após a publicação deste trabalho pioneiro, houve um forte interesse pela área e outros trabalhos começaram a surgir. W. Kickert e H. Van Nauta Lemke, pesquisadores do Lab. de Controle de Sistemas da Universidade de Amsterdan, estudaram a aplicação dos conjuntos nebulosos no controle de temperatura em uma coluna de água e os resultados foram publicados em "Application of a Fuzzy Controller in a Warm Water Plant" ([Naut76]).

Além do interesse acadêmico várias aplicações industriais foram realizadas. B. Rutherford e R. Tong desenvolveram, para a British Steel Corporation, sistemas para controle de processos de fabricação de aços. P. King desenvolveu sistemas para controle de processos em fornos de fabricação de cimento, seus resultados aparecem em "An Analysis of Human Decision-Making in Cement Kiln Control and Implications for Automation" ([King80]).

Recentemente, J. Bernard em "Use of a Rule-Based Systems for Process Control" ([Bern88]) descreve a aplicação dos conjuntos nebulosos no controle do sistema de potência de um reator nuclear (MITRR - II) no M.I.T. - U.S.A.. Também, pesquisadores japoneses, com destaque para M. Sugeno do Instituto de Tecnologia de Tóquio, vem desenvolvendo algumas aplicações inovadoras e importantes. Entre elas: sistemas de controle automático de direção de automóveis e o Fuzzy Guzzy para controle do metrô de Tóquio ([Suge85b]).

Para informação adicionais sobre novas aplicações consulte-se publicações da área: "IEEE Conference on Decision and Control", "Congress on Instrumentation and Automation - ISA" da Instrumentation for Society American e "Fuzzy Sets and Systems" entre outros.

## ORGANIZAÇÃO DESTE TEXTO

Este trabalho é um estudo sobre a aplicação da teoria dos conjuntos nebulosos ao controle de sistemas. Os controladores que utilizam esta abordagem são denominados controladores baseados em regras nebulosas - CBRN. O nosso estudo é centrado em dois pontos básicos: a análise e o projeto destes controladores. Na análise estudaremos a arquitetura do CBRN e a função de cada um dos seus módulos. A parte do texto referente a projeto descreve a experiência de alguns pesquisadores e a do próprio autor no desenvolvimento destes sistemas.

O texto está estruturado em três partes. Na primeira, Capítulos 1 e 2, são estudados alguns fundamentos matemáticos e cognitivos básicos. O Capítulo 1 descreve o que são conjuntos nebulosos e algumas de suas operações, a lógica nebulosa e seu sistema de inferência. O Capítulo 2 é destinado à discussão sobre alguns aspectos da representação do conhecimento e sua implementação utilizando a lógica nebulosa.

Na segunda parte do texto; Capítulos 3, 4 e 5, os CBRN's são estudados atuando no processo de forma análoga à dos controladores digitais diretos. No Capítulo 3 é apresentada a arquitetura do CBRN, discutida a função de cada um dos seus módulos funcionais e seu algoritmo de inferência. O Capítulo 4 analisa um modelo heurístico do CBRN e estabelece uma relação entre ele e o controlador PI. O Capítulo 5 finaliza esta parte discutindo alguns pontos básicos a serem considerados em um projeto. A discussão é ilustrada com a apresentação de resultados, obtidos em simulação, da implementação do CBRN controlando um processo simples. Estes resultados são comparados com os obtidos por um controlador PI, o qual é bastante usado na indústria.

Na última parte, o CBRN é estudado atuando como um sistema de supervisão de um controlador PI. Sua função é corrigir o controlador PI, mantendo o sistema com um bom

desempenho. O conjunto de regras do supervisor é baseado na experiência de um operador no ajuste do controlador PI. Aqui, também, é apresentado um algoritmo de inferência que possui a capacidade de incorporar dúvidas do operador sobre qual ação de controle tomar. Este algoritmo é uma extensão do apresentado no Capítulo 3.

Apesar dos CBRN serem utilizados para controlar processos complexos, os exemplos aqui estudados são relativamente simples. Isto deve-se à não disponibilidade de processos desta natureza com especialistas em nosso ambiente de trabalho. Além disto, a simulação de processos complexos levaria a um excessivo consumo de tempo para o desenvolvimento das regras heurísticas de controle. Desta forma, o autor optou por ilustrar as aplicações com sistemas relativamente simples, o que não leva a perda de generalidade.

Ao final de cada uma das três partes deste texto é apresentado um item de conclusão, no qual discute-se os resultados apresentados, vantagens na utilização desta abordagem, limitações encontradas e sugestões para superá-las. Por último são apresentadas as conclusões gerais sobre o trabalho desenvolvido.

## PARTE I

### FUNDAMENTOS TEÓRICOS

#### CAPÍTULO 1

#### INTRODUÇÃO AOS CONJUNTOS E A LÓGICA NEBULOSA

Este capítulo tem por finalidade apresentar ao leitor uma introdução à teoria dos conjuntos e da lógica nebulosa, teoria que vem sendo cada vez mais utilizada para descrever e analisar sistemas complexos.

A representação dos conjuntos pode ser classificada de duas formas: os representados por funções matemáticas e os que são constituídos por um grupo discreto de elementos descrevendo um dado universo; aqui, apenas trataremos desta última representação. Os elementos de um conjunto serão representados através de um par ordenado (elemento, pertinência), sendo que a forma como a pertinência é definida irá nos dizer se o conjunto é do tipo nebuloso ou não.

A primeira parte deste capítulo destina-se a introduzir ao leitor alguns dos conceitos fundamentais desta teoria bem como descrever algumas operações básicas com os conjuntos nebulosos.

As seções seguintes mostram a lógica nebulosa apresentando inicialmente sua definição e estrutura sintática. O capítulo é finalizado descrevendo algumas operações básicas e a regra de inferência da composição, a qual é o principal método de inferência utilizado neste trabalho. O leitor poderá notar que muitas das operações

descritas nesta seção são derivadas ou tomadas das operações sobre os conjuntos nebulosos.

## 1.1 DEFINIÇÃO DE CONJUNTOS NEBULOSOS

A teoria dos conjuntos nebulosos foi originalmente proposta por Lofti Zadeh ([Zade65]) como uma forma alternativa de se modelar sistemas complexos. Segundo Zadeh, estes sistemas podem ser melhor estudados descrevendo o seu funcionamento através de um conjunto de regras (vide Capítulo 2), ao invés de equações diferenciais ([Zade73]).

A abordagem usada pela teoria dos conjuntos nebulosos baseia-se na suposição de que os elementos-chaves no pensamento humano não são objetos que possuem uma classificação exata, mas sim conjuntos com fronteiras difusas, isto é, classes de objetos nos quais a transição de um estado para outro é gradual e não abrupta.

Vamos agora definir uma função de pertinência  $\mu$ , a qual toma valores no conjunto  $\{0,1\}$ , associada ao símbolo de pertinência  $\in$ , dos conjuntos ordinários, e depois vamos formalmente definir um conjunto nebuloso reescrevendo o domínio de  $\mu$ .

Seja  $H$  um conjunto e  $A$  um subconjunto de  $H$ :

$$A \subset H .$$

Tomemos um elemento  $\{a\}$  pertencente ao conjunto  $A$ :

$$a \in A \text{ e } A \subset H .$$

O símbolo  $\in$  é usado para indicar que  $\{a\}$  é um elemento de  $A$ ; sua negação é o símbolo  $\notin$ . Podemos então usar uma função de pertinência binária que expresse o fato de  $\{a\}$  pertencer

ou não ao conjunto A:

$$\mu_A(u) = \begin{cases} 1 & \text{se } a \in A \\ 0 & \text{se } a \notin A . \end{cases}$$

Aqui  $\mu_A(a)$  toma valores no conjunto M:

$$\mu_A(a) : u \rightarrow M ,$$

onde:

$$M = \{ 0, 1 \} .$$

Podemos definir um conjunto nebuloso redefinindo M, tomando-o não como um conjunto, mas como um intervalo. Assim um conjunto nebuloso é um grupamento de pares ordenados ([Kauf76], [Nego78a]):

$$\{(a, \mu_A(a)), \forall a \in A \} ,$$

onde  $\mu_A(a)$  é sua função de pertinência:

$$\mu_A(a) : A \rightarrow M ,$$

onde:

$$M = [ 0, 1 ] .$$

Ou seja,  $\mu_A(a)$  toma valores em  $0 \leq \mu_A(a) \leq 1$ .

De agora em diante, neste texto usaremos as letras A, B, C etc para representar os conjuntos nebulosos.

## 1.2 ALGUMAS OPERAÇÕES BÁSICAS COM CONJUNTOS NEBULOSOS

Dois operadores serão extensamente usados ao longo deste trabalho:  $\vee$  como o operador max (dados dois números retorna o maior deles) e  $\wedge$  como o operador min (dados dois números retorna o menor deles). Por questão de simplicidade nos exemplos a seguir omitiremos a representação dos elementos preservando apenas suas pertinências. Tendo definido o que são conjuntos nebulosos vamos estabelecer agora as operações básicas executadas sobre eles.

Seja  $U$  uma coleção de objetos denotados genericamente por  $\langle u \rangle$ , sendo  $U$  denominado de universo do discurso ou domínio. Seja  $\underline{A}$ ,  $\underline{B}$ ,  $\underline{C}$  e  $\underline{H}$  conjuntos nebulosos com domínio em  $U$ . Em todos os exemplos descritos a seguir os conjuntos  $\underline{A}$ ,  $\underline{B}$ ,  $\underline{C}$  e  $\underline{H}$  possuem quatro elementos.

### a) Inclusão:

Seja  $\underline{H}$  um conjunto nebuloso com  $\underline{A} \subset \underline{H}$  e  $\underline{B} \subset \underline{H}$ , dizemos que o conjunto nebuloso  $\underline{B}$  inclui  $\underline{A}$  se:

$$\{ \forall u \in U \mid \mu_{\underline{A}}(u) \leq \mu_{\underline{B}}(u) \} .$$

Exemplo:

Sejam  $\underline{A}$ ,  $\underline{B}$  e  $\underline{C}$  três conjuntos nebulosos:

$$\underline{A} = \{ 0.1, 0.6, 0.3, 0.0 \} ,$$

$$\underline{B} = \{ 0.3, 0.7, 0.3, 0.1 \} ,$$

$$\underline{C} = \{ 0.5, 0.4, 0.2, 0.0 \} .$$

Podemos dizer então que  $\underline{A} \subset \underline{B}$  mas é falso afirmar que  $\underline{C} \subset \underline{A}$  ou  $\underline{C} \subset \underline{B}$ .

### b) Igualdade:

Seja  $\underline{H}$  um conjunto nebuloso com  $\underline{A} \subset \underline{H}$  e  $\underline{B} \subset \underline{H}$ , dizemos

que dois conjuntos nebulosos  $\underline{A}$  e  $\underline{B}$  são iguais se e somente se:

$$\{\forall u \in U \mid \mu_{\underline{A}}(u) = \mu_{\underline{B}}(u)\}.$$

Exemplo:

Sejam  $\underline{A}$ ,  $\underline{B}$  e  $\underline{C}$  três conjuntos nebulosos:

$$\underline{A} = \{0.1, 0.6, 0.3, 0.0\},$$

$$\underline{B} = \{0.1, 0.6, 0.3, 0.0\},$$

$$\underline{C} = \{0.5, 0.4, 0.2, 0.0\}.$$

Podemos dizer então que  $\underline{A} = \underline{B}$  mas é falso afirmar que  $\underline{C} = \underline{A}$  ou  $\underline{C} = \underline{B}$ .

c) Complementação:

Seja  $\underline{H}$  um conjunto nebuloso com  $\underline{A} \subset \underline{H}$  e  $\underline{B} \subset \underline{H}$ , dizemos que um conjunto é complemento de outro se:

$$\{\forall u \in U \mid \mu_{\underline{B}}(u) = 1 - \mu_{\underline{A}}(u)\}.$$

Exemplo:

Seja  $\underline{A}$  um conjunto nebuloso:

$$\underline{A} = \{0.1, 0.6, 0.3, 0.0\},$$

o conjunto nebuloso  $\underline{B}$  que é o complemento de  $\underline{A}$ , ou seja  $\neg \underline{A}$ , é:

$$\underline{B} = \{0.9, 0.4, 0.7, 1.0\}.$$

d) Interseção:

Seja  $\underline{H}$  um conjunto nebuloso com  $\underline{A} \subset \underline{H}$  e  $\underline{B} \subset \underline{H}$ , dizemos que a interseção de dois conjuntos é definido por:

$$\{ \forall u \in U \mid \mu_{\underline{A} \cap \underline{B}}(u) = \wedge [\mu_{\underline{A}}(u); \mu_{\underline{B}}(u)] \} .$$

Exemplo:

Sejam  $\underline{A}$  e  $\underline{B}$  dois conjuntos nebulosos:

$$\underline{A} = \{0.1, 0.6, 0.3, 0.0\} ,$$

$$\underline{B} = \{0.3, 0.7, 0.3, 0.1\} .$$

A intersecção de  $\underline{A}$  e  $\underline{B}$  é dada por:

$$\underline{A} \cap \underline{B} = \{0.1, 0.6, 0.3, 0.0\} .$$

e) União:

Seja  $\underline{H}$  um conjunto nebuloso com  $\underline{A} \subset \underline{H}$  e  $\underline{B} \subset \underline{H}$ , dizemos que a união entre dois conjuntos nebulosos é definida como:

$$\{ \forall u \in U \mid \mu_{\underline{A} \cup \underline{B}}(u) = \vee [\mu_{\underline{A}}(u); \mu_{\underline{B}}(u)] \} .$$

Exemplo:

Sejam  $\underline{A}$  e  $\underline{B}$  dois conjuntos nebulosos:

$$\underline{A} = \{0.1, 0.6, 0.3, 0.0\} ,$$

$$\underline{B} = \{0.3, 0.7, 0.3, 0.1\} .$$

A união de  $\underline{A}$  e  $\underline{B}$  é dada por:

$$\underline{A} \cup \underline{B} = \{0.3, 0.7, 0.3, 0.1\} .$$

f) Relação:

Uma relação nebulosa entre dois conjuntos  $\underline{A}$  e  $\underline{B}$ , com respectivo domínio em  $U$  e  $V$ , é um conjunto nebuloso  $R$  formado pelo produto cartesiano  $\underline{A} \times \underline{B}$  na forma de uma coleção de pares ordenados  $(u, v)$ ,  $u \in U$  e  $v \in V$ , na forma:

$$\{ \forall (u, v) \in U, V \mid \mu_R(u, v) = \underline{A} \times \underline{B} \}$$

onde  $\mu_R(u,v)$  é dado por:

$$\mu_R(u,v) = \bigwedge_u [\mu_A(u); \mu_B(v)] .$$

Exemplo:

Sejam  $A$  e  $B$  dois conjuntos nebulosos:

$$A = \{0.1, 0.6, 0.3, 0.0\} ,$$

$$B = \{0.3, 0.7, 0.3, 0.1\} .$$

A relação  $R$  entre dois conjuntos  $A$  e  $B$  é:

$$\mu_{A \times B}(u,v) = \begin{pmatrix} 0.1 & 0.1 & 0.1 & 0.1 \\ 0.3 & 0.6 & 0.3 & 0.1 \\ 0.3 & 0.3 & 0.3 & 0.1 \\ 0.0 & 0.0 & 0.0 & 0.0 \end{pmatrix}$$

### 13 DEFINIÇÃO DE LÓGICA NEBULOSA

A lógica nebulosa é um sistema formal baseado na teoria dos conjuntos nebulosos, o que permite a representação de informações de natureza vaga e imprecisa. Como todo sistema de dedução formal ela possui um conjunto de regras de inferência que possibilita realizar deduções sobre um conjunto de informações.

Quando o conjunto de regras de inferência da lógica nebulosa manipula informações de natureza vaga e imprecisa, este conjunto de regras de representação e dedução é denominado de sistema de raciocínio aproximado. Este sistema faz uso de variáveis lingüísticas (vide Capítulo 2) ao invés de variáveis numéricas, as primeiras podem ser interpretadas como adjetivos dados aos elementos do discurso, em oposição a sua quantificação exata, como no exemplo ([Bald78]):

variáveis numéricas:

José tem 1.90 metros.

variáveis lingüísticas:

José é alto<sub>( $\mu = k$ )</sub> ,

onde alto é uma variável lingüística e  $\mu$  é a sua função de pertinência. Esta variável pode referir-se a diferentes indivíduos; neste caso deverá sofrer uma nova restrição para cada novo elemento. A restrição é dada por  $\mu$ , e assim, alto pode ser usado para classificar vários elementos diferentes.

Define-se a lógica nebulosa como um sistema de dedução formal através da especificação de um conjunto de regras sintáticas para a representação de informações e realização de inferências.

Uma expressão na lógica nebulosa pode ser vista como uma proposição  $P$  representada em linguagem natural ou em uma linguagem artificial; esta proposição possui restrições que são dadas por  $\mu_1, \dots, \mu_k$ , através dos diferentes conjuntos nebulosos existentes. Como no exemplo anterior onde  $\mu$  restringe o termo "alto" na proposição "José é alto" com o valor  $\mu = k$ , sendo que em geral as restrições às variáveis são estabelecidas de forma implícita ([Nego85]).

Supondo-se que o conhecimento humano possa ser expresso sob a forma de regras de produção ([Clan83]), a forma canônica de  $P$  é dada por ([Zade83b]):

$$P \equiv x \text{ é } A,$$

onde  $A$  é o valor nebuloso assumido pelo termo  $x$ , o qual é uma variável lingüística, e o símbolo  $\equiv$  significa "deve ser interpretado como" ou "por definição".

A abordagem utilizada como forma de definir a Lógica Nebulosa tem sua validade restrita ao escopo deste texto. O alfabeto utilizado será composto de:

i) símbolos lógicos:

- $\neg$  negação,
- $\wedge$  conjunção,
- $\vee$  disjunção,
- $\rightarrow$  implicação;

ii) símbolos não-lógicos

- $x$  variável lingüística (será formalmente definida no Capítulo 2),
- $A$  valor lingüístico assumido por  $x$ ,
- $\epsilon$  verbo da proposição nebulosa,
- $\mu$  restrição ao valor lingüístico  $A$  (expressa implicitamente).

Uma frase nebulosa é:

i) uma proposição  $P$ .

ii) se  $P_1$  e  $P_2$  são frases nebulosas, então  $\neg P_1$ ,  $P_1 \vee P_2$ ,  $P_1 \wedge P_2$  e  $P_1 \rightarrow P_2$  são também frases nebulosas.

## 1.4 OPERAÇÕES BÁSICAS DA LÓGICA NEBULOSA

Uma vez definida a estrutura sintática do nosso sistema apresentaremos algumas operações lógicas e regras de inferências. O leitor deverá observar que as operações definidas a seguir são derivadas daquelas definições dadas nos itens anteriores.

Nos exemplos a seguir os conjuntos nebulosos  $\underline{A}$ ,  $\underline{B}$  e  $\underline{C}$  referem-se respectivamente aos valores lingüísticos  $A$ ,  $B$  e  $C$ .

a) Regra de Conjunção —  $(P_1 \wedge P_2)$

Seja  $\underline{H}$  um conjunto nebuloso com  $A \in \underline{H}$  e  $B \in \underline{H}$ , a operação

de conjunção é definida como:

$$\begin{array}{l} P_1 = x \text{ é } A \\ P_2 = x \text{ é } B \\ \hline (P_1 \wedge P_2) = x \text{ é } [A \cap B] \end{array}$$

onde  $A \cap B$  é a interseção entre os conjuntos nebulosos  $A$  e  $B$  definida por:

$$\mu_{A \cap B}(u) = \wedge [\mu_A(u); \mu_B(u)] \quad u \in U$$

b) Regra de Disjunção —  $(P_1 \vee P_2)$

Seja  $H$  um conjunto nebuloso com  $A \subset H$  e  $B \subset H$ , a operação de disjunção é definida por:

$$\begin{array}{l} P_1 = x \text{ é } A \\ P_2 = x \text{ é } B \\ \hline (P_1 \vee P_2) = x \text{ é } [A \cup B] \end{array}$$

onde  $A \cup B$  é a união entre os conjuntos nebulosos  $A$  e  $B$  definida por:

$$\mu_{A \cup B}(u) = \vee [\mu_A(u); \mu_B(u)] \quad u \in U$$

c) Regra de Negação —  $(\neg P)$

Seja  $H$  um conjunto nebuloso com  $A \subset H$ , a operação de negação é definida como:

$$\begin{array}{l} \neg (u \text{ é } A) \\ \hline x \text{ é } [\neg A] \end{array}$$

onde  $\neg \underline{A}$  é a negação da classificação; a negação não é tomada sobre a variável  $x$  mas sim sob sua classificação, dada por:

$$\mu_{\neg \underline{A}}(u) = 1 - \mu_{\underline{A}}(u) .$$

$$u \in U$$

d) Regra de Implicação —  $(P_1 \rightarrow P_2)$

Seja  $\underline{H}$  um conjunto nebuloso com  $\underline{A} \subset \underline{H}$  e  $\underline{B} \subset \underline{H}$ , a implicação é definida por:

$$P_1 = x \text{ é } A$$

$$P_2 = y \text{ é } B$$

---


$$(P_1 \rightarrow P_2) = [x, y] \text{ é } [\underline{A} \times \underline{B}]$$

A regra da implicação estabelece uma dependência para a satisfação de  $P_2$  em termos de  $P_1$ . Esta dependência é estabelecida pela função  $\mu_{\underline{A} \times \underline{B}}(u, v)$  onde  $(u, v)$  indicam quais são as variáveis usadas na implicação.

$$\mu_{\underline{A} \times \underline{B}}(u, v) = \wedge [\mu_{\underline{A}}(u); \mu_{\underline{B}}(v)] ,$$

$$u \in U, v \in V$$

que como foi visto anteriormente descreve a relação existente entre dois conjuntos nebulosos.

Exemplo

Seja a implicação  $P_1 \rightarrow P_2$ , e  $\underline{A}$  e  $\underline{B}$  os respectivos conjuntos nebulosos assumidos por  $x$  e  $y$  em  $P_1$  e  $P_2$ :

$$\underline{A} = \{1.0, 0.8\} ,$$

$$\underline{B} = \{0.6, 0.9, 1.0\} ,$$

como a implicação é dada pelo produto cartesiano temos:

$$P_1 \rightarrow P_2 = \begin{pmatrix} 0.6 & 0.9 & 1.0 \\ 0.6 & 0.8 & 0.8 \end{pmatrix}$$

Existem várias abordagens propostas na literatura sobre como implementar a operação de implicação ([Mizu78]), a escolhida neste trabalho foi o método de Mamdani ([Mamd77a]). O porquê desta escolha é apresentado no Apêndice B.

e) Regra da Composição —  $\{ P_1' \cdot (P_1 \rightarrow P_2) \}$

Seja  $H$  um conjunto nebuloso com  $\underline{A} \subset H$  e  $\underline{B} \subset H$ , a regra de inferência da composição é definida por:

$$\frac{\begin{array}{l} P_1' = \quad x \quad \text{é } A' \\ P_1 \rightarrow P_2 = \quad [x,y] \quad \text{é } R \end{array}}{P_2' = \quad y \text{ é } [\underline{A}' \cdot R]}$$

O símbolo  $\cdot$  expressa a composição entre a variável  $x$  e uma relação binária  $R$ , obtendo-se então a variável  $y$ , onde:

$$\mu_{\underline{A}' \cdot R}(v) = \bigvee_u \wedge [\mu_{\underline{A}'}(u); \mu_R(u,v)] ,$$

onde  $R$  é a implicação vista anteriormente:

$$\mu_R(u,v) = \wedge_v [\mu_{\underline{A}}(u); \mu_{\underline{B}}(v)] ,$$

então:

$$\mu_{\underline{B}}(v) = \mu_{\underline{A}' \cdot R}(v) ,$$

substituindo temos:

$$\mu_{\underline{B}}(v) = \bigvee_u [ \wedge [\mu_{\underline{A}}(u); \mu_{\underline{B}}(v)] ] .$$

$u \in U, v \in V$

Na expressão a seguir fazemos uso de dois termos diferentes:  $P_1'$  e  $P_2'$ , os quais são respectivamente o antecedente ocorrido e o conseqüente deduzido, e  $P_1 \rightarrow P_2$  é a relação existente entre ambos os conjuntos nebulosos. Deve-se notar dois aspectos na regra da composição.

Em primeiro lugar, quando  $P_1$  e  $P_1'$  são iguais obtém-se o próprio  $P_2$  atuando, a regra de inferência da composição, como uma identidade:

$$P_2 = P_1 \circ (P_1 \rightarrow P_2) .$$

Exemplo:

Seja  $H$  um conjunto nebuloso com  $A \subset H$  e  $B \subset H$ , tendo os seguintes valores:

$$\underline{A} = \{0.6, 0.4, 0.9\} ,$$

$$\underline{B} = \{0.3, 0.6, 0.7\} ,$$

e sua relação é:

$$P_1 \rightarrow P_2 = R_{\underline{AB}} = \begin{pmatrix} 0.3 & 0.6 & 0.6 \\ 0.3 & 0.4 & 0.4 \\ 0.3 & 0.6 & 0.7 \end{pmatrix}$$

sendo  $P_1 = P_1'$  e aplicando a regra da composição temos:

$$P_2 = \begin{pmatrix} 0.6 \\ 0.4 \\ 0.9 \end{pmatrix} \circ \begin{pmatrix} 0.3 & 0.6 & 0.6 \\ 0.3 & 0.4 & 0.4 \\ 0.3 & 0.6 & 0.7 \end{pmatrix} = \begin{pmatrix} 0.3 \\ 0.6 \\ 0.7 \end{pmatrix} .$$

O segundo aspecto é que quando  $P_1$  é uma aproximação de  $P_1'$  temos que  $P_2'$ , obtido pela regra da composição:

$$P_2' = P_1' \circ (P_1 \rightarrow P_2) ,$$

é também uma aproximação do termo  $P_2$  utilizado na implicação  $P_1 \rightarrow P_2$ .

Exemplo:

Seja  $H$  um conjunto nebuloso com  $A \subset H$ ,  $A' \subset H$  e  $B \subset H$ .  
Tendo  $A$  e  $B$  os valores do exemplo anterior e  $A'$  os valores:

$$A' = \{0.5 \ 0.4 \ 0.6\}.$$

A inferência  $P_2' = P_1' \cdot (P_1 \rightarrow P_2)$ , resulta em:

$$P_2' = \begin{pmatrix} 0.5 \\ 0.4 \\ 0.6 \end{pmatrix} \cdot \begin{pmatrix} 0.3 & 0.6 & 0.6 \\ 0.3 & 0.4 & 0.4 \\ 0.3 & 0.6 & 0.7 \end{pmatrix} = \begin{pmatrix} 0.3 \\ 0.6 \\ 0.6 \end{pmatrix},$$

sendo que o termo inferido  $B'$  é uma aproximação de  $B$ . Assim pode-se ver a regra da composição como uma extensão aproximada do *modus ponens* ([Lee00b]).

## CAPITULO 2

### ALGUNS ASPECTOS A SEREM CONSIDERADOS SOBRE SISTEMAS BASEADOS EM CONHECIMENTO

Este capítulo é dedicado a discutir alguns conceitos importantes sobre sistemas baseados em conhecimento. Começaremos por caracterizar base de conhecimento como um modelo a ser aplicado em um determinado domínio de um problema. Do ponto de vista da ação, estes modelos guardam diferenças básicas com os outros tipos de modelos frequentemente usados: os modelos tradicionais descrevem "o que" acontece quando interagem com o meio, mas não "o porque" e nem "como" acontece a interação com o meio. Este tipo de sistema recebe a denominação de sistema qualitativo. Esta abordagem vem ganhando uma importância crescente na pesquisa sobre Inteligência Artificial e será extensamente usada ao longo deste texto.

Como todo tipo de modelo este é, também, uma abstração do processo de solução de problemas, assim alguns pontos devem ser observados na sua construção: a expressividade e a eficiência do paradigma escolhido para sua construção.

A escolha do paradigma de representação de conhecimento é, em geral, uma função do tipo de conhecimento com o qual se está trabalhando e do problema que desejamos resolver. Deste modo, discutiremos a lógica nebulosa como uma forma de

representar conhecimento de senso comum, que é o tipo de conhecimento com o qual iremos trabalhar.

O capítulo é finalizado mostrando a aplicação da lógica nebulosa na construção de um modelo qualitativo de solução de problemas, o qual recebe uma denominação especial: o modelo linguístico.

## 2.1 O CONHECIMENTO E SUA REPRESENTAÇÃO

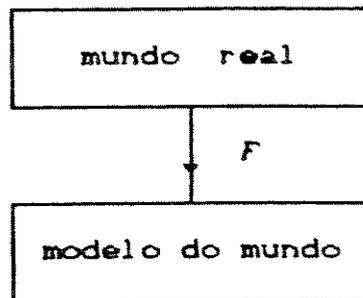
Apesar de já vir sendo estudado desde a antigüidade pelos filósofos gregos, ainda não se conseguiu definir o que seja conhecimento. A abordagem que se tem adotado é a de se caracterizar o conhecimento dentro de um universo bem definido: o domínio do problema a ser estudado.

Para que o conhecimento possa ser utilizado por uma máquina encarregada de processá-lo, ele deve ser codificado em algum tipo de estrutura de dados, sendo que vários são os tipos de estruturas de dados possíveis de se usar ([Nils82], [Rich82]). Ao codificar-se este conhecimento em uma estrutura de dados apropriada estamos construindo uma representação do conhecimento e criando um modelo do mundo real que será usado para resolver uma determinada tarefa, tal como: diagnóstico, controle, planejamento etc.

Vejamos um exemplo: ao se representar o conhecimento de um especialista em resolver uma dada tarefa, escolheu-se a lógica como paradigma, sendo o conhecimento codificado sob a forma de regras e fatos ([Kowa79]). A lógica é, então, a forma usada para representar a estrutura de dados; o conjunto de regras e fatos que formam esta estrutura de dados representam uma abstração do processo de solução de problema usado pelo especialista, ou seja, um modelo.

Assim, na Figura 2.1 podemos identificar a função  $F$  como sendo a lógica, e o conjunto de regras e fatos a imagem ao se

mapear o mundo real através da função  $F$  ([Cian80]).

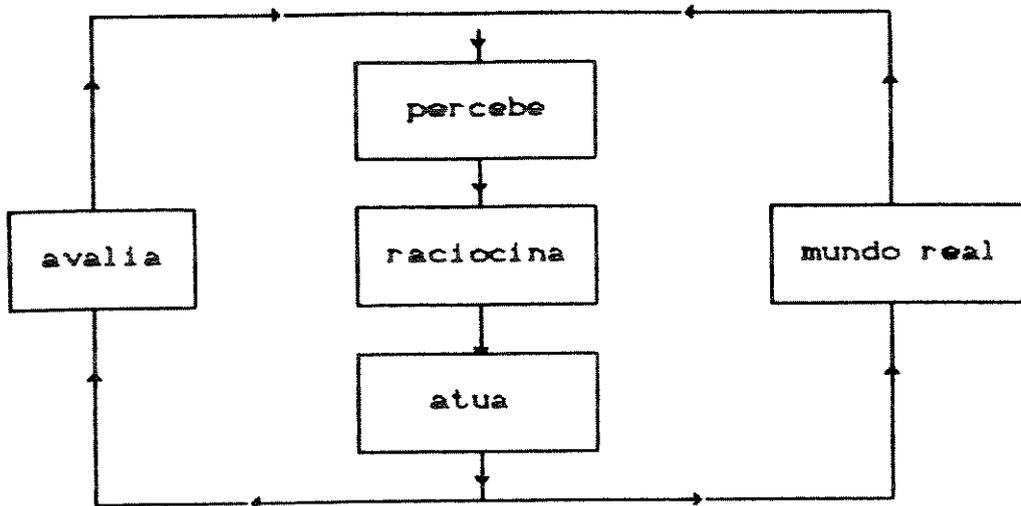


Base de conhecimento como modelo do domínio.

Figura 2.1

Sistemas baseados em conhecimento têm a sua atuação decidida sobre um modelo do mundo, e não no mundo real propriamente dito. Ao atuar sobre o mundo o sistema deverá provocar mudanças, percebê-las e, se necessário, atuar novamente sobre o mundo real construindo desta forma um processo iterativo. A Figura 2.2 mostra este processo. Vale notar novamente o fato de que a atuação dá-se sobre o mundo real mas o raciocínio é sobre um modelo deste ([Newe82]).

Este paradoxo, mundo real - modelo do mundo, obriga que o mapeamento executado pela função  $F$  seja o mais correto e amplo possível para não comprometer a ação sobre o mundo real. Aqui discutiremos apenas os aspectos relevantes na escolha da função  $F$ , a realização do mapeamento através de  $F$  será objeto do Capítulo 5.



Arquitetura de um sistema inteligente.

Figura 2.2

Dois são os fatores que devem ser observados na escolha da função  $F$ . O primeiro aspecto é a expressividade do paradigma de representação de conhecimento escolhido e relaciona-se com o seu poder de representação, isto é, o que a representação pode expressar.

O segundo é a eficiência notacional do paradigma escolhido e diz respeito à forma e à estrutura da representação, bem como ao impacto que esta estrutura causa nas operações do sistema, ou seja, a eficiência computacional das várias formas de inferência.

Como exemplo vejamos a escolha de se representar conhecimento de origem heurística em "frames" ou regras. Ambas as formas possuem a mesma eficiência notacional, ou seja, ambos fornecem condições semelhantes de se representar este tipo de conhecimento. Mas ao se examinar a expressividade das representações, concluímos que a representação sob a forma de regras é a mais adequada pois a codificação e visualização do conhecimento neste paradigma realiza-se de forma mais fácil e agradável.

Até agora temos discutido aspectos relevantes à

representação de conhecimento sem nos aprofundarmos em um paradigma específico, o que será feito na seção seguinte, quando discutiremos a lógica nebulosa. O interesse neste paradigma em particular é motivado pelo tipo de conhecimento com que trabalharemos neste texto: conhecimento de senso comum. Este é, em geral, o tipo de conhecimento encontrado nos operadores de processos industriais na execução de tarefas de controle. A informação expressa por eles é de natureza vaga e imprecisa, sendo entendida corretamente apenas pelos indivíduos que partilham do mesmo meio e perdendo o significado fora dele. Estas informações são expressas pelos operadores, que neste trabalho são os especialistas, na forma de regras ([Zade75a] e [Zade75b]):

Se

a pressão está alta

Então

abaixe um pouco a temperatura.

Veremos agora como a lógica nebulosa pode representar este tipo de conhecimento mantendo uma boa expressividade e eficiência notacional.

## 2.2 O USO DA LÓGICA NEBULOSA PARA REPRESENTAÇÃO DE SENSO COMUM

Pode-se dizer que senso comum é a caracterização feita sobre certos fatos que um certo grupo de indivíduos possuem, caracterização esta que é oriunda da cultura do grupo sobre o meio ([Sowa86]). Assim, podemos dizer que os pigmeus acham todos os homens franceses altos, já que nenhum pigmeu é maior que 1,60 m. Tal fato é senso comum entre os pigmeus mas obviamente não entre outras raças.

Expressões do tipo "João é muito inteligente", "o tempo está frio" ou "os homens franceses são altos", representam expressões do senso comum de um certo grupo de indivíduos em um determinado mundo sobre conceitos como: inteligência, temperatura e tamanho.

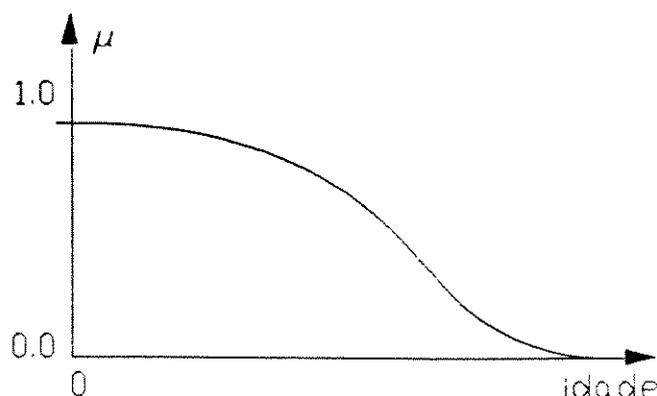
As técnicas convencionais de representação de conhecimento, como o cálculo de predicados, não são bem sucedidas na representação deste tipo de conhecimento por duas razões ([Kauf88]). A primeira é pelo fato de que, em geral, os predicados usados na construção de frases de senso comum possuem uma conotação vaga e imprecisa. Esta conotação não pode expressar-se convenientemente pela lógica tradicional através do uso de predicados cujo valor semântico se restringe a falso ou verdadeiro, mas sim através de um predicado cujo valor verdade possui uma variação gradativa, ao contrário de abrupta ([Rega78]).

Vejamus um exemplo: "Paulo é um homem alto". O significado do predicado alto é vago e impreciso, não sendo o de uma semântica bivalor, mas sim, um valor verdade que sofre restrições ( $\mu = 0.0$  até  $\mu = 1.0$ ). Desta forma o predicado alto pode ser empregado para classificar indivíduos com diferentes tamanhos, sendo que quanto maior for sua altura mais forte é o seu valor verdade, até o valor limite de  $\mu = 1.0$ . Esta abordagem para a representação de informações de senso comum guarda semelhanças com a que é utilizada por nós em nosso dia a dia.

O segundo fator é que, além dos predicados, também os quantificadores são vagos e imprecisos; isto pode ser ilustrado no exemplo: "Muitos homens franceses não são altos". O quantificador "muito" não pode ser representado pelos dois quantificadores do cálculo de predicados  $\forall$  (para todo) e  $\exists$  (existe ao menos um), pois o quantificador "muito" não possui a conotação de todos, como em "todos os homens franceses não são altos", e nem a conotação da existência de ao menos um, como em "existe ao menos um homem francês que não é alto". Pelo contrário, assume a conotação da existência de um número bem maior do que um. Como pode ser visto, o

emprego dos quantificadores da lógica tradicional leva a construção de uma frase com significado diferente da inicial.

A abordagem neste trabalho para representar conhecimento de senso comum baseia-se na criação de um conjunto de valores representando o predicado nebuloso em questão. Este conjunto sofre restrições sobre os valores que representa ([Nego85]). Pode-se ver a representação do predicado "jovem" na Figura 2.3.



Conjunto nebuloso representando o predicado "jovem".

Figura 2.3

Assim o predicado "jovem" possui uma restrição  $\mu$  conforme a idade do indivíduo. Sendo esta restrição implícita, isto possibilita o uso do predicado "jovem" na construção de várias frases para diferentes indivíduos com diferentes idades.

O conhecimento de um indivíduo sobre algo pode ser descrito por um conjunto de regras de produção do tipo ([Eric84]):

$$S_1 \rightarrow S_2 ,$$

que pode ser, por sua vez, descrito como:

$$\begin{aligned} S_1 &= x_1 \text{ é } A_1, \\ S_2 &= x_2 \text{ é } A_2, \end{aligned}$$

onde  $x_1$  e  $x_2$  são variáveis e  $A_1$  e  $A_2$  são valores nebulosos assumidos por estas variáveis ([Nego85]).

A restrição  $\mu$  é estabelecida em relação a  $A_1$  e  $A_2$ . Desta forma o conjunto de operações descritas no Capítulo 1 sobre a lógica nebulosa é possível de ser usada aqui também ([Zade83b]).

Assim, se quisermos dizer que Pedro é jovem, sendo que ele possui 10 anos, teremos:

$$S = \text{Pedro é jovem } [\mu_{\text{jovem}}(10 \text{ anos}) = 0.9].$$

Isto é dado pelo mapeamento:

$$\mu_{\text{jovem}}(\text{idade}) : \text{Idade} \rightarrow \text{Jovem}.$$

Como pode ser observado, a lógica nebulosa nos fornece uma forma simples e eficiente de representar conhecimento de senso comum.

A representação e dedução de informações em lógica nebulosa obedece ao exposto no Capítulo 1, portanto pouparei o leitor de pontos já vistos anteriormente, o que seria uma repetição desnecessária.

Neste texto não será abordado o problema dos quantificadores nebulosos, o leitor interessado deverá consultar as referências citadas ([Zade83a]).

Agora que já temos conceituado sistemas baseados em conhecimento e a utilização da lógica nebulosa para representar conhecimento de senso comum, veremos como utilizar o exposto nas seções 1 e 2 deste capítulo para construir um sistema baseado em conhecimento usando a lógica nebulosa.

## 2.3 MODELAMENTO LINGÜÍSTICO

Neste trabalho estamos interessados em construir um sistema que simule a ação de controle de um operador de processos industriais. Este sistema é, na realidade, um modelo de solução de problemas, em nosso caso o de controlar um processo ([Hunt84]).

Este modelo difere da abordagem usual que é baseada principalmente em equações e algoritmos matemáticos, por fazer uso de informações de origem heurística e, em geral, codificadas em um paradigma de representação de conhecimento, no nosso caso regras ([Clan89]). Sistemas baseados em conhecimento são modelos, tendo suas ações são baseadas em uma abstração do mundo real, fazendo parte do modelo apenas os fatos relevantes na solução do problema. Este tipo de modelo recebe o nome de modelo qualitativo ([Forb84], [Buck89]).

Modelos qualitativos possuem características que fazem seu uso vantajoso em relação aos modelos tradicionais ([Kick79]):

- i) eliminação de problemas de convergência encontrado nos algoritmos numéricos,
- ii) melhor visualização do efeito das variáveis sobre a dinâmica do sistema,
- iii) faz uso de associassões heurísticas dispensando formulações matemáticas precisas.

Na construção de nosso modelo qualitativo, devido aos motivos já expostos, usaremos a lógica nebulosa. Tais modelos recebem uma denominação especial, para diferenci-los dos demais modelos qualitativos: modelo lingüístico ([Kick78b]).

As diferenças entre estes dois tipos de modelos residem em:

- i) uso de conhecimento de natureza vaga,
- ii) uso da lógica nebulosa para representar, e
- iii) uso de processamento numérico.

O item iii mostra um fator ainda não comentado: em nosso trabalho necessitaremos de processamento numérico e não simbólico, devido à natureza do trabalho em si e ao tempo de processamento, em geral grande da computação simbólica; o leitor poderá verificar o porque destas duas afirmações ao longo deste texto.

O modelo linguístico é descrito na forma de regras de produção:

Se  
           a pressão está alta  
 Então  
           abaixe um pouco a temperatura.

onde "pressão" e "temperatura" são chamadas de variáveis linguísticas, e "baixa" e "aumentar" são os valores assumidos por estas variáveis. Assim, uma variável linguística é uma variável que tem o seu valor tomado em um dos conjuntos nebulosos do universo do discurso ([Kick78b]).

Vamos agora definir formalmente o que é uma variável linguística. Uma variável linguística pode ser definida como uma quintupla  $(\Delta, T(\Delta), U, G, M)$  em que  $\Delta$  é o nome da variável e  $T$  é o conjunto de termos que  $\Delta$  assume. Vejamos um exemplo: se definirmos idade como sendo uma variável linguística o conjunto  $T(\Delta)$  de valores assumidos pela variável idade pode ser definido como:

$T(\text{idade}) = \text{velho} \vee \text{novo} \vee \text{adolescente} \vee \text{muito velho} \dots$

O termo  $U$  é o universo do discurso, isto é, o intervalo de existência em que a variável idade toma valores. Vale notar

que o intervalo de existencia dos termos de  $T(\Delta)$  são iguais.  $G$  é a regra sintática que gera os termos em  $T(\Delta)$ , a qual definiremos ser sempre da forma:

$$G \equiv x \text{ é } A,$$

onde  $x = \Delta$  e  $A$  é um dos elementos do conjunto  $T(\Delta)$ , ou seja  $T(\Delta) = A$ . A semântica de uma variável lingüística é dada por  $M$ , a qual associa a cada valor lingüístico de  $T(\Delta)$  o seu correspondente significado,  $M(U)$ , onde  $M(U)$  denota a restrição sofrida pelo elemento  $u$ , com  $u \in U$ , em um conjunto nebuloso de  $T(\Delta)$  ([Zade75a]).

O significado de um valor lingüístico  $T(\Delta)$  é caracterizado pela função de pertinência  $\mu$ , a qual realiza o mapeamento:

$$\mu : U \rightarrow [0, 1] ;$$

o qual associa a cada elemento do discurso uma restrição. Desta forma, a restrição do predicado "jovem", mostrado na Figura 2.3, atribui um valor de  $\mu = 0.7$  para uma pessoa de 20 anos e  $\mu = 0.2$  para uma pessoa 40 anos.

Nosso modelo lingüístico é, então, um conjunto de informações, em geral de natureza heurística, codificado na forma de regras ( $G$ ), em que as variáveis lingüísticas ( $\Delta$ ) tomam valores nos conjuntos nebulosos ( $T(\Delta)$ ) através de  $M$ . A simulação do modelo é feita através da execução de um algoritmo nebuloso, ou seja, através da execução de uma sequência ordenada de regras usando as inferências descritas no Capítulo 1 ([Kick79]).

Este modelo é construído usando-se a lógica nebulosa como paradigma e oferece, no propósito deste trabalho, um ambiente mais realista e flexível para simulação do raciocínio humano que os outros paradigmas ([Zade75b]).

Nos capítulos seguintes veremos como é a arquitetura do modelo lingüístico, seu funcionamento e sua construção.

## CONCLUSÃO DA PARTE I

Os dois primeiros capítulos destinaram-se a apresentar ao leitor os principais aspectos sobre a teoria dos conjuntos e da lógica nebulosa, assim como alguns pontos importantes sobre sistemas baseados em conhecimento.

A teoria dos conjuntos e da lógica nebulosa baseia-se na premissa de que o pensamento humano não se realiza na forma da lógica bivalor verdadeiro/falso, mas sim através de uma transição gradativa entre estes dois valores.

Um dos principais problemas relativo aos sistemas baseados em conhecimento é que a sua ação é determinada em relação a um modelo do mundo real; isto obriga que o modelo represente com exatidão o mundo real. Uma das formas de tentar satisfazer esta condição é a cuidadosa escolha do paradigma de representação de conhecimento a ser adotado.

A escolha da lógica nebulosa como sendo este paradigma, baseou-se na flexibilidade para representar o tipo de conhecimento com que trabalharemos neste texto e em sua eficiência computacional.

Quando a lógica nebulosa é usada para modelar as ações de um fenômeno físico, esta técnica é denominada de modelamento lingüístico. As diversas vantagens oferecidas por esta técnica poderão ser analisadas neste texto.

## PARTE II

### ANÁLISE E SÍNTESE DE CONTROLADORES BASEADOS EM REGRAS NEBULOSAS

#### CAPÍTULO 3

#### ANÁLISE DE CONTROLADORES BASEADOS EM REGRAS NEBULOSAS

Após termos visto na Parte I deste texto, Capítulos 1 e 2, alguns fundamentos teóricos importantes, vamos passar agora à aplicação destes conceitos no desenvolvimento de um sistema baseado em conhecimento para problemas de engenharia controle.

Estes controladores são, na realidade, sistemas especialistas, os quais têm por finalidade obter uma resposta, que seria a mesma tomada por um perito no controle de um determinado processo industrial e que faça o processo funcionar com um índice de desempenho desejado. Por ser um sistema baseado em conhecimento, codificado sob a forma de regras e desempenhar uma função análoga aos controladores tradicionais, denominaremos este sistema de Controladores Baseados em Regras Nebulosas ou, em uma forma mais abreviada, CBRN.

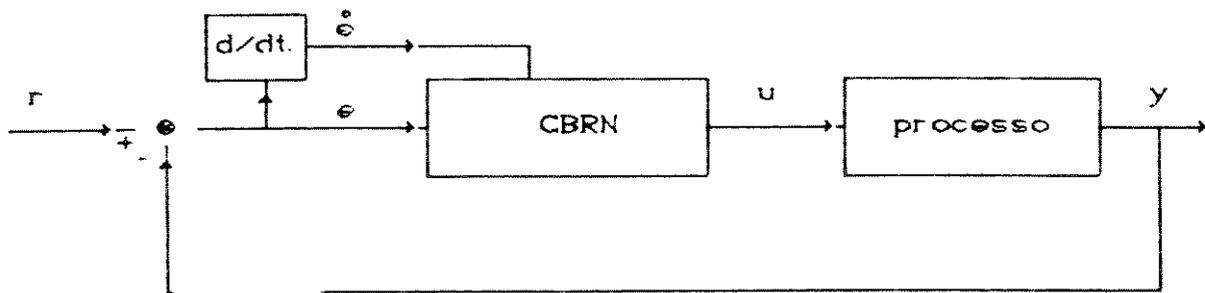
Em primeiro lugar veremos uma arquitetura geral de um sistema que faz uso de um CBRN mostrando a sua posição e função na malha de controle.

Em seguida examinaremos mais detalhadamente a arquitetura de um CBRN, onde serão descritos cada um de seus módulos e as respectivas funções desempenhadas por eles; estas descrições serão feitas por meio de comparações

funcionais com os elementos dos controladores tradicionais.

### 3.1 ARQUITETURA DE UM CONTROLADOR BASEADO EM REGRAS NEBULOSAS

Um sistema de controle que use um CBRN possui o esquema dos sistemas clássicos de controle, sendo que o CBRN está em uma posição e função semelhante aos controladores convencionais ([King77]). A Figura 3.1 mostra este esquema.



Sistema de controle utilizando um CBRN.

Figura 3.1

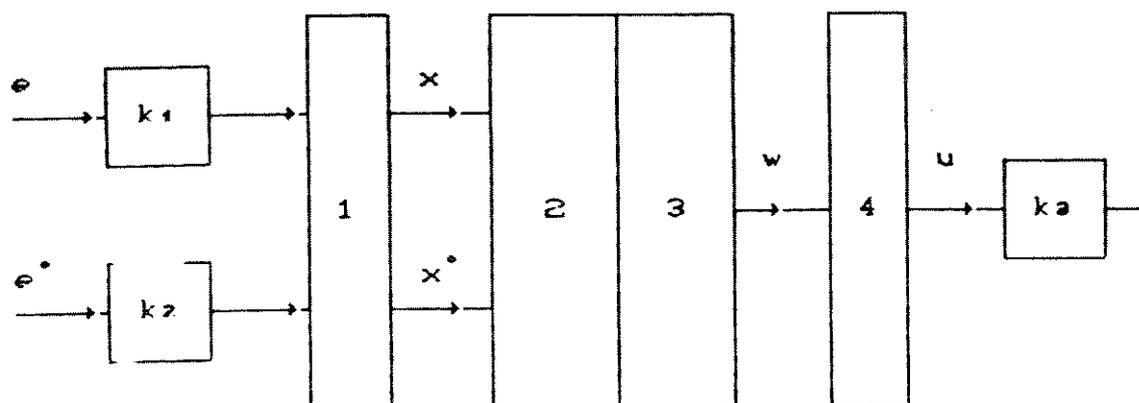
onde:

- r — sinal de referência,
- e — erro ( $r - y$ ),
- $e'$  — derivada do erro,
- u — sinal de controle.
- y — sinal de saída do processo.

A principal diferença entre os controladores

tradicionais e o CBRN, reside no fato de que neste último o procedimento para determinar a ação do controle é baseado num conjunto de regras heurísticas e não em um algoritmo matemático ([Spin90]).

O CBRN pode ser dividido em várias partes como descrito na Figura 3.2 ([Proc79]).



Arquitetura do CBRN.

Figura 3.2

onde:

- 1 - sistema de conversão do  $e$  e  $e'$  para a forma nebulosa,
- 2 - conjunto de regras,
- 3 - algoritmo de inferência,
- 4 - sistema de conversão de  $w$  para a forma determinística.

O algoritmo de trabalho do CBRN baseia-se no seguinte conjunto de passos executados sequencialmente sem interrupções ([Mamd77b], [Curt90]):

- amplifica o erro  $e$  e sua variação  $e'$ ,
- converte-os da forma determinística para um forma nebulosa,
- interpreta o conjunto de regras,
- converte o valor inferido na forma nebulosa para determinística,
- amplifica o sinal de controle obtido e volta novamente ao começo da sequência de passos.

## 3.2 COMPONENTES DO CONTROLADOR BASEADO EM REGRAS NEBULOSAS

Como visto na Figura 3.2 podemos dividir o CBRN em 4 estruturas funcionais:

- 1 - os amplificadores de sinal  $k_1$ ,  $k_2$  e  $k_3$ ,
- 2 - a parte que converte o sinal de um forma determinística para uma forma nebulosa,
- 3 - o conjunto de regras e o seu sistema de inferência, e
- 4 - a parte que converte o sinal nebuloso para determinístico.

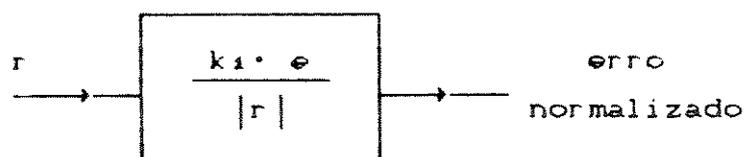
### 3.2.1 AMPLIFICADORES

Os amplificadores tem o seu funcionamento no CBRN análogo ao do sistema de controle convêncional. Sua função é tornar o CBRN de uso mais geral; isto é feito sintonizando os parâmetros  $k_n$ , sendo que os dois primeiros amplificadores ajustam os sinais de entrada ao conjunto de regras, enquanto que o último ajusta a ação de controle do CBRN ao processo em

questão ([Braa70]).

Em uma analogia com controladores PI (Proporcional + Integral), o conjunto de regras do CBRN é o algoritmo do controlador PI que produz a ação de controle. Os dois amplificadores  $k_1$ ,  $k_2$  são análogos aos ganhos  $k_I$  e  $k_P$  do controlador PI, então alterar  $k_1$ ,  $k_2$  significa uma atitude semelhante à de sintonizar  $k_I$  e  $k_P$  ([Maed88]). O leitor poderá constatar o porque destas afirmações no Capítulo 4, quando discutiremos um modelo para o CBRN.

Outra importante função executada por este módulo é a normalização do  $e$  e de sua  $e'$ . Este procedimento permite fixar antecipadamente o intervalo de existência destes dois sinais. O procedimento de normalização é ilustrado na Figura 3.3.



Procedimento de normalização do sinal.

Figura 3.3

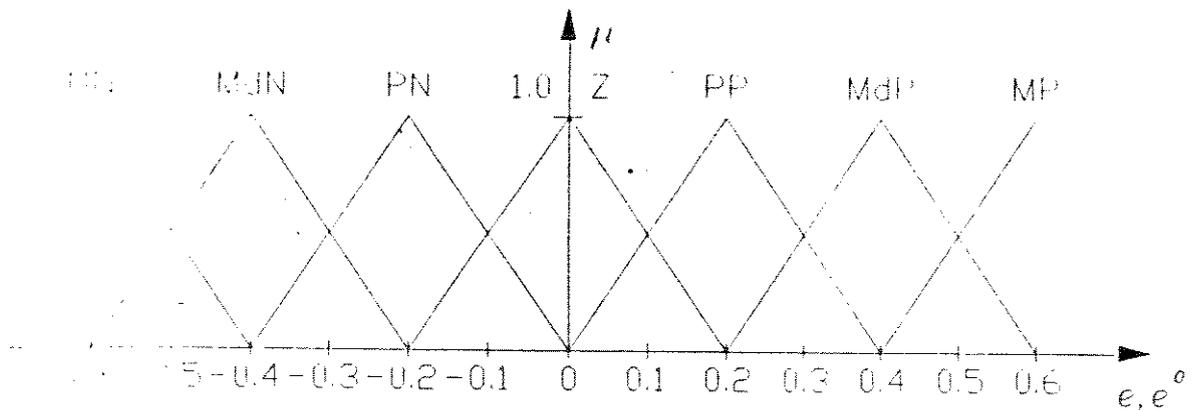
### 3.2.2 CONVERSÃO DE SINAL DETERMINÍSTICO PARA SINAL NEBULOSO

O funcionamento deste módulo do CBRN é semelhante ao dos conversores A/D. Como ainda não existe hardware que produza sinais nebulosos para processamento deste tipo de informação, é necessária a introdução de um módulo que faça esta tarefa através do software; com a introdução deste bloco no controlador. Algumas semelhanças com os conversores A/D

comuns são observadas:

- i) efeito de quantização e,
- ii) conversão de tipos de sinais.

Em relação ao último item, o sistema de conversão de sinal determinístico para sinal nebuloso, atua da seguinte forma: o sinal de entrada é medido e em seguida recebe várias classificações. Estas classificações são atribuições de pertinências ao sinal em cada um dos conjuntos nebulosos, por exemplo: muito positivo (MP), zero (Z), pouco negativo (PN), e etc. Estas pertinências expressam o quanto verdadeira é cada uma das respectivas classificações. A Figura 3.4 mostra um grupo hipotético de conjuntos nebulosos.



Exemplo de um grupo de conjuntos nebulosos.

Figura 3.4

Em relação ao item i, o efeito de quantização existe simplesmente por questão de praticidade: nos conversores A/D por exemplo, quanto menor for o intervalo de quantização mais onerosa será a implementação do conversor, o que exige um critério de custo/benefício. O mesmo fato acontece em relação

ao processo de conversão de sinal determinístico para nebuloso. Muitas implementações de conjuntos nebulosos tem as suas funções de pertinência  $\mu$  descritas por equações matemáticas de computação onerosa. Isto faz com que o sistema tenha a sua execução penalizada gastando muito tempo de processamento e restringindo sua disponibilidade de aplicação. Em face a isto a abordagem usada para classificar o sinal, é a utilização de níveis de quantização através da implementação de uma tabela de conversão, ao invés de calcular as funções de pertinência  $\mu$  ([Naut76] , [Lee90a]). A Figura 3.5 ilustra uma tabela de conversão de sinal de um CBRN que faz uso de quantização. Os conjuntos nebulosos utilizados são os descritos na Figura 3.4.

$10^{-1}$	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6
MP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.7	1.0
MdP	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.7	1.0	0.7	0.3
PP	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.7	1.0	0.7	0.3	0.0	0.0
Z	0.0	0.0	0.0	0.0	0.3	0.7	1.0	0.7	0.3	0.0	0.0	0.0	0.0
PN	0.0	0.0	0.3	0.7	1.0	0.7	0.3	0.0	0.0	0.0	0.0	0.0	0.0
MdP	0.3	0.7	1.0	0.7	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
MP	1.0	0.7	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

Conjuntos nebulosos discretizados.

Figura 3.5

Todos os conjuntos nebulosos utilizados neste trabalho são normais, ou seja:

$$\forall u, \exists u \{ \vee \mu(u) \} = 1 .$$

Este procedimento de conversão de sinais pode ser interpretado como uma operação do tipo:

$$\mu(u) = F(u) .$$

### 3.2.3 REGRAS E INFERÊNCIAS

Dentre os paradigmas existentes para se representar conhecimento, o de regras de produção é o que melhor adapta-se a descrever procedimentos heurísticos por ser a forma natural com que se expressa o especialista ([Wate78], [Wiel87]). Assim, as regras que usaremos, modeladas pela lógica nebulosa, são descritas na forma de regras de produção. A Figura 3.6 apresenta algumas das regras utilizadas por E. Mamdani ([Mamd77a]) em seu trabalho pioneiro.

Se pressão é CMN OU MdN E var. da pressão é PN  
Então calor é MP

OU

Se pressão é PN E var. da pressão é PP  
Então calor é MP

OU

Se pressão é Z E var. da pressão é (MP OU MdP)  
Então calor é MdP

OU

Se pressão é Z E var. da pressão é (MN OU MdN)  
Então calor é MdN

Exemplo de um grupo de regras de controle.

Figura 3.6

O processo de obtenção de uma resposta do conjunto de regras da Figura 3.6 é conhecido como um algoritmo nebuloso, por processar informações codificadas de forma nebulosa ([Zade68]). Mostraremos dois tipos de algoritmos nebulosos. O primeiro, um algoritmo para sistema de um sinal de entrada e um de saída, aqui descrito apenas por razões didáticas devido à sua simplicidade. O segundo, um algoritmo para sistemas com dois sinais de entrada e um de saída, que é o algoritmo utilizado nas várias implementações de CBRN's realizadas por outros autores e também neste texto.

O método de inferência utilizado neste texto é o método proposto por Ebrahim Mamdani ([Mamd74]). O Apêndice B apresenta o método de inferência proposto por Lofti Zadeh ([Zade73]).

### 3.2.3.1 ALGORITMO DE INFERÊNCIA PARA SISTEMAS DE UMA ENTRADA E UMA SAÍDA

Seja o conjunto de regras  $S$  descrito por:

$$\begin{array}{l}
 S_1 : \text{ Se } x \text{ é } A_1 \text{ Então } y \text{ é } B_1 \\
 \quad \quad \quad \text{OU} \\
 S_2 : \text{ Se } x \text{ é } A_2 \text{ Então } y \text{ é } B_2 \\
 \quad \quad \quad \text{OU} \\
 \quad \quad \quad \vdots \quad \quad \quad \vdots \\
 \quad \quad \quad \text{OU} \\
 S_n : \text{ Se } x \text{ é } A_n \text{ Então } y \text{ é } B_n
 \end{array}$$

Forma canônica do conjunto de regras do tipo  
 $A \rightarrow B$ .

Figura 3.7

Onde  $x$  e  $y$  são variáveis linguísticas;  $A$  e  $B$  são os valores linguísticos assumidos por  $x$  e  $y$  no respectivo domínio de  $U$  e  $V$  ([Zade75a]).

A regra  $S_i$  é uma implicação simples, descrita pela relação  $R$  entre as variáveis  $x$  e  $y$ , ou seja, pelo produto cartesiano  $U \times V$ :

$$\mu_{A \times B}(u, v) = \wedge_u [ \mu_A(u); \mu_B(v) ] .$$

O resultado da interpretação da regra  $S_i$  é obtido usando-se a regra de inferência da composição ([Zade88]):

antecedente\_1: Se  $x$  é  $A$  Então  $y$  é  $B$   
 antecedente\_2:  $x$  é  $A'$

---

consequente:  $y$  é  $B'$

ou seja:

$$B'_i = A' \cdot (A_i \rightarrow B_i) .$$

O conjunto de regras  $S$  tem a forma:

$$S = \{ S_1, S_2, \dots, S_n \} .$$

e como cada uma das  $S_i$  regras estabelece uma relação  $R_i$ , temos:

$$S = \{ R_1, R_2, \dots, R_n \} .$$

Cada um das  $S_i$  regras é estabelecida como uma relação  $R$  interligadas por meio do conectivo OU ( $\vee$ ) as regras  $S_{i-1}$  e  $S_{i+1}$ , temos então:

$$R = \{ R_1 \vee R_2 \vee \dots \vee R_n \} ,$$

que pode ser reescrita em uma notação mais concisa como:

$$R = \bigvee_{i=1 \dots n} R_i$$

Cada uma das  $S_i$  regras são instanciadas pelo antecedente  $A'$ , obtendo-se o consequente  $B'_i$ . A expressão final do conjunto de regras levando em consideração cada um dos termos  $B'_i$  tem, então, a forma ([Lee90b]):

$$B' = (A') \circ \left\{ \bigvee_{i=1 \dots n} R_i \right\}$$

Lema 1:  $(A') \circ \left\{ \bigvee_{i=1 \dots n} R_i \right\} = \bigvee_{i=1 \dots n} \{ (A') \circ R_i \}$

Prova:

$$B' = (A') \circ \bigvee_{i=1 \dots n} R_i$$

$$B' = (A') \circ \bigvee_{i=1 \dots n} (A'_i \rightarrow B'_i)$$

As funções de pertinências  $\mu_A$  e  $\mu_B$  tomam valores no domínio  $U$  e  $V$  respectivamente; ou seja,  $u \in U$  e  $v \in V$ .

$$\mu_{B'}(v) = \mu_{A'}(u) \circ \bigvee_{u,v} [\mu_{R_1}(u,v); \mu_{R_2}(u,v); \dots; \mu_{R_n}(u,v)]$$

$$\mu_{B'}(v) = \bigvee_u \wedge \{ \mu_{A'}(u); \bigvee_{u,v} [\mu_{R_1}(u,v); \dots; \mu_{R_n}(u,v)] \}$$

$$\mu_{B'}(v) = \bigvee_u \bigvee_{u,v} \{ \wedge [\mu_{A'}(u), \mu_{R_1}(u,v)]; \wedge [\mu_{A'}(u), \mu_{R_2}(u,v)]; \dots; \wedge [\mu_{A'}(u), \mu_{R_n}(u,v)] \}$$

$$\mu_{B'}(v) = \bigvee_{u,v} \{ [\mu_{A'}(u) \circ \mu_{R_1}(u,v)]; [\mu_{A'}(u) \circ \mu_{R_2}(u,v)]; \dots; [\mu_{A'}(u) \circ \mu_{R_n}(u,v)] \}$$

reescrevendo:

$$B' = [(A') \circ R_1] \vee [(A') \circ R_2] \vee \dots \vee [(A') \circ R_n]$$

$$B' = \bigvee_{i=1 \dots n} (A') \circ R_i,$$

$$B' = \bigvee_{i=1 \dots n} (A') \circ (A_i \rightarrow B_i),$$

$$B' = \bigvee_{i=1 \dots n} B'_i.$$

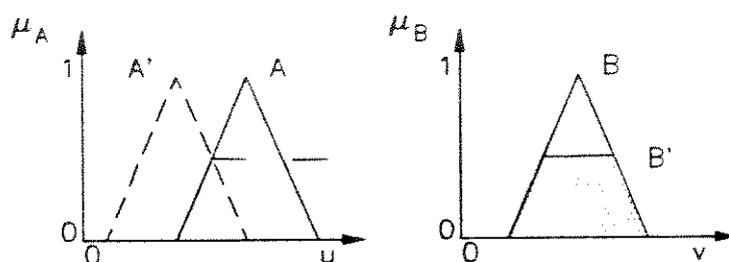
■

O Lema 1 mostra que a regra de inferência da composição  $(\circ)$  e o conectivo OU  $(\bigvee)$  são comutativos. A obtenção de  $B'$  é dada por:

$$\mu_{B'}(v) = \bigvee_v \left\{ \bigvee_u \left[ \wedge \left[ \mu_{A'}(u); \mu_{A_i \rightarrow B_i}(u,v) \right] \right] \dots \dots \bigvee_u \left[ \wedge \left[ \mu_{A'}(u); \mu_{A_i \rightarrow B_i}(u,v) \right] \right] \right\},$$

$u \in U, v \in V$

Este tipo de inferência pode ser visto geometricamente na Figura 3.8 abaixo.



Exemplo geométrico de inferência para regras  $A \rightarrow B$ .

Figura 3.8

### 3.2.3.2 ALGORITMO DE INFERÊNCIA PARA SISTEMAS DE DUAS ENTRADAS E UMA SAÍDA

Este tipo de algoritmo é utilizado na interpretação de um conjunto de regras do tipo descrito na Figura 3.9, o qual é uma evolução do tipo de regra visto anteriormente.

$$\begin{array}{l}
 S_1 : \text{ Se } x \text{ é } A_1 \quad \text{E} \quad y \text{ é } B_1 \quad \text{Então } z \text{ é } C_1 \\
 \qquad \qquad \qquad \qquad \qquad \qquad \text{OU} \\
 S_2 : \text{ Se } x \text{ é } A_2 \quad \text{E} \quad y \text{ é } B_2 \quad \text{Então } z \text{ é } C_2 \\
 \qquad \qquad \qquad \qquad \qquad \qquad \text{OU} \\
 \qquad \qquad \qquad \qquad \qquad \qquad \vdots \\
 \qquad \qquad \qquad \qquad \qquad \qquad \text{OU} \\
 S_n : \text{ Se } x \text{ é } A_n \quad \text{E} \quad y \text{ é } B_n \quad \text{Então } z \text{ é } C_n
 \end{array}$$

Forma canônica do conjunto de regras do tipo  
 $(A \wedge B) \rightarrow C$ .

Figura 3.9

Como o exposto na seção anterior  $z$  é uma variável lingüística,  $C$  seu respectivo valor lingüístico, que é definido no domínio  $W$ . O algoritmo de interpretação da regra é:

$$\begin{array}{l}
 \text{antecedente}_1: \quad \text{Se } x \text{ é } A \quad \text{E} \quad y \text{ é } B \quad \text{Então } z \text{ é } C \\
 \text{antecedente}_2: \quad \quad \quad x \text{ é } A', \quad \quad y \text{ é } B' \\
 \hline
 \text{consequente:} \quad \quad \quad z \text{ é } C'
 \end{array}$$

O consequente  $C'$  pode ser deduzido dos antecedentes 1 e 2 usando a regra de inferência da composição. Então temos que:

$$C' = (A', B') \circ [(A \wedge B) \rightarrow C],$$

vamos estabelecer dois importantes enunciados:

$$\text{Lema 2: } [\wedge (A', B') \rightarrow C] = \wedge [(A \rightarrow C); (B \rightarrow C)]$$

*Prova:*

Sejam  $A$  e  $B$  dois conjuntos nebulosos de  $n$  elementos cada, e  $C$  um conjunto nebuloso com  $m$  elementos:

$$\begin{aligned} A &= \{a_1, a_2, \dots, a_n\}, \\ B &= \{b_1, b_2, \dots, b_n\}, \\ C &= \{c_1, c_2, \dots, c_m\}. \end{aligned}$$

as implicações  $A \rightarrow C$  e  $B \rightarrow C$  são dadas pelo produto cartesiano:

$$A \rightarrow C = \{ (a_1 \wedge c_1); (a_1 \wedge c_2); \dots; (a_1 \wedge c_m); \\ (a_2 \wedge c_1); (a_2 \wedge c_2); \dots; (a_2 \wedge c_m); \\ (a_n \wedge c_1); (a_n \wedge c_2); \dots; (a_n \wedge c_m) \},$$

$$B \rightarrow C = \{ (b_1 \wedge c_1); (b_1 \wedge c_2); \dots; (b_1 \wedge c_m); \\ (b_2 \wedge c_1); (b_2 \wedge c_2); \dots; (b_2 \wedge c_m); \\ (b_n \wedge c_1); (b_n \wedge c_2); \dots; (b_n \wedge c_m) \},$$

a expressão  $(A \rightarrow C) \wedge (B \rightarrow C)$ , a qual por facilidade de notação denominaremos de  $\lambda$ , tem a forma:

$$\lambda = \{ [(a_1 \wedge c_1) \wedge (b_1 \wedge c_1)]; [(a_1 \wedge c_2) \wedge (b_1 \wedge c_2)]; \\ \dots; [(a_1 \wedge c_m) \wedge (b_1 \wedge c_m)]; \dots; [(a_n \wedge c_1) \wedge \\ (b_n \wedge c_1)]; \dots; [(a_n \wedge c_m) \wedge (b_n \wedge c_m)] \};$$

evidenciando os elementos de  $C$ :

$$\lambda = \{ [(a_1 \wedge b_1) \wedge c_1]; [(a_1 \wedge b_1) \wedge c_2]; \dots; [(a_1 \wedge b_1) \wedge c_m]; \\ [(a_2 \wedge b_2) \wedge c_1]; [(a_2 \wedge b_2) \wedge c_2]; \dots; [(a_2 \wedge b_2) \wedge c_m]; \\ [(a_n \wedge b_n) \wedge c_1]; [(a_n \wedge b_n) \wedge c_2]; \dots; [(a_n \wedge b_n) \wedge c_m] \}.$$

A expressão  $\lambda$  pode ser reduzida, então, a forma:

$$\lambda = (A \wedge B) \rightarrow C$$

■

$$\begin{aligned} \text{Lema 3: } (A', B') \circ [(A_i \wedge B_i) \rightarrow C_i] \\ = [A' \circ (A_i \rightarrow C_i)] \wedge [B' \circ (B_i \rightarrow C_i)] \end{aligned}$$

Prova:

$$C'_i = (A', B') \circ [(A_i \wedge B_i) \rightarrow C_i],$$

utilizando o resultado do Lema 2, temos ([Lee90b]):

$$C'_i = (A', B') \circ [(A_i \rightarrow C_i) \wedge (B_i \rightarrow C_i)],$$

$$\mu_{C'_i}(w) = \bigvee_{u,v} \wedge \{ (\mu_{A'}; \mu_{B'}) ; [\mu_{A_i \rightarrow C_i}(u,w) \wedge \mu_{B_i \rightarrow C_i}(v,w)] \}$$

aplicando cada um dos valores linguísticos às suas respectivas partes na premissa da regra S, temos:

$$\mu_{C'_i}(w) = \bigvee_{u,v} \wedge \{ [\mu_{A'} \wedge \mu_{A_i \rightarrow C_i}(u,w)] ; [\mu_{B'} \wedge \mu_{B_i \rightarrow C_i}(v,w)] \},$$

$$\mu_{C'_i}(w) = \wedge \{ [\mu_{A'} \circ \mu_{A_i \rightarrow C_i}(u,w)] ; [\mu_{B'} \circ \mu_{B_i \rightarrow C_i}(v,w)] \}.$$

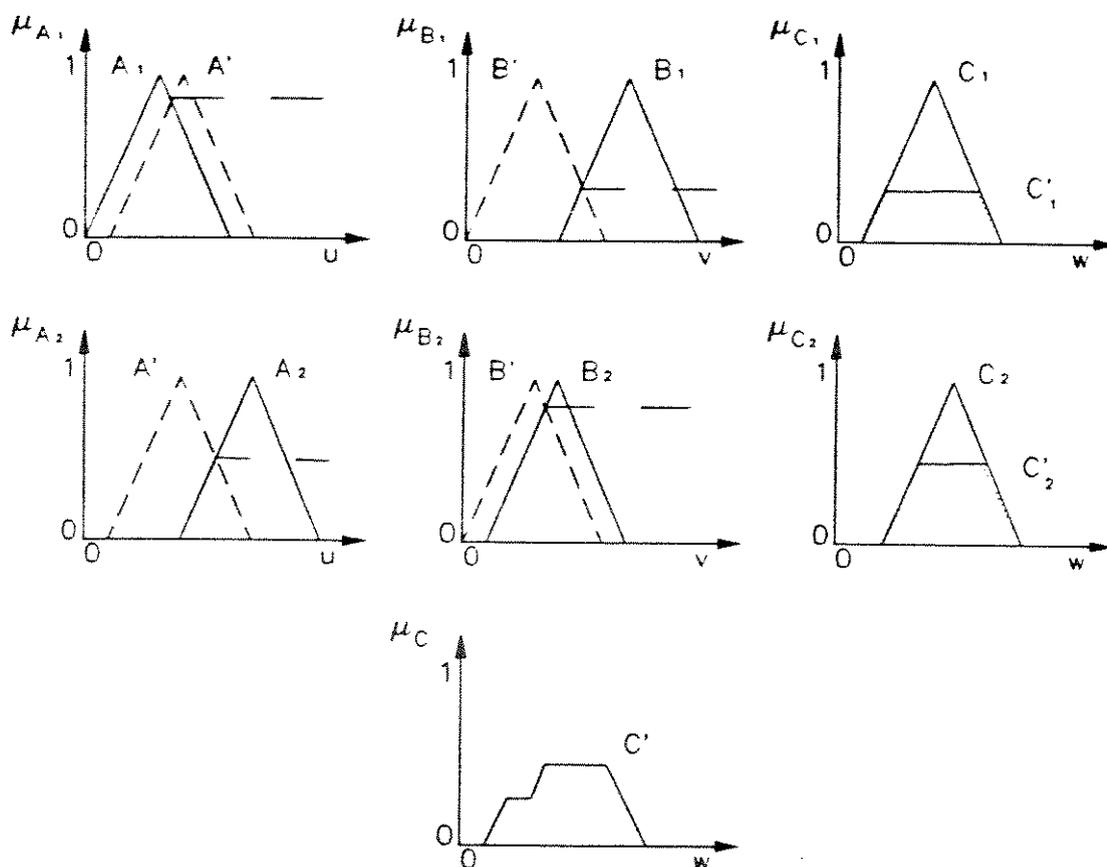
■

Esta expressão mostra que a regra S contém duas implicações  $R_{A \rightarrow B}$  e  $R_{B \rightarrow C}$  interligadas pelo conectivo  $\wedge$ . Generalizando o resultado do Lema 1, mostrado na seção anterior, o qual leva em consideração todo o conjunto de regras interligado pelo conectivo  $\vee$ , temos a expressão final de  $C'$ :

$$\mu_{C'_i}(w) = \{ [\mu_{A'_i} \circ \mu_{A_i \rightarrow C'_i}(u, w)) \wedge (\mu_{B'_i} \circ \mu_{B_i \rightarrow C'_i}(v, w))]\}$$

$$C' = \bigvee_{i=1 \dots n} C'_i$$

Seu resultado pode ser visualizado através do exemplo geométrico da Figura 3.10.



Exemplo geométrico de infêrencias de duas regras  $(A \wedge B) \rightarrow C$ , interligadas pelo conectivo  $\vee$ .

Figura 3.10

### 3.2.4 CONVERSÃO DO SINAL NEBULOSO PARA SINAL DETERMINÍSTICO

Como pode ser visto, o sinal  $z$  ainda está na forma nebulosa, pois é resultado da interpretação do conjunto de regras nebulosas. Para que este sinal possa ser utilizado por um sistema, ele necessita ser convertido para uma forma determinística. Esta operação pode ser interpretada como:

$$w = F^{-1}(\mu(w)) .$$
$$w \in W$$

Várias são as formas de se obter esta conversão ([Suge85b]). A seguir veremos três destas formas:

i) Soma dos Pesos:

$$w_o = \frac{\sum_{i=1}^n w_i \mu_i(w)}{\sum_{i=1}^n \mu_i(w)}$$

ii) Centro de Gravidade:

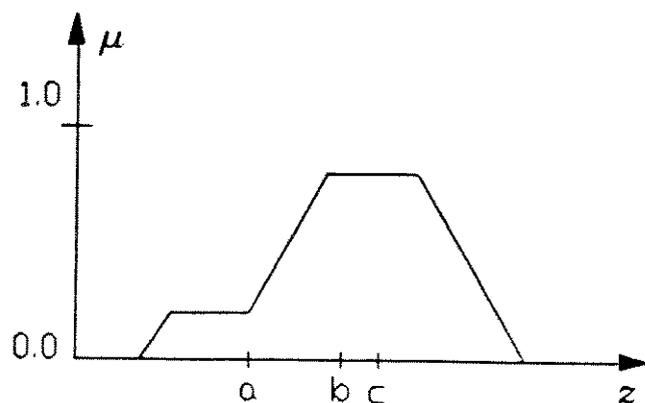
$$w_o = \frac{\int_{-\infty}^{\infty} z \cdot \mu_C(w) dw}{\int_{-\infty}^{\infty} \mu_C(w) dw}$$

iii) Máximo valor de  $\mu_i(w)$ .

Esta última escolha foi usada na implementação das primeiras experiências com CBRN, onde o valor de  $w$  corresponderia ao máximo pico de  $\mu(w)$  ([Mamd77b]). Neste trabalho usaremos a forma ii, o centro de gravidade.

A Figura 3.11 mostra, através de um exemplo geométrico, o efeito da escolha das três formas de obter-se o valor

determinístico de  $\mu$ . Os pontos A, B e C correspondem máximo valor de  $\mu'$ , centro de gravidade e soma dos pesos.



Comparação dos três procedimentos de conversão.

Figura 3.11

## CAPÍTULO 4

### ESTUDO DE UM MODELO PARA O CONTROLADOR BASEADO EM REGRAS NEBULOSAS

Após termos visto como é a estrutura de um CBRN, e discutido alguns de seus aspectos mais importantes, vamos agora estudar um modelo para o mesmo.

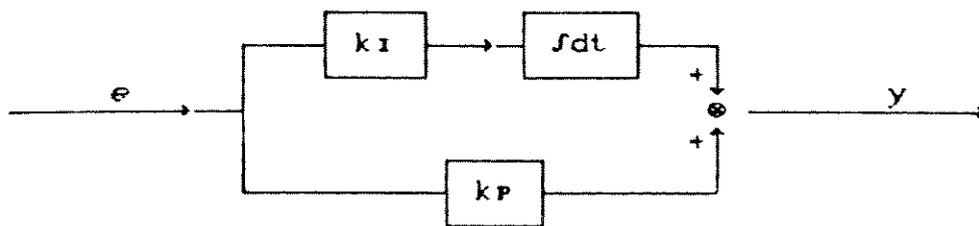
O CBRN não é um controlador que possua um algoritmo matemático para a construção da ação de controle como nos controladores convencionais. Como vimos anteriormente, sua ação de controle é baseada em um modelo qualitativo e não quantitativo. Assim, torna-se inviável obter-se um modelo matemático do mesmo.

A abordagem usada neste capítulo é baseada em uma interpretação heurística deste controlador. O modelo obtido é comparado com o modelo de um controlador PI, utilizando-se de alguns aspectos e propriedades da estrutura do CBRN, com os quais já se tomou contato anteriormente; o capítulo é finalizado mostrando a relação existente entre estes dois tipos de controladores.

No Capítulo 5 será apresentado a resposta de um sistema sendo controlado por um CBRN e um controlador PI.

## 4.1 UM MODELO BASEADO EM HEURÍSTICA

Várias implementações de CBRNs tem sido realizadas ([Mamd74], [Kick78a], [King77], [Tong80], [Maie85]). Nestas implementações observou-se que a atuação do CBRN, possui uma ação de controle semelhante ao tradicional controlador PI ([Mamd77b], [Naut76]), embora suas estruturas sejam diferentes; a Figura 4.1 mostra a estrutura do controlador PI.



Arquitetura de um controlador PI.

Figura 4.1

A ação de controle  $y$  do controlador PI pode ser vista através de sua equação no plano  $s$  ([Iser81]):

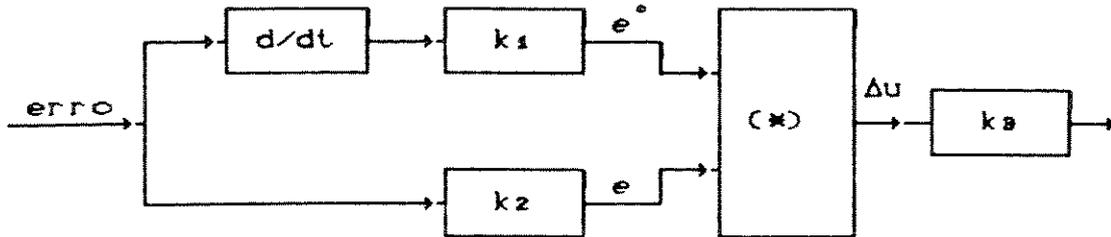
$$\frac{Y(s)}{E(s)} = k_P + k_I \frac{1}{s}$$

$$s Y(s) = (k_I + s k_P) E(s) .$$

Reescrevendo a equação anterior sob a forma de equação a diferença temos:

$$y_{kT} - y_{kT-T} = k_P [e_{kT} - e_{kT-T}] + k_I e_{kT} .$$

A Figura 4.2 mostra a arquitetura do CBRN com a identificação funcional de cada um dos seus módulos.



(\*) Sistema de Regras e Inferência  
Arquitetura funcional de um CBRN.

Figura 4.2

Denominaremos por  $f$  uma função não-linear que, recebendo  $e$  e  $e^\circ$  como argumento, retorna o valor  $\Delta u$ . A função  $f$  é representado pelo conjunto de regras e inferências da Figura 4.2 ([Gupt88]). Lembrando que  $\Delta u$  é a variação do sinal de saída então:

$$\Delta u_{Tk} = u_{Tk} - u_{Tk-T}$$

Pode-se agora, em função da Figura 4.2, obter-se a expressão:

$$\Delta u = f(k_2 e, k_1 e^\circ),$$

que expandida toma a forma:

$$u_{kT} - u_{kT-T} = f(k_1 (e_{kT} - e_{kT-T}), k_2 e_{kT}).$$

Reescrevendo a equação do controlador PI, utilizando a função  $f'$  como a soma aritmética, fica claro a semelhança entre ambas equações:

$$y_{kT} - y_{kT-T} = f'(k_P (e_{kT} - e_{kT-T}), k_I e_{kT}) .$$

$$u_{kT} - u_{kT-T} = f(k_I (e_{kT} - e_{kT-T}), k_D e_{kT}) .$$

Assim, podemos interpretar a função  $f$  como sendo encarregada de realizar um mapeamento em que o sinal  $u_{kT}$ , sinal de saída do CBRN, tenha o mesmo efeito na resposta do processo que o sinal  $u_{kT}$ , sinal de saída do controlador PI ([Mota90], [Kick78a]).

## 4.2 A RELAÇÃO ENTRE O CBRN E O CONTROLADOR PI

A execução do conjunto de regras do CBRN não sofre encadeamento, sendo cada regra executada uma vez e o resultado final uma composição do consequente de cada regra; assim, este conjunto de regras pode ser descrito sob a forma de uma tabela ([Ruth76]). A Figura 4.3 mostra uma matriz de regras, onde os mnemônicos Z, MdP, PN e etc, significam zero, medianamente positivo e pouco negativo e assim por diante. Este conjunto de regras foi proposto por MacVicar-Whelan ([Macv76]) para simular a ação do controlador PI. Ela possui sete valores lingüísticos;  $U_{-3}$ ,  $U_{-2}$ , ...,  $U_3$ , para o sinal de saída.

Deve-se observar uma certa simetria da tabela da Figura 4.3 em relação a sua diagonal principal, apresentando-se dividida em várias regiões que representam a mesma saída  $u$ . As fronteiras destas regiões não são contínuas, isto deve-se ao processo de classificação dos sinais em um número fixo de conjuntos ([Ruth76], [Tong77]).

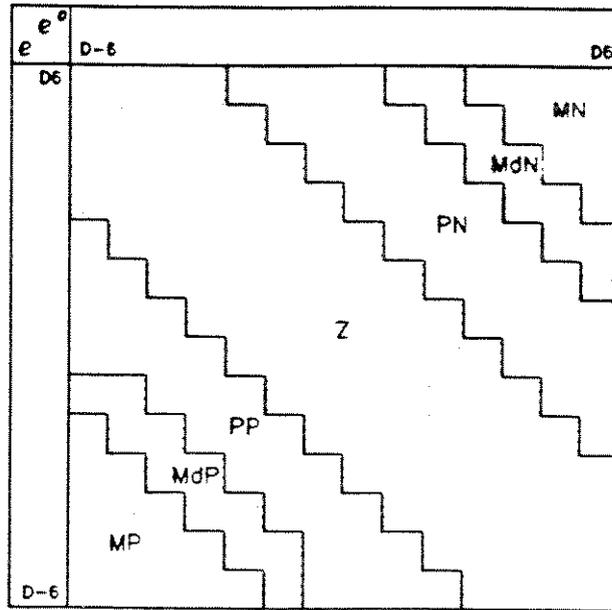
$e$ $e^*$	MN	MdN	PN	Z	PP	MdP	MP
MP	Z	PN	MdN	MN	MN	MN	MN
MdP	PP	Z	PN	MdN	MdN	MN	MN
PP	MdP	PP	Z	PN	PN	MdN	MN
Z	MdP	MdP	PP	Z	PN	MdN	MdN
PN	MP	MdP	PP	PP	Z	PN	MdN
MdN	MP	MP	MdP	MdP	PP	Z	PN
MN	MP	MP	MP	MP	MdP	PP	Z

Regras na forma de matriz de controle.

Figura 4.3

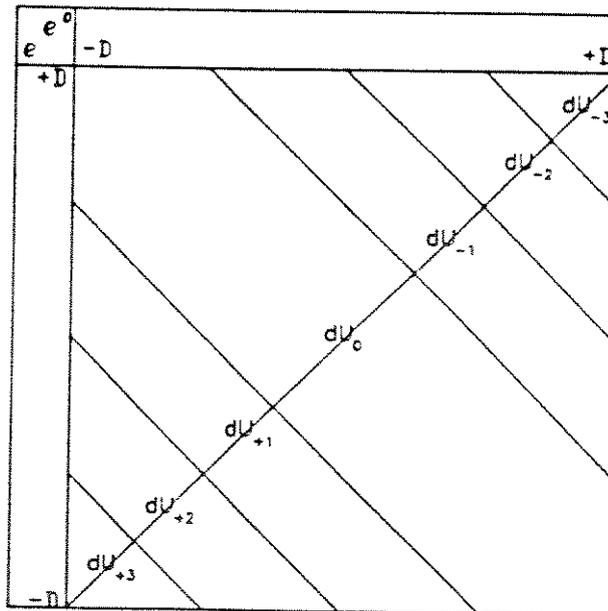
Dobrando-se o número de conjuntos de classificação dos sinais do  $e$  e da  $e^*$ , mas mantendo-se fixo o seu intervalo de existencia, e fixando MP e MN como os conjuntos mais extremos, a descontinuidade da ação apresentada na Figura 4.3 diminuirá.

Este processo de aumentar o número de conjuntos pode ser repetido até as fronteiras das regiões não serem mais descontínuas. Este procedimento é mostrado pelas Figuras 4.4 e 4.5.



Efeito do aumento do número de conjuntos nebulosos.

Figura 4.4



Matriz de controle quando o número de conjuntos nebulosos tende ao infinito.

Figura 4.5

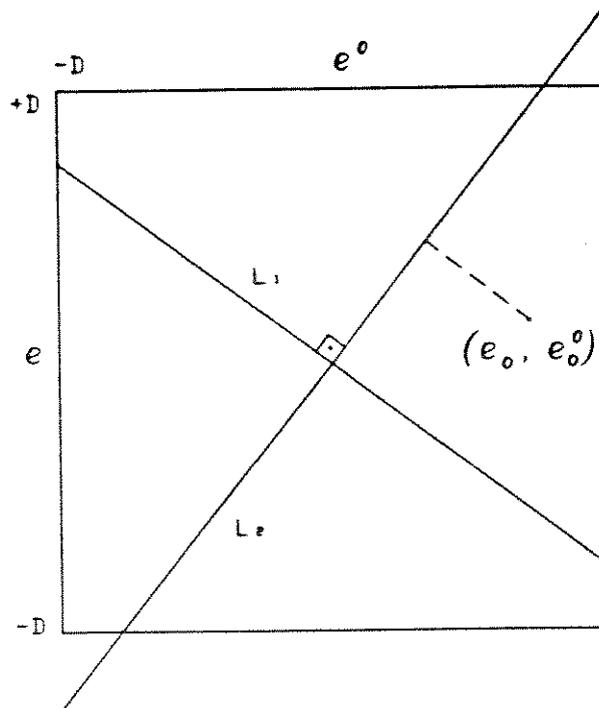
Assumindo-se que:

i) a política de controle é tal que sua fronteira seja linear e, portanto

ii) o nível de descontinuidade seja zero;

uma estrutura linear de um controlador PI pode ser definida.

Vamos supor que a política de controle tem uma inclinação  $-C$ , como ilustrado na Figura 4.6.



Inclinação  $C$  da política de controle.

Figura 4.6

Iremos determinar a ação de controle para um dado ponto de operação  $(e_0, e_0^0)$ . Considere a linha reta com inclinação  $-C$  dada pela política de controle. Sua equação é dada por:

$$(e - e_0) / (e^\circ - e_0^\circ) = -C .$$

Sua perpendicular, que passa pela origem (o ponto central da reta), tem a equação:

$$e / e^\circ = 1 / C .$$

Resolvendo ambas as equações anteriores obtemos:

$$e = [ 1 / (1 + C^2) ] [ e_0 + C e_0^\circ ] ,$$

e

$$e^\circ = [ C / (1 + C^2) ] [ e_0 + C e_0^\circ ] .$$

Calculando a distância do ponto de trabalho à origem obtemos:

$$[ e^2 + (e^\circ)^2 ]^{1/2} = [ 1 / (1 + C^2) ]^{1/2} [ e_0 + C e_0^\circ ] .$$

Assumindo que esta distância é linearmente proporcional a variação da ação de controle  $\Delta u$  ([Braa79]):

$$\Delta u = k [ 1 / (1 + C^2) ]^{1/2} [ e_0 + C e_0^\circ ] ,$$

onde  $k$  é a constante de proporcionalidade da equação.

Desenvolvendo a expressão de  $\Delta y$  anterior teremos:

$$\Delta u = k [ 1 / (1 + C^2) ]^{1/2} e_0 + k C [ 1 / (1 + C^2) ]^{1/2} e_0^\circ .$$

Retornando a equação do controlador PI:

$$y_{kT} - y_{kT} = k_P (e_{kT} - e_{kT-1}) + k_I e_{kT} ,$$

temos que o termo:

$$k_P (e_{kT} - e_{kT-1}) + k_I e_{kT} ,$$

é responsável pela modificação da ação de controle, e devido a isto o denominaremos por  $\Delta y$ , ou seja:

$$y_{kT} - y_{kT-1} = \Delta y_{kT} .$$

Igualando as ações de controle  $\Delta y$  e  $\Delta u$  e comparando a expressão obtida para  $\Delta u$  com  $\Delta y$  obtemos:

$$k_P = k C [1 / (1 + C^2)]^{1/2} ,$$

e a expressão do ganho integral será:

$$k_I = k C^2 [1 / (1 + C^2)]^{1/2} .$$

Na escolha de  $k_P$  considerou-se que: a inclinação  $C$  é menor do que 1, já que por questões de estabilidade nos sistemas de controle que usam controladores PI, o ganho proporcional é maior do que o ganho integral. Este dois fatos guiaram a obtenção da expressão final de  $k_P$  e  $k_I$ .

## CAPÍTULO 5

### IMPLEMENTAÇÃO DE UM CONTROLADOR BASEADO EM REGRAS NEBULOSAS

Após termos estudado a arquitetura do CBRN, veremos agora alguns aspectos relevantes a serem considerados na sua implementação.

Como descrito nos capítulos anteriores, os CBRNs são, na realidade, Sistemas Especialistas executando funções análogas a dos controladores industriais. Desta forma, apesar de possuímos um modelo do CBRN, os métodos tradicionais da Engenharia de Controle não podem ser aplicados a este tipo de sistema.

Os procedimentos a serem considerados no projeto do CBRN são fortemente baseados em heurísticas, não existindo um procedimento analítico que possa ser usado. Este capítulo descreve alguns pontos básicos a serem observados, baseando-se em informações obtidas em publicações de outros autores e principalmente na experiência do autor na realização deste trabalho.

Aqui, daremos também continuidade à discussão iniciada no Capítulo 2 sobre a escolha do tipo da função  $f$ , encarregada de realizar o mapeamento da interpretação do mundo real para um modelo deste. Iremos descrever como realizar este mapeamento adotando como procedimento o uso de métodos de Aquisição de Conhecimento utilizados na

Inteligência Artificial.

O capítulo é finalizado mostrando os resultados obtidos por simulação na implementação de um CBRN controlando um processo, em nosso exemplo uma planta de primeira ordem com atraso de transporte, estes resultados serão comentados no final da Parte II deste texto.

## 5.1 SELEÇÃO DOS PARÂMETROS DO CBRN E DA LINGUAGEM DE CONTROLE.

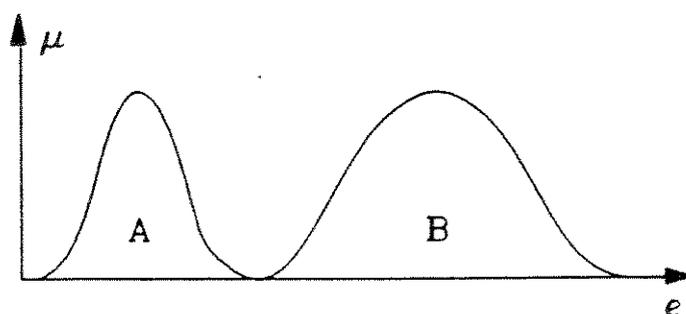
O primeiro passo no projeto de um CBRN é conhecer o processo a controlar (na Engenharia do Conhecimento esta fase é descrita como identificação da tarefa). Após feito isto, o passo seguinte é especificar a linguagem de controle utilizada pelo especialista, em nosso caso os operadores do processo. Denominamos de linguagem de controle ao conjunto de regras, expressas utilizando-se de variáveis lingüísticas, a qual traduz como os operadores humanos controlam o processo. Este procedimento assemelha-se ao utilizado no desenvolvimento de Sistemas Especialistas [(Eric84), (Bern88)].

A linguagem de controle será por nos representada utilizando a lógica nebulosa, já que o conhecimento dos operadores são expressos em frases do tipo senso comum, veja Capítulo 2. Assim, as variáveis de controle correspondem as variáveis lingüísticas, A segundo a notação apresentada no Capítulo 2, e os termos usados pelos especialistas para classificar os estados dos sinais de controle são valores  $T(\Delta)$  assumidos pelas variáveis lingüísticas  $\Delta$ .

A especificação da linguagem de controle deverá também identificar o conjunto de valores determinístico, U, assumido por cada um dos valores lingüísticos.

A resolução que a função de pertinência assume, define o

tipo de sensibilidade que um determinado valor lingüístico possui. A Figura 5.1 mostra uma função de pertinência com alta resolução (A) e outra com baixa (B). As funções que possuem baixa resolução fazem com que o conjunto de regras tenha uma grande sensibilidade a este valor lingüístico e, as que possuem alta resolução uma sensibilidade menor. Isto deve-se ao fato de que quando uma função de pertinência possui alta resolução é necessário grande variação no sinal para provocar pequenas variações na medida da função de pertinência. Em face a isto, durante o projeto de um CBRN é recomendado o uso de um meio termo entre a alta e baixa resolução na função de pertinência. Este efeito será verificado quando estivermos estudando a implementação de um CBRN no final deste capítulo ([Czog81]).



Conjuntos nebulosos quanto a sua resolução.

Figura 5.1

Além da forma da função de pertinência devemos identificar o número de valores lingüísticos utilizados. Este número deverá ser um compromisso entre a flexibilidade (muitos valores lingüísticos) e a simplicidade (poucos valores lingüísticos).

A implementação da função de pertinência  $\mu$  pode ser realizada de duas formas: ela pode ser contínua ou discreta

ao longo da variável de base. Para mostrar que ambas as abordagens são possíveis de utilização, neste capítulo a função  $\mu$  será implementada usando de discretização e no capítulo seguinte usaremos da primeira abordagem de implementação da função  $\mu$ .

Assumindo-se que a função de pertinência é da forma discreta, podemos definir uma função  $\lambda$  encarregada de realizar o mapeamento entre o valor da variável de base e o seu correspondente valor de pertinência. Esta função é descrita utilizando um fator de escala (o ganho do sistema), o intervalo de existência da variável de base e o número de discretizações sofrido pela função  $\mu$ . A função  $\lambda$  é definida como:

$$\lambda = \delta (E_{\text{eat}} - E_{\text{min}}) + 1 ,$$

onde o fator  $\delta$  é definido como:

$$\delta = (n_k - 1) / (E_{\text{max}} - E_{\text{min}}) ,$$

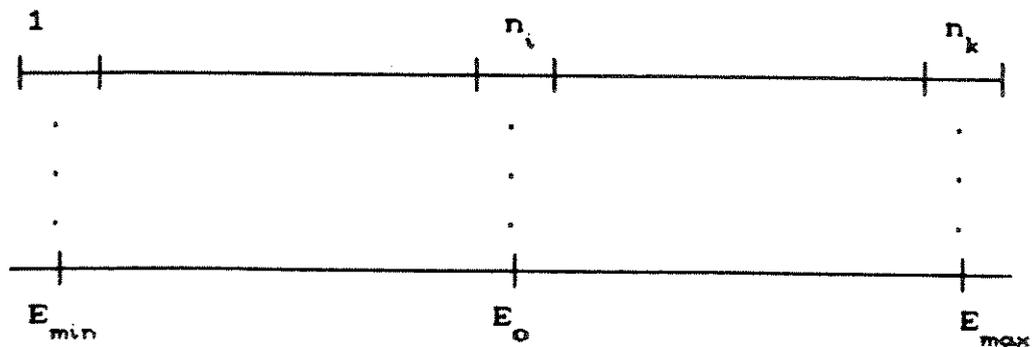
→ onde  $n_k$  é o número total de intervalos discretizados da variável de base e,

→  $E_{\text{max}}$  e  $E_{\text{min}}$  o valor máximo e mínimo das variáveis de base.

O termo  $E_{\text{eat}}$  é definido da forma:

$$E_{\text{eat}} = \max ( E_{\text{min}} ; \min [ E_{\text{max}} ; E_0 ] ) ,$$

onde  $E_0$  é o valor medido na variável de base. A Figura 5.2 ilustra o discutido anteriormente.



Exemplo de conversão de um domínio contínuo para outro discreto.

Figura 5.2

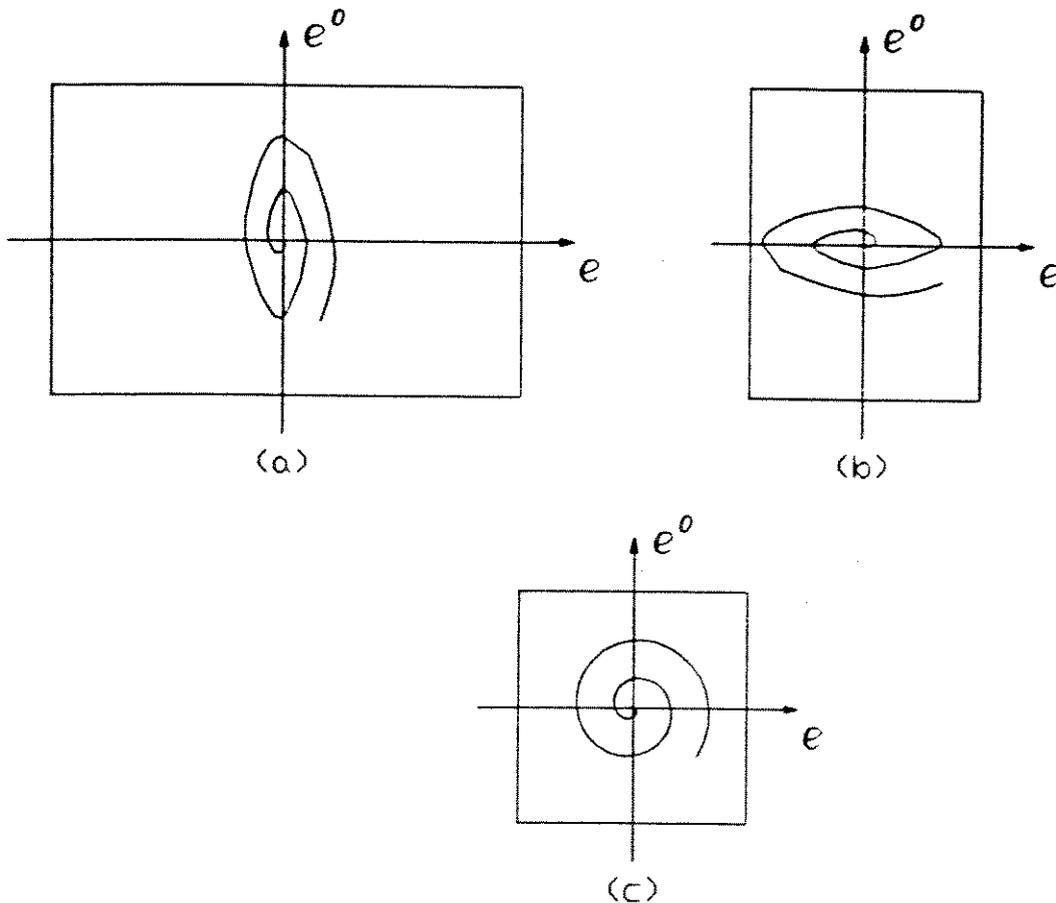
A função  $\mu$  de cada um dos conjuntos nebulosos deve cobrir todo o intervalo de existência da variável de base. Isto assegura que qualquer valor medido na variável da base pode ser descrito por um conjunto nebuloso. A seguir, mostraremos alguns exemplos do efeito da escolha do domínio da variável de base. A Figura 5.3 mostra uma matriz de regras de controle.

				$e'$			
	Z	Z	Z	PP	MdP	MdP	MdP
	Z	Z	Z	PP	MdP	MdP	MP
	PN	Z	Z	Z	PP	PP	MP
—	MdP	PN	PN	Z	PP	PP	MdP
	MN	PN	PN	Z	Z	Z	PP
	MN	MdN	MdN	PN	Z	Z	Z
	MN	MdN	MdN	PN	Z	Z	Z
							$e$

Exemplo de um conjunto de regras de controle na forma de plano cartesiano.

Figura 5.3

Na Figura 5.4 temos o plano de fase ( $e, e^{\circ}$ ) para a ação de controle, usando o conjunto de regras anterior. A parte (a) da figura mostra o erro  $e$  com o domínio muito grande e a variação do erro  $e^{\circ}$  com domínio ideal. A parte (b) mostra o inverso e a parte (c) uma escolha ideal do domínio de  $e$  e da  $e^{\circ}$  ([Tong84]).



Exemplo do efeito da escolha do domínio do  $e, e^{\circ}$  na resposta do sistema.

Figura 5.4

## 5.2 PROJETO DO CONJUNTO DE REGRAS.

Na obtenção do conjunto de regras para o CBRN serão utilizadas técnicas de Aquisição de Conhecimento da Inteligência Artificial os procedimentos de aquisição de conhecimento usados na construção da base de conhecimento do CBRN são denominados métodos diretos ([Hart86]). Estes métodos visam obter do especialista conhecimentos por ele articulado diretamente. Este tipo de método inclui entrevista, questionário, repertorio grid e análise de protocolo ([Kidd87]).

Para poder aplicar os métodos de aquisição de conhecimento, o projetista do CBRN deverá primeiramente observar a tarefa que o especialista executa e como ele a resolve. Assim o projetista saberá como entrevistar, o que questionar e como analisar os protocolos de informação.

Dentre os métodos de aquisição de conhecimento, a entrevista é o método mais simples e utilizado. Nela o especialista revelará como a tarefa é realizada e os fatores que são levados em consideração. A entrevista apresenta a vantagem de que as informações são obtidas de forma livre, pois o especialista detém o controle do processo de informação. Entretanto, entrevistar consome tempo e no estágio final do processo de aquisição pode gerar informações repetidas ([Hoff87]).

O método de aquisição de conhecimento conhecido como questionário, por outro lado, tem a vantagem de ser uma forma mais eficiente de se obter informações. Este método força o especialista a responder questões específicas, e evita obtenção de informações repetidas. Outra vantagem do questionário é que este auxilia o especialista a expressar-se, pois é de fato conhecido a deficiência que estes tem de narrar seu próprio conhecimento.

A técnica de grau de repertório (tradução adotada pelo autor para o termo em inglês "repertory grid") é aplicada em seguida a de entrevista. Sua finalidade é capturar os

conceitos, a definição de conceito usado neste texto é a apresentada por Sowa ([Sowa86]). Esta técnica é aplicada usando-se o método de restrições pessoais desenvolvido por G. Kelly ([Kelly86]), onde o especialista é exposto sempre a três alternativas devendo escolher uma delas e explicar o porque da sua restrição as demais. Maiores detalhes sobre este método pode ser encontrado em consulta a bibliografia citada. O objetivo na aplicação deste método é descobrir as variáveis lingüísticas usadas pelo especialista.

O método de análise de protocolos ([Eric84]) consiste no projetista observar o especialista em realizar a tarefa e também narrando o porque de suas ações. A análise de protocolo deverá ser o último método a ser empregado, pois para entender a narrativa do especialista o projetista deverá já estar familiarizado com o universo do especialista e com a própria tarefa em si.

Em face do já discutido podemos estabelecer um procedimento para a utilização dos métodos de aquisição de conhecimento ([Wrig87]):

- i) inspeção ao local onde é realizada a tarefa: o objetivo é conhecer a própria tarefa em si,
- ii) começar o processo de aquisição de conhecimento usando entrevista e, ao perceber redundância de informações obtidas, deve-se trocar para o método de questionário, enfatizando dúvidas e procedimentos não esclarecidos com a utilização do método anterior,
- iii) uso da técnica de grau de repertório para descobrir as variáveis lingüísticas empregadas,
- iv) por fim o método de análise de protocolos deverá ser utilizado para conferir as informações obtidas do especialista no item (ii).

### 5.3 AJUSTE DOS PARÂMETROS.

Na Engenharia de Controle os parâmetros usados no projeto de um controlador são normalmente especificados em termos de ganho, tempo de amortecimento, tempo de subida etc, sendo que os valores das constantes do controlador podem ser determinadas por medidas, antes do sistema realmente entrar em funcionamento ([Zieg42]). Os ajustes finais podem ser realizados utilizando-se de testes ao degrau, por exemplo.

No projeto do CBRN, em contraste com a Engenharia de Controle, não existem métodos analíticos de ajustes dos parâmetros. Estes são ajustados usando-se de tentativa e erro. Entretanto algumas observações podem ser realizadas e, se levadas em consideração, podem diminuir o esforço na tarefa de calibração do controlador.

O CBRN possui quatro parâmetros de ajustes ([Mamd75]):

- i) o conjunto das variáveis de base,
- ii) a função de pertinência  $\mu$ ,
- iii) os ganhos  $e$ ,
- iv) o conjunto de regras.

O conjunto das variáveis de base representam os valores reais que o processo manipula. A obtenção destes valores não é algo difícil que possa levar a obtenção de dados errados, desta forma uma vez obtidos e confirmados através de inspeção aos equipamentos de medida, por exemplo; o projetista não deve proceder a mudanças.

A função de pertinência  $\mu$  descreve a restrição com que o operador do processo classifica os sinais de entrada em cada valor lingüístico, ou seja, em cada conjunto nebuloso. Ela deve ser utilizada como um procedimento de ajuste fino na resposta do sistema, sua alteração equivale a modificar a sensibilidade do sistema a um dado valor lingüístico. Este procedimento de ajuste da função de pertinência é semelhante ao procedimento de troca do algoritmo de integração do

controlador PI por outro, para obter maior acuidade no sinal de integração ([Li80]).

Os ganhos do CBRN devem ser ajustados na forma ilustrada na seção 2 deste capítulo. Entretanto algum ajuste fino pode ser requerido quando o sistema estiver em funcionamento e o projetista deve fazê-lo.

A última possibilidade de ajuste do CBRN é a modificação do conjunto de regras. Esta medida somente deve ser tentada se o sistema apresentar um índice de desempenho por demais degradado e nenhuma outra tentativa de ajuste tenha obtido sucesso. Neste caso um novo conjunto de regras deve ser tentado junto com o especialista e um procedimento de ajuste através dos itens ii e iii deve ser repetido, se necessário ([Hal88]).

#### 5.4 EXEMPLO DE IMPLEMENTAÇÃO DE UM CBRN.

Considere um sistema dinâmico cuja equação diferencial é do tipo:

$$a_0 \frac{dy}{dt} + y = x$$

onde  $a_0$  é sua constante de tempo. Iremos agora, a título de exemplo, aplicar o CBRN no controle de um processo com estas características.

O CBRN fará uso do erro e sua variação como parâmetros para o disparo do conjunto de regras:

$$\longrightarrow \text{erro } (e) = (\text{referencia}) - (\text{saída}) e,$$

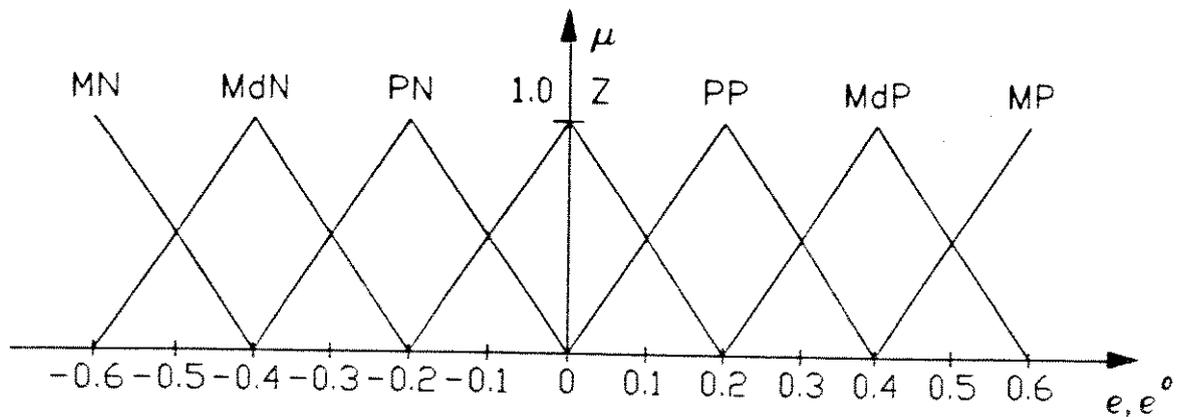
$$\longrightarrow \text{variação do erro } (e^\circ) = e_k - e_{k-1}.$$

A ação controle é dada por:

$$\rightarrow q_k = q_{k-1} + \Delta q ,$$

onde  $\Delta q$  é o valor obtido pelo processo de inferência no CBRN e  $q_{k-1}$  é a ação de controle na amostragem anterior.

O conjunto de regras usado na implementação deste CBRN é apresentado no Apêndice C e os conjuntos nebulosos usados por elas são descritos na Figura 5.5.



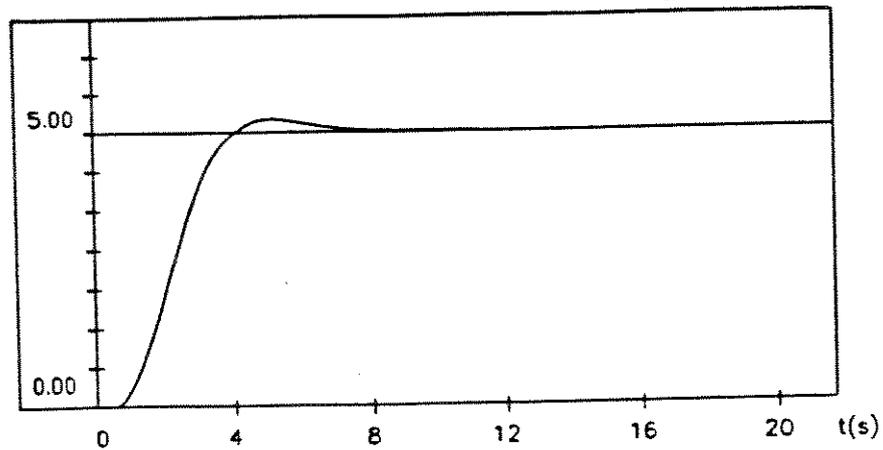
Conjuntos nebulosos dos sinais  $e$ ,  $e'$  e do sinal de saída  $y$ .

Figura 5.5

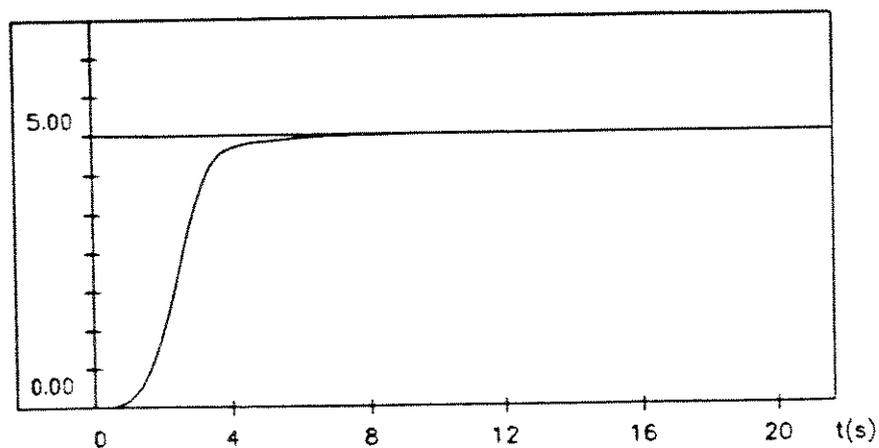
A Figura 5.6 mostra a resposta do sistema a função de transferência do tipo:

$$G(s) = \frac{e^{-T_d s}}{(T_p + s)}$$

com  $T_d = 0.2$  e  $T_p = 2.0$  sendo controlada pelo CBRN, que usa o conjunto de regras descrito anteriormente. A Figura 5.7 mostra a resposta do mesmo sistema tendo por controlador um PI.

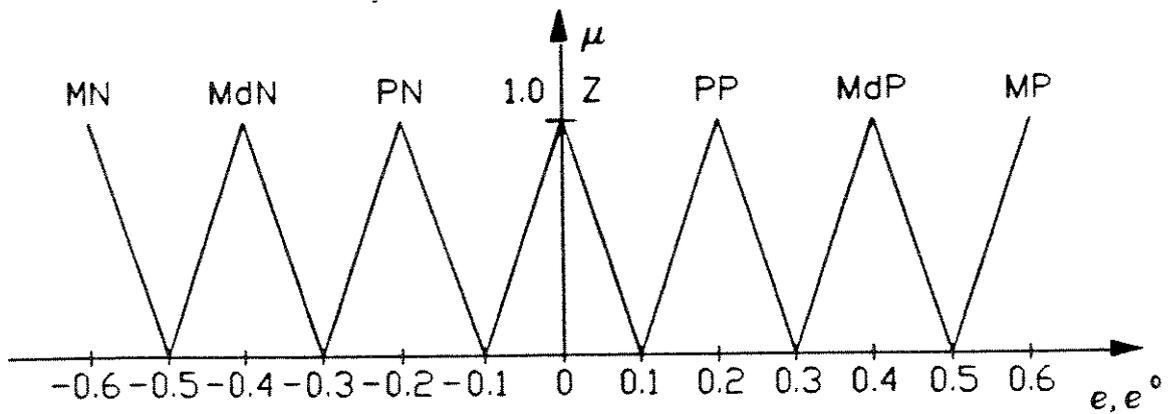


Resposta do sistema com um CBRN.  
 Figura 5.6

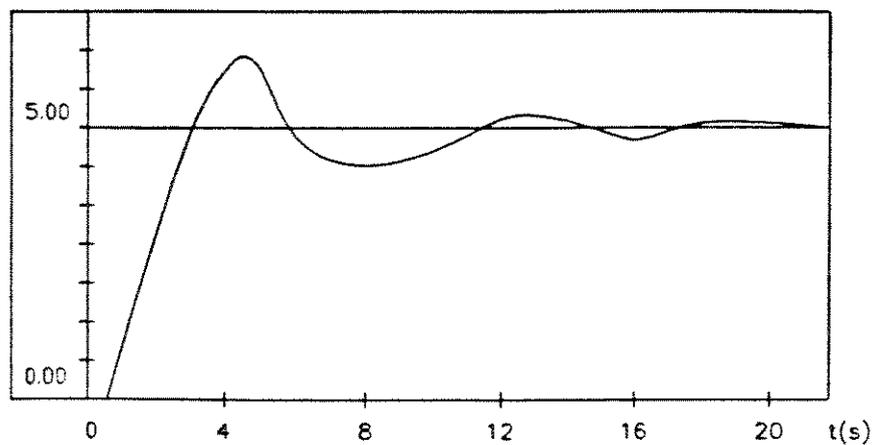


Resposta do sistema com um PI.  
 Figura 5.7

A figura 5.8 mostra os conjuntos nebulosos modificados, agora mais separados sem sobrepor-se muito. A resposta do sistema usando estes novos conjuntos nebulosos e com o mesmo conjunto de regras é mostrado na Figura 5.9.

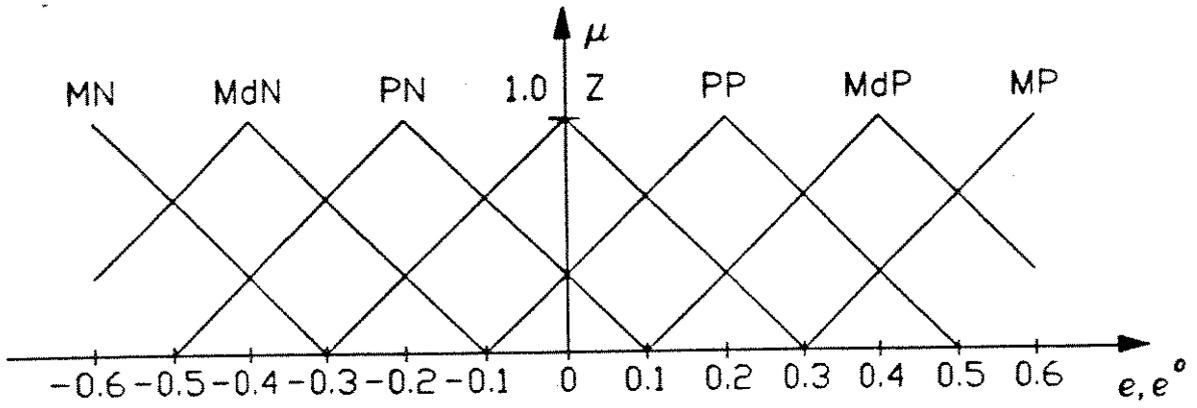


Conjuntos nebulosos modificados, não sobrepondo-se.  
 Figura 5.8

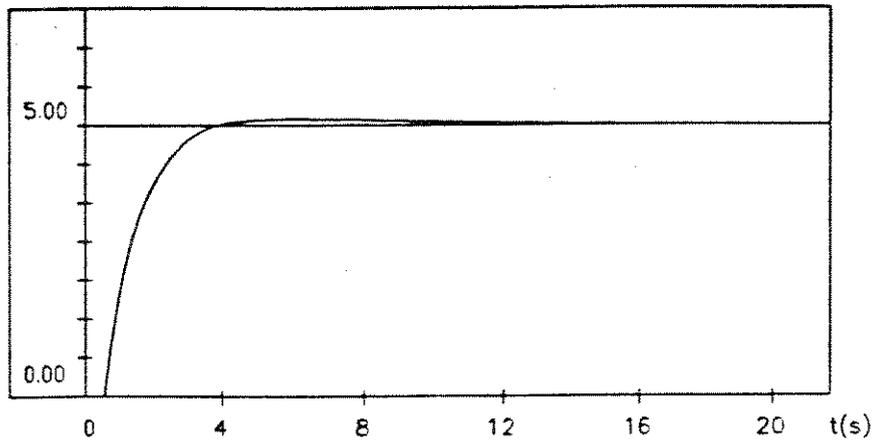


Resposta do sistema com os conjuntos nebulosos não sobrepondo-se.  
 Figura 5.9

A Figura 5.10 mostra os conjuntos nebulosos, agora, mais sobrepostos. A resposta do sistema com estes novos conjuntos nebulosos mas mantendo o mesmo conjunto de regras é descrito na Figura 5.11.



Conjuntos nebulosos modificados, sobrepondo-se mais.  
 Figura 5.10



Resposta do sistema com os conjuntos nebulosos sobrepondo-se mais.

Figura 5.11

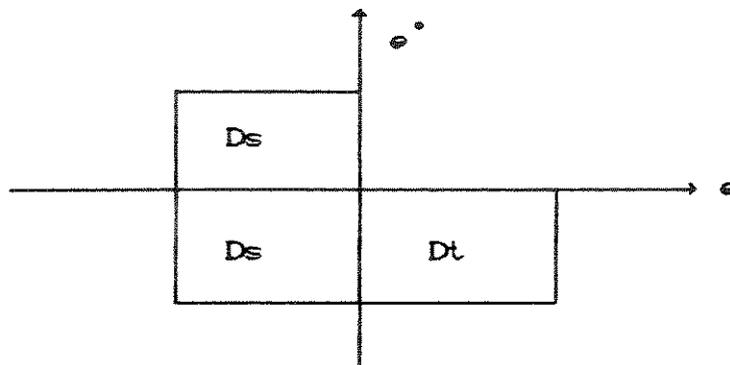
## CONCLUSÃO DA PARTE II

Na Parte II deste texto foi mostrado a arquitetura de um CBRN, discutido o seu algoritmo de inferência e sua função como módulo de controle. O Capítulo 4 foi destinado ao estudo de um modelo para o CBRN, este modelo faz uso de uma interpretação heurística do CBRN. Usando-se de algumas considerações podemos realizar um mapeamento entre o conjunto de regras do CBRN e as constantes do controlador PI. Esta relação mostra o porque da semelhança na resposta destes dois controladores.

O Capítulo 5 mostrou algumas linhas básicas de projeto a serem utilizadas no CBRN. Alguns destes métodos discutidos são baseados em técnicas de Aquisição de Conhecimento e tem-se mostrado útil durante o projeto. Um quadro resumo é:

fator	critério	típico
$\mu$	um compromisso entre alta e baixa resolução	
$n_k$	suficiente para descrever as variações de $\mu$	5 à 10
E	depende do processo, deve ser suficiente para acomodar as variáveis de base	
num de $\mu$	o suficiente para permitir o operador expressar-se em seu universo	5 à 7

Um quadro resumo sobre procedimento de ajuste do conjunto de regras do CBRN e seu efeito na resposta do sistema é:



Esta figura mostra o efeito do ajuste do conjunto de regras sobre diferentes índices de desempenho, onde Dt é o efeito de diminuir o tempo de subida e Ds é o efeito de

diminuir a sobre-elevação do sinal.

O CBRN mostrou possuir uma boa desempenho no controle de processos, sendo especialmente recomendado nos casos em que o modelo do processo é difícil de obter-se de forma acurada e precisa. Nos casos em que o modelo do processo é bem conhecido a utilização dos controladores convencionais tem mostrado-se mais vantajoso devido a já existir métodos de projeto bem desenvolvidos.

A principal limitação na utilização do CBRN é em relação a sistemas cuja dinâmica varia com o tempo, como mostrado nos dois últimos exemplos do Capítulo 5. Isto ocorre porque a mudança de algum parâmetro do sistema faz com que o conjunto de regras não represente mais a ação que o especialista tomaria, já que a dinâmica do sistema agora é outra tornando-se desconhecida para ele. Uma forma possível de se ultrapassar esta limitação é a utilização do CBRN em um módulo hierárquico que torne o sistema mais flexível.

## PARTE III

### O CONTROLADOR BASEADO EM REGRAS NEBULOSAS COMO MÓDULO DE SUPERVISÃO

#### CAPÍTULO 6

#### ANÁLISE E PROJETO DE UM SUPERVISOR UTILIZANDO REGRAS NEBULOSAS

Nos capítulos anteriores mostrou-se o CBRN atuando diretamente sobre o processo, de uma forma análoga aos controladores digitais diretos. Esta abordagem é especialmente recomendada quando não se conhece acuradamente o modelo do processo a controlar.

Entretanto o CBRN apresentou limitações de desempenho quando a constante de tempo do sistema variava no tempo. Uma forma de ultrapassar esta limitação é a utilização do CBRN em uma estrutura hierárquica capaz de dar ao sistema maior flexibilidade.

Esta estrutura será apresentada com o CBRN atuando como supervisor e um controlador PI como módulo escravo. Esta abordagem nos permitirá trabalhar com o controlador PI, o qual é largamente utilizado na indústria, em situações adversas a sua utilização ou seja: nos casos em que o modelo do processo não é conhecido com uma boa acuidade e seus parâmetros sofrem substanciais variações.

Na construção do supervisor, o qual é uma generalização do CBRN descrito na Parte II, apresentaremos um algoritmo de inferência, que possui a capacidade de incorporar dúvidas do operador sobre qual ação de controle a executar. Este

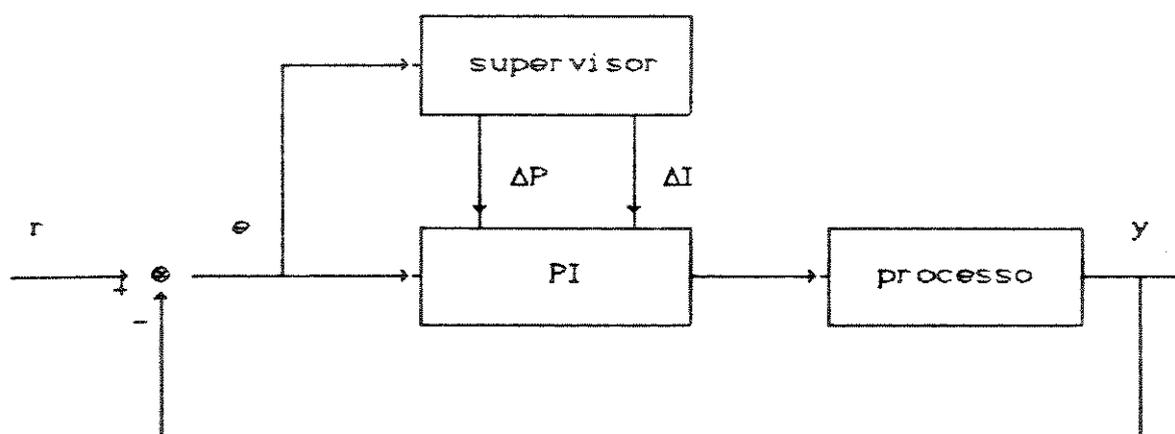
algoritmo é uma generalização do algoritmo de inferência descrito no Capítulo 3.

O capítulo é finalizado mostrando, por meio de simulação, como a utilização do supervisor permite o emprego do controlador PI em ambientes onde sua utilização não é normalmente recomendada.

## 6.1 ARQUITETURA DO SISTEMA

O módulo supervisor descrito neste capítulo possui uma função semelhante ao módulo de supervisão usado em sistemas de controle hierárquico. A Figura 6.1 mostra a arquitetura de um sistema fazendo uso deste módulo de supervisão.

Sua função no sistema é a de efetuar correções nos parâmetros do controlador PI, caso seja necessário. Para isto, o sistema recebe como parâmetro de entrada o erro, a partir do qual calcula sua derivada. Estes são os dois sinais usados pelo supervisor para disparar o conjunto de regras.

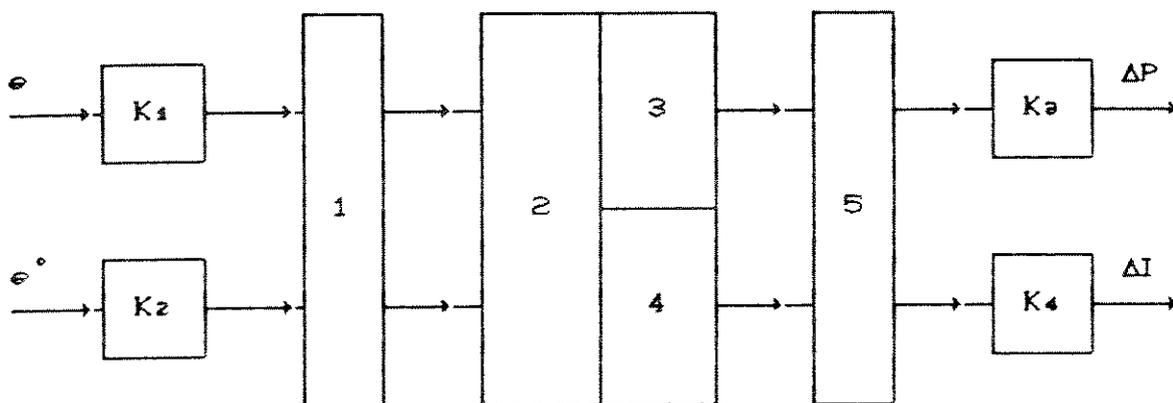


Arquitetura do sistema de controle utilizando o supervisor.

Figura 6.1

A cada instante do intervalo de amostragem o supervisor irá receber estes dois sinais, e de posse destes, ele deverá proceder a tomada de decisão de efetuar ou não correções no controlador; caso faça a correção deverá também determinar o valor desta. Este processo é executado com o supervisor disparando um conjunto de regras lingüísticas de controle, as quais traduzem a heurística no processo de ajuste do controlador PI por um experiente operador humano ([Peel88]).

O supervisor é constituído por vários módulos, com formas e funções semelhantes ao de um CBRN (descrito nos capítulos anteriores). A Figura 6.2 mostra a arquitetura do supervisor.



Arquitetura funcional do supervisor.

Figura 6.2

Deve-se notar a existência de dois conjuntos de regras distintos, um para cada parâmetro do controlador PI, sendo ambos processados pelo mesmo algoritmo de inferência ([Ollé88], [Shao88]).

Cada módulo possui as seguintes funções:

1 - converte o sinal determinístico para uma forma

- nebulosa,
- 2 - algoritmo de inferência,
  - 3 - conjunto de regras do ganho proporcional,
  - 4 - conjunto de regras do ganho integral,
  - 5 - converte o sinal na forma nebulosa para determinística.

Os valores de saída  $\Delta P$  e  $\Delta I$  são obtidos pelo processo de inferência e são os valores a serem adicionados aos parâmetros atuais do controlador PI. Estes parâmetros são ajustados na forma:

$$P_k = P_o + \Delta P_k$$

$$I_k = I_o + \Delta I_k .$$

onde  $P_o$  e  $I_o$  são os valores do controlador PI previamente ajustados,  $\Delta P_k$  e  $\Delta I_k$  são as respectivas correções a serem feitas no controlador no instante  $k$ , determinadas pelo processo de inferência; e  $P_k$  e  $I_k$  são os parâmetros do controlador no instante  $k$  depois de efetuada as correções.

O módulo de supervisão pode ser visto por duas abordagens diferentes. A primeira é feita considerando o supervisor como um módulo de controle hierárquico, usando o modelo visto no Capítulo 4, onde discutiu-se a semelhança entre o modelo heurístico do CBRN e o do controlador PI. Esta analogia entre os dois modelos justifica a semelhança encontrada entre as respostas dos sistemas com o CBRN e com o controlador PI. Este fato explica a escolha feita pelo controlador PI como o módulo de atuação do sistema hierárquico supervisor-controlador PI ([Grah88], [Krij88]).

A segunda abordagem pode ser feita interpretando o supervisor como um sistema especialista, monitorando um controlador PI; esta será a abordagem por nós adotada.

## 6.2 O ALGORITMO DE INFERÊNCIA

Em um CBRN, o projeto de controle é baseado no uso de um conjunto de regras lingüísticas de controle na forma de implicações; estas regras, como visto anteriormente ao longo deste trabalho, representam a ação de controle tomada por um experiente operador humano.

O conjunto de regras do módulo de supervisão é acionado por dois sinais: o erro e sua derivada. Neste capítulo faremos uso da versão rápida do algoritmo de inferência apresentado no Apêndice A. Temos, então, que as pertinências são dadas por:

$$e = \mu_{A_1}(u_0), \mu_{A_2}(u_0), \mu_{A_3}(u_0), \dots, \mu_{A_m}(u_0),$$

$$e^\circ = \mu_{B_1}(v_0), \mu_{B_2}(v_0), \mu_{B_3}(v_0), \dots, \mu_{B_n}(v_0).$$

Supondo que seja  $m$  o número de conjuntos nebulosos disponíveis para se classificar o erro, e  $n$  o da derivada do erro; o plano  $e$ - $e^\circ$  é desta forma dividido em  $(m \times n)$  regiões, e o número de ações de controle  $q$  é dado pelo número de conjuntos nebulosos usados para se classificar o sinal de saída; sendo assim, as  $(m \times n)$  regiões do plano  $e$ - $e^\circ$  possuem no máximo  $q$  diferentes regiões, onde  $q \leq (m \times n)$ .

O conjunto de regras é constituído por implicações do tipo:

$$\begin{array}{l} \text{Se } e \text{ é } A_i \text{ E } e^\circ \text{ é } B_j \\ \text{Então } z \text{ é } C_{ij}. \end{array}$$

onde  $C_{ij}$  representa a ação de controle tomada pelo operador quando as duas premissas da múltipla implicação são interpretadas.

Muitas vezes o operador humano não possui certeza de qual ação ele vai tomar; este fato pode ser representado admitindo-se, na conclusão da regra, mais do que uma ação de

controle sob a forma de uma combinação linear nebulosa de  $C_1, C_2, \dots, C_q$  do tipo:

$$\mu_{C_{ij}}(w) = a_1 \mu_{C_1}(w) \vee a_2 \mu_{C_2}(w) \vee \dots \vee a_q \mu_{C_q}(w),$$

onde os  $a_i$  são os pesos de cada termo na combinação linear nebulosa.

Experiências realizadas pelo autor deste trabalho mostraram que os operadores de processos expressam a ação de controle que vão tomar, quando possuídos de dúvidas, através de uma combinação linear restrita a apenas duas variáveis ([McCo88]). Esta combinação linear é da forma :

$$\delta\% \text{ em } C_i \text{ ou } (1 - \delta)\% \text{ em } C_j,$$

assim, restringiremos a combinação linear nebulosa descrita anteriormente a:

$$\mu_{C_{ij}}(w) = a_{ij} \mu_{C_i}(w) \vee (1 - a_{ij}) \mu_{C_j}(w),$$

onde  $a_{ij}$  toma valores em  $0 \leq a_{ij} \leq 1$ .

Como pode ser visto na Parte II deste trabalho o conjunto de regras não é executado usando-se de encadeamento. Cada uma das regras da base de conhecimento é executada, e a conclusão do sistema é uma composição da conclusão de cada uma das regras. Em uma analogia com os processos de inferências tradicionais, às regras, em nosso sistema, que levam para a composição do resultado um valor de pertinência  $\mu = 0,0$  são análogas as regras que nos sistemas tradicionais, não tem as suas premissas satisfeitas e portanto falham ([Buck89]).

Como as regras não sofrem encadeamento no processo de inferência, então elas podem ser representadas por uma matriz ([Chen80]), a qual denominaremos de matriz de controle, ou seja, MC.

Fazendo  $m$  e  $n$ , os números de conjuntos nebulosos para classificação do erro e sua derivada respectivamente, iguais a

cinco podemos construir MC como uma matriz quadrada 5x5. A Figura 6.3 mostra esta construção para MC.

MC =

$e \quad e'$	$B_1$	$B_2$	$B_3$	$B_4$	$B_5$
$A_1$	$C_{11}$	$C_{12}$	$C_{13}$	$C_{14}$	$C_{15}$
$A_2$	$C_{21}$	$C_{22}$	$C_{23}$	$C_{24}$	$C_{25}$
$A_3$	$C_{31}$	$C_{32}$	$C_{33}$	$C_{34}$	$C_{35}$
$A_4$	$C_{41}$	$C_{42}$	$C_{43}$	$C_{44}$	$C_{45}$
$A_5$	$C_{51}$	$C_{52}$	$C_{53}$	$C_{54}$	$C_{55}$

Matriz de controle MC 5x5.

Figura 6.3

Assim, o elemento  $C_{23}$  da matriz MC deve ser interpretado como:

Se  $e$  é  $A_2$  E  $e'$  é  $B_3$   
Então  $z$  é  $C_{23}$ .

sendo que  $C_{23}$  é uma combinação linear do tipo:

$$\mu_{C_{ij}}(w) = a_{ij} \mu_{C_i}(w) \vee (1 - a_{ij}) \mu_{C_j}(w).$$

Devido ao fato de que cada elemento  $C_{ij}$  de MC é uma combinação linear nebulosa com:

$$(1 \leq i \leq q) \text{ e } (1 \leq j \leq q),$$

podemos reescrever, então, MC como uma combinação de matrizes na forma:

$$MC = M_1 \mu_{C_1}(w) \vee M_2 \mu_{C_2}(w) \vee \dots \vee M_q \mu_{C_q}(w),$$

onde cada  $M_k$  é uma matriz formada apenas por elementos  $a_{ij}$  vistos anteriormente. Assim, seja  $C_{12}$  a conclusão da regra:

$$(A_1 \wedge B_2) \rightarrow C_{12},$$

temos que:

$$\mu_{C_{12}}(w) = a_{12} \mu_{C_1}(w) \vee (1 - a_{12}) \mu_{C_2}(w).$$

Podemos dizer, então, que o elemento  $e_{12}$  da matriz  $M_1$  tem o valor  $a_{12}$  visto na expressão anterior, e o mesmo elemento na matriz  $M_2$  tem o valor  $(1 - a_{12})$ . Os elementos  $e_{12}$  das demais matrizes  $M_k$  tem o valor zero.

A soma das  $M_k$  matrizes obedecem a propriedade:

$$M_1 + M_2 + \dots + M_q = [1],$$

onde [1] é uma matriz com todos os seus elementos iguais a 1.

Como as regras são do tipo:

$$(A_i \wedge B_j) \rightarrow C_{ij},$$

o sistema de inferência desenvolvido durante os capítulos anteriores permanece válido. Sendo a implicação descrita como uma relação, vide Capítulo 1.

Por motivo de clareza e facilidade de compreensão, demonstraremos o processo de inferência para este tipo de regra baseando-se na versão rápida do algoritmo de inferência, neste capítulo também faremos uso dos conjuntos nebulosos degenerados ( $A'$  e  $B'$ ). O leitor poderá encontrar maiores detalhes sobre este procedimento no Apêndice A.

A regra da composição esbelece que:

$$C'_{ij} = (A', B') \circ (A_i \wedge B_j) \rightarrow C_{ij}$$

Assim, usando-se os resultados do Apêndice A, temos que a conclusão da implicação anteriormente descrita é dada pela forma:

$$\mu_{C'_{ij}}(w) = \{ \mu_{A'_i}(u_0) \wedge \mu_{B'_j}(v_0) \wedge \mu_{C_{ij}}(w) \},$$

Utilizando o resultado do Lema 1 apresentado no Capítulo 3, temos o conjunto de regras na forma:

$$\begin{aligned} \mu_{C'}(w) = & [\mu_{A'_1}(u_0) \wedge \mu_{B'_1}(v_0) \wedge \mu_{C_{11}}(w)] \vee [\mu_{A'_2}(u_0) \wedge \mu_{B'_1}(v_0) \wedge \\ & \mu_{C_{21}}(w)] \vee \dots \vee [\mu_{A'_m}(u_0) \wedge \mu_{B'_1}(v_0) \wedge \mu_{C_{m1}}(w)] \vee \\ & [\mu_{A'_1}(u_0) \wedge \mu_{B'_2}(v_0) \wedge \mu_{C_{21}}(w)] \vee \dots \vee [\mu_{A'_1}(u_0) \wedge \\ & \mu_{B'_n}(v_0) \wedge \mu_{C_{1n}}(w)]. \end{aligned}$$

Esta expressão pode ser reescrita como:

$$\mu_{C'}(w) = \vee \left\{ [\mu_{A'_i}(u_0) \wedge \mu_{B'_j}(v_0) \wedge \mu_{C_{ij}}(w)] \right\},$$

$i = 1 \dots m$   
 $j = 1 \dots n$

que por sua vez pode ser colocada na forma matricial:

$$\mu_{C'}(w) = [\mu_{A'_1} \dots \mu_{A'_m}] \wedge \begin{pmatrix} \mu_{C_{11}} & \vee \dots \vee \mu_{C_{1n}} \\ \vdots & \\ \mu_{C_{m1}} & \vee \dots \vee \mu_{C_{mn}} \end{pmatrix} \wedge \begin{pmatrix} \mu_{B'_1} \\ \vdots \\ \mu_{B'_n} \end{pmatrix},$$

onde as operações de adição (+) e multiplicação (\*) foram substituídas pelos operadores max ( $\vee$ ) e min ( $\wedge$ ), respectivamente.

Uma notação curta para a equação anterior permite escrevê-la na forma :

$$\mu_{C'}(w) = M_A \wedge M_C \wedge M_B,$$

onde  $M_A$  e  $M_B$  são matrizes  $(1 \times m)$  e  $(n \times 1)$  respectivamente.

Desmembrando  $\mu_{C'_{ij}}$ , através da interpretação da implicação usada por nós, podemos reescrever a regra  $\mu_{C'_{ij}}$  como:

$$\mu_{C'_{ij}}(w) = \{ \mu_{A'_i}(u_0) \wedge \mu_{B'_j}(v_0) \wedge [\mu_{C'_v}(w) \vee \mu_{C'_k}(w)] \},$$

onde  $1 \leq w \leq q$  e  $1 \leq k \leq q$ , e que pode ser reescrita como:

$$\mu_{C'_{ij}}(w) = \{ [\mu_{A'_i}(u_0) \wedge \mu_{B'_j}(v_0) \wedge a_{ij} \mu_{C'_{ij}}(w)] \vee$$

$$[\mu_{A'_i}(u_0) \wedge \mu_{B'_j}(v_0) \wedge (1 - a_{ij}) \mu_{C'_{ij}}(w)] \}.$$

Como MC é uma matriz constituída apenas por  $\mu_{C'_{ij}}$ , podemos reescreve-la como uma combinação linear nebulosa na forma:

$$\mu_C(w) = M_A^t \wedge \{ \mu_{C'_1}(w)M_1 \vee \mu_{C'_2}(w)M_2 \vee \dots \vee \mu_{C'_q}(w)M_q \} \wedge M_B.$$

Fazendo:

$$\Psi_i = M_A^t \wedge M_i \wedge M_B,$$

e tendo:

$$M_A^t = [\mu_{A'_1}(u_0) \dots \mu_{A'_5}(u_0)]^t,$$

e

$$M_B = [\mu_{B'_1}(v_0) \dots \mu_{B'_5}(v_0)],$$

podemos reescrever  $\mu_C(w)$  como:

$$\mu_C(w) = \Psi_1 \mu_{C'_1}(w) \vee \dots \vee \Psi_i \mu_{C'_i}(w) \vee \dots \vee \Psi_q \mu_{C'_q}(w).$$

Pode-se observar na expressão que  $\mu_C(w)$  é uma

combinação linear nebulosa, onde  $\Psi_i$  é o peso de cada termo na combinação obtido pelo processo de inferência.

Como  $\mu_C(w)$  é, ainda, um conjunto nebuloso, torna-se necessário uma forma de obter um valor determinístico que possa ser, então, usado pelo controlador PI; isto é conseguido adotando-se a regra do centro de gravidade como processo de conversão.

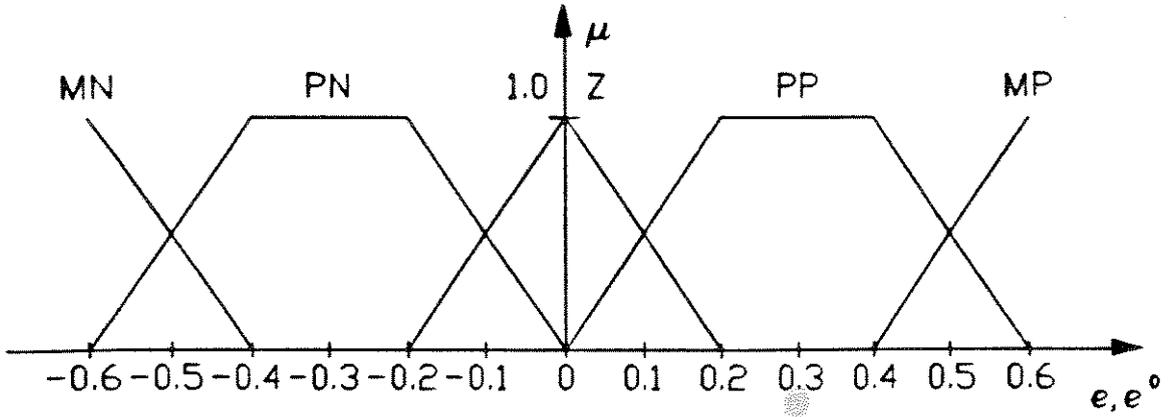
Pode-se verificar que este algoritmo, é uma generalização do algoritmo estudado nos capítulos anteriores; para verificar, basta apenas fazer o valor de  $a_{ij} = 1.0$  ou  $a_{ij} = 0.0$ . Assim, as matrizes  $M_k$  teriam apenas como elementos 1.0 ou 0.0 e teríamos então o algoritmo anterior. Deve-se observar que:

$$M_1 + M_2 + \dots + M_q = [1] ,$$

permanece válida e, portanto, poder-se-ia realizar as inferências dos capítulos anteriores usando-se esta estrutura de inferência.

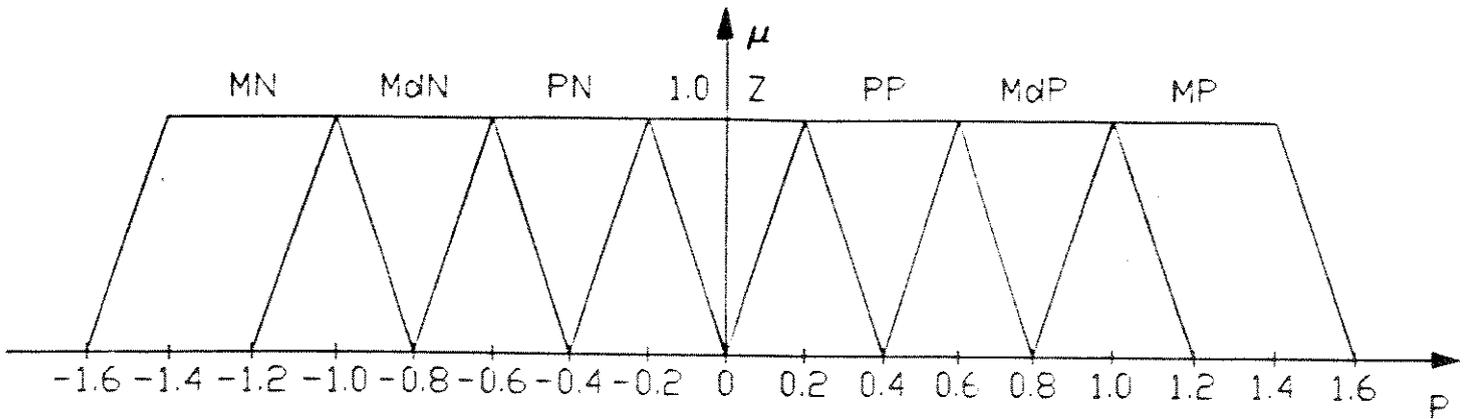
### 6.3 IMPLEMENTAÇÃO DO SUPERVISOR E RESULTADOS DE SIMULAÇÃO

A seguir a Figura 6.4 mostra os conjuntos nebulosos em que são classificados os sinais do  $e$  e da  $e^\circ$ . As Figuras 6.5 e 6.6 mostram os conjuntos nebulosos em que são classificados os sinais de saída do ganho proporcional e integral respectivamente.



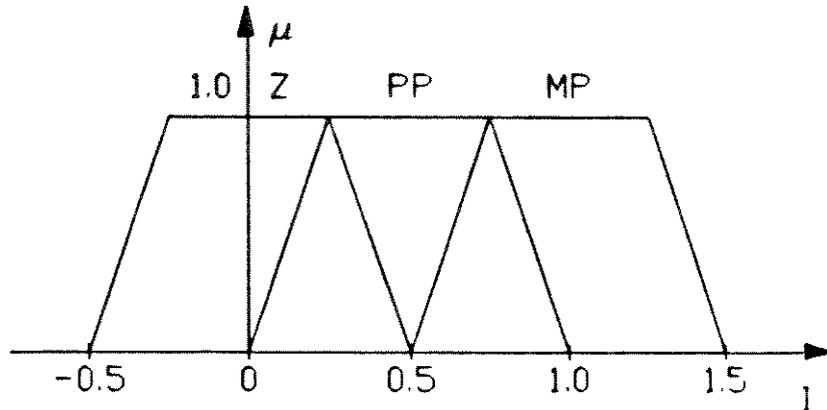
Conjuntos nebulosos utilizados na classificação do  $e$  e da  $e^\circ$ .

Figura 6.4



Conjuntos nebulosos utilizados na classificação do sinal de saída  $P$ .

Figura 6.5



Conjuntos nebulosos utilizados na classificação do sinal de saída I.

Figura 6.6

A seguir serão mostrados os resultados da simulação com e sem o supervisor para vários tipos de sinais de referência, quando o processo é do tipo:

$$G_p(s) = \frac{K_{dc} e^{-T_d s}}{(sT_p + 1)}$$

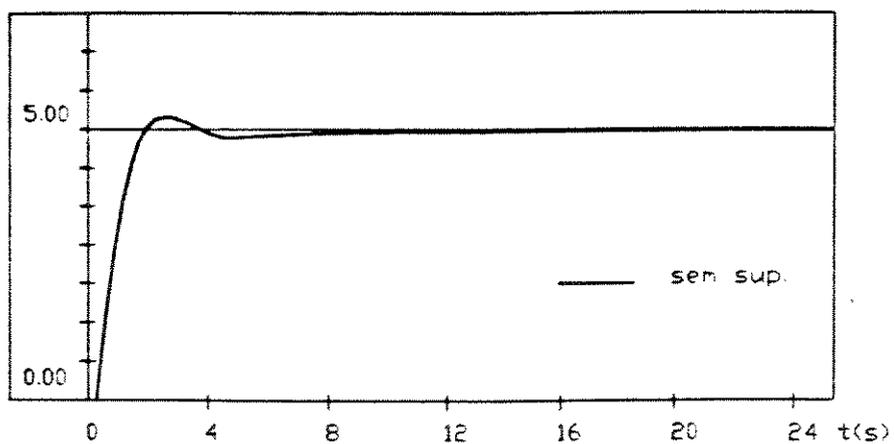
O controlador PI foi ajustado usando-se o método de Ziegler-Nichols ([Zieg42]), fazendo  $P_o = 1.2$  e  $I_o = 1.1$ ; e assumiu-se inicialmente que os parâmetros de  $G_p(s)$  são:  $T_p = 1.0$  e  $T_d = 0.32$ . O sistema é submetido a uma referência de amplitude máxima igual a 5.

A seguir é mostrado uma tabela com os parâmetros do sistema.

— Processo:	
tempo morto ( $T_d$ )	= 0.32 seg.
constante de tempo ( $T_p$ )	= 1.0 seg.
— Controlador PI:	
ganho proporcional ( $P_o$ )	= 1.2

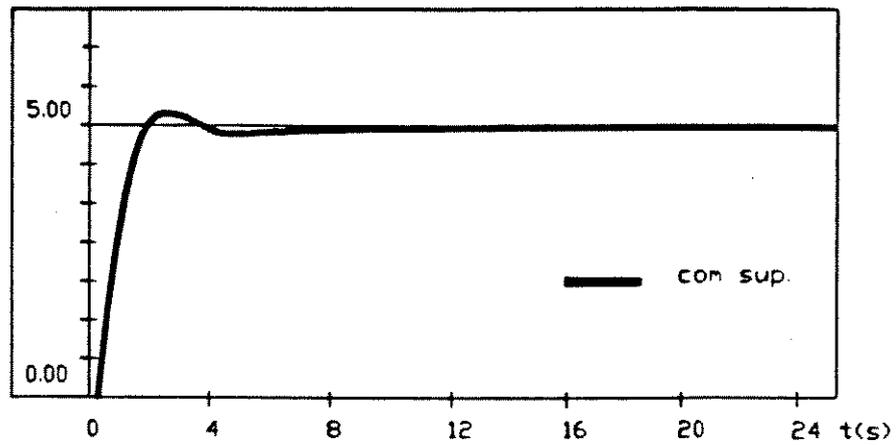
ganho integral ( $I_0$ )	= 1.1
— Supervisor:	
ganho do erro ( $K_e$ )	= 0.6
ganho da derivada do erro ( $K_e'$ )	= 1.1
ganho do integrador ( $K_i$ )	= 5.6
ganho do proporcional ( $K_p$ )	= 2.5

As Figuras 6.7 e 6.8 mostram as respostas do sistema sem e com o supervisor respectivamente, quando submetido a um sinal degrau.



Resposta do sistema sem o supervisor ao sinal degrau.

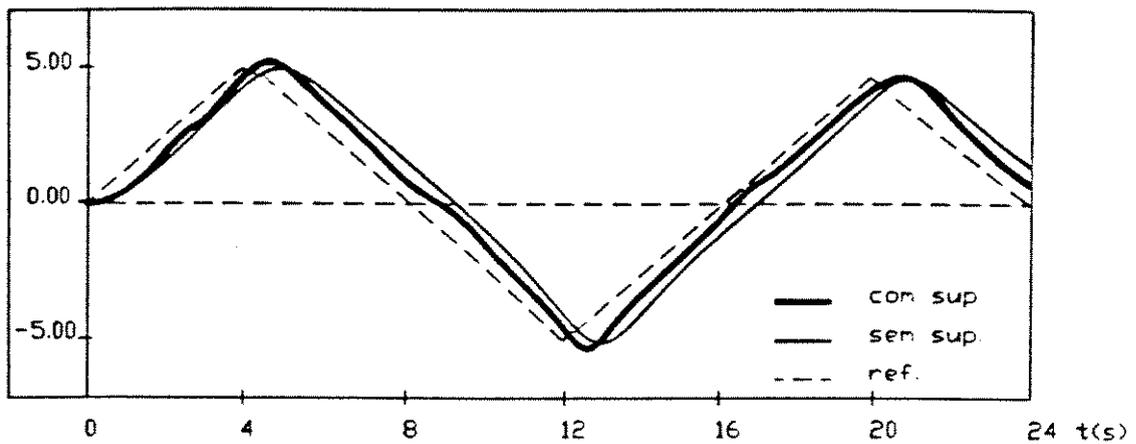
Figura 6.7



Resposta do sistema com o supervisor ao sinal degrau.

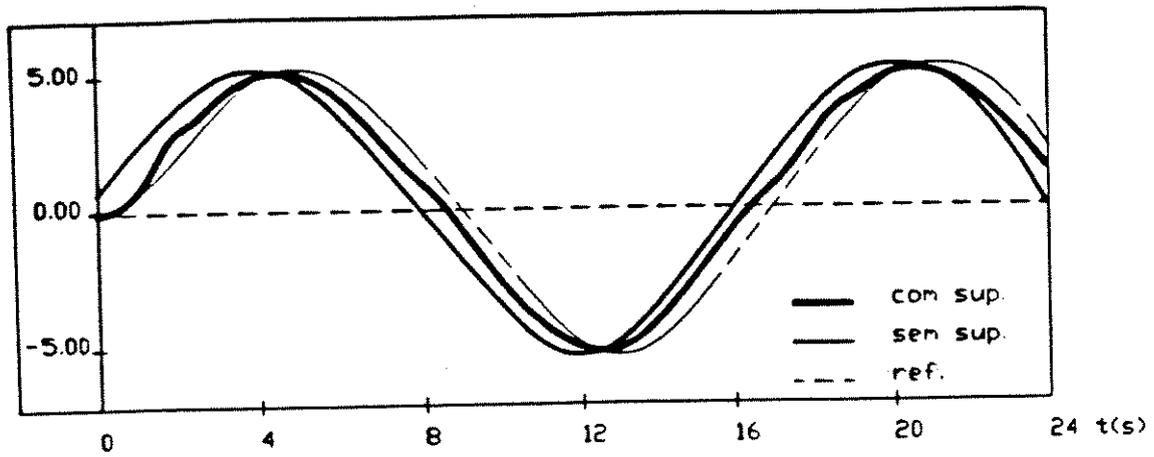
Figura 6.8

As Figuras 6.9 e 6.10 mostram respectivamente as resposta do sistema, com e sem o supervisor, aos sinais rampa e senoide.



Resposta do sistema com e sem o supervisor ao sinal rampa.

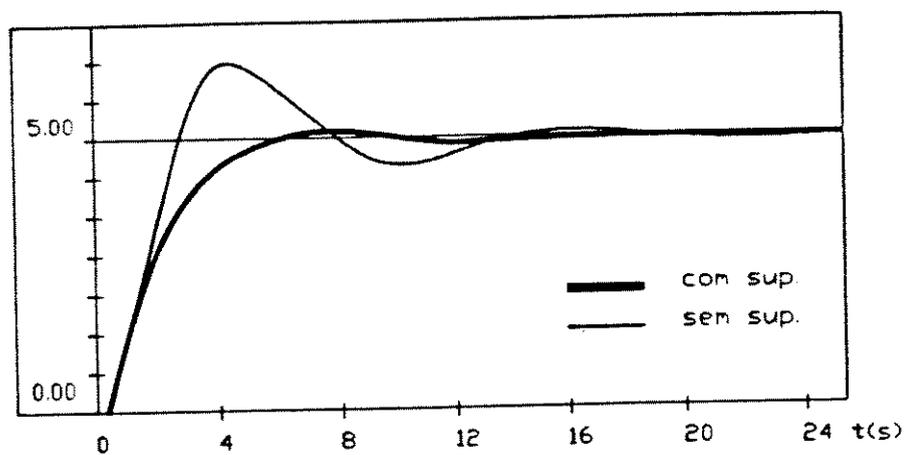
Figura 6.9



Resposta do sistema com e sem o supervisor ao sinal senoidal.

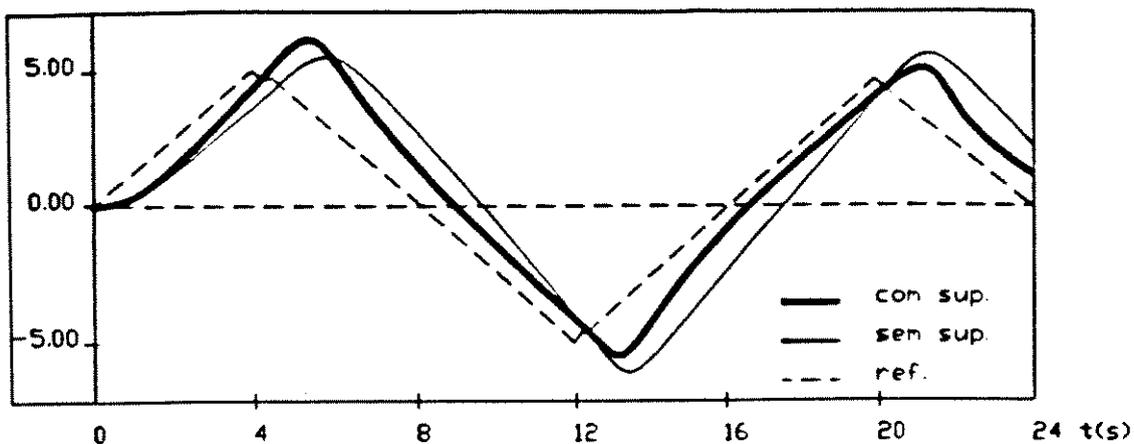
Figura 6.10

As Figuras a seguir, 6.11, 6.12 e 6.13; mostram as respectivas respostas ao degrau, rampa e senoide com o processo  $G_p(s)$  tendo sofrido uma variação em sua constante de tempo ( $T_p$ ) de 1 para 4 seg.. Os demais parâmetros do sistema permaneceram inalterados.



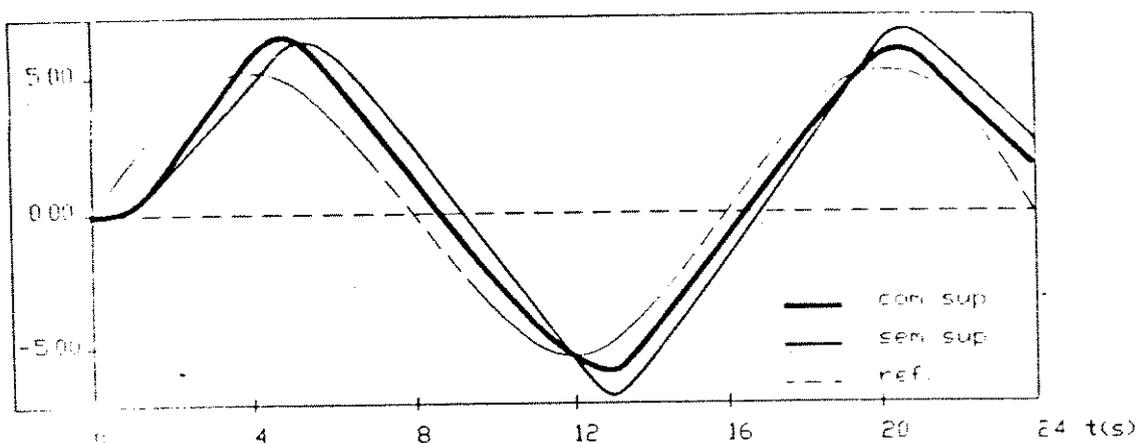
Resposta do sistema com e sem o supervisor ao sinal degrau com  $T_p = 4.0\text{seg.}$

Figura 6.11



Resposta do sistema com e sem o supervisor ao sinal rampa com  $T_p = 4.0$ seg..

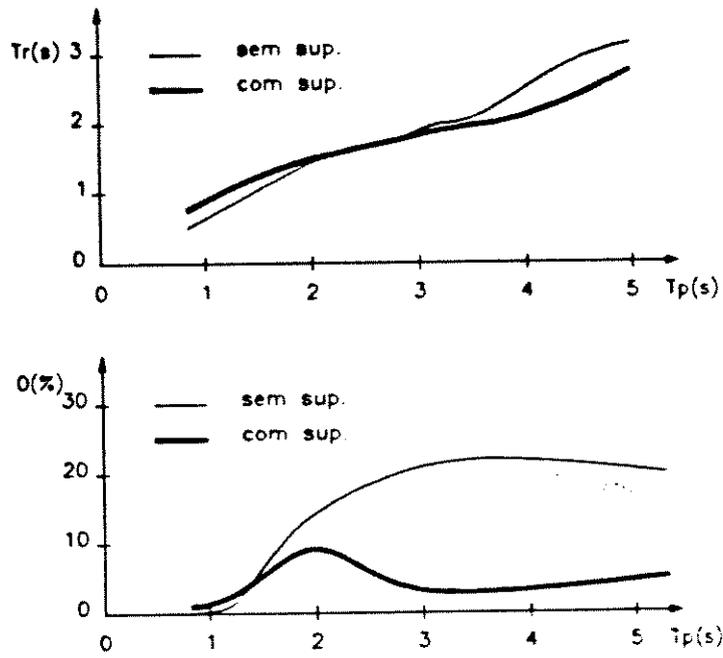
Figura 6.12



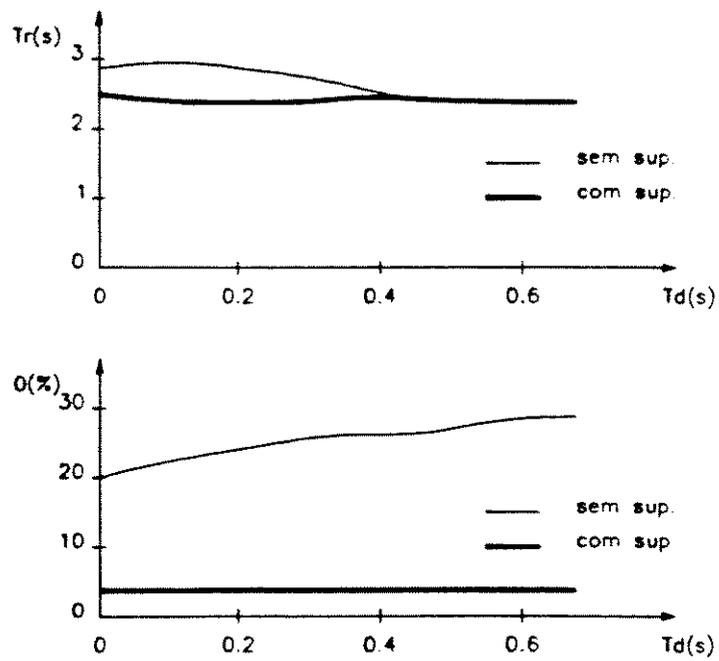
Resposta do sistema com e sem o supervisor ao sinal senoidal com  $T_p = 4.0$ seg..

Figura 6.13

Um estudo sobre a sensibilidade do sistema em relação a variação dos parâmetros  $T_d$  e  $T_p$ , são mostrados nas Figuras 6.14 e 6.15. Os símbolos  $O(\%)$  e  $T_r$  significam respectivamente sobrelevação percentual em relação a referência e tempo de ajuste.



Sensibilidade do sistema a  $T_p$ , com  $T_d = 0.32\text{seg.}$ .  
 Figura 6.14

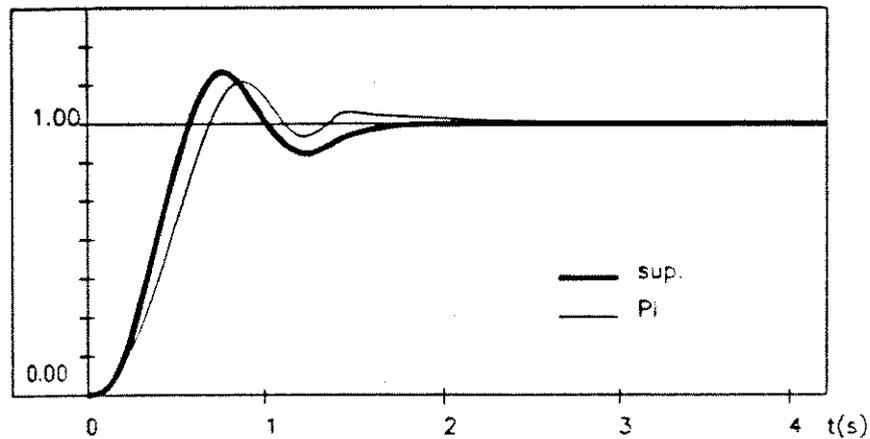


Sensibilidade do sistema a  $T_d$ , com  $T_p = 4.0\text{seg.}$ .  
 Figura 6.15

Foram realizados, também, testes com sistemas de segunda ordem. A Figura 6.16 mostra a resposta do supervisor controlando o processo:

$$G_p(s) = \frac{K_{dc}}{(sT_{p1}+1)(sT_{p2}+1)},$$

com  $T_{p1} = 1.0\text{seg}$  e  $T_{p2} = 4.0\text{seg}$ . Para esta função  $G_p(s)$  os parâmetros do controlador PI foram reajustados para os seguintes valores:  $K_P = 8.0$  e  $K_I = 1.2$ .



Resposta do sistema a uma planta de segunda ordem.

Figura 6.16

Como pode ser visto o supervisor não apresenta um bom desempenho quando  $G_p(s)$  é um sistema de segunda ordem. Isto deve-se ao fato de que o conjunto de regras do supervisor foi projetado tomando-se como modelo a reação de  $G_p(s)$  como um sistema de primeira ordem.

## 6.4 A UTILIZAÇÃO DE UM CONJUNTO DE META-REGRAS DE CONTROLE PARA MONITORAMENTO DE UM SISTEMA DE SEGUNDA ORDEM

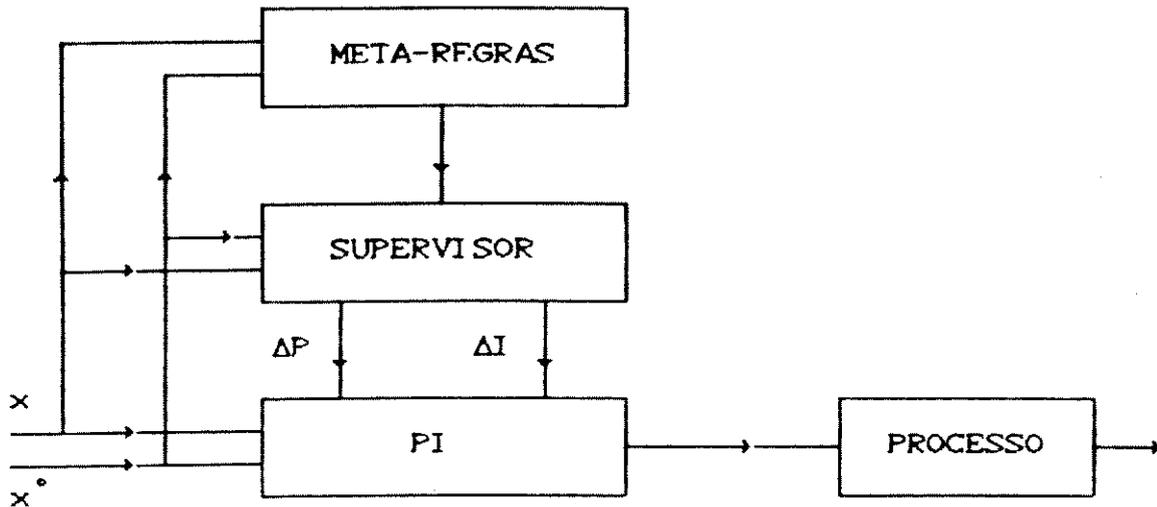
Como foi visto na seção anterior, o supervisor não apresentou um bom desempenho para sistemas de segunda ordem. Tentativas de ajustar-se o conjunto de regras para monitorar um sistema de segunda ordem foram realizadas. Contudo, observou-se que este novo conjunto de regras, embora funcionasse adequadamente para um sistema de segunda ordem, quando utilizado para monitorar um sistema de primeira ordem degradava consideravelmente a resposta.

Baseando-se nesta experiência, podemos dizer que o procedimento de ajuste dos parâmetros do controlador PI para sistemas de primeira e segunda ordem são disjuntos, ou seja, existem dois conjuntos de regras diferentes entre si, que descrevem este procedimento para cada um dos dois tipos de sistemas.

Separou-se, então, o conjunto de regras em dois subconjuntos: um para o sistema de primeira ordem e outro para o sistema de segunda ordem. Cada um destes dois subconjuntos de regras possui, por sua vez, dois grupos de regras: um para o procedimento de ajuste do parâmetro do ganho proporcional e outro para o ganho integral.

O problema agora reside em como controlar a utilização destes dois subconjuntos de regras pelo algoritmo de inferência. Para decidir qual dos dois subconjuntos de regras será acionado pelo algoritmo de inferência, projetou-se um grupo de regras para resolver esta situação de conflito. Este conjunto de regras utiliza informações sobre o estado do  $e$  e  $e'$  para determinar qual dos subconjuntos de regras serão utilizados. Este conjunto de regras é, na realidade, um conjunto de meta-regras, uma vez que elas detêm o controle sobre a execução de outras regras. Pode-se dizer, a muito grosso modo, que este conjunto de meta-regras de controle possui a função de identificar se  $Gp(s)$  é um sistema de primeira ou segunda ordem.

O novo módulo de supervisão possui a arquitetura ilustrada na Figura 6.17.



Arquitetura do supervisor utilizando um conjunto de meta-regras de controle.

Figura 6.17

As meta-regras têm a forma:

SE condição-1 E/OU condição-2  
Então executa subconjunto-i

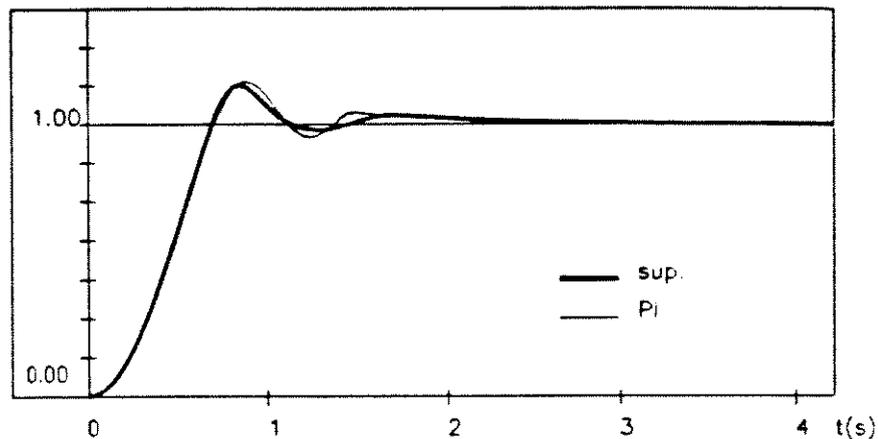
onde condição-1 e condição-2 são expressões matemáticas do tipo:  $((e^o / e) > 0,0)$ . Como pode ser visto o conjunto de meta-regras não está na forma nebulosa. O supervisor é agora formado por um sistema híbrido de representação de conhecimento: utiliza-se de regras na forma booleana e na forma nebulosa.

Um importante fato a notar é que, apesar do número total de regras para o ajuste dos parâmetros P e I terem sido dobradas, o esforço computacional do sistema permanece

inalterado. Isto deve-se ao fato de que apenas metade das regras nebulosas são executadas pelo algoritmo de inferência. A outra metade não é considerada pelo sistema, já que o conjunto de meta-regras de controle determinou sua não consideração pelo algoritmo de inferência.

Este procedimento de seleção feito pelas meta-regras de controle e a execução do algoritmo de inferência é, da mesma forma que na versão anterior do supervisor, executado continuamente.

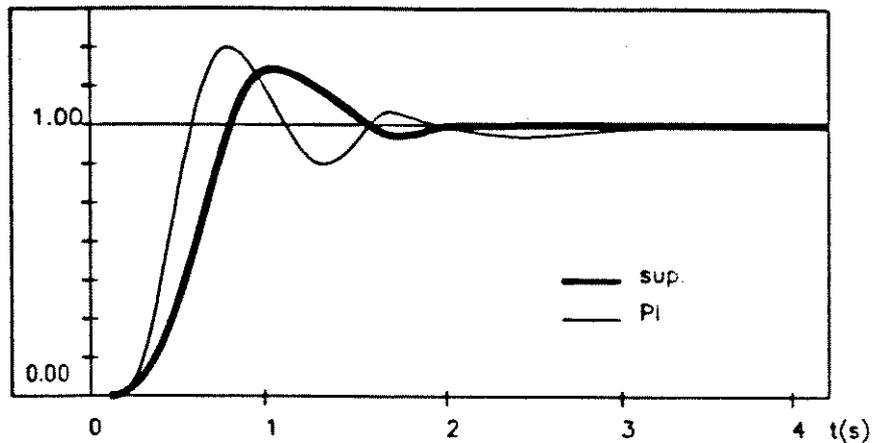
Na Figura 6.18 temos a resposta do supervisor para a mesma função de segunda ordem de  $G_p(s)$ , apresentada no final da seção 6.3.



Resposta ao sinal degrau para o novo supervisor

Figura 6.18

O comportamento do sistema quanto a variações de  $T_p$  pode ser ilustrado na Figura 6.19, quando  $T_{p2}$  tem seu valor alterado para  $T_{p2} = 6.0 \text{seg.}$



Resposta ao sinal degrau do novo supervisor  
com  $G_p(s)$  alterado,  $T_{pz} = 6.0\text{seg.}$

Figura 6.19

Estes testes mostram que a nova versão do supervisor apresenta grandes possibilidades de utilização. Contudo, um melhor estudo do novo conjunto de regras é necessário. Os testes realizados foram relativamente simples, apenas com variação de um dos polos de  $G_p(s)$  ao longo do eixo real. Um estudo mais profundo do conjunto de meta-regras e dos conjuntos de regras nebulosas, com as consequentes modificações, faz-se necessário para o sistema conseguir monitorar processos com variação simultânea em ambos os polos ou processos com polos complexos.

## CONCLUSÃO DA PARTE III

Nesta parte do texto foi apresentado uma generalização do algoritmo de inferência mostrado no Capítulo 3. Este algoritmo manipula regras nebulosas que incorporam dúvidas em relação a ação de controla a tomar.

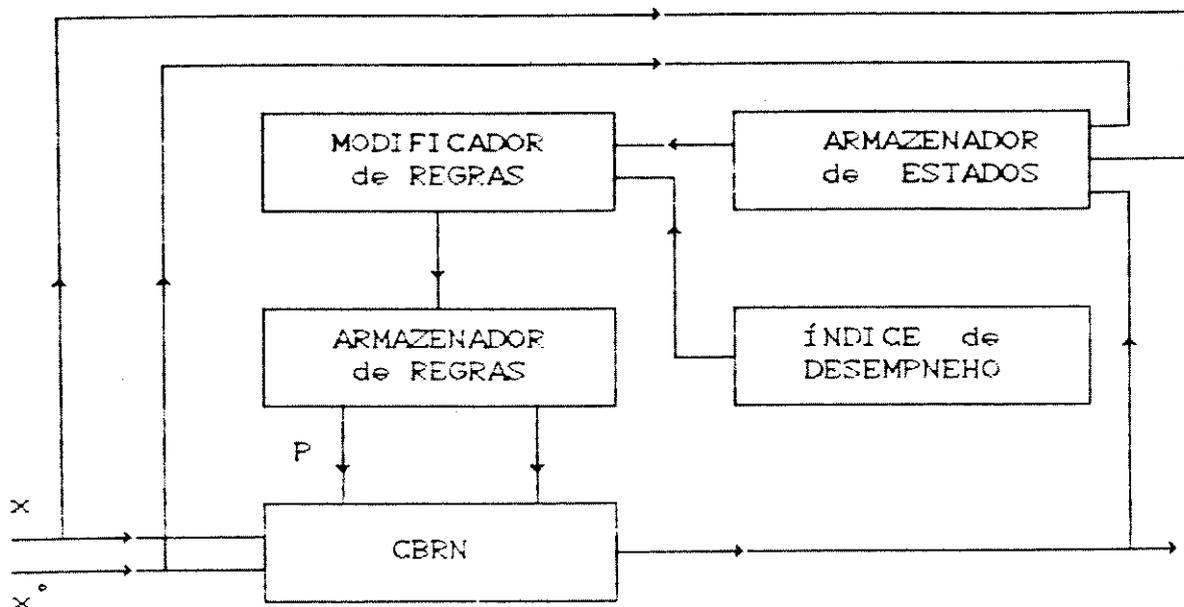
O supervisor foi desenvolvido como forma de superar as limitações encontradas no uso do CBRN. Este supervisor, o qual é uma generalização do CBRN, possui um conjunto de regras que procede as correções nos parâmetros do controlador PI, quando necessário.

Nos testes realizados a limitação apresentada pelo CBRN foi superada em parte com a utilização deste módulo de supervisão. Contudo duas restrições foram notadas: o sistema não apresenta bom desempenho para processos de segunda ordem, salvo aqueles que possuem um dos polos na origem; nem quando a constante de tempo do sistema sofre uma variação muito grande em relação ao valor pré-ajustado.

Com relação a primeira restrição, foi utilizado um conjunto de meta-regras de controle como forma de superar esta limitação. Esta abordagem mostrou ser possível e promissora, contudo um melhor refinamento e estudo deste conjunto de regras faz-se necessário.

Apesar destas limitações o supervisor mostrou ser um sistema de uso possível devido a sua simplicidade, rapidez e eficiência (dentro das limitações apresentadas). Contudo, este sistema pode ser substancialmente melhorado com a utilização de um módulo de aprendizagem. Este sistema seria uma versão auto-organizada do supervisor apresentado na seção 6.4.

A estrutura deste sistema auto-organizado ([Proc79] e [Shao88]) é apresentada na Figura 6.20. Ele é melhor visualizado se dividido em dois níveis hierárquicos. O mais baixo é composto pelo algoritmo de inferência e um conjunto de regras, ou seja, é a antiga versão do supervisor. O nível hierárquico mais alto é formado por um sistema encarregado de modificar o conjunto de regras, de um segundo com a função de avaliar o índice de desempenho e, por fim, de uma memória de trabalho onde fica armazenado o histórico do sistema.



Estrutura de um sistema auto-organizado

Figura 6.20

O sistema de índice de desempenho é formado por um grupo de regras nebulosas na forma:

Se  $e$  é  $A$  E  $e^{\circ}$  é  $B$

Então Variação no CBRN é  $\Delta C$ .

Este conjunto de regras é interpretado por um algoritmo de inferência da mesma forma que um CBRN convencional. A modificação, se necessária, a ser realizada no conjunto de regras do CBRN é então determinada. Este ajuste no conjunto de regras é realizada pelo módulo de modificação de regras.

Desta forma o novo supervisor, na realidade um sistema auto-organizado, suportaria o monitoramento de diferentes funções de  $Gp(s)$ , para as quais não teria sido previamente ajustado.

## CONCLUSÃO GERAL DO TRABALHO

A utilização dos CBRN's, mostrou ser uma abordagem importante como alternativa no controle de sistemas, nos casos em que a tarefa de controle é executada por experientes operadores humanos. Um CBRN captura, em seu conjunto de regras, a heurística destes operadores.

Entretanto, estes operadores não têm o seu conhecimento disposto em uma forma estruturada e organizada. Isto faz com que o processo de aquisição de conhecimento dos operadores seja uma atividade cansativa, demorada e muito suscetível a erros.

Esta restrição é, hoje, a responsável pela limitação em sua utilização apenas nos casos descritos acima. Em resposta a isto, muitos pesquisadores vem trabalhando em áreas como controlabilidade e análise de estabilidade destes controladores.

Uma outra abordagem na tentativa de superar estas limitações é o estudo de módulos de aprendizagem para o CBRN, capaz de torna-lo um sistema auto-organizado. Esta abordagem é muito promissora e importante por dois motivos. O primeiro, por permitir a utilização do CBRN em sistemas cujas dinâmicas sofrem grandes variações. O segundo, o fato da acuidade e da precisão no projeto do conjunto de regras não serem tão necessários. Isto deve-se à possibilidade do sistema, com o tempo, "conhecer" a dinâmica do processo e "aprender a controlá-la"; ou seja, o sistema possui a capacidade de alterar o seu conjunto de regras.

## APÊNDICE A

### UMA VERSÃO RÁPIDA DO ALGORITMO DE INFERÊNCIA

Apresentou-se no Capítulo 3 o algoritmo de inferência utilizado na interpretação do conjunto de regras do CBRN. Contudo, quando  $A'$  na inferência:

antecedente 1 : Se  $x$  é  $A$  Então  $y$  é  $B$   
antecedente 2 :  $x$  é  $A'$

---

consequente :  $y$  é  $B'$

é um conjunto nebuloso degenerado, uma grande simplificação no algoritmo anterior é possível quando  $A'$  é um conjunto nebuloso degenerado ([Mand77a]).

#### *Definição:*

Um conjunto nebuloso degenerado é um conjunto em que todos os seus elementos são iguais a 0.0, com exceção de  $\mu(u_0) = 1.0$ .

■

Como vimos, a inferência para um sistema de duas variáveis linguísticas tem a seguinte forma:

$$B' = A' \circ (A \rightarrow B),$$

onde  $A$  e  $A'$  são conjuntos nebulosos com  $m$  elementos, e  $B$  um conjunto nebuloso com  $n$  elementos. Por questão de praticidade os seus elementos serão aqui representados na forma:

$$\begin{aligned} A &= \{a_1, a_2, \dots, a_m\}, \\ B &= \{b_1, b_2, \dots, b_n\}, \\ A' &= \{a'_1, a'_2, \dots, a'_m\}. \end{aligned}$$

A implicação  $(A \rightarrow B)$  possui a seguinte forma:

$$(A \rightarrow B) = \{(a_1 \wedge b_1), (a_1 \wedge b_2), \dots, (a_1 \wedge b_n), (a_2 \wedge b_1), (a_2 \wedge b_2), \dots, (a_2 \wedge b_n), (a_m \wedge b_1), (a_m \wedge b_2), \dots, (a_m \wedge b_n)\},$$

esta expressão é o produto cartesiano  $U \times V$ . Denominando

$$(a_i \wedge b_j) = r_{ij},$$

a inferência  $[A' \circ (A \rightarrow B)]$  tem a seguinte forma:

$$[A' \circ (A \rightarrow B)] = \{(a'_1 \wedge r_{11}) \vee (a'_2 \wedge r_{21}) \vee \dots \vee (a'_m \wedge r_{m1}); (a'_1 \wedge r_{12}) \vee (a'_2 \wedge r_{22}) \vee \dots \vee (a'_m \wedge r_{m2}); (a'_1 \wedge r_{1m}) \vee (a'_2 \wedge r_{2m}) \vee \dots \vee (a'_m \wedge r_{nm})\},$$

como  $A'$  é um conjunto nebuloso degenerado, temos que apenas o elemento  $a'_k$  é diferente de zero. Desta forma, contribuem para a expressão  $[A' \circ (A \rightarrow B)]$  apenas este elemento  $a'_k$ , já que todos os outros elemento  $a'_i$  são zero. Então, a expressão anterior reduz-se à forma:

$$[A' \circ (A \rightarrow B)] = \{(a'_k \wedge r_{k1}) \vee (a'_k \wedge r_{k2}) \vee \dots \vee (a'_k \wedge r_{km})\}.$$

Pela definição de conjunto degenerado temos  $a'_k = 1.0$ . O termo  $r_{kj}$  esta no intervalo fechado  $[0.0; 1.0]$ , então, temos que:

$$[A' \circ (A \rightarrow B)] = \{r_{k1}, r_{k2}, \dots, r_{km}\},$$

onde  $r_{kj} = (a_k \wedge b_j)$ .

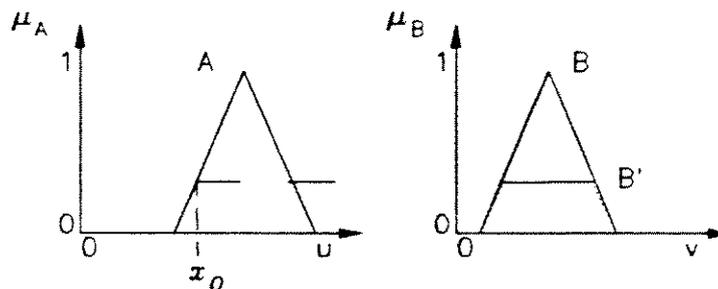
O elemento  $a_k$  representa a medida da pertinência do elemento  $u_0$  no conjunto nebuloso  $A$ , ou seja,  $\mu_A(u_0) = a_k$ . Desta forma a expressão final de  $B'_i$  tem a forma:

$$B'_i = [A' \circ (A_i \rightarrow B_i)],$$

$$B'_i = \wedge [\mu_A(u_0); B_i],$$

$$B' = \vee_{i=1, \dots, n} B'_i.$$

A Figura A.1 mostra este tipo de inferência.



Exemplo geométrico de inferência para regras  $A \rightarrow B$  com o novo algoritmo.

Figura A.1

Esta expressão é importante por permitir-nos grande economia de esforço computacional. Vamos considerar agora um segundo tipo de regras, regras de três variáveis linguísticas:

$$(A_i \wedge B_i) \rightarrow C_i.$$

Utilizando o Lema 3 do Capítulo 3, temos que:

$$C_i' = (A' \cdot B') \cdot [(A_i \wedge B_i) \rightarrow C_i] ,$$

que pode ser reescrito como:

$$C_i' = [A' \cdot (A_i \rightarrow C_i)] \wedge [B' \cdot (B_i \rightarrow C_i)] .$$

Tomando  $A'$  e  $B'$  como conjuntos nebulosos degenerados, temos que:

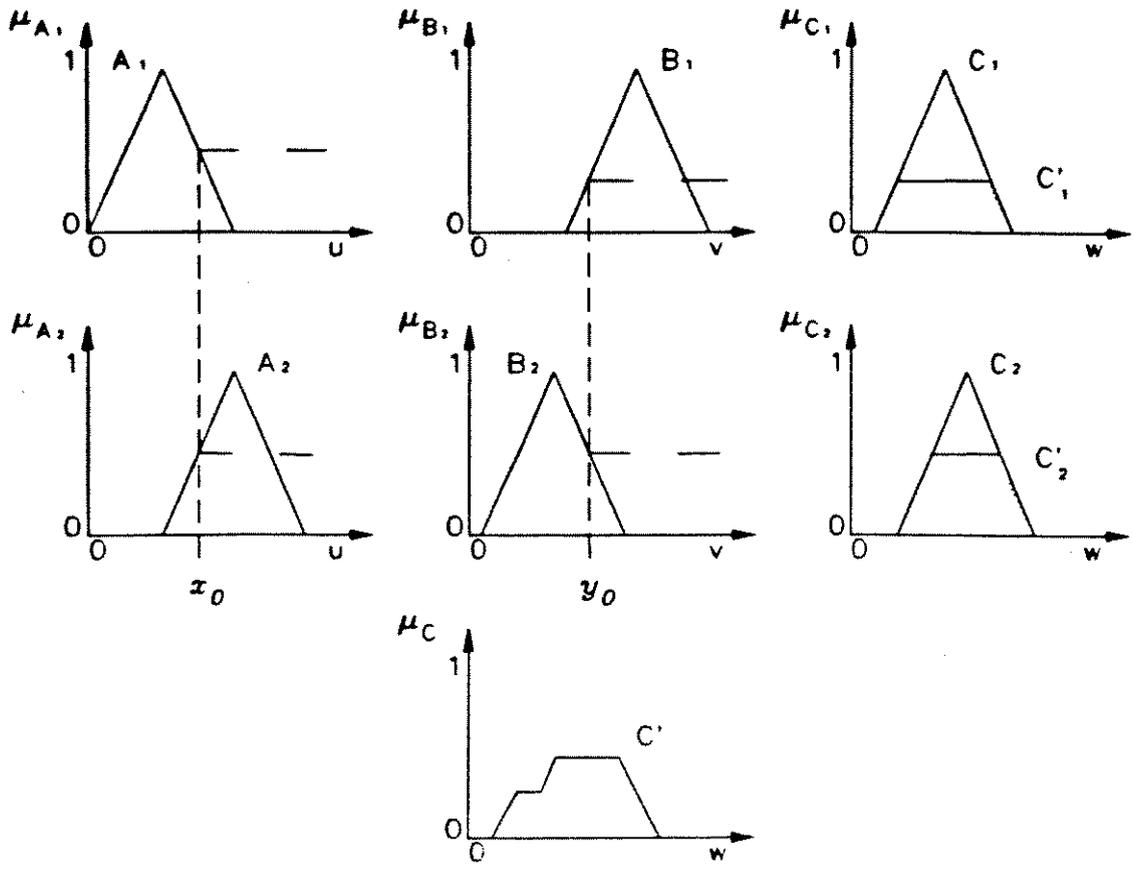
$$C_i' = [\mu_A(u_0) \wedge C_i] \wedge [\mu_B(v_0) \rightarrow C_i] ,$$

que equivale a:

$$C_i' = [\mu_A(u_0) \wedge \mu_B(v_0) \wedge C_i] ,$$

$$C' = \bigvee_{i=1 \dots n} C_i' .$$

A Figura A.2 mostra este algoritmo de inferência para um conjunto de duas regras.



Exemplo geométrico de inferência para regras  $(A \wedge B) \rightarrow C$  utilizando o novo algoritmo.

Figura A.2

Deve-se comparar estes exemplos com os apresentados no Capítulo 3, e comprovar sua maior eficiência.

## APÊNDICE B

### O MÉTODO DE INFERÊNCIA DE ZADEH

Várias são as abordagens propostas para a interpretação da implicação nebulosa ( $\rightarrow$ ) ([Mizu78]). Neste trabalho foi utilizado o método proposto por E. Mamdani ([Mamd74]). Este método possui algumas diferenças em relação ao trabalho original de L. Zadeh ([Zade73]). Mostraremos a seguir esta diferença.

Seja a inferência

antecedente 1 : Se  $x$  é  $A$  Então  $y$  é  $B$

antecedente 2 :  $x$  é  $A'$

---

consequente :  $y$  é  $B'$

que pode ser representada pela regra da composição:

$$B' = A' \cdot (A \rightarrow B) .$$

Segundo Zadeh, a implicação deve ter a forma:

$$\mu_{A \rightarrow B} = [ 1 \wedge ( 1 - \mu_A(u) + \mu_B(v) ) ] ,$$

a regra da composição tem, então, a forma:

$$\mu_{B'}(v) = \bigvee_u \{ \mu_A(u) \wedge [ 1 \wedge ( 1 - \mu_A(u) + \mu_B(v) ) ] \}.$$

Se  $A'$  for um conjunto nebuloso degenerado a expressão anterior pode ser simplificada de forma semelhante ao procedimento descrito no Apêndice A. Então,  $B'$  toma a forma:

$$\mu_{B'}(v) = \bigvee_u \{ \mu_A(u_0) \wedge [ 1 \wedge ( 1 - \mu_A(u) + \mu_B(v) ) ] \}.$$

Consideremos agora a forma de inferência na qual a hipótese da implicação nebulosa é da forma:

antecedente 1 : Se  $x$  é  $A$  então Se  $y$  é  $B$  Então  $z$  é  $C$

antecedente 2 :  $x$  é  $A'$ ,  $y$  é  $B'$

---

consequente :  $z$  é  $C'$

que pode ser escrita como:

$$C' = (A', B') \circ [(A \rightarrow B) \rightarrow C].$$

No Capítulo 3 apresentamos o *Lema 2*:

$$(A \wedge B) \rightarrow C = (A \rightarrow C) \wedge (B \rightarrow C),$$

esta propriedade é válida, como ficou demonstrada, quando a implicação é dada pelo método de Mamdani. Quando a implicação é estabelecida na forma proposta por Zadeh o *Lema 2* toma a forma:

$$(A \wedge B) \rightarrow C = (A \rightarrow C) \vee (B \rightarrow C).$$

Esta expressão pode ser demonstrada de forma análoga a anterior.

Utilizando o *Lema 3*, ele não leve em consideração a forma como é descrita a implicação, temos que:

$$(A', B') \bullet [(A \rightarrow B) \rightarrow C] = [A' \bullet (A \rightarrow C)] \vee [B' \bullet (B \rightarrow C)].$$

ou seja:

$$\mu_{C_i}(w) = \left\{ \bigvee_u \left\{ \mu_{A_i}(u) \wedge [1 \wedge (1 - \mu_{A_i}(u) + \mu_{C_i}(w))] \right\} \right. \\ \left. \bigvee \left\{ \bigvee_v \left\{ \mu_{B_i}(v) \wedge [1 \wedge (1 - \mu_{B_i}(v) + \mu_{C_i}(w))] \right\} \right\} \right\}.$$

Supondo  $A'$  e  $B'$  sejam conjuntos nebulosos degenerados temos:

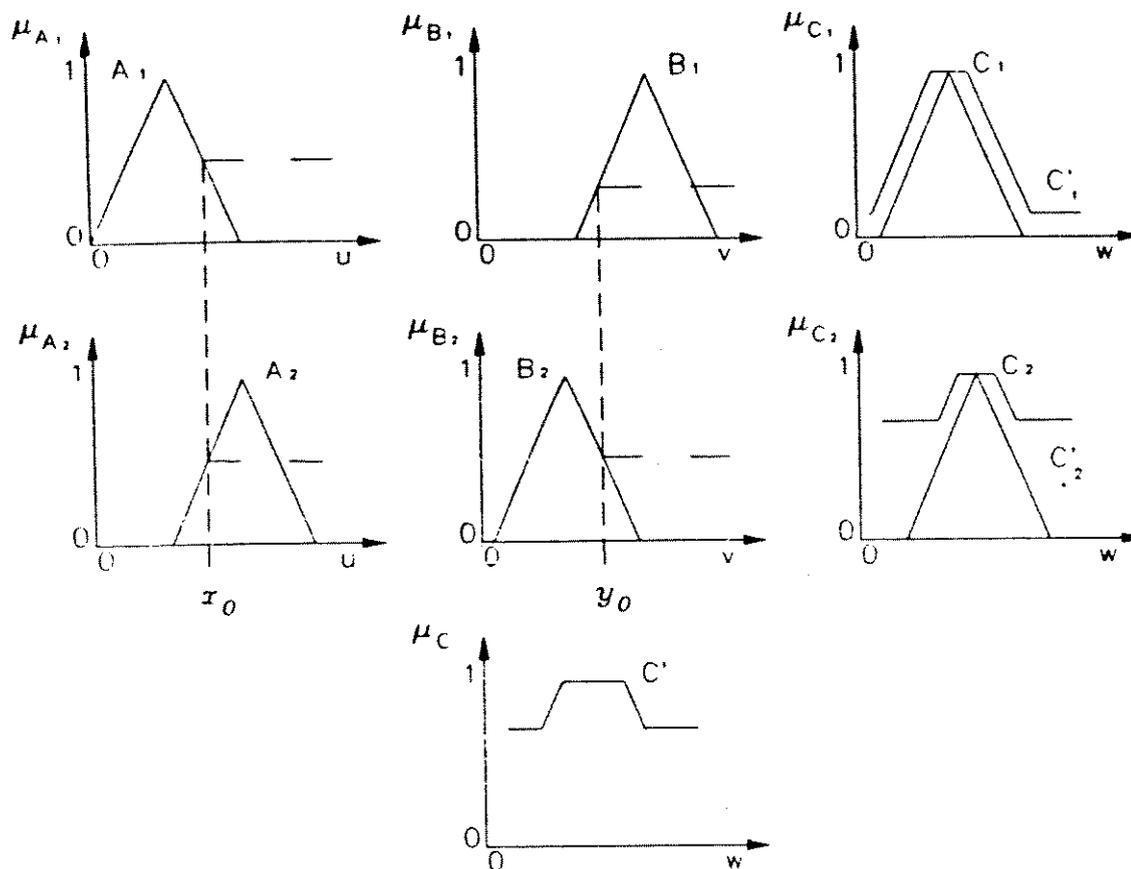
$$\mu_{C_i}(w) = [1 \wedge (1 - \mu_{A_i}(u_0) + \mu_{C_i}(w))] \vee [1 \wedge (1 - \mu_{B_i}(v_0) + \mu_{C_i}(w))],$$

que equivale a:

$$\mu_{C_i}(w) = 1 \wedge \{ 1 - [\mu_{A_i}(u_0) + \mu_{B_i}(v_0)] + \mu_{C_i}(w) \},$$

$$\mu_C(w) = \bigvee_{i=1 \dots n} \mu_{C_i}(w).$$

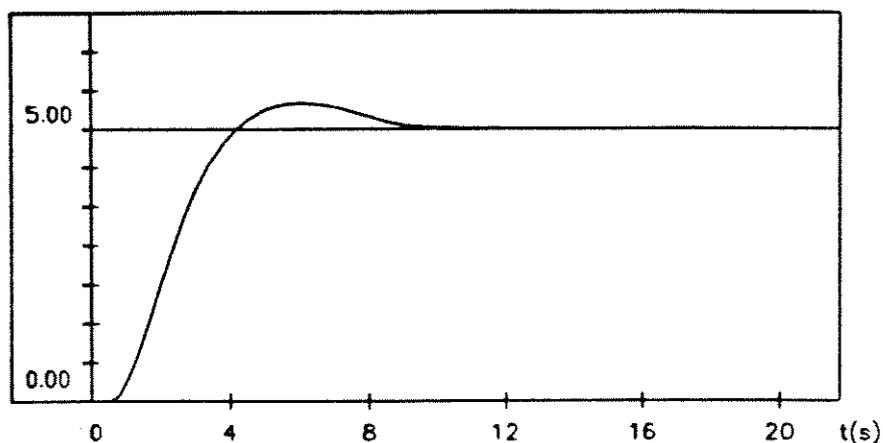
A Figura B.1 mostra um exemplo geométrico deste tipo de inferência.



Exemplo geométrico de inferência para regras do tipo  $(A \wedge B) \rightarrow C$  utilizando o método de Zadeh.

Figura B.1

A Figura B.2 apresenta a resposta do CBRN utilizando a abordagem de Zadeh no controle do processo  $G_p(s)$  apresentado no Capítulo 5. Os conjuntos nebulosos são os mesmos da Figura 5.5 e o conjunto de regras o descrito no Apêndice C.



Resposta ao degrau do CBRN utilizando o método de Zadeh.

Figura B.2

A diferença entre as respostas do CBRN usando este método e a vista no Capítulo 3 deve-se, justamente, as diferentes formas de se interpretar a implicação ( $\rightarrow$ ).

No método de Mamdani a conclusão da regra é interpretada como uma "intersecção" entre os conjuntos nebulosos da premissa e da conclusão.

Já no método de Zadeh, a expressão:

$$\mu_{C_i}(w) = [1 \wedge [1 - (\mu_A(u_0) \wedge \mu_B(v_0)) + \mu_{C_i}(w)]] ,$$

é o complemento da intersecção entre os elementos da premissa adicionados da conclusão, com este complemento sendo limitado em 1. Devido a isto temos que:

$$\mu_{C_i}(w) \geq \mu_{C_i}(w) .$$

Podemos dizer que a conclusão da regra  $\mu_C$ , contém um grau de certeza maior do que o existente nas premissas  $\mu_A$  e  $\mu_B$ , e no termo  $\mu_C$ ; o que é um contra senso. Esta forma de interpretar-se as regras é a responsável por esta diferença entre as respostas.

## APÊNDICE C

### LISTAGEM DOS CONJUNTOS DE REGRAS UTILIZADOS NOS CAPÍTULOS 5 E 6

O conjunto de regras utilizado no Capítulo 5 é apresentado a seguir, são 13 regras onde Z, MN, MdP etc são mnemônicos significando respectivamente Zero, Muito Negativo, Medianamente Positivo etc.

Se erro é MN E der. do erro é Z  
Então y é MN

OU

Se erro é MdN E der. do erro é Z  
Então y é MdN

OU

Se erro é PN E der. do erro é Z  
Então y é PN

OU

Se erro é Z E der. do erro é Z  
Então y é Z

OU

Se erro é PP E der. do erro é Z  
Então y é PP

OU

Se erro é MdP E der. do erro é Z  
Então y é MdP

OU

Se erro é MP E der. do erro é Z  
Então y é MdP

OU

Se erro é Z E der. do erro é MN  
Então y é MN

OU

Se erro é Z E der. do erro é MdN  
Então y é MdN

OU

Se erro é Z E der. do erro é PN  
Então y é PN

OU

Se erro é Z E der. do erro é PP  
Então y é PP

OU

Se erro é Z E der. do erro é MdP  
Então y é MdP

OU

Se erro é Z E der. do erro é MP  
Então y é MdP

O supervisor apresentado no Capítulo 6 possui dois conjuntos de regras. O do ganho proporcional  $K_p$  é descrito abaixo.

SE erro é MN E der. do erro é MN  
Então  $K_p$  é Z.

OU

Se erro é MN E der. do erro é PN  
Então  $K_p$  é Z.

OU

Se erro é MN E der. do erro é Z  
Então  $K_p$  é Z.

OU

Se erro é MN E der. do erro é PP  
Então Kp é Z.

OU

Se erro é MN E der. do erro é MP  
Então Kp é Z.

OU

Se erro é PN E der. do erro é MN  
Então Kp é Z.

OU

Se erro é PN E der. do erro é PN  
Então Kp é PN.

OU

Se erro é PN E der. do erro é Z  
Então Kp é 10% MN ou 90% MdN.

OU

Se erro é PN E der. do erro é PP  
Então Kp é PN.

OU

Se erro é PN E der. do erro é MP  
Então Kp é Z.

OU

Se erro é Z E der. do erro é MN  
Então Kp é Z.

OU

Se erro é Z E der. do erro é PN  
Então Kp é PN.

OU

Se erro é Z E der. do erro é Z  
Então Kp é Z.

OU

Se erro é Z E der. do erro é PP  
Então Kp é MdP.

OU

Se erro é Z E der. do erro é MP  
Então Kp é Z.

OU

Se erro é PP E der. do erro é MN  
Então Kp é Z.

OU

Se erro é PP E der. do erro é PN  
Então Kp é MdP.

OU

Se erro é PP E der. do erro é Z  
Então Kp é 90% MdP ou 10% MP.

OU

Se erro é PP E der. do erro é PP  
Então Kp é MdN.

OU

Se erro é PP E der. do erro é MP  
Então Kp é Z.

OU

Se erro é MP E der. do erro é MN  
Então Kp é Z.

OU

Se erro é MP der. do erro é PN  
Então Kp é Z.

OU

Se erro é MP E der. do erro é Z  
Então Kp é Z.

OU

Se erro é MP E der. do erro é PP  
Então Kp é Z.

OU

Se erro é MP E der. do erro é MP  
Então Kp é Z.

O conjunto de regras do ganho integral é visto a seguir.

Se erro é MN E der. do erro é MN  
Então Ki é Z.

OU

Se erro é MN E der. do erro é PN  
Então Ki é Z.

OU

Se erro é MN E der. do erro é Z  
Então Ki é Z.

OU

Se erro é MN E der. do erro é PP  
Então Ki é Z.

OU

Se erro é MN E der. do erro é MP  
Então Ki é Z.

OU

Se erro é PN E der. do erro é MN  
Então Ki é Z.

OU

Se erro é PN E der. do erro é PN  
Então Ki é PN.

OU

Se erro é PN E der. do erro é Z  
Então Ki é PP.

OU

Se erro é PN E der. do erro é PP  
Então Ki é PN.

OU

Se erro é PN E der. do erro é MP  
Então Ki é Z.

OU

Se erro é Z E der. do erro é MN  
Então Ki é Z.

OU

Se erro é Z E der. do erro é PN  
Então Ki é PP.

OU

Se erro é Z E der. do erro é Z  
Então Ki é Z.

OU

Se erro é Z E der. do erro é PP  
Então Ki é PP.

OU

Se erro é Z E der. do erro é MP  
Então Ki é Z.

OU

Se erro é PP E der. do erro é MN  
Então Ki é Z.

OU

Se erro é PP E der. do erro é PN  
Então Ki é MdP.

OU

Se erro é PP E der. do erro é Z  
Então Ki é PP.

OU

Se erro é PP E der. do erro é PP  
Então Ki é MdN.

OU

Se erro é PP E der. do erro é MP  
Então Ki é Z.

OU

Se erro é MP E der. do erro é MN  
Então Ki é Z.

OU

Se erro é MP E der. do erro é PN  
Então Ki é Z.

OU

Se erro é MP E der. do erro é Z  
Então Ki é Z.

OU

Se erro é MP E der. do erro é PP  
Então Ki é Z.

OU

Se erro é MP E der. do erro é MP  
Então Ki é Z.

## BIBLIOGRAFIA

[Bald78] - Baldwin, J. F.; "Fuzzy Logic and its Application to Fuzzy Reasoning", *Advances in Fuzzy Sets Theory and Application*, Gupta, M., Ragade, R. e Yager, R. (eds); North-Holland, 1978.

[Bern88] - Bernard, J. A.; "Use of a Rule-Based Systems for Process Control", *IEEE Control Systems Magazine*, vol. 8, num. 5, 1988.

[Braa79] - Braae, M. e Rutherford, D. A.; "Selection of Parameters for a Fuzzy Logic Controller", *Fuzzy Sets and Systems*, vol. 12, 1979.

[Buck89] - Buckley, J. J. e Tucker, D. M.; "Second Generation Fuzzy Expert Systems", *Fuzzy Sets and Systems*, vol. 31, num 3, 1988.

[Chen80] - Chen, G.; "An Approach to Fuzzy Controller Algorithms", *Information and Control*, vol. 9, num. 3, 1980.

[Chen89] - Chen, Y. Y. e Tsao, T. L.; "A Description of the Dynamical Behavior of Fuzzy Systems", *IEEE Trans. on Systems, Man and Cybernetics*, vol. 19, num.4, 1988.

[Clan83] - Clancey, W. J.; "The Epistemology of a Rule-Based Expert System: A Framework for Explanation", *Artificial Intelligence*, vol. 20, num. 3, 1983.

[Clan89] - Clancey, W. J.; "Viewing Knowledge Bases as Qualitative Models", *IEEE Expert*, Summer 1989.

[Curt90] - Curtis, J. W., Youngblood, R. & Vachtsevanos, G.; "Fuzzy Logic Control: New Approaches and Applications", *Proceedings of Instruments Society of America 1990 (ISA90) Conference on Advances in Instrumentation and Control*, Louisiana, U.S.A. vol. 45, part 3, 1990.

[Czog81] - Czogala, E. & Hirota, K.; "Some Problems Concerning the Construction of Algorithms of Decision-Making in Fuzzy Systems", *Intern. Journal on Man-Machine Studies*, vol 15, 1981.

[Eric84] - Ericsson, K. A. & Simon, H.; "Protocol Analyses: Verbal Report as Data", *MIT Press*, 1984.

[Forb84] - Forbus, K. D.; "Qualitative Process Theory", *Artificial Intelligence*, vol. 24, num. 2, 1984.

[Gupt88] - Gupta, M. M.; "Design of Cognitive Controller", *IEEE Proceedings of the 27<sup>th</sup> Conference on Decision and Control*, Texas, U. S. A., 1988.

[Grah88] - Graham, B. P. & Newell, R. B.; "Fuzzy Adaptive Control of a first-order Process", *Fuzzy Sets Systems*, vol. 31, 1989.

[Hall88] - Hall, L. O., Friedmann, M. & Kandel, A.; "On the Validation and Testing of Fuzzy Expert Systems", *IEEE Trans. on Systems, Man and Cybernetics*, vol. 18, num. 6, 1988.

[Hart86] - Hart, A.; "Knowledge Acquisition for Expert Systems", *McGraw-Hill*, 1986.

[Hoff87] - Hoffman, R. R.; "The Problem of Extracting the Knowledge of Expert from the Perspective of Experimental Psychology", *AI Magazine*, vol. 3, 1987.

[Hunt84] - Hunt, R. M. e Rouse, W. B.; "A Fuzzy-Based Model of Human Problem Solving", *IEEE Trans. on Systems, Man and Cybernetics*, vol. 14, num. 1, 1984.

[Iser81] - Isermann, R.; *Digital Control Systems*, Springer-Verlag, 1981.

[Kauf75] - Kaufmann, A.; *Introduction to the Theory of Fuzzy Sets*, vol. 1, Academic Press, 1975.

[Kauf88] - Kaufmann, A.; "Theory of Expertons and Fuzzy Logic", *Fuzzy Sets and Systems*, vol. 28, num 3, 1988.

[Kel155] - Kelly, G. A.; "Psychology of Personal Constructs", *W. W. Norton Press*, 1955.

[Kidd87] - Kidd, A.; "Knowledge Acquisition for Expert Systems", Kidd, A. (ed), *Plenum Press*, 1987.

[Kick78a] - Kickert, W. J. M. e Mamdani, E.; "Analyses of Fuzzy Logic Controller", *Fuzzy Sets and Systems*, vol. 1, num. 1, 1978.

[Kick78b] - Kickert, W. J. M.; "Toward an Analyses of Linguistics Modelling", *Fuzzy Sets and Systems*, vol. 3, 1978.

[Kick79] - Kickert, W. J. M.; "An Exemple of Linguistic Modelling", *Advances in Fuzzy Sets Theory and Application*, Gupta, M., Ragade, R. e Yager, R. (eds); North-Holland, 1978.

[King77] - King, P.J. e Mamdani, E.; "The Application of Fuzzy Control Systems to Industrial Processes", *Automatica*, vol. 13, 1977.

[King80] - King, P. e Umbers, I.; "An Analysis of Human Decision-Making in Cement Kiln Control and Implications for Automation", *Int. J. Man-Machine Studies*, vo. 12, 1980.

[Mamd77a] - Mamdani, E.; "Application of Fuzzy Logic to Aproximate Reasoning using Linguistic Synthesis", *IEEE Trans. on Computer*, vol. 12, num. 26, 1977.

[Mamd77b] - Mamdani, E.; "Application of Fuzzy Sets Theory to Control Systems: A Survey", *Fuzzy Automata and Decision Processes*, Gupta, M., Saridis, G. e Gaines, B. (eds), North-Holland, 1977.

[McCo88] - McCoy, M. S. e Levary, R.; "Augmenting Knowledge Acquisition Processes of Expert Systems with Human Performance Modelling Techniques", *IEEE Trans. on Systems, Man and Cybernetics*, vol. 18, num. 3, 1988.

[Mizu78] - Mizumoto, M. Fukami, S.; "Some Methods of Fuzzy Reasoning", *Advances in Fuzzy Sets Theory and Application*, Gupta, M., Ragade, R. e Yager, R. (eds), North-Holland, 1978.

[Mota90] - Mota, A. S. e Andrade Netto, M. L.; "Controladores Baseados em Regras Nebulosas", *Anais do 8º Congresso da Sociedade Brasileira de Automação*, Pará, Brasil, 1990.

[Naut76] - van Nauta Lemke, H. R. e Kickert, W. J. M.; "Application of a Fuzzy Controller in a Warm Water Plant", *Automatica*, vol. 12, 1976.

[Naut85] - van Nauta Lemke, H. R. e De-Zhao, W.; "Fuzzy PID Supervisor", *IEEE Proceedings of the 25º Conference on Decision and Control*, Florida, U. S. A., 1985.

[Nego78a] - Negoita, C. V.; "On Fuzzy Sets", *Advances in Fuzzy Sets Theory and Application*, Gupta, M., Ragade, R. e Yager, R. (eds); North-Holland, 1978.

[Nego78b] - Negoita, C. V.; "On the Stability of Fuzzy Systems", *Proceedings IEEE International Conference on Cybernetics and Society*, 1978.

[Kowa70] - Kowalski, R.; "Logic for Problem Solving",  
Elsivier Science Publishing, 1987.

[Krij88] - Krijgsman, A. J., Verbruggen, H., e Bruijn, P. M.;  
"Knowledge-Based Control", *IEEE Proceedings of the 27<sup>th</sup>  
Conference on Decision and Control*, Texas, U.S.A., 1988.

[Lee90a] - Lee, C. C.; "Fuzzy Logic in Control Systems: Fuzzy  
Logic Controller - Part I", *IEEE Trans. on Systems, Man and  
Cybernetics*, vol. 20, num. 2, 1990.

[Lee90b] - Lee, C. C.; "Fuzzy Logic in Control Systems: Fuzzy  
Logic Controller - Part II", *IEEE Trans. on Systems, Man and  
Cybernetics*, vol. 20, num. 2, 1990.

[Li89] - Li, Y. F. e Lau, C.C.; "Development of a Fuzzy  
Algorithms for Servo Systems", *IEEE Control Systems  
Magazine*, vol. 9, num. 3, 1989.

[Macv76] - MacVicar-Whelan, P. J.; "Fuzzy Set for Man-Machine  
Interaction", *Intern. Journal on Man-Machine Studies*, vol. 8,  
1976.

[Maed88] - Maeda, M. e Murakami, S.; "A Design for a Fuzzy  
Logic Controller", *Information Sciences*, vol. 45, 1988.

[Maie85] - Maiers, J. e Sherif, Y. S.; "Application of Fuzzy  
Set Theory", *IEEE Trans. on Systems, Man and Cybernetics*,  
vol. 15, num. 1, 1985.

[Mamd74] - Mamdani, E.; "Application of Fuzzy Algorithms for  
Control of Simple Dynamic Plant", *Proceedings of the IEE*,  
vol. 121, num. 12, 1974.

[Mamd75] - Mamdani, E., e Assilian, S.; "An Experiment in  
Linguistic Synthesis with a Fuzzy Logic Controller", *Intern.  
Journal on Man-Machine Studies*, vol. 7, 1975.

[Nego85] - Negoite , C. V.; *Expert Systems and Fuzzy Systems*, Benjamin Cummings, 1985.

[Newe82] - Newell, A.; "Knowledge Level", *Artificial Intelligence*, vol. 18, num. 1, 1982.

[Nils82] - Nilsson. N.; *Principles of Artificial Intelligence*, Tioga Publishers, 1982.

[Oll88] - Ollero, A. e García - Cerezo, A. J.; "Direct Digital Control, Auto-Tuning and Supervision using Fuzzy Logic", *Fuzzy Sets and Systems*, vol. 30 , num. 2, 1988.

[Peel88] - Peel, D. e Morris, A. J.; "Artificial Intelligence for Process Regulation and Servo Control", *Industrial Digital Control Systems - IEE Control Engineering Series num. 29* (2ª edição), Warwick, K. e Rees, D. (eds); Peter Peregrinus Ltd, 1988.

[Port87] - Porter, B., Jones, A. H. e McKeown, C. B.; "Real-Time Expert Tunner for PI Controller", *IEE Proceedings on Automatic Control and Applications*, vol.1 34, num. 4, 1987.

[Proc79] - Procyk, T. J. e Mamdani, E.; "A Linguistic Self-Organising Processes Controller", *Automatica*, vol. 15, 1979.

[Rega78] - Regade, R. K. e Gupta, M.; "Fuzzy Set Theory and Application : a Synthesis", *Advances in Fuzzy Sets Theory and Application*, Gupta, M., Ragade, R. e Yager, R. (eds); North-Holland, 1978.

[Rich82] - Rich, E.; *Artificial Intelligence*, MacGraw-Hill, 1982.

[Ruth76] - Rutherford, D. A. e Bloore, G. C.; "The Implementation of Fuzzy Algorithms for Control", *Proceedings of the IEEE*, April 1976.

[Shao88] - Shao, S.; "Fuzzy Self-Organizing Controller and its Application for Dynamic Processes", *Fuzzy Sets and Systems*, vol. 26, 1988.

[Sowa86] - Sowa, J. F.; "Conceptual Structures Information Processing in the Mind and Machine", Addison-Wesley, 1986.

[Spin90] - Spinrad, M. e Ray, J.; "Fuzzy Logic Issues for Process Control", *Proceedings of Instruments Society of America 1990 (ISA90) Conference on Advances in Instrumentation and Control*, Louisiana, U.S.A., vol. 45, part 3, 1990.

[Suge85a] - Sugeno, M.; "An Introductory Survey of Fuzzy Control", *Information and Science*, num. 36, 1985.

[Suge85b] - Sugeno, M. e Nishida, M; "Fuzzy Control of Model Car", *Fuzzy Sets and Systems*, vol.16, 1985.

[Tong77] - Tong, R. M.; "A Control Engenering Review of Fuzzy Systems", *Automatica*, vol. 13, 1977.

[Tong78] - Tong, R. M.; "The Construction and Evaluation of Fuzzy Models", *Advances in Fuzzy Sets Theory and Application*, Gupta, M., Ragade, R. e Yager, R. (eds); North-Holland, 1978.

[Tong80] - Tong, R. M. e Beck, M. B.; "Fuzzy Control of The Activade Sludge Wastewater Treatment Process", *Automatica*, vol. 16, 1980.

[Tong84] - Tong, R. M.; "A Retrospective view of Fuzzy Control Systems", *Fuzzy Sets and Systems*, vol. 14, 1984.

[Wate78] - Waterman, D. A. e Hayes-Roth, F.; "An Overview of Pattern-Directed Inference Systems", *Pattern-Directed Inference Systems*, Waterman, D. A. e Hayes-Roth, F. (eds), Academic Press, 1978.

[Wie187] - Wieling, B. e Breuker, J.; "Model of Expertise", *Advances in Artificial Intelligence II*, Du Boulay, B., Hogg, D. e Stells, L. (eds), North-Holland, 1987.

[Wrig87] - Wright, G. e Ayton, P.; "Eliciting and Modelling Expert Knowledge", *Decision-Suport Systems*, vol. 3, Elsvier Science Publishing, 1987.

[Zade65] - Zadeh, L. A.; "Fuzzy Sets", *Information and Control*, vol. 8, 1965.

[Zade68] - Zadeh, L. A.; "Fuzzy Algorithms", *Information and Control*, vol. 12, 1968.

[Zade73] - Zadeh, L. A.; "Outline of a New Approach to the Analyses of Complex Systems and Decision Processes", *IEEE Trans. on Systems, Man and Cybernetics*, vol. 3, num. 1, 1973.

[Zade75a] - Zadeh, L. A.; "The Concept of Linguistic Variable and its Application to Approximate Reasoning Part I", *Information and Science*, vol. 8, 1975.

[Zade75b] - Zadeh, L. A.; "The Concept of Linguistic Variable and its Application to Approximate Reasoning Part II", *Information and Science*, vol. 8, 1975.

[Zade83a] - Zadeh, L. A.; "Commonsense Knowledge Representation Based on Fuzzy Logic", *IEEE Computer Magazin*, October, 1983.

[Zade83b] - Zadeh, L. A.; "The Role of Fuzzy Logic in the Management of Uncertainty in Expert Systems", *Fuzzy Sets and Systems*, vol. 11, 1983.

[Zade84a] - Zadeh, L. A.; "Making Computer Think like People", *IEEE Spectrum*, August 1984.

[Zade84b] - Zadeh, L. A.; "Coping with the Imprecision of the Real Word - An Interview with Lotfi A. Zadeh", *Communications of the ACM*, vol. 27, num. 4, 1984.

[Zade88] - Zadeh, L. A.; "Fuzzy Logic", *IEEE Computer Magazin*, April, 1988.

[Zieg42] - Ziegler, J. G. e Nichols, N. B.; "Optimum Setting for Automatic Controllers", *Transaction of the A.S.M.E.*, vol. 64, num. 8, 1942.